



Version 2019.1

The Official Help File



Table of Contents

1	ProcessRobot	15
1.1	About Softomotive	16
1.2	Release History	17
1.3	ProcessRobot Introduction	19
1.4	Hardware requirements	22
1.4.1	Trial Installation Requirements	22
1.4.2	Live Installation Requirements	22
1.5	Installation Guide	24
1.5.1	Installing ProcessRobot	24
1.5.2	Creating a Self-Signed Certificate	43
1.5.3	Configuring Custom Authentication after Installation	51
1.5.4	Installing SQL Server Express	60
1.5.5	Installing Microsoft SQL Express 2016	79
1.5.6	Installing SQL Server Management Studio	83
1.5.7	Setting up the ProcessRobot Database	84
1.5.8	Installing the Web Console	91
1.5.9	Configuring Elastic Search	101
1.5.10	Optional: Securing ProcessRobot Directory from unauthorized access	103
1.5.11	FAQ	105
1.5.12	Install MODI	107
1.5.13	Enabling the Filestream	108
1.5.14	BotStack	113
1.5.15	BotStack Requirements	116
2	Control Desk Components	119
2.1	Control Desk	120
2.1.1	Control Desk Introduction	120
2.1.2	Control Desk Tabs	121
2.1.2.1	Home	121
2.1.2.2	Monitor	121
2.1.2.3	Processes	124
2.1.2.3.1	Run/Stop a Process	126
2.1.2.3.2	Process Properties	128
	Enable/Disable a Process	128
	General	130
	Deployment	131
	Execution	135
	Maker-Checker	136
	Concurrency Policy	138
	Error Handling	139
	Credentials	141
2.1.2.3.3	Processes Security	143
2.1.2.4	Robots	147

2.1.2.4.1	Remote Control	148
2.1.2.5	Queues	151
2.1.2.6	Triggers	152
2.1.2.6.1	What Triggers Are	152
2.1.2.6.2	Trigger Variables	153
2.1.2.6.3	Triggers tab	154
2.1.2.6.4	How to set a Trigger to a Process	155
2.1.2.6.5	Trigger Types	159
	Email Monitor Trigger	159
	Exchange Email Monitor Trigger	161
	Event Log Monitor Trigger	164
	File Monitor Trigger	165
	Idle Monitor Trigger	167
	Ping Trigger	167
	Service Monitor Trigger	168
	Process Monitor Trigger	169
	Database Monitor Trigger	170
	Performance Counter Trigger	172
2.1.2.7	Scheduler	173
2.1.2.7.1	Scheduler Tab	173
2.1.2.7.2	Schedule	174
2.1.2.7.3	Recurring Schedule	176
2.1.2.8	Audit	178
2.1.2.8.1	ProcessRobot Records (Logging)	178
2.1.2.8.2	Audit Tab	179
2.1.2.8.3	Error Video Log	188
2.1.2.8.4	Event Ids	189
2.1.2.9	Settings	190
2.1.2.9.1	General	191
2.1.2.9.2	Roles	193
2.1.2.9.3	Users	195
2.1.2.9.4	Groups	199
2.1.2.9.5	Robots	200
	Add a new SideBot	201
	Add a new Solobot	203
	SoloBot Authentication	208
2.1.2.9.6	Machines	210
2.1.2.9.7	Robot Pools	217
2.1.2.9.8	Queues	220
	How Queues Work	224
2.1.2.9.9	Dashboards	233
2.1.2.9.10	Environments	235
2.1.2.9.11	Global Variables	237
2.1.2.9.12	Credential Manager	241
2.1.2.9.13	Error Handling	248
2.1.2.9.14	License	250
2.2	Control Desk Web Console	251
2.2.1	Structure of Web Console	252
2.2.1.1	Monitor	254
2.2.1.2	Processes	257
2.2.1.3	Robots	260
2.2.1.4	Scheduler	262
2.2.1.5	Audit	268
2.2.1.6	Standard Control Elements	271

3	Process Studio	275
3.1	Process Studio - Introduction	276
3.2	Process Studio	277
3.2.1	ProcesStudio Window	277
3.2.2	Create a new Folder	281
3.2.3	Create a new Process	281
3.2.4	Import/Export a Process	283
3.2.5	The Notification Popup Window	284
3.3	Process Designer	285
3.3.1	The Actions Pane	285
3.3.2	The Workspace	286
3.3.3	The Variables Pane	288
3.3.4	The Variables Visualizer Window	290
3.3.5	The Errors Pane	294
3.3.6	The Control Repository Pane	296
3.3.7	The Image Repository Pane	297
3.3.8	The Search Results Pane	297
3.3.9	Go to Line	299
3.3.10	Functions	301
3.3.11	Managing Functions	302
3.3.12	Adding an Action	303
3.3.13	Configuring an Action	304
3.3.14	Enable/Disable Actions	306
3.3.15	Organize your Actions with Regions	307
3.3.16	Adding Comments	309
3.3.17	The Variables Manager Window	311
3.3.18	Process Designer Options	312
3.3.19	Process Designer Custom Layout	314
3.4	The Debugger	315
3.4.1	Run/Stop/Pause a Process from Process Designer	315
3.4.2	Adding Breakpoints	317
3.4.3	Execute step by step	318
3.4.4	Rename a Variable	319
3.4.5	Full Error View	320
3.5	The Macro Recorder	321
3.5.1	Simulating the activity of a user	321
3.5.2	Recording vs Building a Robot	321
3.5.3	Macro Recorder	322
3.6	The Custom Dialog Designer	329
3.6.1	Designing a Custom Dialog	329
3.7	UI Automation	336
3.7.1	UI Automation Overview	336
3.7.2	UI Elements - Select Controls	338
3.7.3	Manage Control Repository	342
3.7.4	Manage UI Selectors	343
3.7.5	Building UI Selectors	346
3.7.6	UI Data Extraction	350
3.8	Web Automation	351
3.8.1	Web Automation Overview	351
3.8.2	Building a WebAutomation Process	353

3.8.3	The Web Recorder	360
3.8.4	Web Data Extraction Part I	373
3.8.5	Web Data Extraction Part II	378
3.8.6	Web Data Extraction Part III	382
3.8.7	Configure IE for Web Automation	383
3.8.8	Configure Chrome and Firefox for Web Automation	387
3.8.9	Configure Internet Explorer for Servers	391
3.8.10	Actual Internet Explorer VS Automated Browser	393
3.8.11	Web Helpers Offset	393
3.9	User Libraries	394
3.9.1	Why Use User Libraries?	394
3.9.2	Accessing User Library Designer	395
3.9.3	User Library Designer vs. Process Designer	397
3.9.4	Structure of User Library Designer	398
3.9.4.1	The User Actions pane	405
3.9.5	User Actions and Functions	413
3.9.6	Sharing User Libraries	414
3.9.6.1	Importing/Exporting User Libraries	415
3.9.6.2	Publishing User Libraries	415
3.9.7	Editing User Libraries	421
3.9.8	Permissions and User Libraries	422
3.9.9	An Example on User Libraries	423
3.10	Advanced Topics	435
3.10.1	Variables	435
3.10.1.1	Data Types	436
3.10.1.2	Lists	438
3.10.1.3	Data Tables	439
3.10.1.4	Data type Properties	439
3.10.1.5	Persistent Variables	442
3.10.1.6	External Variables	444
3.10.1.7	Custom Date Formats	444
3.10.1.8	Custom Objects	446
3.10.2	Loops	452
3.10.2.1	Simple Loops	452
3.10.2.2	Loop Condition	453
3.10.2.3	For Each	454
3.10.3	Version Control	455
3.10.3.1	Process Lifecycle	455
3.10.3.2	Manage Process Versions	462
3.10.3.3	Save and Commit	465
3.10.3.4	Compare two Process Versions	467
3.10.4	Connecting to a Database	472
3.10.4.1	Connect to Excel or Access Database	473
3.10.4.2	Connection Strings and Drivers	473
3.10.5	Conditionals	474
3.10.5.1	If/Else/EndIf	474
3.10.5.2	If Variations	475
3.10.5.3	Switch-Case	476
3.10.6	Enate Actions	477
3.10.6.1	Authentication on Enate	478
3.10.6.2	Retrieving Work from Enate	480
3.10.6.3	Creating Cases on Enate	481
3.10.6.4	Using Enate Actions in Processes	483
3.10.7	CSS Selectors	485

3.10.8	Email Interaction with Two Step Verification	489
3.10.9	The use of the % sign	501
3.10.10	Exception Handling	502
3.10.11	Exception Block	504
3.10.12	Image Recognition	506
3.10.13	Interactive/Non Interactive Processes	507
3.10.14	Valid Key Codes for the Send Keys Action	508
3.10.15	Error-Exception Handling Levels	510
3.10.16	Terminal Emulation - Guide	514
3.10.17	Interaction with JAVA-based applications	516
3.10.18	Interaction with SAP	517
3.10.19	CaptureFast Actions	523
3.10.20	Integration with CyberArk	526
3.11	Actions Reference	529
3.11.1	System	529
3.11.1.1	Run Application Action	529
3.11.1.2	Run DOS Command Action	531
3.11.1.3	Run VBScript Action	533
3.11.1.4	Run Javascript Action	535
3.11.1.5	Run PowerShell Script	536
3.11.1.6	Run Python Script Action	537
3.11.1.7	Terminate Process Action	539
3.11.1.8	Print Document Action	540
3.11.1.9	Get Default Printer Action	540
3.11.1.10	Set Default Printer Action	541
3.11.1.11	Log Off User Action	542
3.11.1.12	Shutdown Computer Action	543
3.11.1.13	Show Desktop Action	544
3.11.1.14	Lock Workstation Action	545
3.11.1.15	Play Sound Action	546
3.11.1.16	Empty Recycle Bin Action	547
3.11.1.17	Take ScreenShot Action	548
3.11.1.18	Control Screen Saver Action	550
3.11.1.19	Ping Action	550
3.11.1.20	Set Environmental Variable Action	552
3.11.1.21	Get Environmental Variable Action	552
3.11.1.22	Delete Environmental Variable Action	553
3.11.1.23	Get Screen Resolution	554
3.11.1.24	Set Screen Resolution	556
3.11.2	Conditionals	557
3.11.2.1	If Action	557
3.11.2.2	If File Exists Action	559
3.11.2.3	If Folder Exists Action	560
3.11.2.4	If Service Action	561
3.11.2.5	If Process Action	562
3.11.2.6	If Window Action	562
3.11.2.7	If Image Action	566
3.11.2.8	If Text on Screen (OCR)	567
3.11.2.9	If Window Contains Action	571
3.11.2.10	If Web Page Contains Action	573
3.11.2.11	Else Action	576
3.11.2.12	Else If Action	576
3.11.2.13	End If Action	577
3.11.2.14	Switch	578

3.11.2.15	End Switch	580
3.11.2.16	Case	580
3.11.2.17	End Case	583
3.11.2.18	Default Case	583
3.11.2.19	End Default Case	583
3.11.3	Loops	584
3.11.3.1	Loop Action	584
3.11.3.2	Loop Condition Action	586
3.11.3.3	For Each Action	587
3.11.3.4	Exit Loop Action	587
3.11.3.5	End Loop Action	588
3.11.4	Wait	589
3.11.4.1	Wait Action	589
3.11.4.2	Wait for File Action	589
3.11.4.3	Wait for HotKey Action	590
3.11.4.4	Wait for Process Action	593
3.11.4.5	Wait for Service Action	594
3.11.4.6	Wait for Window Action	595
3.11.4.7	Wait for Image Action	598
3.11.4.8	Wait for Text on Screen (OCR)	600
3.11.4.9	Wait for Mouse Action	605
3.11.4.10	Wait for Window Content Action	606
3.11.4.11	Wait for Web Page Content Action	608
3.11.5	Flow Control	611
3.11.5.1	Label Action	611
3.11.5.2	Go To Action	611
3.11.5.3	Run Function Action	612
3.11.5.4	Exit Function Action	613
3.11.5.5	Stop Process Action	614
3.11.5.6	Begin Exception Block Action	615
3.11.5.7	End Exception Block Action	617
3.11.6	Mouse and Keyboard	617
3.11.6.1	Block Input Action	617
3.11.6.2	Get Mouse Position Action	618
3.11.6.3	Move Mouse Action	619
3.11.6.4	Move Mouse to Image Action	621
3.11.6.5	Move Mouse to Text on Screen (OCR)	624
3.11.6.6	Send Mouse Click Action	629
3.11.6.7	Send Keys Action	630
3.11.6.8	Press/Release Key Action	632
3.11.6.9	Set Key State Action	633
3.11.7	Message Boxes	634
3.11.7.1	Display Message Action	634
3.11.7.2	Display Custom Dialog Action	636
3.11.7.3	Display Notification Action	638
3.11.7.4	Display Input Dialog Action	638
3.11.7.5	Display Select Date Dialog	640
3.11.7.6	Display Select From List Dialog Action	642
3.11.7.7	Display Select File Dialog Action	645
3.11.7.8	Display Select Folder Dialog Action	647
3.11.8	Files	649
3.11.8.1	Get Files in Folder Action	649
3.11.8.2	Copy File(s) Action	651
3.11.8.3	Move File(s) Action	652

3.11.8.4	Delete File(s) Action	653
3.11.8.5	Rename File(s) Action	654
3.11.8.6	Read Text from File Action	657
3.11.8.7	Write Text to File Action	659
3.11.8.8	Read From CSV File Action	660
3.11.8.9	Write to CSV File Action	661
3.11.8.10	Get FilePath Part Action	663
3.11.8.11	Get Temporary File Action	664
3.11.9	Folders	665
3.11.9.1	Get Subfolders in Folder Action	665
3.11.9.2	Create Folder Action	667
3.11.9.3	Delete Folder Action	668
3.11.9.4	Empty Folder Action	669
3.11.9.5	Copy Folder Action	670
3.11.9.6	Move Folder Action	672
3.11.9.7	Rename Folder Action	673
3.11.9.8	Get Special Folder Action	674
3.11.10	Compression	675
3.11.10.1	Zip Files Action	675
3.11.10.2	Unzip Files Action	676
3.11.11	Clipboard	678
3.11.11.1	Get Clipboard Text Action	678
3.11.11.2	Set Clipboard Text Action	679
3.11.11.3	Clear Clipboard Contents Action	679
3.11.12	UI and Windows	680
3.11.12.1	Use Desktop Action	680
3.11.12.2	Select Tab in Window Action	682
3.11.12.3	Click Element in Window Action	682
3.11.12.4	Select Menu Option in Window Action	683
3.11.12.5	Drag and Drop Element in Window Action	686
3.11.12.6	Expand / Collapse Tree Node in Window Action	688
3.11.12.7	Windows	690
3.11.12.7.1	Get Window Action	690
3.11.12.7.2	Focus Window Action	692
3.11.12.7.3	Set Window State Action	694
3.11.12.7.4	Set Window Visibility Action	698
3.11.12.7.5	Move Window Action	701
3.11.12.7.6	Resize Window Action	704
3.11.12.7.7	Close Window Action	708
3.11.12.8	Form Filling	710
3.11.12.8.1	Focus Text Field in Window Action	710
3.11.12.8.2	Populate Text Field in Window Action	711
3.11.12.8.3	Press Button in Window Action	712
3.11.12.8.4	Select Radio Button in Window Action	713
3.11.12.8.5	Set Checkbox State in Window Action	714
3.11.12.8.6	Set Dropdown List Value in Window Action	715
3.11.12.9	Data Extraction	717
3.11.12.9.1	Get Details of Window Action	717
3.11.12.9.2	Get Details of Element in Window Action	719
3.11.12.9.3	Get Selected Checkboxes in Window Action	721
3.11.12.9.4	Get Selected Radiobutton in Window Action	722
3.11.12.9.5	Extract Data from Window Action	723
3.11.13	Web Automation	726
3.11.13.1	Launch New Internet Explorer Action	726

3.11.13.2	Launch New Firefox	729
3.11.13.3	Launch New Chrome	734
3.11.13.4	Create New Tab	739
3.11.13.5	Go to Web Page Action	741
3.11.13.6	Click Link on Web Page Action	743
3.11.13.7	Click Download Link on Web Page Action	744
3.11.13.8	Hover Mouse over element on Web Page Action	746
3.11.13.9	Close Web Browser	746
3.11.13.10	Execute Javascript Function On Web	747
3.11.13.11	Web Forms	751
3.11.13.11.1	Focus Text Field on Web Page Action	751
3.11.13.11.2	Populate Text Field on Web Page Action	753
3.11.13.11.3	Set Checkbox State on Web Page Action	755
3.11.13.11.4	Select Radio Button on Web Page Action	757
3.11.13.11.5	Set DropDown List Value on Web Page Action	758
3.11.13.11.6	Press Button on Web Page Action	760
3.11.13.12	Web Data Extraction	762
3.11.13.12.1	Extract Data from Web Page Action	762
3.11.13.12.2	Get Details of Web Page Action	765
3.11.13.12.3	Get Details of Element on Web Page Action	766
3.11.13.12.4	Take Screenshot of Web Page Action	767
3.11.13.13	Direct Web Access	769
3.11.13.13.1	Download from Web Action	769
3.11.13.13.2	Invoke Web Service Action	772
3.11.14	FTP	776
3.11.14.1	Open FTP Connection Action	776
3.11.14.2	Open Secure FTP Connection Action	778
3.11.14.3	Close FTP Connection Action	780
3.11.14.4	List FTP Directory Action	781
3.11.14.5	Change Working Directory Action	782
3.11.14.6	Download File(s) from FTP Action	783
3.11.14.7	Download Folder(s) from FTP Action	785
3.11.14.8	Upload File(s) to FTP Action	786
3.11.14.9	Upload Folder(s) to FTP Action	787
3.11.14.10	Delete FTP File Action	788
3.11.14.11	Rename FTP File Action	789
3.11.14.12	Create FTP Directory Action	790
3.11.14.13	Delete FTP Directory Action	791
3.11.14.14	Synchronize FTP Directory Action	792
3.11.14.15	Invoke FTP Command Action	796
3.11.15	Email	797
3.11.15.1	Retrieve Emails Action	797
3.11.15.2	Process Emails Action	806
3.11.15.3	Send Email Action	808
3.11.16	Exchange	812
3.11.16.1	Connect to Exchange Server	812
3.11.16.2	Retrieve Exchange Email Messages	816
3.11.16.3	Send Exchange Email Message	820
3.11.16.4	Process Exchange Email Messages	822
3.11.17	Cmd Session	824
3.11.17.1	Open Cmd Session	824
3.11.17.2	Close Cmd Session	826
3.11.17.3	Write to Cmd Session	826
3.11.17.4	Read from Cmd Session	828

3.11.17.5	Wait for Text on Cmd Session	829
3.11.18	Cognitive	830
3.11.18.1	Google	830
3.11.18.1.1	Vision	830
	Face Detection	830
	Label Detection	831
	Landmark Detection_2_2	833
	Text Detection	834
	Logo Detection	836
	Image Properties Detection	837
	Safe Search Detection	839
3.11.18.1.2	Natural Language	840
	Analyze Sentiment	840
	Analyze Entities	842
	Analyze Syntax	844
3.11.18.2	IBM	846
3.11.18.2.1	Visual Recognition	846
	Classify Image	846
	Detect Faces	848
3.11.18.2.2	Document Conversion	850
	Convert Document	850
3.11.18.2.3	Tone Analyzer	852
	Analyze Tone	852
3.11.18.2.4	Language Translator	854
	Translate	854
	Identify Language	856
3.11.18.3	Microsoft	858
3.11.18.3.1	Computer Vision	858
	Analyze Image	858
	Describe Image	860
	OCR	861
	Tag Image	863
3.11.18.3.2	Face	865
	Detect Face	865
	Find Similar Face	867
3.11.18.3.3	Text Analytics	869
	Detect Language	869
	Key Phrases	871
	Sentiment	872
3.11.18.3.4	Bing Spell Check	874
	Spell Check	874
3.11.19	Database	875
3.11.19.1	Open SQL Connection	875
3.11.19.2	Execute SQL Statement Action	876
3.11.19.3	Close SQL Connection	878
3.11.20	Excel	878
3.11.20.1	Launch Excel Action	878
3.11.20.2	Attach to Running Excel Action	880
3.11.20.3	Read from Excel Worksheet	880
3.11.20.4	Save Excel	883
3.11.20.5	Write to Excel Worksheet	884
3.11.20.6	Close Excel Action	886
3.11.20.7	Set Active Worksheet Action	887
3.11.20.8	Add New Worksheet Action	888

3.11.20.9	Delete Excel Worksheet Action	889
3.11.20.10	Rename Excel Worksheet Action	890
3.11.20.11	Get Active Excel Worksheet Action	892
3.11.20.12	Get All Excel Worksheets Action	893
3.11.20.13	Activate Cell in Excel Worksheet Action	894
3.11.20.14	Select Cells in Excel Worksheet Action	896
3.11.20.15	Get Selected Cell Range From Excel Worksheet Action	898
3.11.20.16	Copy Cells in Exel Worksheet Action	900
3.11.20.17	Paste Cells to Excel Worksheet Action	903
3.11.20.18	Insert Row in Excel Worksheet Action	904
3.11.20.19	Delete Row from Excel Worksheet Action	905
3.11.20.20	Insert Column to Excel Worksheet Action	906
3.11.20.21	Delete Column from Excel Worksheet Action	907
3.11.20.22	Get First Free Column/Row Action	907
3.11.20.23	Get First Free Row on Column from Excel Worksheet Action	908
3.11.20.24	Run Excel Macro	909
3.11.21	PDF	911
3.11.21.1	Extract Text from PDF	911
3.11.21.2	Extract Text From PDF With OCR	914
3.11.21.3	Extract Images From PDF	917
3.11.21.4	Extract PDF Pages to New PDF	920
3.11.21.5	Merge PDF Files	921
3.11.22	Terminal Emulation	922
3.11.22.1	Open Terminal Session	922
3.11.22.2	Close Terminal Session	924
3.11.22.3	Move Cursor on Terminal Session	925
3.11.22.4	Get Text from Terminal Session	925
3.11.22.5	Set Text on Terminal Session	927
3.11.22.6	Send Key to Terminal Session	929
3.11.22.7	Wait for Text on Terminal Session	930
3.11.23	OCR	931
3.11.23.1	Create Tesseract OCR Engine	931
3.11.23.2	Create MODI OCR Engine	934
3.11.23.3	Get Data with ABBYY FlexiCapture	937
3.11.23.4	Extract Text With OCR	940
3.11.24	CaptureFast	943
3.11.24.1	Connect to CaptureFast Account	943
3.11.24.2	Upload Document to CaptureFast	945
3.11.24.3	Get Results From CaptureFast Document	946
3.11.25	CyberArk	948
3.11.25.1	Get Password From CyberArk	948
3.11.26	Enate	951
3.11.26.1	Login to Enate	951
3.11.26.2	Is Work Available on Enate	952
3.11.26.3	Get More Work from Enate	953
3.11.26.4	Create Case on Enate	953
3.11.26.5	Get Case from Enate	955
3.11.26.6	Update Case on Enate	956
3.11.26.7	Create Ticket on Enate	957
3.11.26.8	Get Ticket from Enate	958
3.11.26.9	Update Ticket on Enate	960
3.11.26.10	Get Action from Enate	961
3.11.26.11	Update Action on Enate	962
3.11.26.12	Update Checklist on Enate	963

3.11.26.13	Reject Work from Enate	963
3.11.26.14	Add File to Enate Item	964
3.11.26.15	Save File from Enate	965
3.11.26.16	Logout from Enate	966
3.11.27	Cryptography	967
3.11.27.1	Encrypt Text	967
3.11.27.2	Decrypt Text	970
3.11.27.3	Encrypt From File	973
3.11.27.4	Decrypt To File	976
3.11.27.5	Hash Text	980
3.11.27.6	Hash From File	981
3.11.27.7	Hash Text With Key	982
3.11.27.8	Hash From File With Key	983
3.11.28	Outlook	985
3.11.28.1	Launch Outlook Action	985
3.11.28.2	Retrieve Email Messages From Outlook Action	986
3.11.28.3	Send Email Through Outlook Action	988
3.11.28.4	Process Email Messages in Outlook Action	989
3.11.28.5	Save Outlook Email Messages Action	991
3.11.28.6	Respond To Outlook Email Message Action	992
3.11.28.7	Close Outlook Action	995
3.11.29	Services	995
3.11.29.1	Start Service Action	995
3.11.29.2	Stop Service Action	996
3.11.29.3	Pause Service Action	997
3.11.29.4	Resume Service Action	998
3.11.30	ProcessRobot Actions	999
3.11.30.1	Start Process Action	999
3.11.30.2	Get Command Line Arguments Action	1000
3.11.30.3	Get Process' Path Action	1001
3.11.30.4	Log Message Action	1002
3.11.30.5	Get Last Exception Action	1003
3.11.30.6	Update KPI Action	1005
3.11.30.7	Get Global Variable	1006
3.11.30.8	Set Global variable	1007
3.11.30.9	Enter Locked Region Action	1007
3.11.30.10	Exit Locked Region Action	1009
3.11.30.11	Add Queue Item in Queue Action	1010
3.11.30.12	Get Queue Item from Queue Action	1012
3.11.30.13	Get Credential	1015
3.11.30.14	Update Credential	1021
3.11.31	Text	1026
3.11.31.1	Get Text Length Action	1026
3.11.31.2	Append Line to Text Action	1027
3.11.31.3	Get Subtext Action	1028
3.11.31.4	Pad Text Action	1030
3.11.31.5	Trim Text Action	1031
3.11.31.6	Change Text Case Action	1032
3.11.31.7	Convert Text to Number Action	1033
3.11.31.8	Convert Number to Text Action	1034
3.11.31.9	Convert Text to DateTime Action	1036
3.11.31.10	Convert DateTime to Text Action	1037
3.11.31.11	Create Random Text Action	1038
3.11.31.12	Join Text Action	1040

3.11.31.13	Split Text Action	1041
3.11.31.14	Parse Text Action	1042
3.11.31.15	Replace Text Action	1044
3.11.31.16	Escape Text for Regular Expression Action	1046
3.11.32	DateTime	1047
3.11.32.1	Get Current Date and Time Action	1047
3.11.32.2	Add to DateTime Action	1048
3.11.32.3	Subtract Dates Action	1049
3.11.33	XML	1050
3.11.33.1	Read XML from File Action	1050
3.11.33.2	Write XML to File Action	1051
3.11.33.3	Execute XPath Expression Action	1052
3.11.33.4	Get XML Element Attribute Action	1053
3.11.33.5	Set XML Element Attribute Action	1055
3.11.33.6	Remove XML Element Attribute Action	1056
3.11.33.7	Get XML Element Value Action	1057
3.11.33.8	Set XML Element Value Action	1059
3.11.33.9	Insert XML Element Action	1060
3.11.33.10	Remove XML Element Action	1061
3.11.34	Variables	1062
3.11.34.1	Set Variable Action	1062
3.11.34.2	Increase Variable Action	1063
3.11.34.3	Decrease Variable Action	1064
3.11.34.4	Truncate Number Action	1064
3.11.34.5	Generate Random Number Action	1065
3.11.34.6	Get Items Count Action	1067
3.11.34.7	Create New List Action	1068
3.11.34.8	Clear List Action	1068
3.11.34.9	Add Item to List Action	1069
3.11.34.10	Remove Item from List Action	1070
3.11.34.11	Sort List Action	1071
3.11.34.12	Shuffle List Action	1073
3.11.34.13	Merge Lists	1073
3.11.34.14	Reverse List Action	1074
3.11.34.15	Remove Duplicate Items from List Action	1075
3.11.34.16	Find Common List Items Action	1076
3.11.34.17	Subtract Lists Action	1077
3.11.34.18	Retrieve DataTable Column into List Action	1078
3.11.34.19	Create New Custom Object	1080
3.11.34.20	Add Property To Custom Object	1080
3.11.34.21	Convert Json to Custom Object	1083
3.11.34.22	Convert Custom Object to Json	1084
3.11.35	User Actions	1087
4	Sidebot	1088
4.1	Sidebot Introduction	1089
4.2	Sidebot Interface Window	1090
4.3	Deploy a Process to a Sidebot	1091
5	Solobot	1094
5.1	Solobot Introduction	1095

6	Insights Dashboards	1097
6.1	Dashboard Viewer	1098
6.1.1	Dashboard Viewer Window	1098
6.1.2	Print Preview	1100
6.2	Dashboard Designer	1101
6.2.1	Dashboard Designer - Introduction	1101
6.2.2	Dashboard Designer Window	1102
6.2.3	Dashboard Elements	1103
6.2.3.1	Grids	1103
6.2.3.1.1	Data Tools	1103
6.2.3.1.2	Design Tools	1104
6.2.3.1.3	Data Items	1105
6.2.3.2	Charts	1105
6.2.3.2.1	Data Tools	1105
6.2.3.2.2	Design Tools	1106
6.2.3.2.3	Data Items	1106
6.2.3.3	Pies	1107
6.2.3.3.1	Data Tools	1107
6.2.3.3.2	Design Tools	1108
6.2.3.3.3	Data Items	1108
6.2.3.4	Text Boxes	1109
6.2.3.4.1	Design Tools	1109
6.2.3.5	Groups	1109
6.2.3.5.1	Data Tools	1109
6.2.3.5.2	Design Tools	1110
6.2.4	KPI Entries	1110
6.2.4.1	Entries	1110
6.2.4.2	Calculated Fields	1111
6.2.5	Data Items	1112
6.2.6	Print Preview	1113
6.3	Creating a Sample Dashboard	1115
6.3.1	Updating the KPI through the Process	1115
6.3.2	Creating The Dashboard	1116
6.3.3	Configuring a Calculated Fields	1118
6.3.4	Developing the Grid	1120
6.3.5	Developing the Chart	1121
6.3.6	Developing the Pie	1122
6.3.7	Completing the Dashboard	1124

1. ProcessRobot

1 ProcessRobot

1.1 About Softomotive

[We Talk Automation](#)

Softomotive offers sharp business solutions, premium support and the kind of professional services you deserve.

The company is one of the leading worldwide providers of Robotic Process Automation products and services, trusted by more than 6.000 companies worldwide.

With 10 years in the software automation market, Softomotive offers the most reliable and scalable RPA solutions, bridging the gap between state-of-the-art technology and unparalleled ease of use.

Our automation solutions are proven to deliver operational efficiency, reduce costs and empower brands.

- Feel free to contact us at: info@softomotive.com

More specifically:

- For Sales matters: Sales@softomotive.com
- For Support, if you are an active SSMP User: Support@softomotive.com
- For Professional Services: professionalservices@softomotive.com

Our Offices:

United Kingdom

LONDON

WeWork Devonshire Square

London, EC2M 4PL, United Kingdom

Phone: +44 207 048 2000

USA

HOUSTON

700 Milam St, Ste 1300

Houston, TX 77002-2736, USA

Phone: +1 346 561 0854

NEW YORK

One World Trade Center

285 Fulton Street Suite 46J

New York, NY 10007 USA

Phone: +1 646 757 4733

Greece

ATHENS

103 Kallirois St

Athens, 117 45, Greece

Phone: +30 218 218 0000

India

BANGALORE

Block 6A, 1st Floor, RMZ Ecoworld

Bellandur 560103

Bangalore Karnataka, India

Phone: +91 80614 29091

1.2 Release History

▣ Version 2019.1

General New Features:

- New Triggers added: Exchange Email Monitor, Database Monitor, Performance Counter
- Custom User and Machine Authentication is now supported
- It is now possible to host multiple SoloBots on a single machine
- A single installation file is used for all ProcessRobot components

Process Designer

- Added the option to remove all breakpoints in a Process
- Controls in the Controls Repository can now be sorted

New Actions:

- "Cryptography" group of Actions enables the encrypting, decrypting and hashing of data and files

- "CaptureFast" group of Actions provides integration with the cloud-based data capturing platform
- "CyberArk" group of Actions offers integration with the CyberArk secure password vault platform
- "Enate" group of Actions provides integration with the Enate service orchestration platform

Corrections and Improvements:

- Credentials in the Credential Manager can now be assigned different values based on the Process Lifecycle
- Security options regarding User permissions added to Credentials in the Credential Manager, User Libraries folders and Dashboards
- Recurring Schedules can be assigned a Start and Expiration date & time
- Added options for managing Process Versions in Version History
- Corrected sorting by Timestamp in Process Version History
- File Monitor Trigger no longer displays errors when deployed to SideBots
- Improved capturing and use of Pager elements in Google Chrome
- The Table Extraction Preview no longer omits the first row when an entire HTML Table is extracted
- Using NT Credentials during SoloBot Autologin no longer produces errors
- Actions inside a Switch block now correctly display an error message when not in a Case block
- Renaming previously undefined variables no longer causes an error
- The "Retrieve Emails" Action can now be configured to only retrieve Read emails
- Simultaneously importing multiple Licenses no longer causes an error
- The "Include Subfolders" option when synchronizing files on SFTP Servers now works as expected
- SoloBots can now use another User's credentials to execute a Process without errors
- HTTP PATCH Method added to "Invoke Web Service" Action
- Information window no longer pops up when the cursor does not hover over the ProcessRobot tray icon
- Improved performance in workstations where Java is not installed
- Using the same Function name in a User Library and a Process that uses it no longer produces an error
- "Occurrence" property added to the "Move Mouse to Image" Action
- "Find Usages" in the Process Designer is no longer case sensitive
- Improvements in UI and Web Automation
- Selector Builder can now be maximized

- Improved SAP Controls
- Improved performance in deleting large Processes and Folders

+ **Version 2018.1**

+ **Version 2017.2**

+ **Version 2017.1**

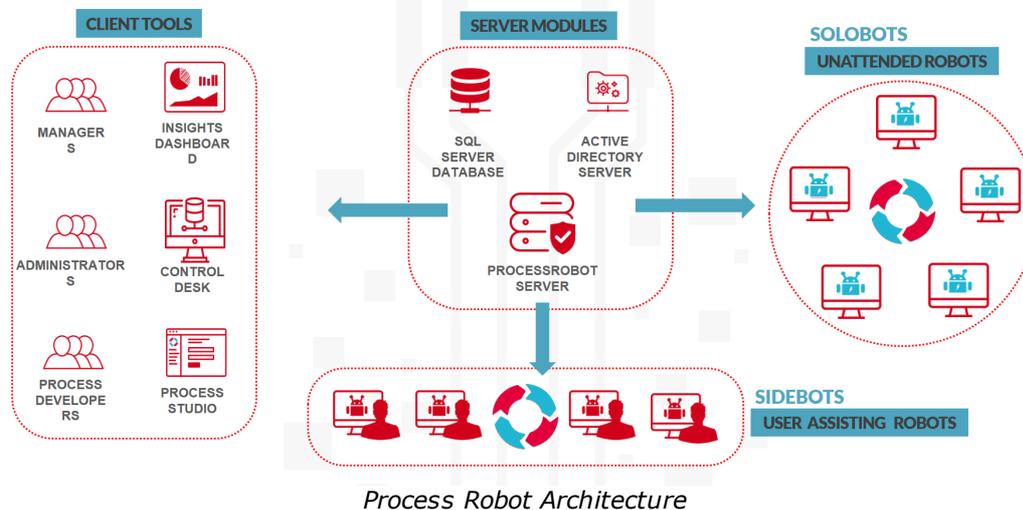
1.3 ProcessRobot Introduction

Process Robot is a powerful yet surprisingly easy to use, Enterprise grade RPA solution.

It is designed to allow you to meet the needs of your enterprise today while being able to scale instantly in the future, no matter the size or complexity of your business.

It consists of the Process Robot Server and a SQL Server and the client tools/Components that can be purchased separately at will (more than once each). The option to buy as many Components as you need, allows Process Robot to adapt to your business structure; helping you make the most of your hardware and human resources.

Below, the ProcessRobot architecture is displayed:



These five Components are:

1. The [Control Desk](#) ¹²⁰

This is the Component through which you can have the general Control & Overview of your ProcessRobot. The Control Desk among many other things does the following:

- Automates distribution of [Processes](#)^[124] and [Robot workload](#)^[151] across the enterprise.
- Allows you to set [Triggers](#)^[154] and [Schedules](#)^[173] that can fire your Processes after the happening of specific events or at set or recurring time intervals.
- Grants you the capability to [allow or deny](#)^[193] specific "Process Robot rights" on different users/employees of your enterprise.
- Is the gateway to advanced features like [Audit](#)^[179] or [Robot Pools](#)^[217].

Please be aware of the fact that the first [user](#)^[195] that will open the Control Desk after the installation it will become automatically an [Administrator](#)^[193].

The Control Desk has also the [Control Desk Web Console](#)^[251] version.

This is a simplified version of the Control Desk that is accessible through the Internet. The Web Console includes amazing Cumulative Graph Flows that allows you to monitor and analyze your workflow over time.

2. The [Process Studio](#)^[276]

Process Studio is where the developer can build, debug and test the Processes which will be assigned later on to run on Robots ([Solobots](#)^[1086] or [Sidebots](#)^[1089]).

Through this Component users can do the following:

- Here is where the "construction" of individual processes takes place, using a uniquely powerful and easy-to-use interactive interface called [Process Designer](#)^[285].
- Process Studio is the gateway not only to the amazing interactive features that allow you to program without writing a single line of code, but to sophisticated features like [User Libraries](#)^[394] that allow you *to build your own user actions*^[394]!
- This is also where your Processes begin their [Lifecycle](#)^[455], created in development and migrating in Review and from there in Production via the Control Desk.
- Process Studio allows you to exchange processes and user actions with other Process Robot users worldwide (your partners for example), effortlessly via its [Import](#)^[283] and [Export](#)^[415] facilities.

[Insights Dashboard](#)^[1098] is a business intelligence analytics tool and key component of Softomotive Process Robot Enterprise RPA platform. Insights Dashboard allows for custom KPI

tracking and monitoring and therefore it is mainly interesting for Senior Managers, who would like to have a picture of the high-level performance of the RPA system.

It consists of two Components:

3. The Process Robot Insights Dashboards (Analytics Tools):

a) The [Insights Dashboards Designer](#)

This allows you to do among many other things the following:

- Customize [your own KPIs](#)
- Visualize KPIs according to the [Environment](#) and [Execution Target](#) they were executed on.
- Perform analytics and real-time monitoring of organizational goals with the help of advanced processes like a custom made ROI calculator.

and

b) Robot [Insights Dashboard Viewer](#)

This allows you to do among other things the following:

- Access the [print preview](#) of all of your Dashboards.
- Export Dashboards as [a PDF or Image file](#).

4. Sidebots

[Sidebots](#) are the Robots that work along with the user for Attended Automation.

5. Solobots:

[Solobots](#) are self-reliant, autonomous independent full time employees (FTE) that can auto-login and work Unattended.

1.4 Hardware requirements

1.4.1 Trial Installation Requirements

Below you can find the minimum hardware and software requirements for a ProcessRobot trial installation. Please note that since a trial installation is usually needed for demo purposes or in a proof of concept phase, you can easily forego or trade some of the live installation requirements in order to have the environment work in readily available hardware.

Minimum Installation Requirements

A minimum installation can be setup and work in one physical machine, with at least 8GB of RAM (however more is recommended), which will have separate Virtual Machines (VMs) installed, as follows:

1. An active directory server with MS SQL Express 2016, SQL Server Management Studio, ProcessRobot, Control Desk, Process Studio, Dashboard Designer, Dashboard Viewer, and SideBot: 5GB of Ram and 40GB of Hard Disk Space (including windows installation)
2. A VM with Windows 7,8 or 10 in which SoloBot will run: 3GB of Ram and 30GB of hard disk space (including windows installation)

Recommended Installation Requirements

As described above, ProcessRobot can work in one physical machine if absolutely required, however this environment will provide only minimum functionality. The recommended trial installation hardware configuration, is the following:

1. An Active Directory Server (2008, 2012 R2 or 2016): 6 GB Ram, 35 H.D Space
2. A VM with SQL Server Express 2016, SQL Server Management Studio, ProcessRobot Server, Control Desk, Process Studio, Dashboard Viewer and Dashboard Designer: 6GB Ram, 30GB H.D Space
3. A VM with windows 7,8 or 10 with SideBot: 4 GB Ram, 30GB H.D space
4. A VM with Windows 7, 8 or 10 with SoloBot: 4 GB Ram, 30GB H.D space

1.4.2 Live Installation Requirements

In a production installation, hardware requirements are higher than in [Trial Installation](#)²², as ProcessRobot will have to deal with a larger amount of robots and processes, handling real data.

Module: ProcessRobot Server

Hardware:

- RAM: 16GB
- HDD 5GB + Database Size

Software:

- Microsoft Windows Server 2008 or later
- Microsoft .Net Framework v4.0 or later
- Microsoft SQL Server 2012 or later (any edition)
- Active Directory

Modules: Solobot and Sidebot

Hardware:

- RAM: 4GB
- HDD 2GB

Software:

- Microsoft Windows 7 or later (32 or 64 bit edition) or Microsoft Windows Server 2008 or later
- Microsoft .Net Framework v4.0 or later
- Internet Explorer 9 or later
- Microsoft Excel 2003 or later (for Excel functionality)

Note: The minimum hardware requirements for the Robot modules depend on the processes that they'll execute. As a general rule, if a HW/SW configuration is enough for a human employee to perform the same process(es) with ease, ProcessRobot will be able to run the automated version of this process(es) with no problems.

Module: Control Desk

Hardware:

- RAM: 8GB
- HDD 4GB

Software:

- Microsoft Windows 7 or later (32 or 64 bit edition) or Microsoft Windows Server 2008 or later
- Microsoft .Net Framework v4.0 or later
- Internet Explorer 9 or later

Module: Process Studio

Hardware:

- RAM: 8GB
- HDD 2GB

Software:

- Microsoft Windows 7 or later (32 or 64 bit edition) or Microsoft Windows Server 2008 or later
- Microsoft .Net Framework v4.0 or later
- Internet Explorer 9 or later

- Microsoft Excel 2003 or later (for Excel functionality)

Module: Insights Dashboard

Hardware:

- RAM: 8GB
- HDD 2GB

Software:

- Microsoft Windows 7 or later (32 or 64 bit edition) or Microsoft Windows Server 2008 or later
- Microsoft .Net Framework v4.0 or later

1.5 Installation Guide

1.5.1 Installing ProcessRobot

As of version 2019.1, ProcessRobot installation is now carried out using a single installer, whether the target machine will be used for the ProcessRobot Server, Client Tool(s) (Control Desk, Process Studio, Dashboards) or a Robot (SideBot, SoloBot).

This guide describes the steps required for installing the Server, all Client Tools, and a SideBot on the target machine.

To install a single component on a Client machine, for example Process Studio, the required steps will be the same as those described in this guide, but will not include those specific to installing the Server.

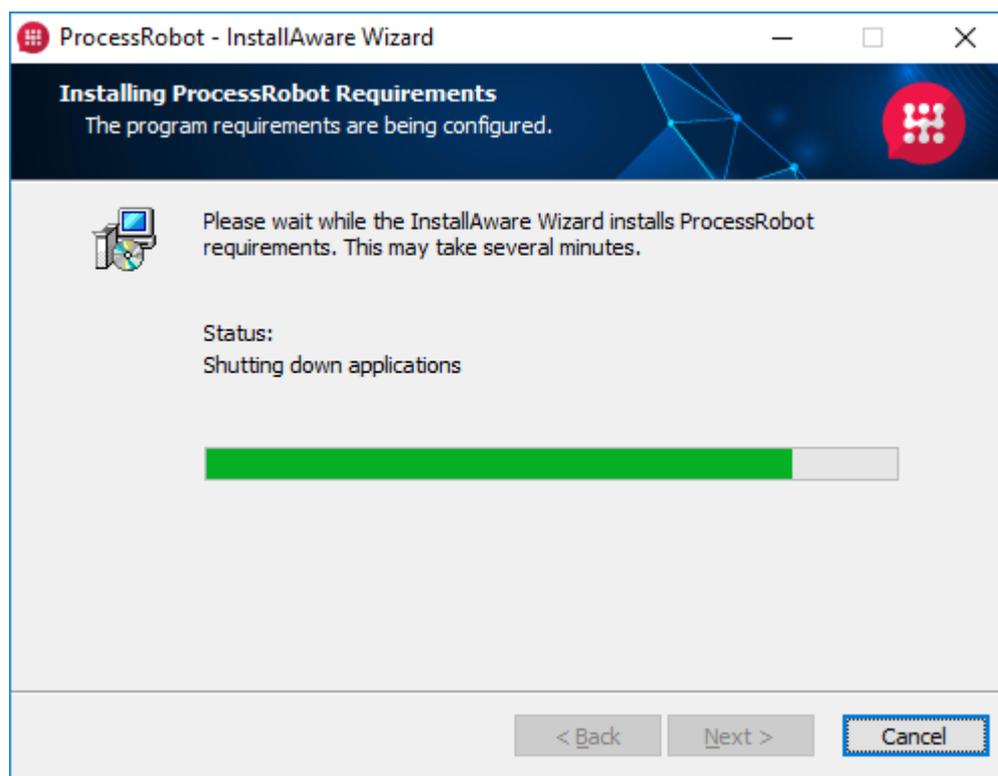
Server installation

The ProcessRobot Server makes use of at least one SQL Server database as the main Database. Optionally, a second database can be used for Logs. If SQL Server has not been set up on the Server machine, the new ProcessRobot installer can be used to install SQL Server Express, for Proof of Concept demonstrations, quick trials, etc. For live production installations, the standard version of SQL Server is required (see [Live Installation Requirements](#)^[22]).

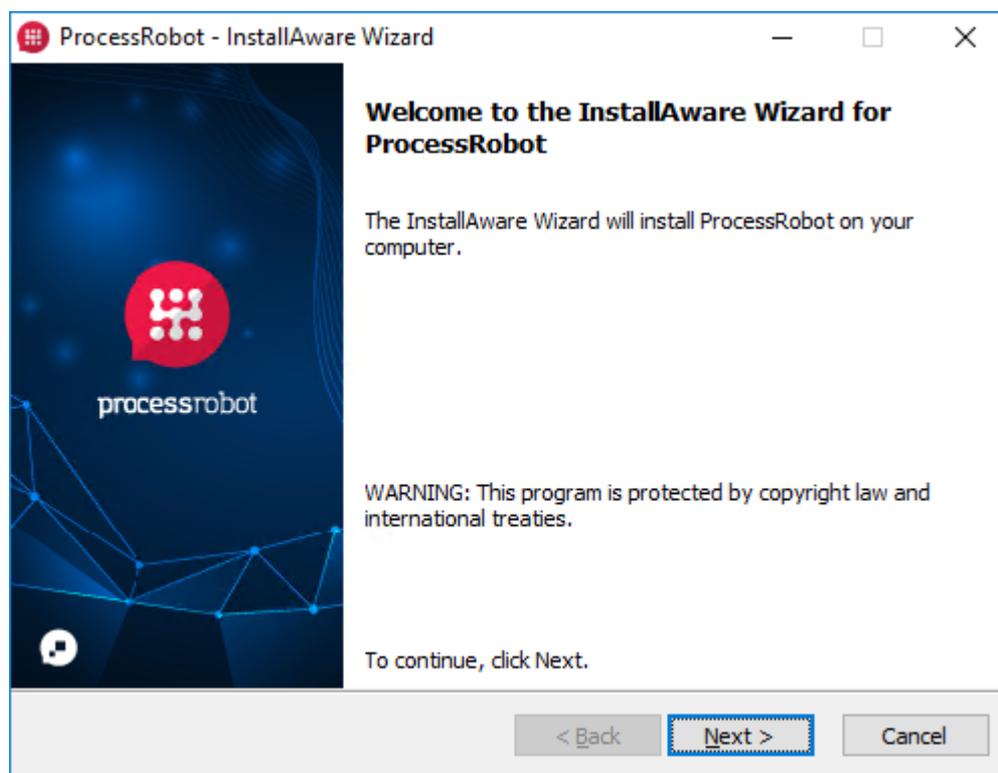
This guide describes the installation steps to be followed when SQL Server and the database to use for ProcessRobot are already in place. For the additional steps to follow when

Launch the ProcessRobot installer .exe or .msi file.

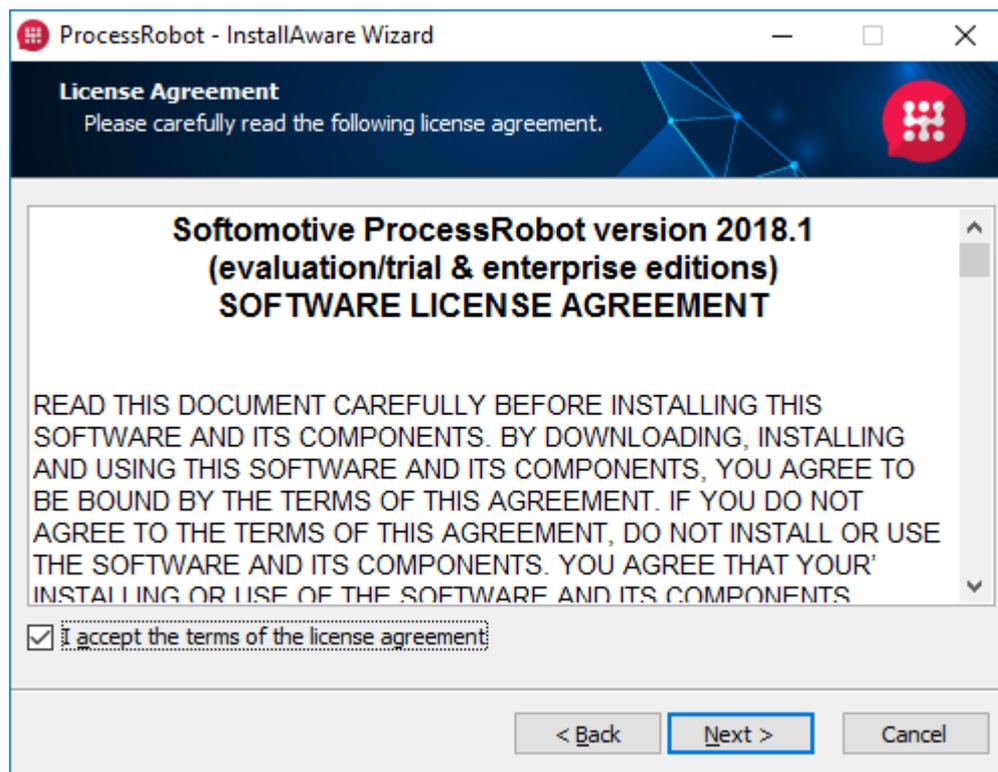
The installation wizard will begin by setting up some requirements for ProcessRobot.



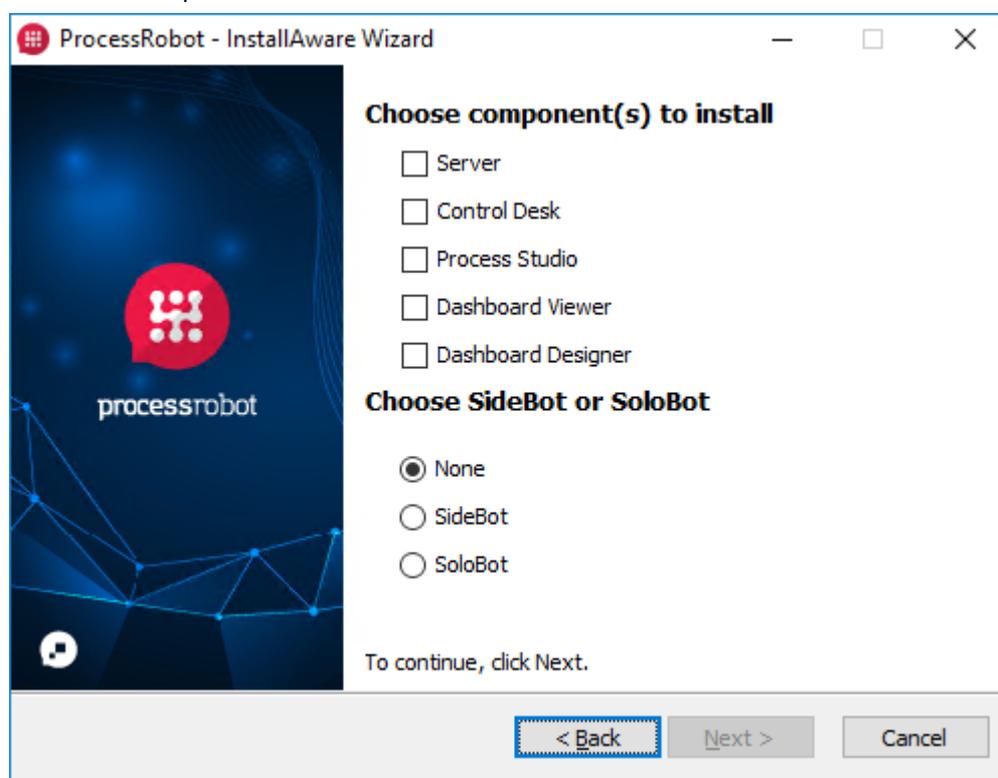
Click 'Next' to proceed with the installation.



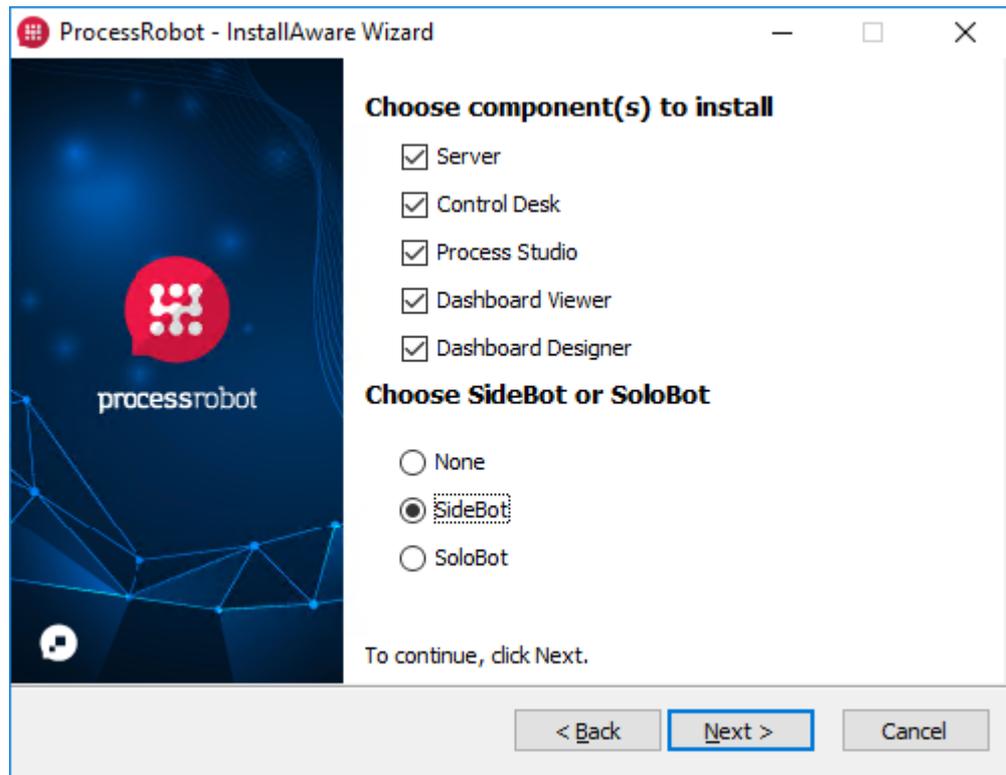
Accepting the terms of the license agreement is necessary to proceed.



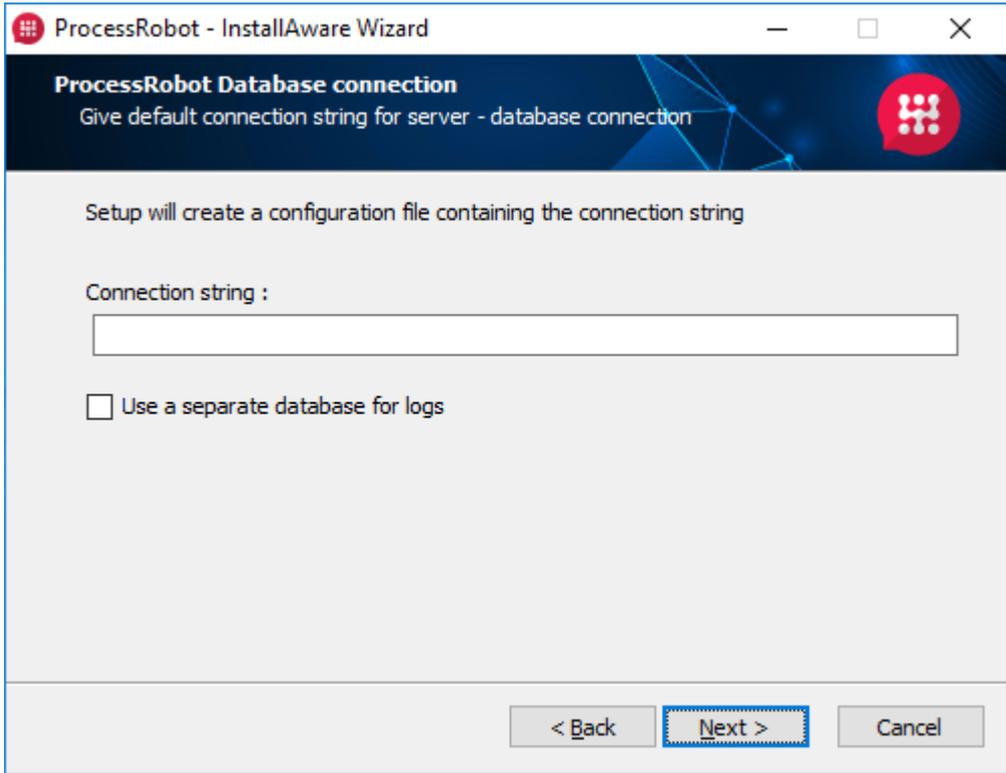
Choose which components to install:



For example, in this Server installation, also include all Client Tools plus a SideBot:



Enter the Connection String for the existing SQL Server database to be used for ProcessRobot. Optionally select if a second database should be used for Logs. If this option is not selected, Logs will be stored in the same database as the other ProcessRobot data.



ProcessRobot - InstallAware Wizard

ProcessRobot Database connection
Give default connection string for server - database connection

Setup will create a configuration file containing the connection string

Connection string :

Use a separate database for logs

< Back Next > Cancel

Enter the Connection String in the format:

Password= <password>; Persist Security Info=True; User ID=<dbUserName>; Initial Catalog= <dbName>; Data Source=<serverName\SQLEXPRESS>;

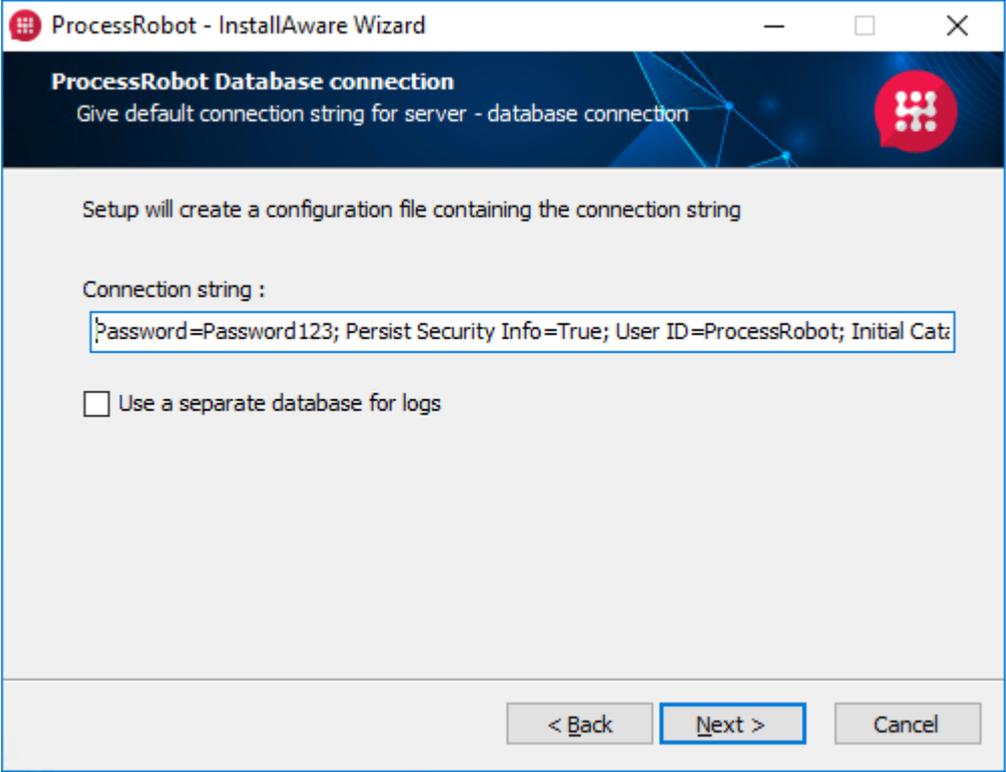
Where:

<password> is the database user's password

<dbUserName> is the database user name

<dbName> is the name of the database to be used for ProcessRobot

<serverName\SQLEXPRESS>: serverName is the name of the Server machine, and SQLEXPRESS is the correct instance of SQL Server. By default, this should be SQLEXPRESS, but if multiple instances of SQL Server exist on the machine, the instance may have a different name, eg SQLEXPRESS01.



ProcessRobot - InstallAware Wizard

ProcessRobot Database connection
Give default connection string for server - database connection

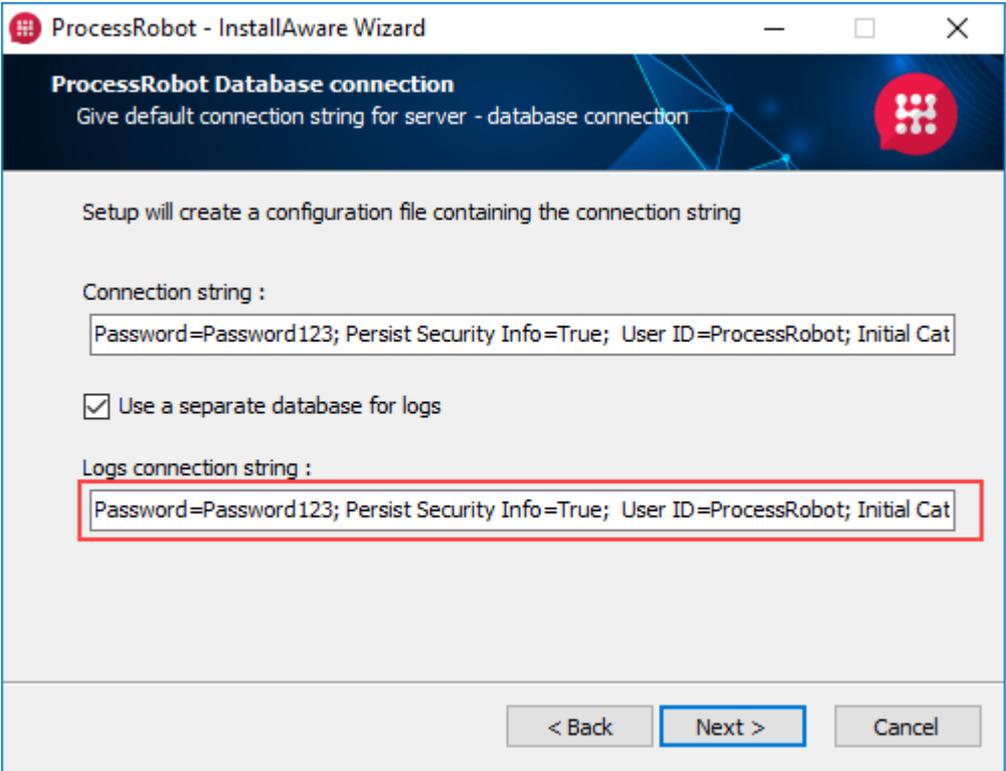
Setup will create a configuration file containing the connection string

Connection string :
`Password=Password123; Persist Security Info=True; User ID=ProcessRobot; Initial Cat`

Use a separate database for logs

< Back Next > Cancel

When selecting to use a separate database for Logs, a second connection must be entered.



ProcessRobot - InstallAware Wizard

ProcessRobot Database connection
Give default connection string for server - database connection

Setup will create a configuration file containing the connection string

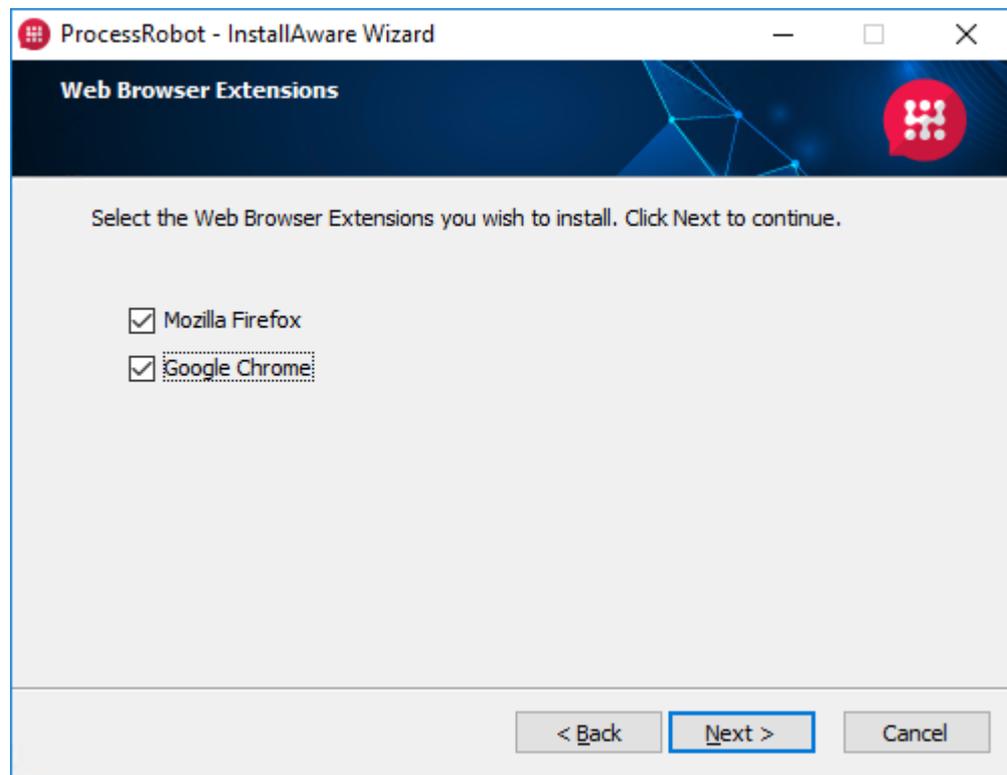
Connection string :
`Password=Password123; Persist Security Info=True; User ID=ProcessRobot; Initial Cat`

Use a separate database for logs

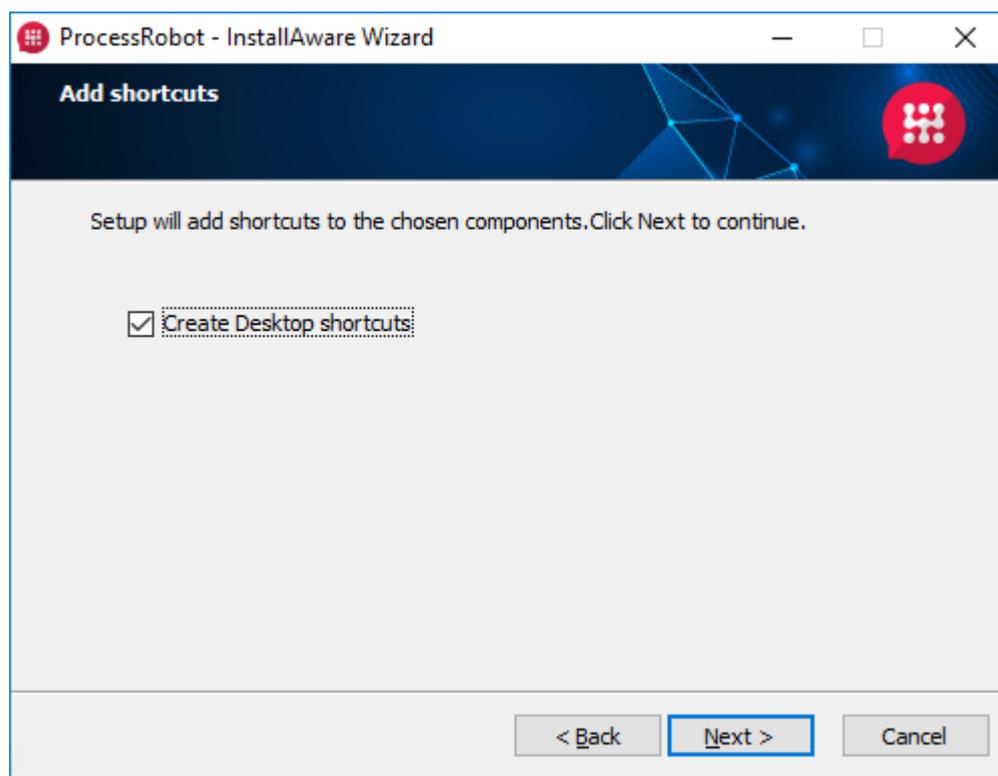
Logs connection string :
`Password=Password123; Persist Security Info=True; User ID=ProcessRobot; Initial Cat`

< Back Next > Cancel

Select whether to install the Softmotive Browser Extensions for Chrome and Firefox. These are necessary for allowing ProcessRobot to perform web automation tasks on those browsers. Both Extensions are recommended.



Choose whether to create Desktop shortcuts for the Client Tools / Robot.



Enter the IP address, host name or Fully Qualified Domain Name (FQDN) of the Server machine and the port number.

The number entered should correspond to a port that is not currently in use, and which complies with ITSEC and IANA (<https://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xhtml>).

A quick way to see which ports are currently in use is to run the following command as an Administrator in the Command Prompt:

```
netstat -a -b
```

As an example, this guide will use port number 6090.

ProcessRobot - InstallAware Wizard

ProcessRobot Server Address
Give default server address

Setup will create a configuration file containing the server's IP address and port

Address

Port

< Back Next > Cancel

In cases where the ProcessRobot Server and Client machines belong to the same Active Directory domain, the host name should suffice.

ProcessRobot - InstallAware Wizard

ProcessRobot Server Address
Give default server address

Setup will create a configuration file containing the server's IP address and port

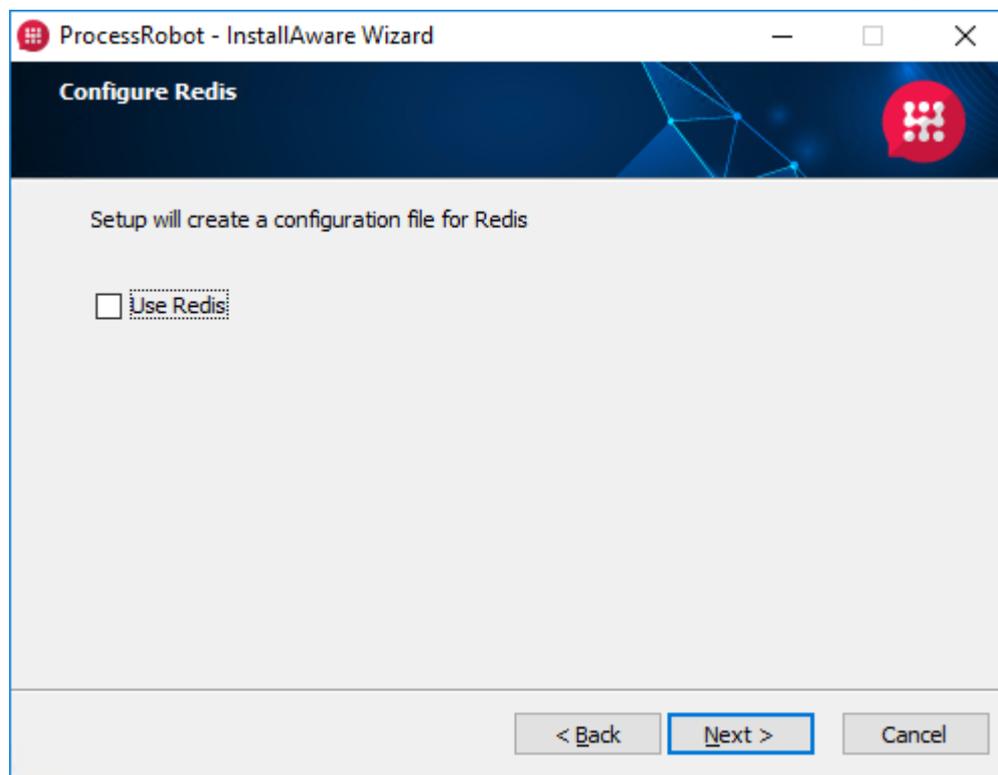
Address

Port

< Back Next > Cancel

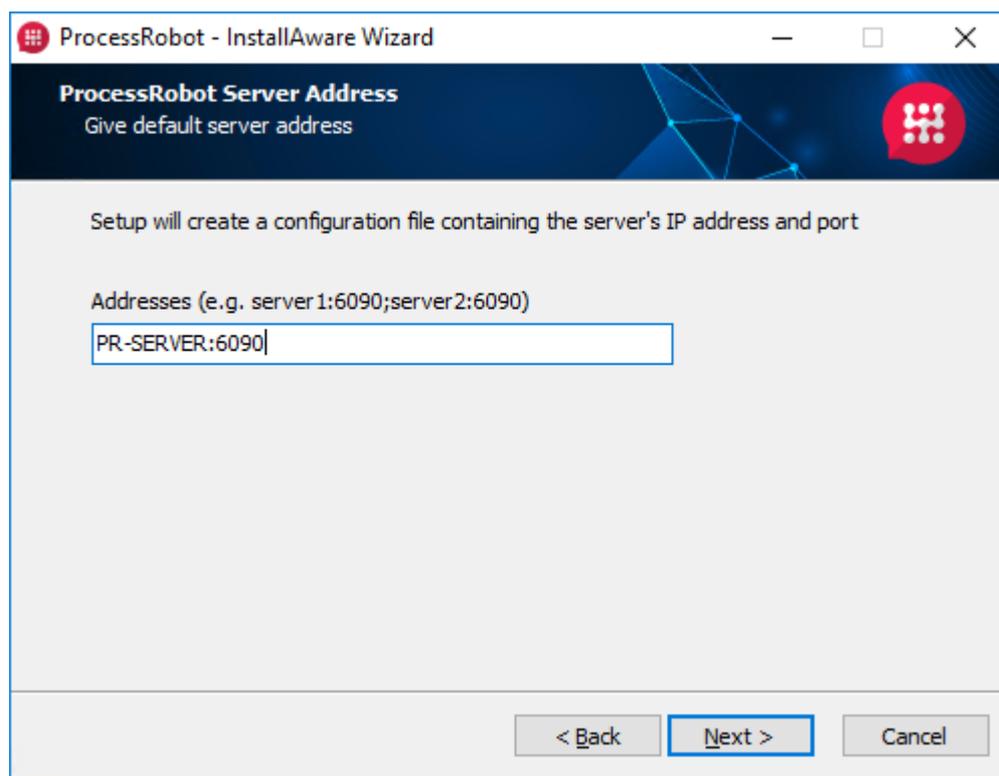
Note that if there are issues with the DNS, the other ProcessRobot components will not be able to communicate with the Server using only the host name or FQDN. DNS issues can be identified by pinging the host name or FQDN. In such cases, either the Server's IP address must be used instead, or the DNS issues must be resolved.

Select whether to use a Redis deployment. This guide will proceed without enabling this option.

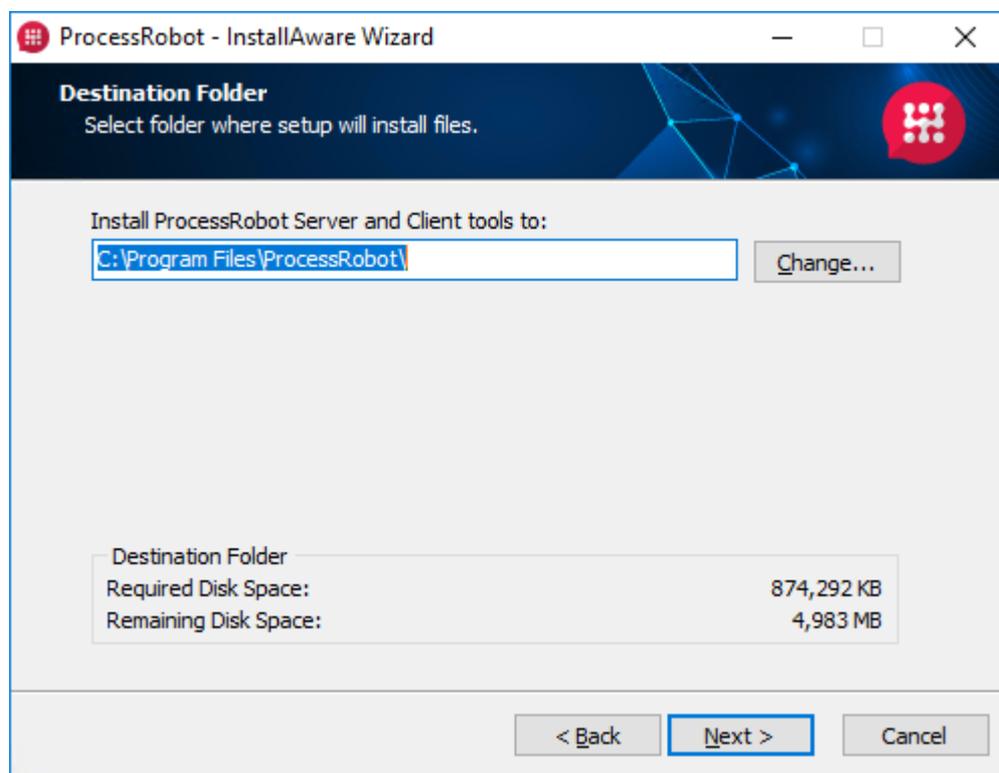


When Client Tools or a Robot are installed, it is necessary to also enter the Server address and port number previously entered in the format `<server>:<port>`. The `<server>` value can be entered in any of the three ways described previously, for example:

- 10.1.182.106:6090 (IP address)
- PR-SERVER:6090 (host name)
- PR-SERVER.processrobot.com:6090 (FQDN)

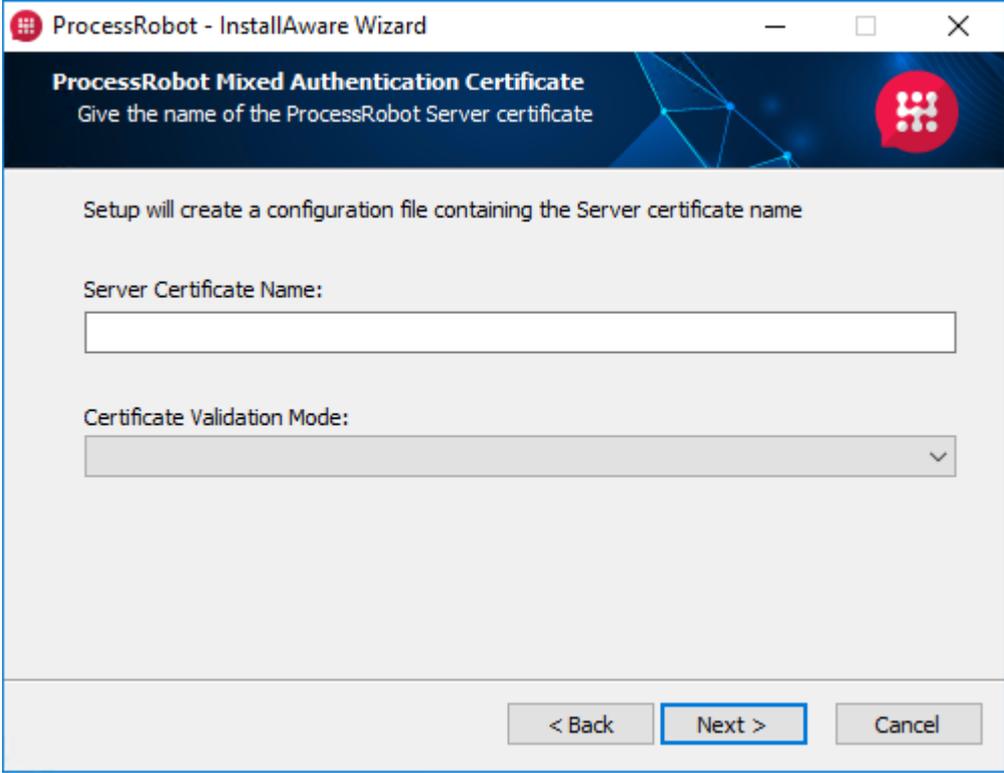


Select the destination file path for the ProcessRobot installation. The default location is: C:\Program Files\ProcessRobot.



Optionally enter a Certificate Name to use for ProcessRobot Custom Authentication. The Certificate must already be set up on the Server machine. For more details about how to set up Certificates, read [this article](#)^[43].

To enable Custom Authentication at another time, refer to the article [Configuring Custom Authentication after Installation](#)^[51].



ProcessRobot - InstallAware Wizard

ProcessRobot Mixed Authentication Certificate
Give the name of the ProcessRobot Server certificate

Setup will create a configuration file containing the Server certificate name

Server Certificate Name:

Certificate Validation Mode:

< Back Next > Cancel

Once the Certificate Name has been entered, there are several Certificate Validation Modes to choose from. For Self-Signed Certificates, choose "None".

For Production grade deployments, the Mode will likely be different. In order to learn more about the Certificate validation modes and decide which one fits your environment, please visit: <https://docs.microsoft.com/en-us/dotnet/api/system.servicemodel.security.x509certificatevalidationmode?view=netframework-4.8>

ProcessRobot - InstallAware Wizard

ProcessRobot Mixed Authentication Certificate
Give the name of the ProcessRobot Server certificate

Setup will create a configuration file containing the Server certificate name

Server Certificate Name:
AKADDC.processrobot.demo

Certificate Validation Mode:
None
PeerTrust
Chain Trust
Peer Or Chain Trust

< Back Next > Cancel

ProcessRobot - InstallAware Wizard

ProcessRobot Mixed Authentication Certificate
Give the name of the ProcessRobot Server certificate

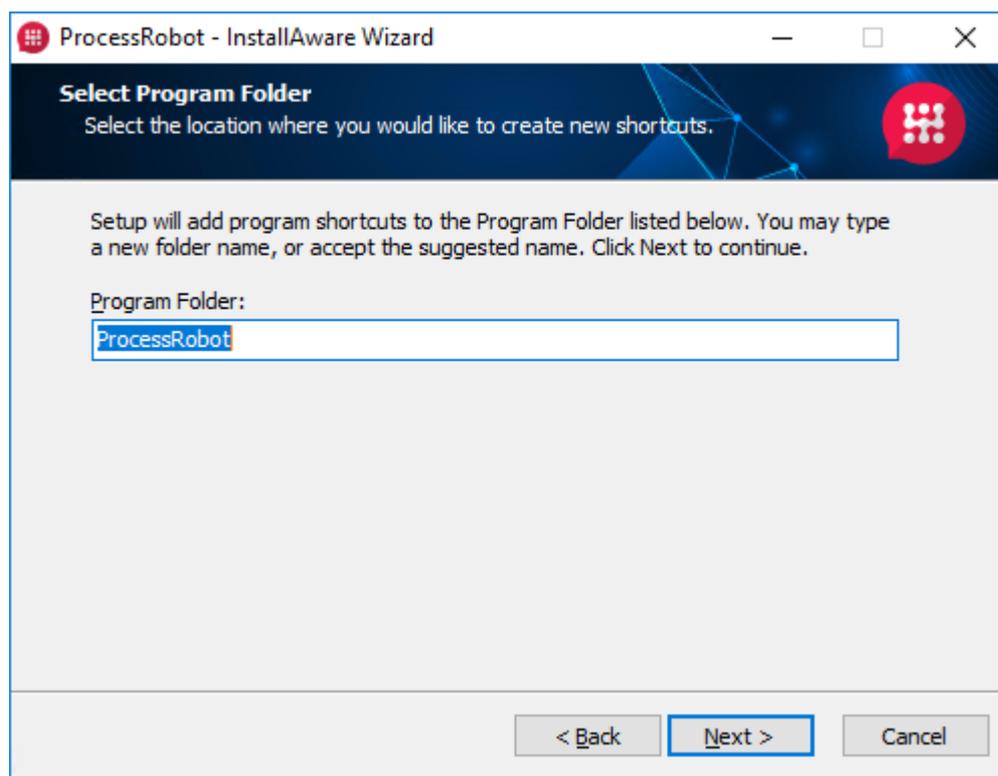
Setup will create a configuration file containing the Server certificate name

Server Certificate Name:
AKADDC.processrobot.demo

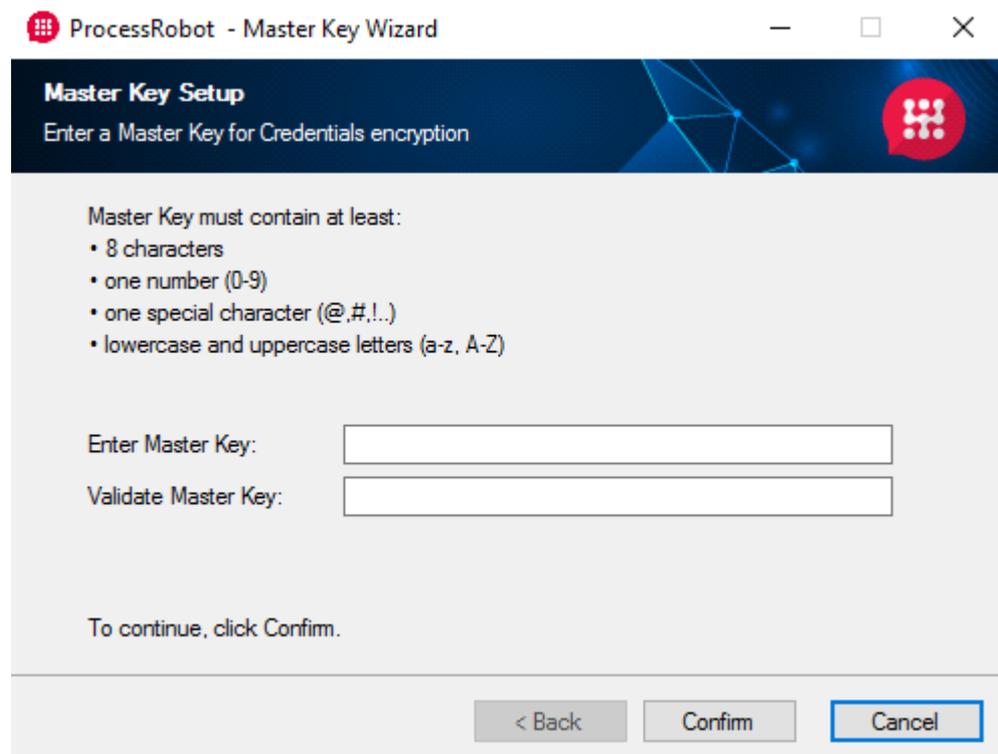
Certificate Validation Mode:
None
PeerTrust
Chain Trust
Peer Or Chain Trust

< Back Next > Cancel

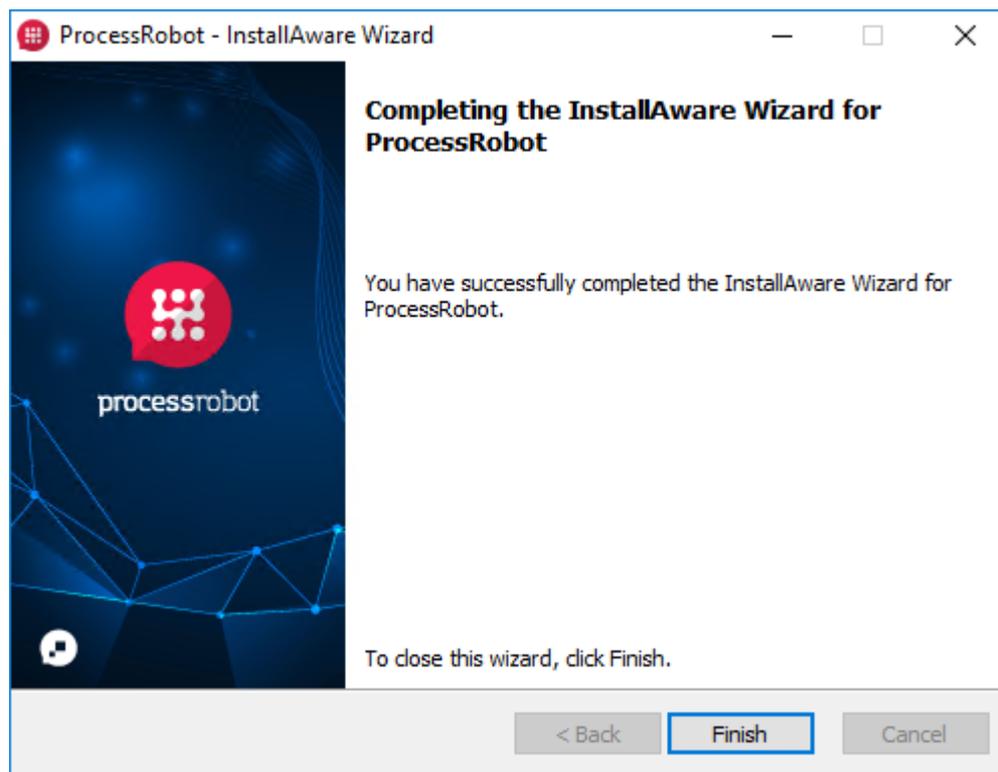
Set the Program Folder name.



Set a Master Key, which is used to encrypt all Credentials stored in ProcessRobot.



Click "Finish" to exit the ProcessRobot installer.



Once the installation is finished, in case the "ProcessRobot Server" service does not start, navigate to the installation folder (usually to C:\Program Files\ProcessRobot\Server) and execute the file ProcessRobotDatabaseUpgrader.exe. This will setup the ProcessRobot database automatically for so that the service will start.

To verify that everything went well, check if the ProcessRobot service is up and running. To do so, open the Windows Task Manager. Under the "Services" tab, find the ProcessRobot Server service.

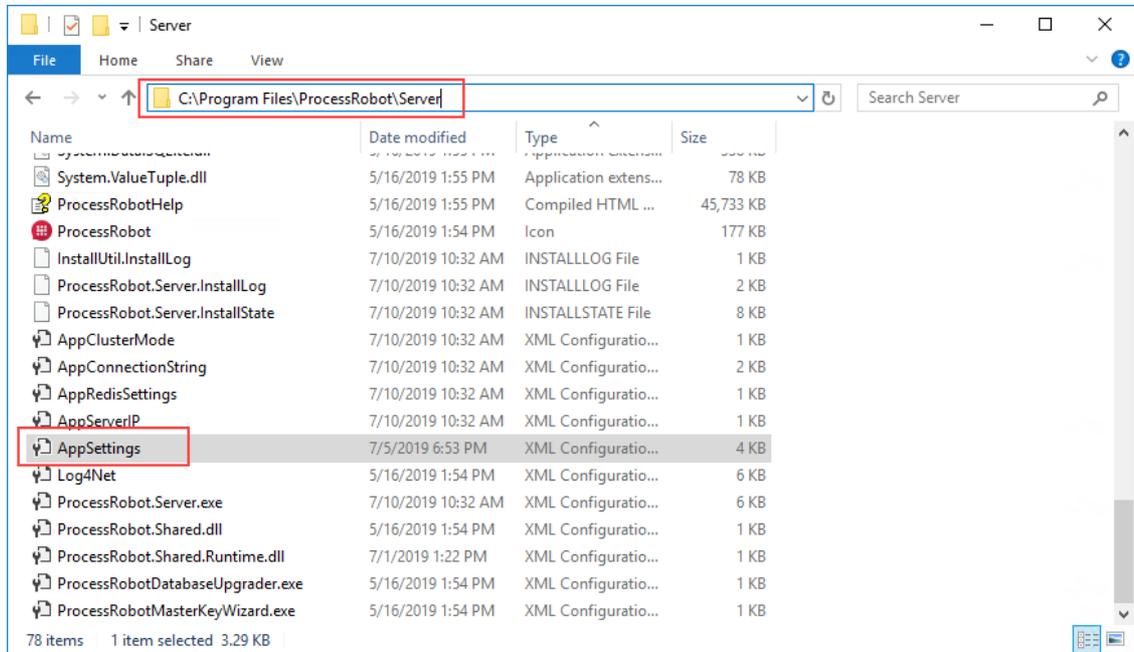
Power	Manages p...	Running	Automatic	Local Syste...
Print Spooler	This service ...	Running	Automatic	Local Syste...
Printer Extensions and Notif...	This service ...		Manual	Local Syste...
Problem Reports and Soluti...	This service ...		Manual	Local Syste...
ProcessRobot Server	Controls th...	Running	Automatic	Local Syste...
Program Compatibility Assi...	This service ...	Running	Automatic	Local Syste...
Quality Windows Audio Vid...	Quality Win...		Manual	Local Service
Radio Management Service	Radio Mana...		Manual	Local Service

Allow the ProcessRobot Server to communicate through the firewall by either allowing the Port # or the ProcessRobot.Server.exe.

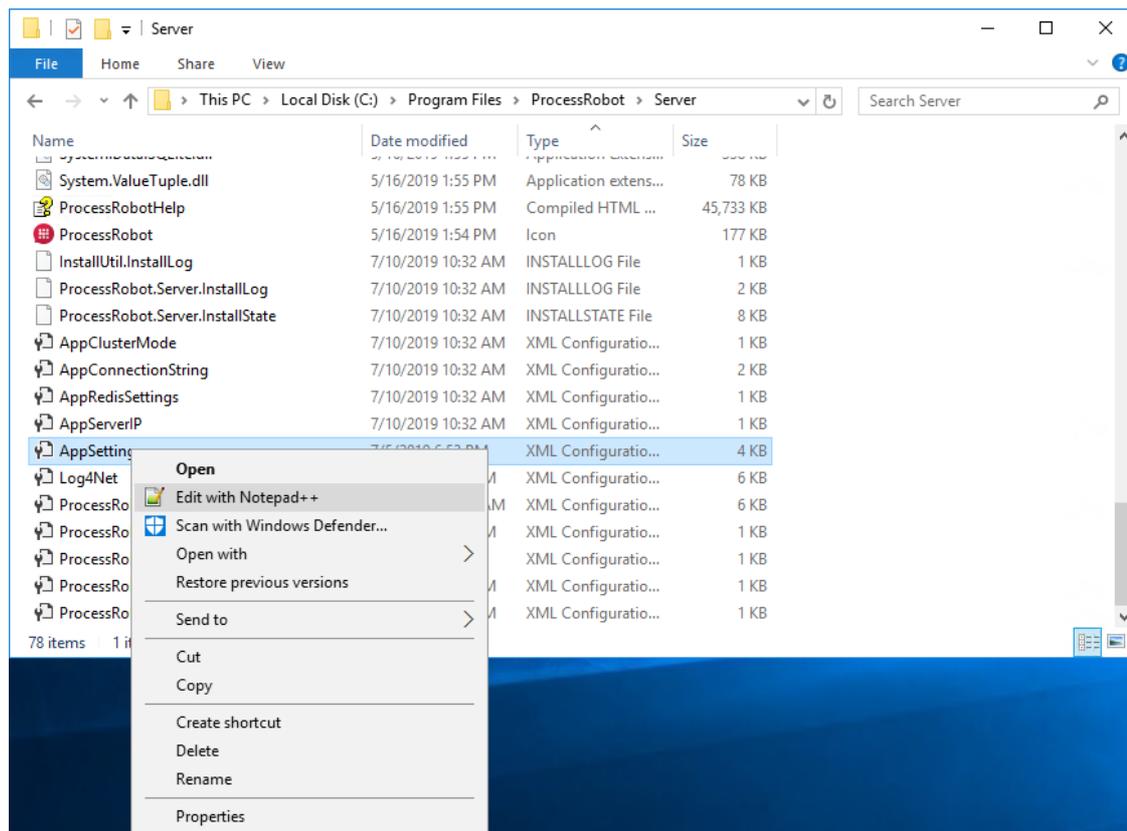
Custom Authentication

To enable custom authentication using the Certificate, some additional actions are required beyond the installer steps.

Navigate to the file path where the Server components have been installed, by default C:\Program Files\ProcessRobot\Server. Locate the "AppSettings.config" file.



Edit the file with a text editor like Microsoft Notepad or Notepad++.



Locate in the file the line with following:

```
<add key="SupportMixedAuthentication" value ="false"/>
```

```
57
58 <add key="SupportMixedAuthentication" value ="false"/>
59
60 <!-- Valid Options: CurrentUser, LocalMachine -->
61 <add key="CertificateStoreLocation" value ="LocalMachine"/>
62
63 <!-- Valid Options: AddressBook, AuthRoot, CertificateAutho
64 <add key="CertificateStoreName" value ="My"/>
65
66 <!-- Valid Options: FindByThumbprint, FindBySubjectName, Fi
67 FindByTimeNotYetValid, FindByTimeExpired, FindByTemplateName
68 <add key="CertificateFindBy" value ="FindBySubjectName"/>
69
70 <add key="CertificateFindValue" value ="ProcessRobot"/>
71
72 <add key="LogoutAfterProcessesCompletedLag" value="10000"/>
73
74 <add key="LogsPageSize" value ="1000"/>
75
76 </applicationConfiguration>
77
```

Normal text file

Change the value of "false" to "true".

```
57
58 <add key="SupportMixedAuthentication" value ="true"/>
59
60 <!-- Valid Options: CurrentUser, LocalMachine -->
61 <add key="CertificateStoreLocation" value ="LocalMachine"/>
62
63 <!-- Valid Options: AddressBook, AuthRoot, CertificateAuthori
64 <add key="CertificateStoreName" value ="My"/>
65
66 <!-- Valid Options: FindByThumbprint, FindBySubjectName, Find
67 FindByTimeNotYetValid, FindByTimeExpired, FindByTemplateName,
68 <add key="CertificateFindBy" value ="FindBySubjectName"/>
69
70 <add key="CertificateFindValue" value ="ProcessRobot"/>
71
72 <add key="LogoutAfterProcessesCompletedLag" value="10000"/>
73
74 <add key="LogsPageSize" value ="1000"/>
75
76 </applicationConfiguration>
77
```

Normal text file

Locate the line with the following:

```
<add key="CertificateFindValue" value ="ProcessRobot"/>
```

```
65
66     <!-- Valid Options: FindByThumbprint, FindBySubjectName, FindBySubjectDistinguishec
67     FindByTimeNotYetValid, FindByTimeExpired, FindByTemplateName, FindByApplicationPoli
68     <add key="CertificateFindBy" value ="FindBySerialNumber"/>
69
70     <add key="CertificateFindValue" value ="ProcessRobot"/>
71
72     <add key="LogoutAfterProcessesCompletedLag" value="10000"/>
73
74     <add key="LogsPageSize" value ="1000"/>
75
76 </applicationConfiguration>
77
```

Normal text file

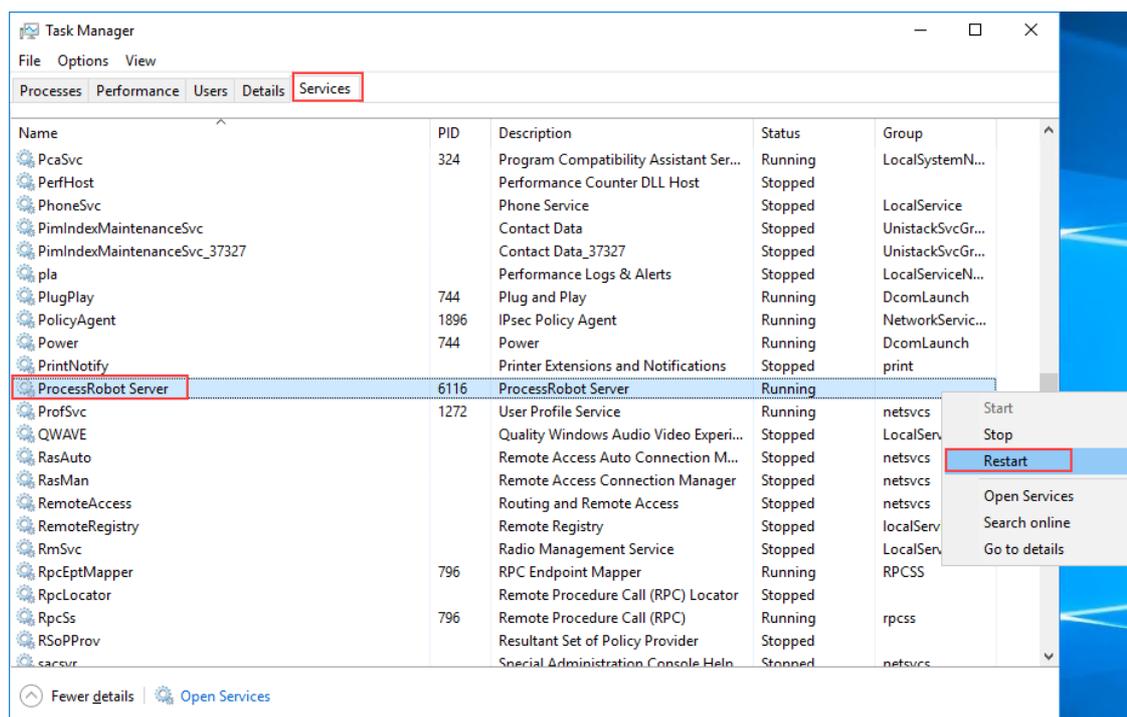
Change the value of "ProcessRobot" to the Certificate Name, as entered in the equivalent field in the ProcessRobot installer.

```
66     <!-- Valid Options: FindByThumbprint, FindBySubjectName, FindBySubject
67     FindByTimeNotYetValid, FindByTimeExpired, FindByTemplateName, FindByAp
68     <add key="CertificateFindBy" value ="FindBySubjectName"/>
69
70     <add key="CertificateFindValue" value ="AKADCDC.processrobot.demo"/>
71
72     <add key="LogoutAfterProcessesCompletedLag" value="10000"/>
73
74     <add key="LogsPageSize" value ="1000"/>
75
76 </applicationConfiguration>
77
```

Normal text file length

Save and close the file.

Restart the ProcessRobot Server service through the Windows Task Manager for these changes to take effect.



This completes the configuration of Custom Authentication.

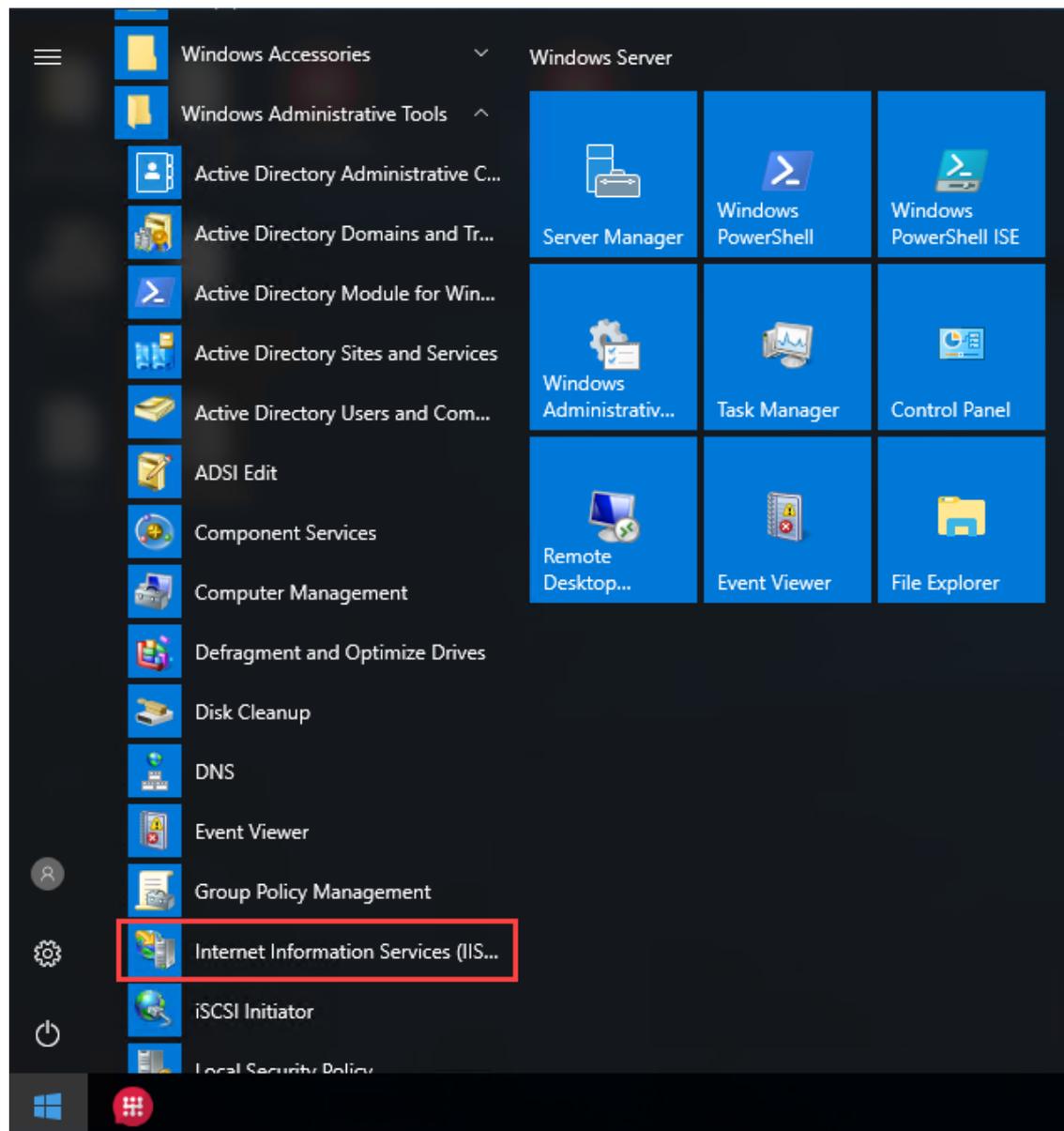
1.5.2 Creating a Self-Signed Certificate

Creating a Self-Signed Certificate through Internet Information Services (IIS)

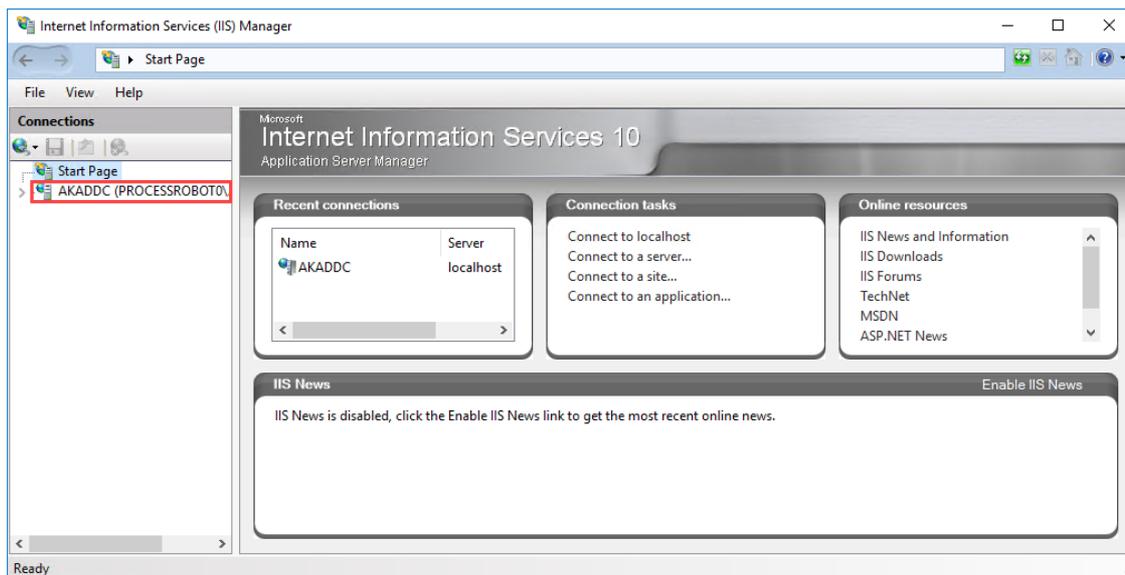
To create a Self-Signed Certificate for use with ProcessRobot using Windows, Internet Information Services (IIS) must be installed on the Server machine. If IIS is not installed on the machine, alternative solutions like using OpenSSL are available.

This guide describes the steps required to create the Certificate using IIS.

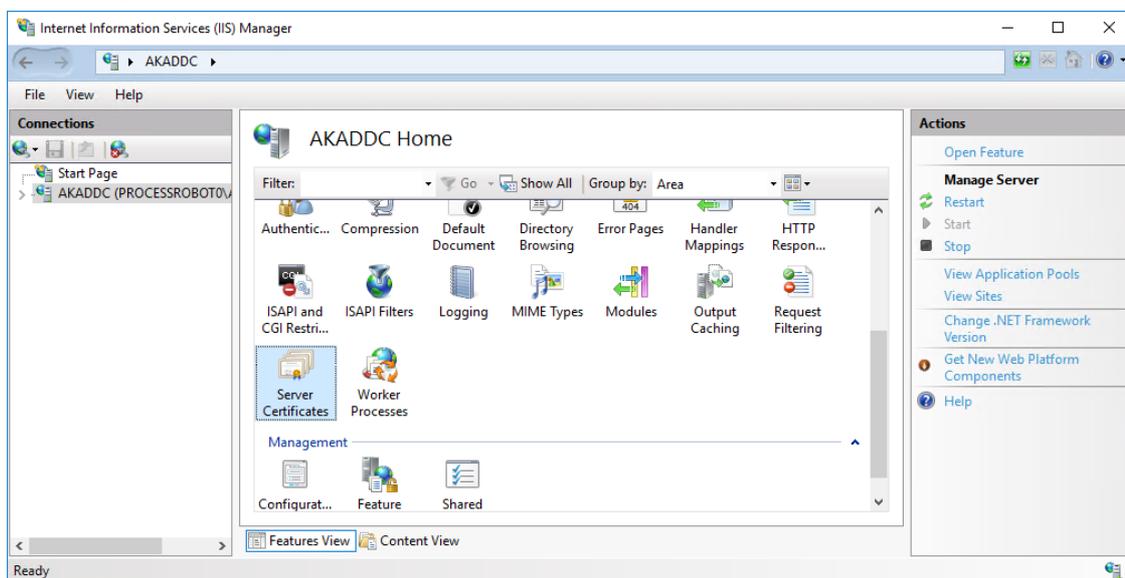
Launch IIS.



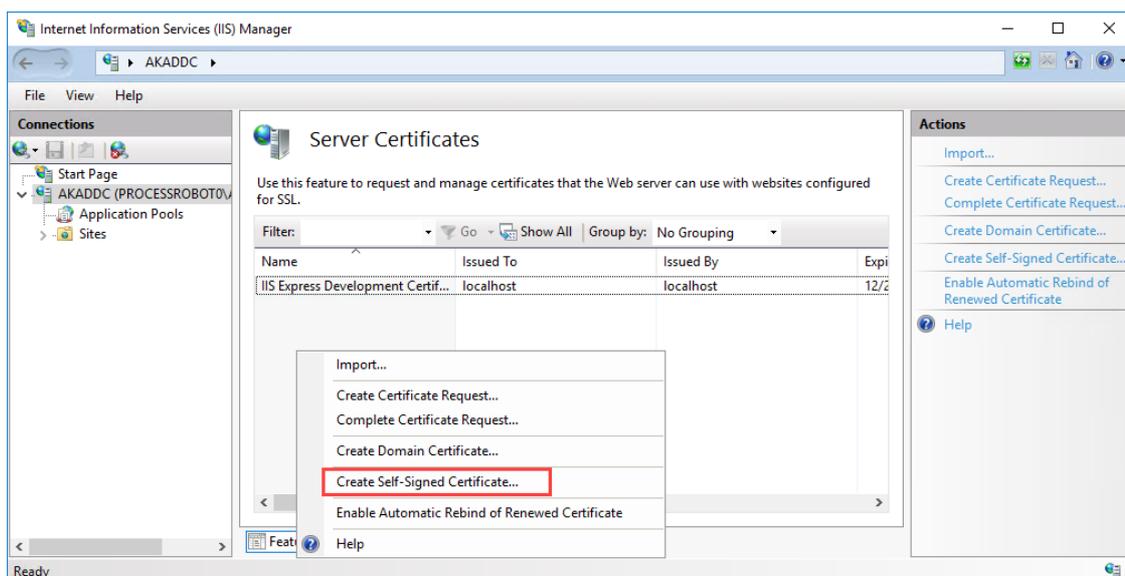
In IIS, select the name of the localhost



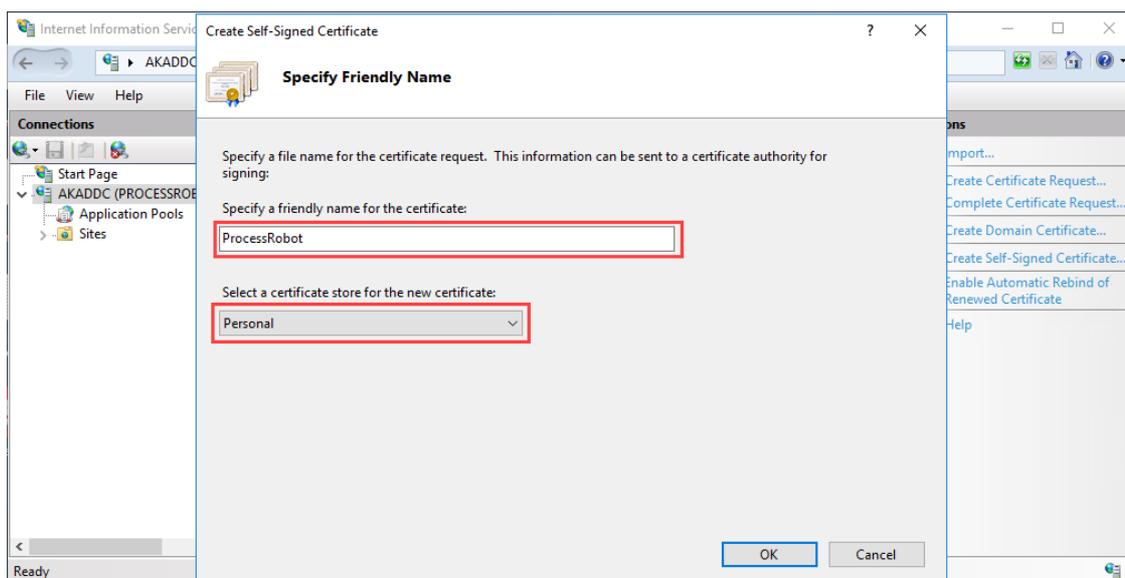
Then navigate to and open "Server Certificates"



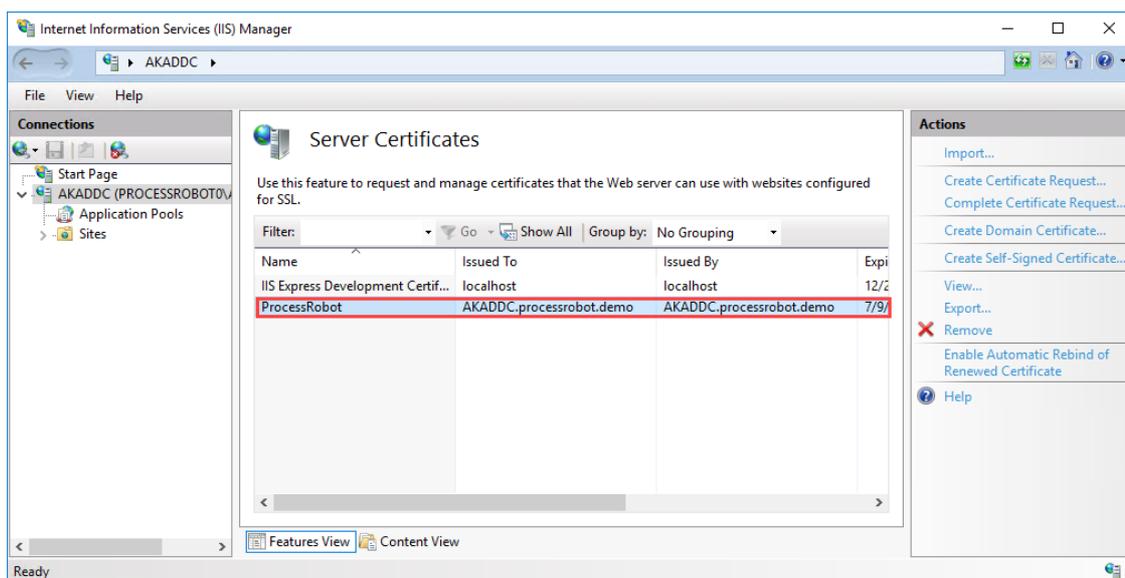
In "Server Certificates", right click in a blank area, and select "Create Self-Signed Certificate..."



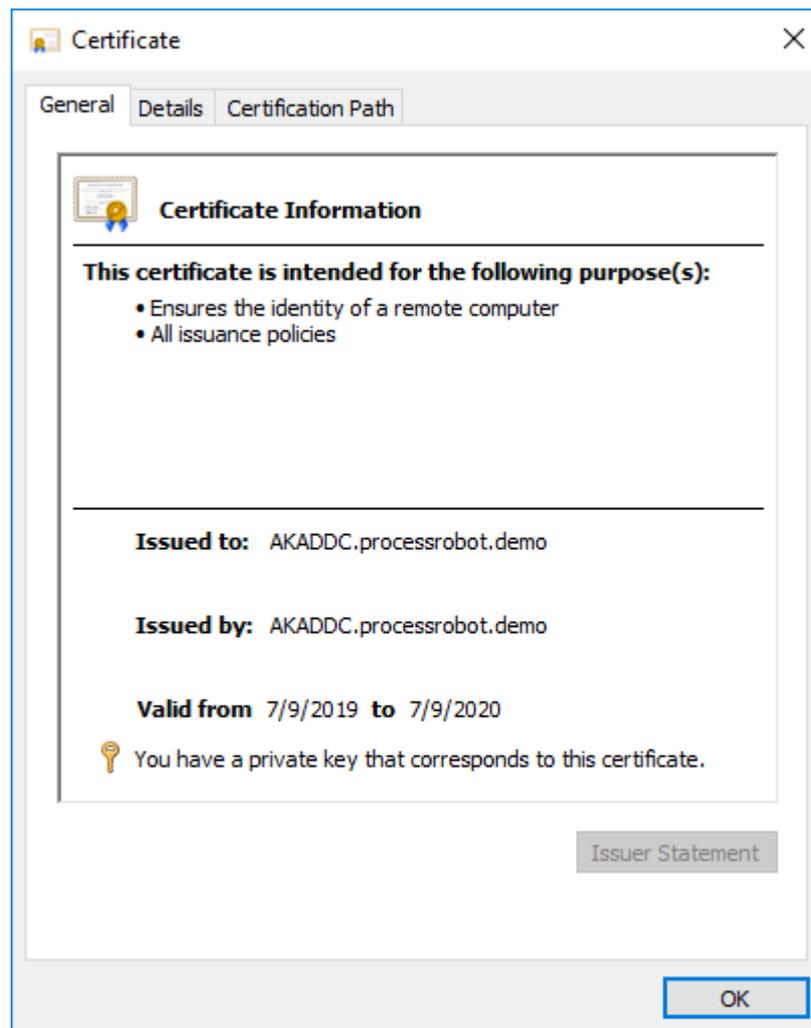
Specify a friendly name for the Certificate, eg "ProcessRobot". Important: it is strongly recommended that only one certificate exists on the Server machine, to avoid conflicts when ProcessRobot attempts to access the Certificate. Set the Certificate Store to "Personal". Click "OK" to save the Certificate and close the window.



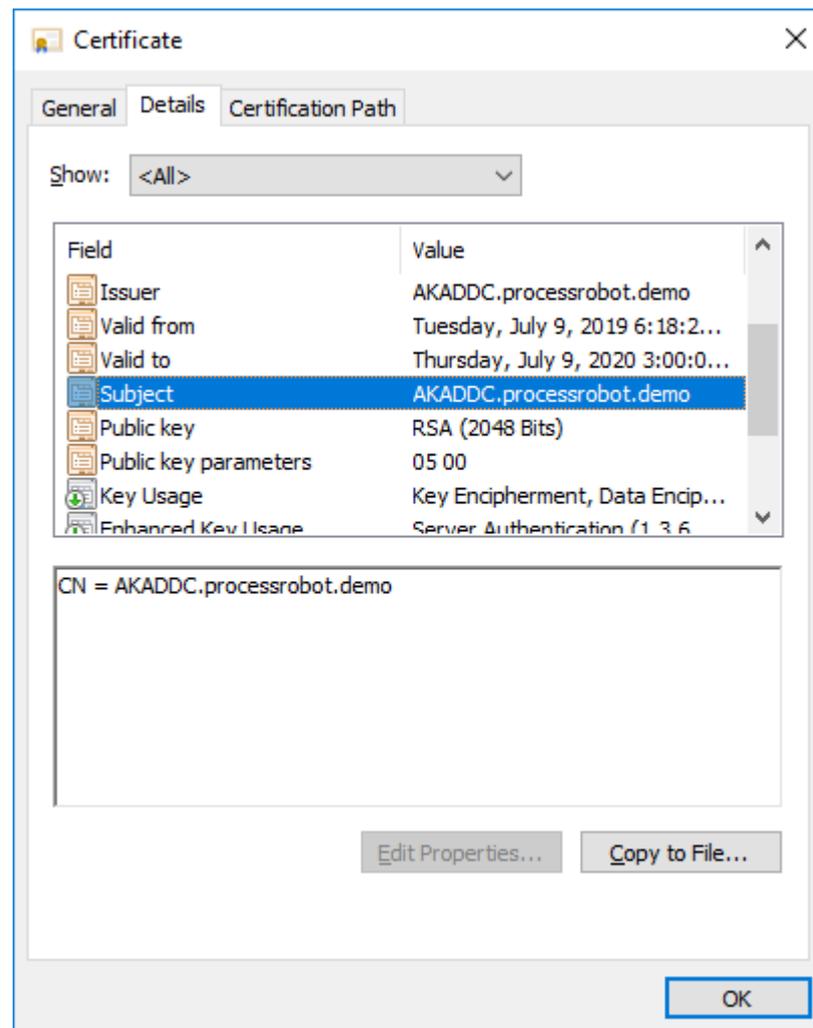
Find the newly created Certificate in "Server Certificates" and double click it to open its settings.



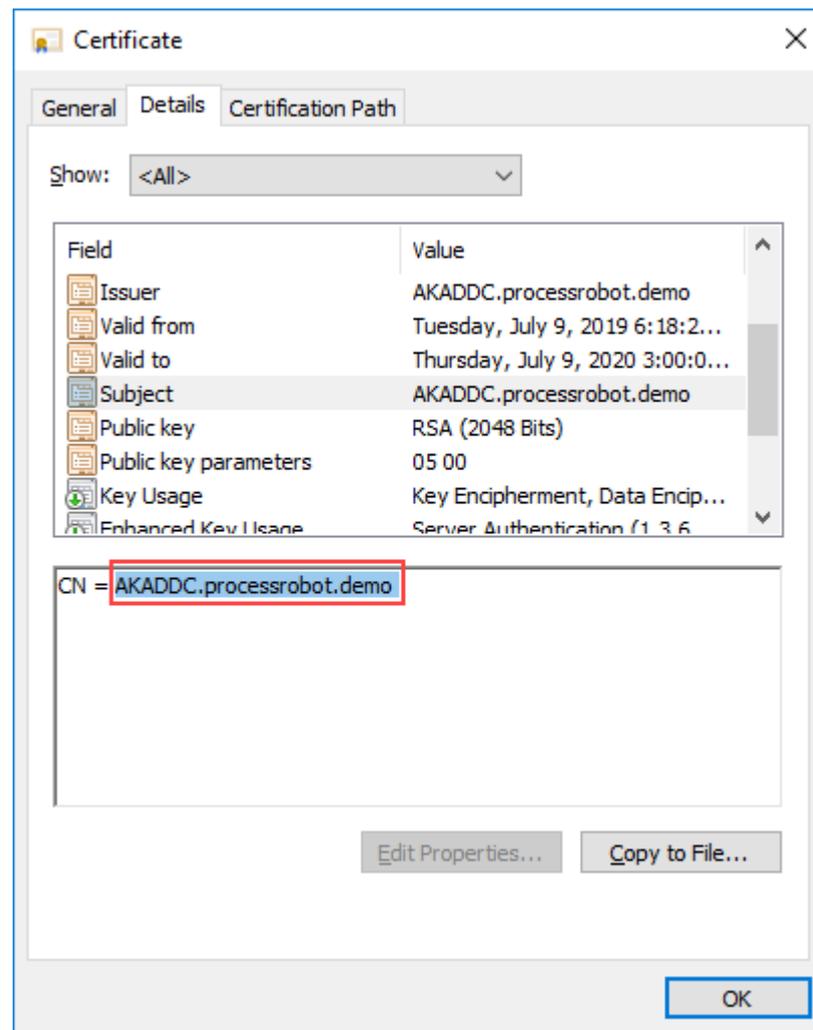
Click "Details"



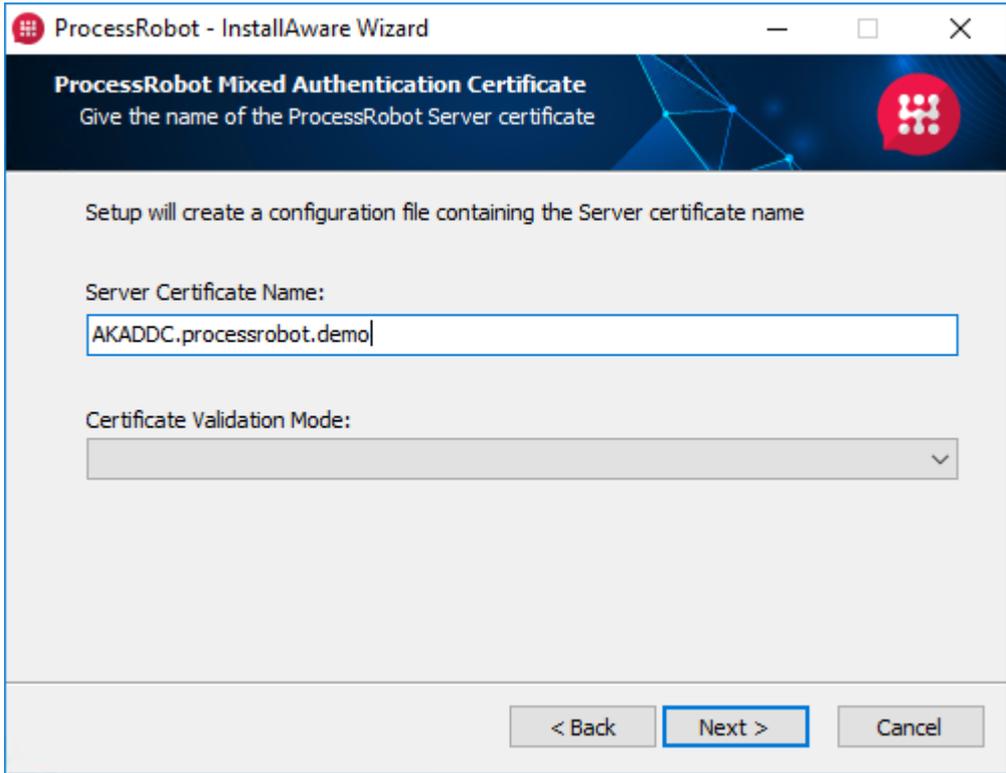
Click "Subject"



Copy the value for CN = <value>



Paste the CN value into the appropriate field of the ProcessRobot installer



ProcessRobot - InstallAware Wizard

ProcessRobot Mixed Authentication Certificate
Give the name of the ProcessRobot Server certificate

Setup will create a configuration file containing the Server certificate name

Server Certificate Name:

Certificate Validation Mode:

< Back Next > Cancel

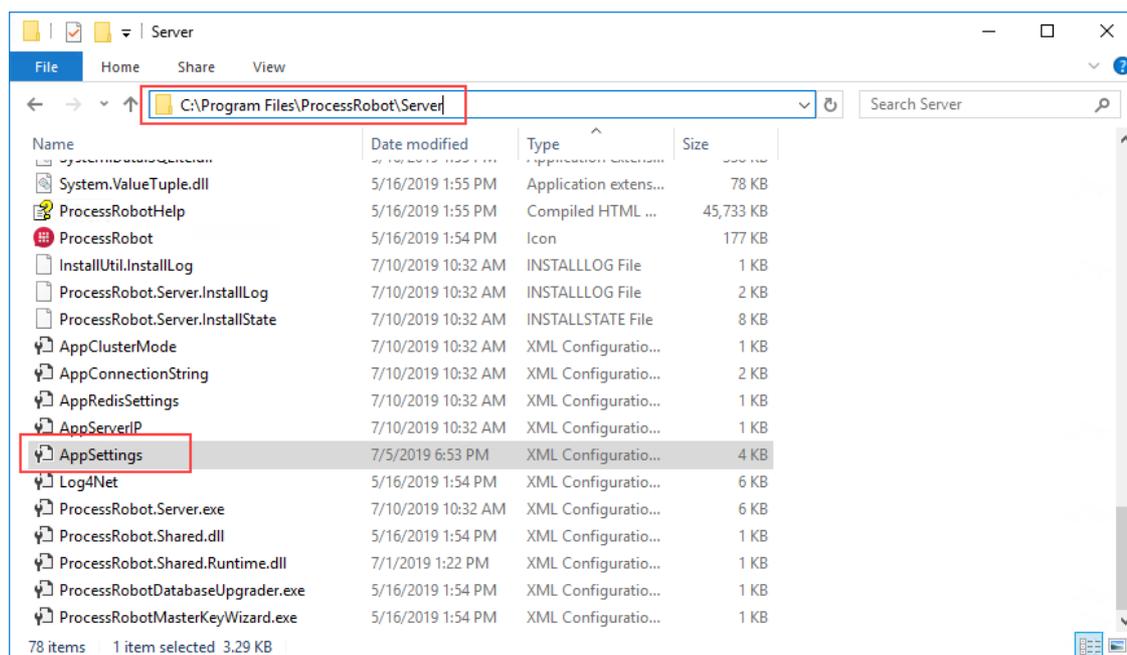
1.5.3 Configuring Custom Authentication after Installation

If Custom Authentication has not been configured when installing ProcessRobot, this can be configured later.

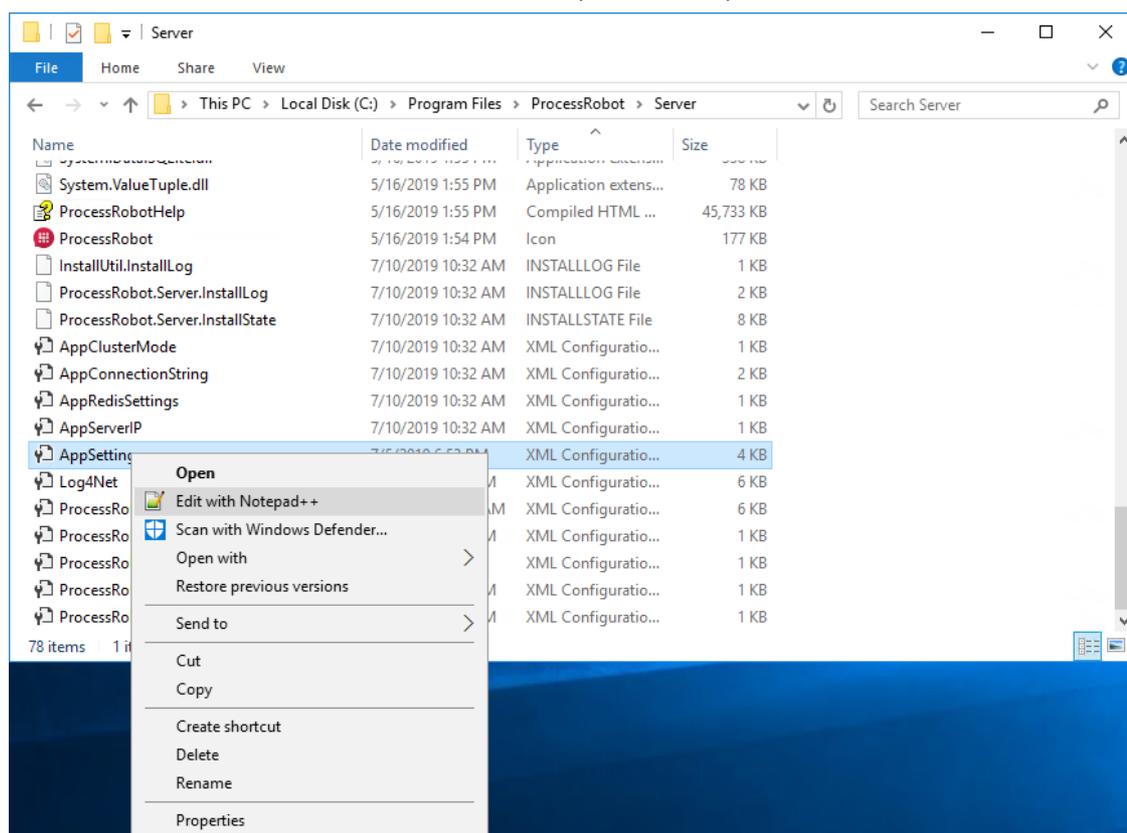
The first requirement is that a certificate has been installed on the Server machine. Once this in place, certain .config files will have to be edited and the appropriate values entered on the Server and Client machines.

On the Server machine:

Navigate to the file path where the Server components have been installed, by default C:\Program Files\ProcessRobot\Server. Locate the "AppSettings.config" file.



Edit the file with a text editor like Microsoft Notepad or Notepad++.



Locate in the file the line with following:

```
<add key="SupportMixedAuthentication" value ="false"/>
57
58 <add key="SupportMixedAuthentication" value ="false"/>
59
60 <!-- Valid Options: CurrentUser, LocalMachine -->
61 <add key="CertificateStoreLocation" value ="LocalMachine"/>
62
63 <!-- Valid Options: AddressBook, AuthRoot, CertificateAutho
64 <add key="CertificateStoreName" value ="My"/>
65
66 <!-- Valid Options: FindByThumbprint, FindBySubjectName, Fi
67 FindByTimeNotYetValid, FindByTimeExpired, FindByTemplateNar
68 <add key="CertificateFindBy" value ="FindBySubjectName"/>
69
70 <add key="CertificateFindValue" value ="ProcessRobot"/>
71
72 <add key="LogoutAfterProcessesCompletedLag" value="10000"/>
73
74 <add key="LogsPageSize" value ="1000"/>
75
76 </applicationConfiguration>
77
```

Normal text file

Change the value of "false" to "true".

```
57
58 <add key="SupportMixedAuthentication" value ="true"/>
59
60 <!-- Valid Options: CurrentUser, LocalMachine -->
61 <add key="CertificateStoreLocation" value ="LocalMachine"/>
62
63 <!-- Valid Options: AddressBook, AuthRoot, CertificateAuthori
64 <add key="CertificateStoreName" value ="My"/>
65
66 <!-- Valid Options: FindByThumbprint, FindBySubjectName, Find
67 FindByTimeNotYetValid, FindByTimeExpired, FindByTemplateName,
68 <add key="CertificateFindBy" value ="FindBySubjectName"/>
69
70 <add key="CertificateFindValue" value ="ProcessRobot"/>
71
72 <add key="LogoutAfterProcessesCompletedLag" value="10000"/>
73
74 <add key="LogsPageSize" value ="1000"/>
75
76 </applicationConfiguration>
77
```

Normal text file

Locate the line with the following:

```
<add key="CertificateFindValue" value ="ProcessRobot"/>
65
66 <!-- Valid Options: FindByThumbprint, FindBySubjectName, FindBySubjectDistinguishe
67 FindByTimeNotYetValid, FindByTimeExpired, FindByTemplateName, FindByApplicationPoli
68 <add key="CertificateFindBy" value ="FindBySerialNumber"/>
69
70 <add key="CertificateFindValue" value ="ProcessRobot"/>
71
72 <add key="LogoutAfterProcessesCompletedLag" value="10000"/>
73
74 <add key="LogsPageSize" value ="1000"/>
75
76 </applicationConfiguration>
77
```

Normal text file

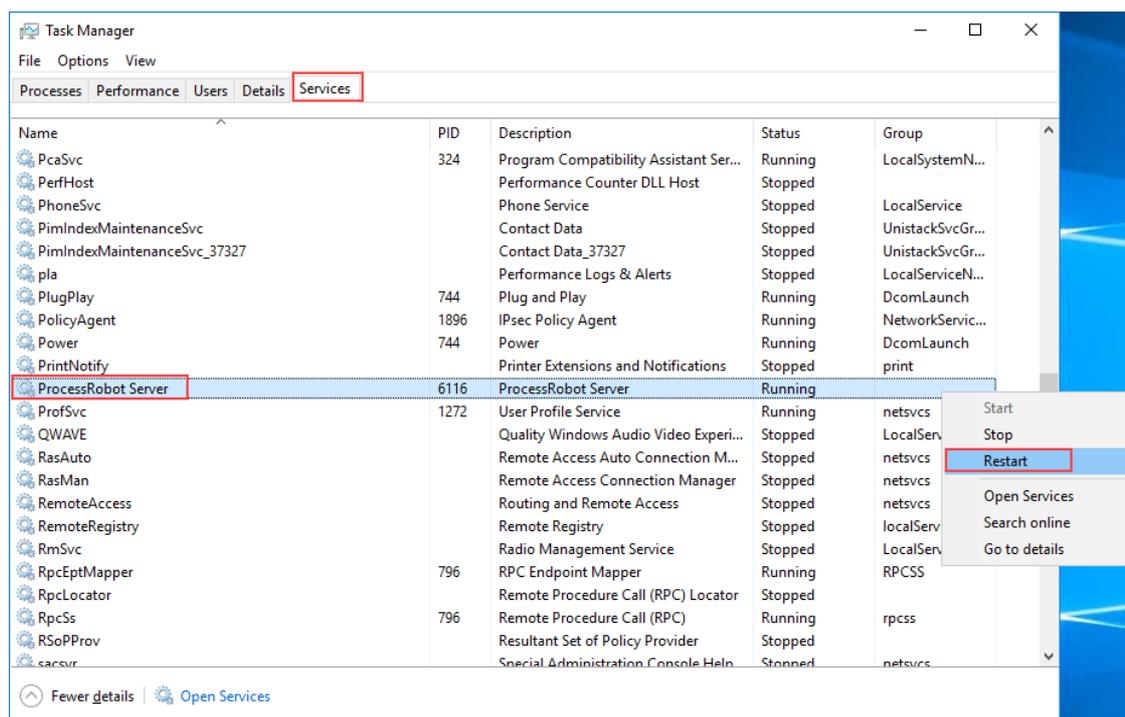
Change the value of "ProcessRobot" to the Certificate Name, as entered in the equivalent field in the ProcessRobot installer.

```
66 <!-- Valid Options: FindByThumbprint, FindBySubjectName, FindBySubject
67 FindByTimeNotYetValid, FindByTimeExpired, FindByTemplateName, FindByAp
68 <add key="CertificateFindBy" value ="FindBySubjectName"/>
69
70 <add key="CertificateFindValue" value ="AKADDC.processrobot.demo"/>
71
72 <add key="LogoutAfterProcessesCompletedLag" value="10000"/>
73
74 <add key="LogsPageSize" value ="1000"/>
75
76 </applicationConfiguration>
77
```

Normal text file length

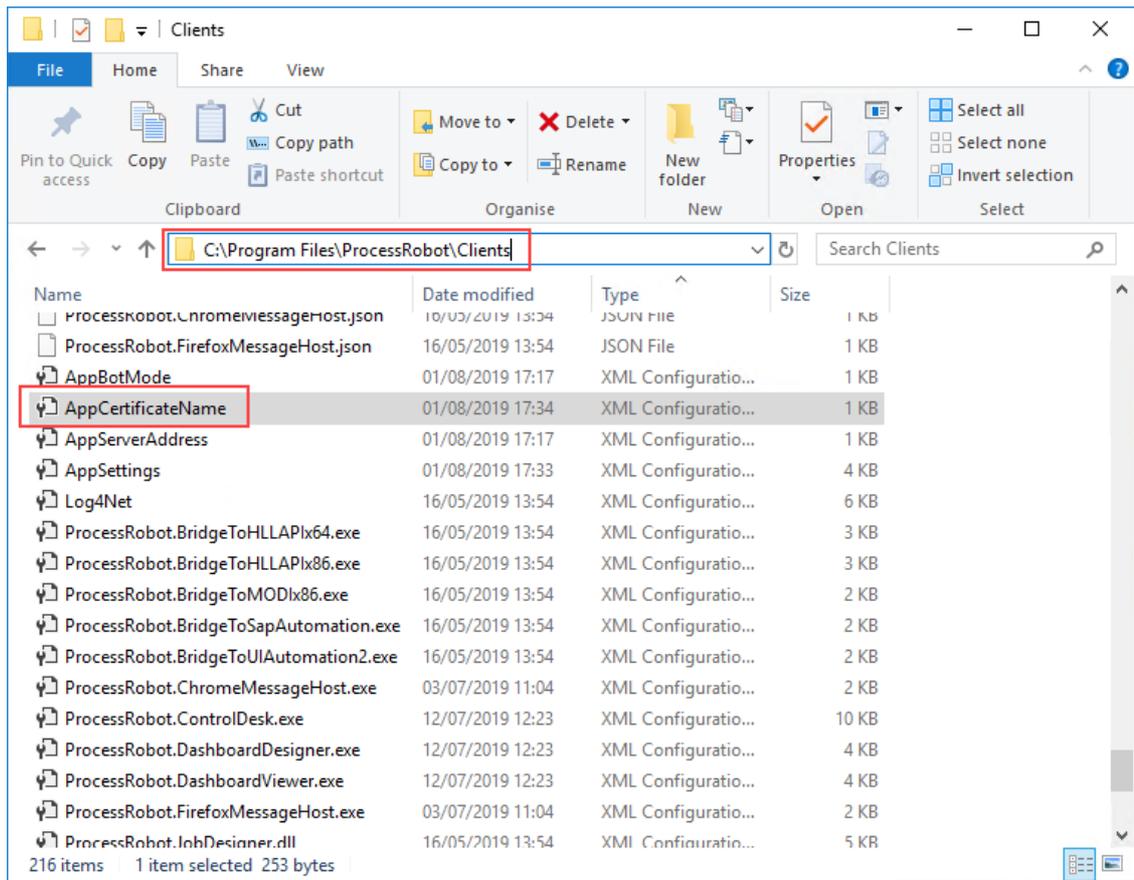
Save and close the file.

Restart the ProcessRobot Server service through the Windows Task Manager for these changes to take effect.

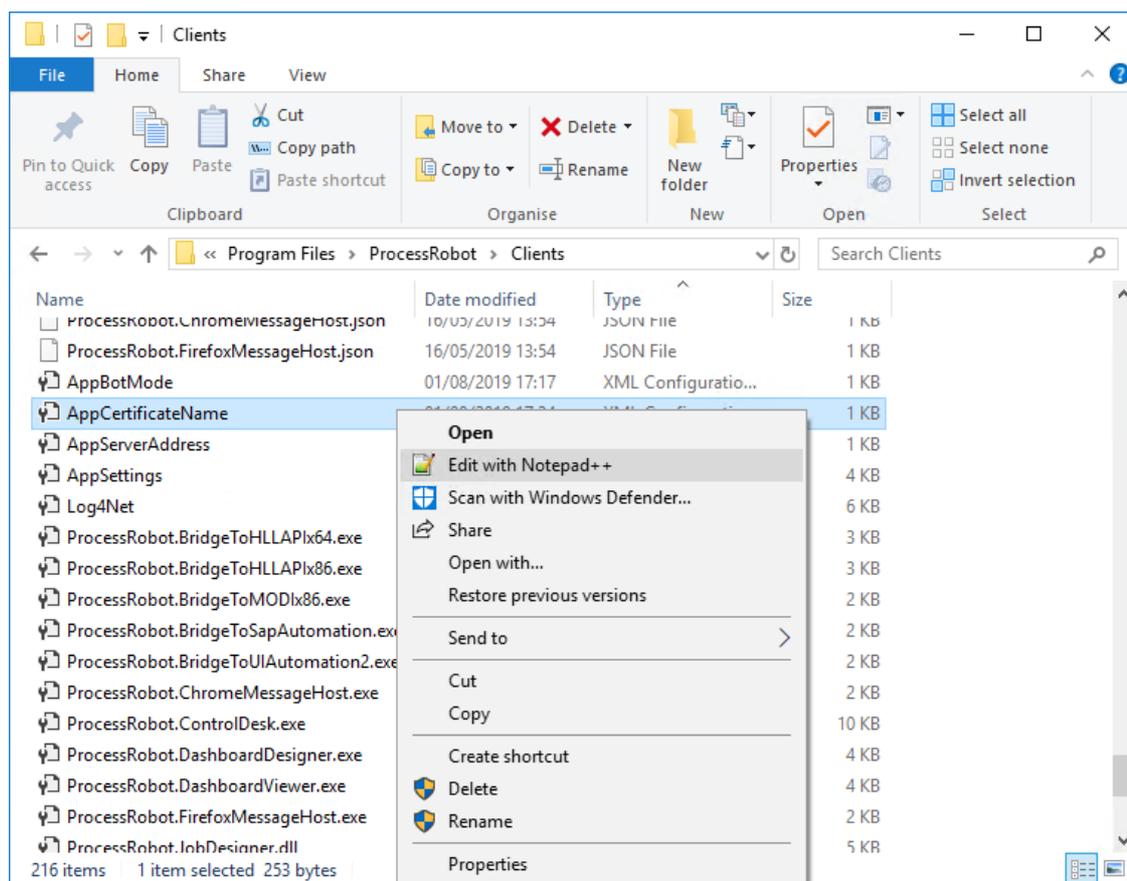


On the Client machines:

Navigate to the file path where the Client components have been installed, by default C:\Program Files\ProcessRobot\Clients. Locate the "AppCertificateName.config" file.



Edit the file with a text editor like Microsoft Notepad or Notepad++.



Edit the following lines of the file:

```
<add key="CertificateName" value =""/>
```

```
<add key="CertificateValidationMode" value =""/>
```

In the first of the above lines, enter the name of the Certificate used as the value. In the second line, enter the Validation Mode used with the Certificate as the value.

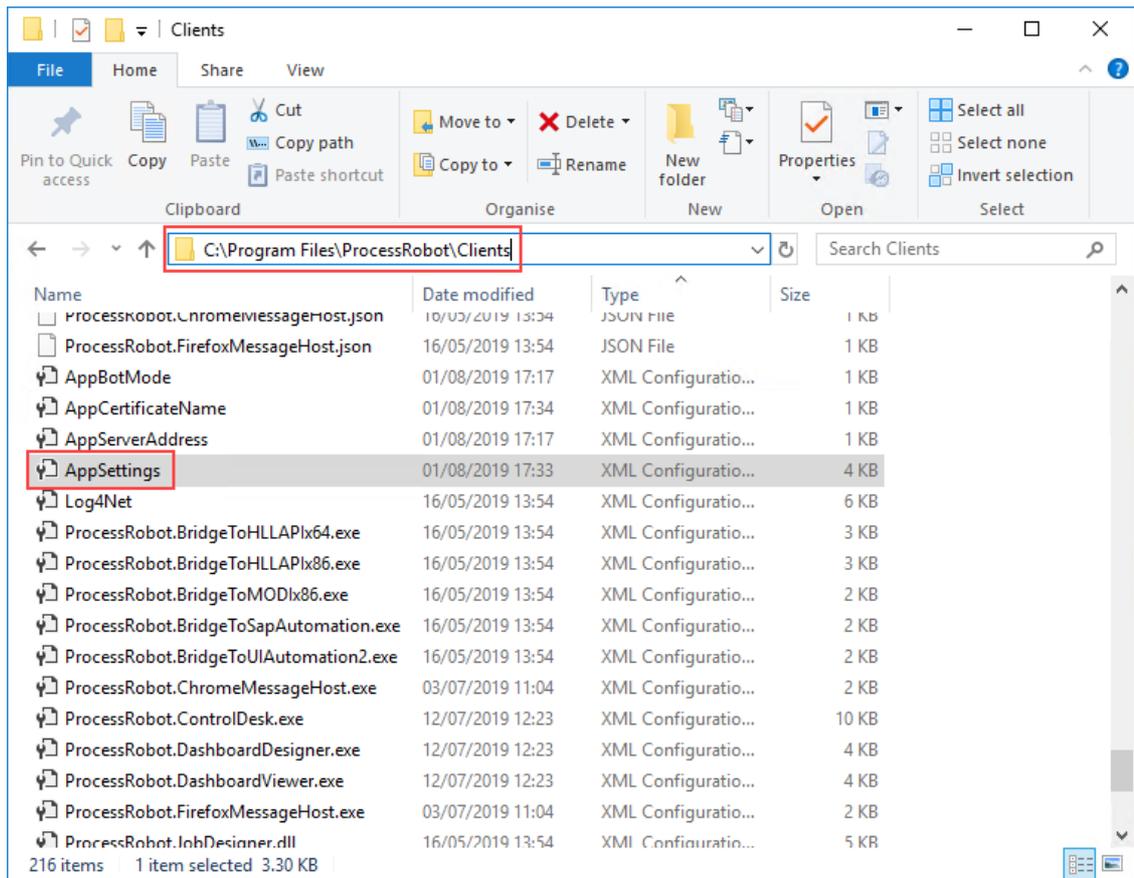
```

1 <certificateConfiguration>
2 <add key="CertificateName" value ="AKADDC.processrobot.demo"/>
3 <!-- Valid Options: None, PeerTrust, Chain Trust, Peer Or Chain Trust -->
4 <add key="CertificateValidationMode" value ="None"/>
5 </certificateConfiguration>
6

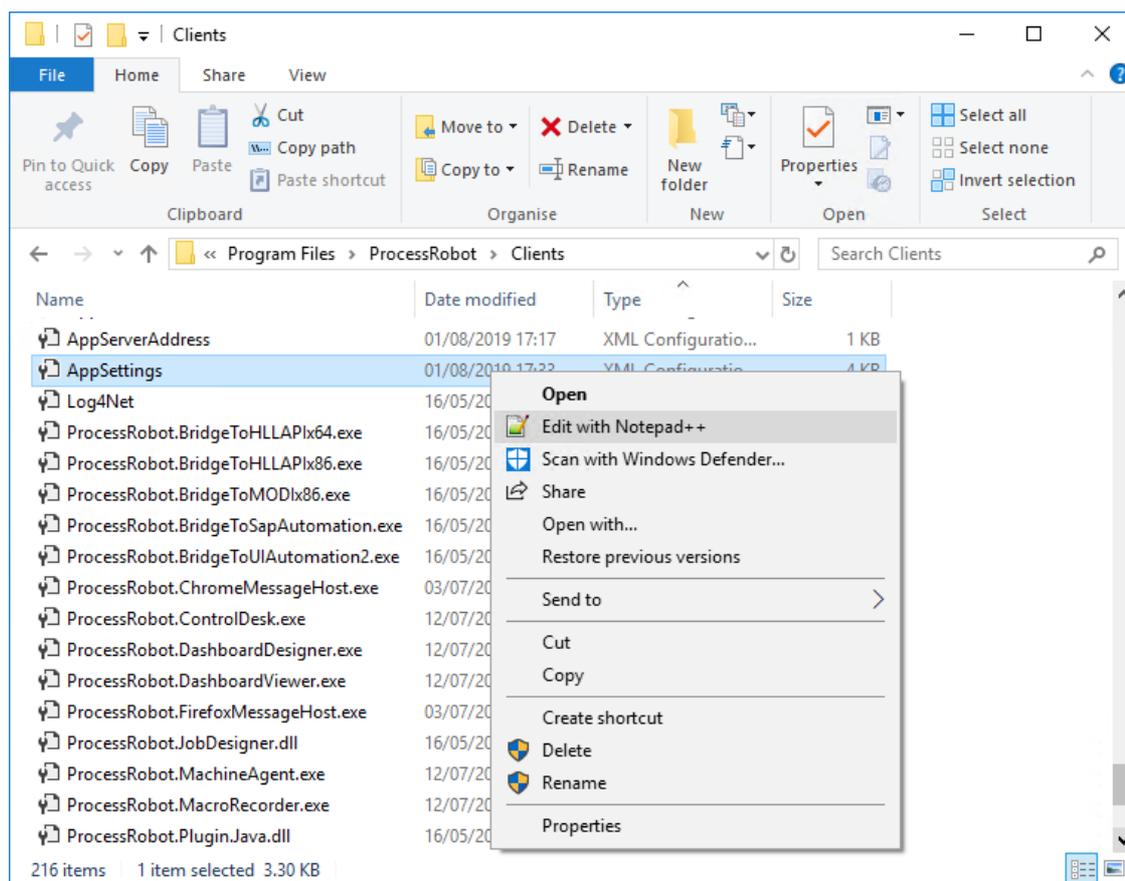
```

Save and close the file.

Next, navigate back to the parent directory of this file, by default C:\Program Files\ProcessRobot\Clients and locate the "AppSettings.config" file.



Edit the file with a text editor like Microsoft Notepad or Notepad++.



Edit the following lines of the file:

```
<add key="SupportMixedAuthentication" value ="false"/>
```

```
<add key="CertificateFindValue" value ="ProcessRobot"/>
```

In the first of the above lines, edit the value of "false" to "true", for "SupportMixedAuthentication". In the second line, replace "ProcessRobot" with the Certificate name as the value of "CertificateFindValue".

```

57
58 <add key="SupportMixedAuthentication" value ="true"/>
59
60 <!-- Valid Options: CurrentUser, LocalMachine -->
61 <add key="CertificateStoreLocation" value ="LocalMachine"/>
62
63 <!-- Valid Options: AddressBook, AuthRoot, CertificateAuthority, Disallowed, My, Root, TrustedPeople, TrustedPublisher -->
64 <add key="CertificateStoreName" value ="My"/>
65
66 <!-- Valid Options: FindByThumbprint, FindBySubjectName, FindBySubjectDistinguishedName, FindByIssuerName,
67 FindByIssuerDistinguishedName, FindBySerialNumber, FindByTimeValid,
68 FindByTimeNotYetValid, FindByTimeExpired, FindByTemplateName, FindByApplicationPolicy, FindByCertificatePolicy,
69 FindByExtension, FindByKeyUsage, FindBySubjectKeyIdentifier -->
70 <add key="CertificateFindBy" value ="FindBySubjectName"/>
71 <add key="CertificateFindValue" value ="AKADDC.processrobot.demo"/>

```

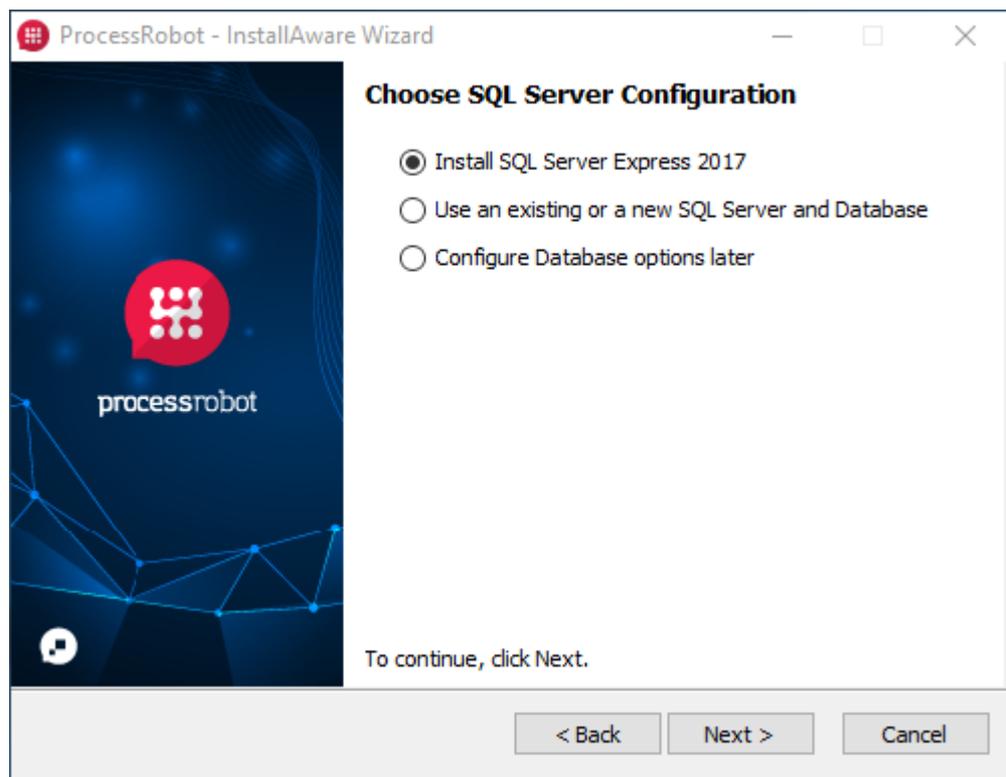
Save and close the file.

Custom authentication has now been configured for the Client machine as well as the Server.

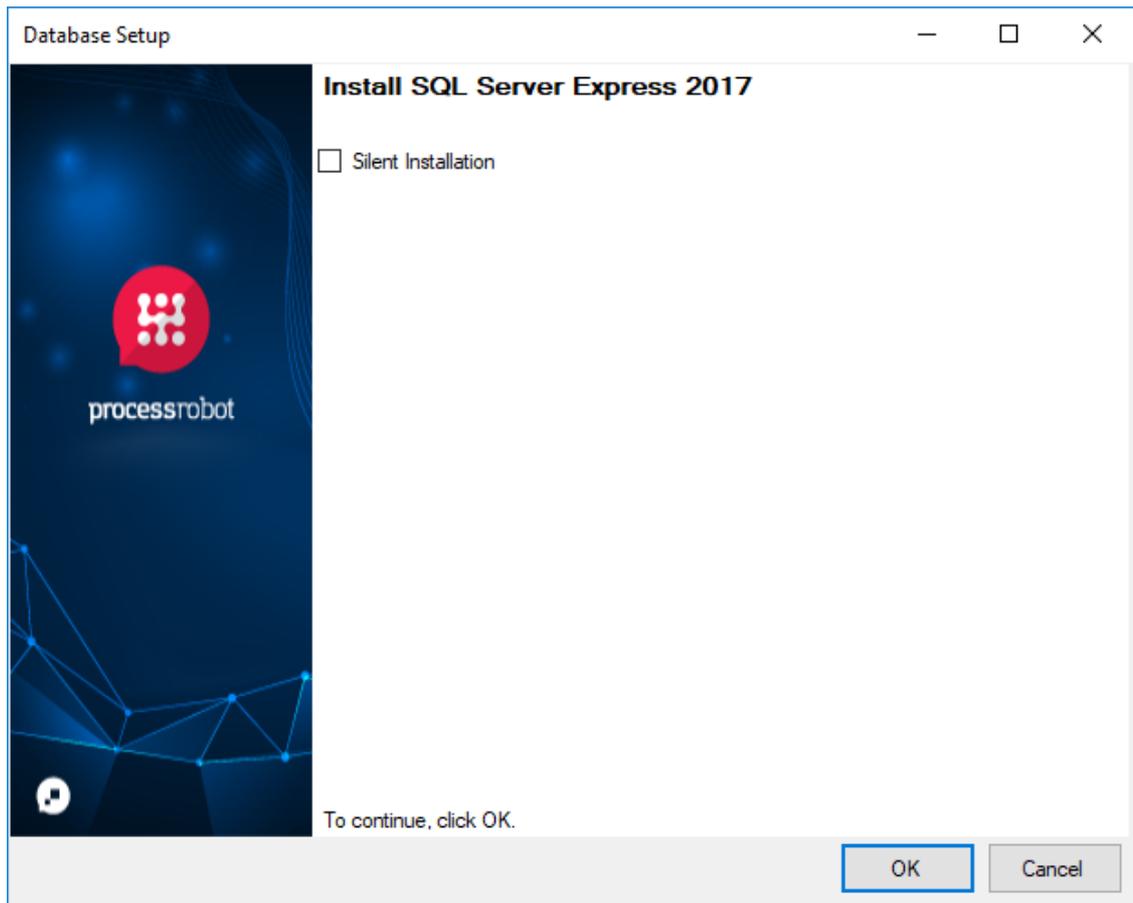
1.5.4 Installing SQL Server Express

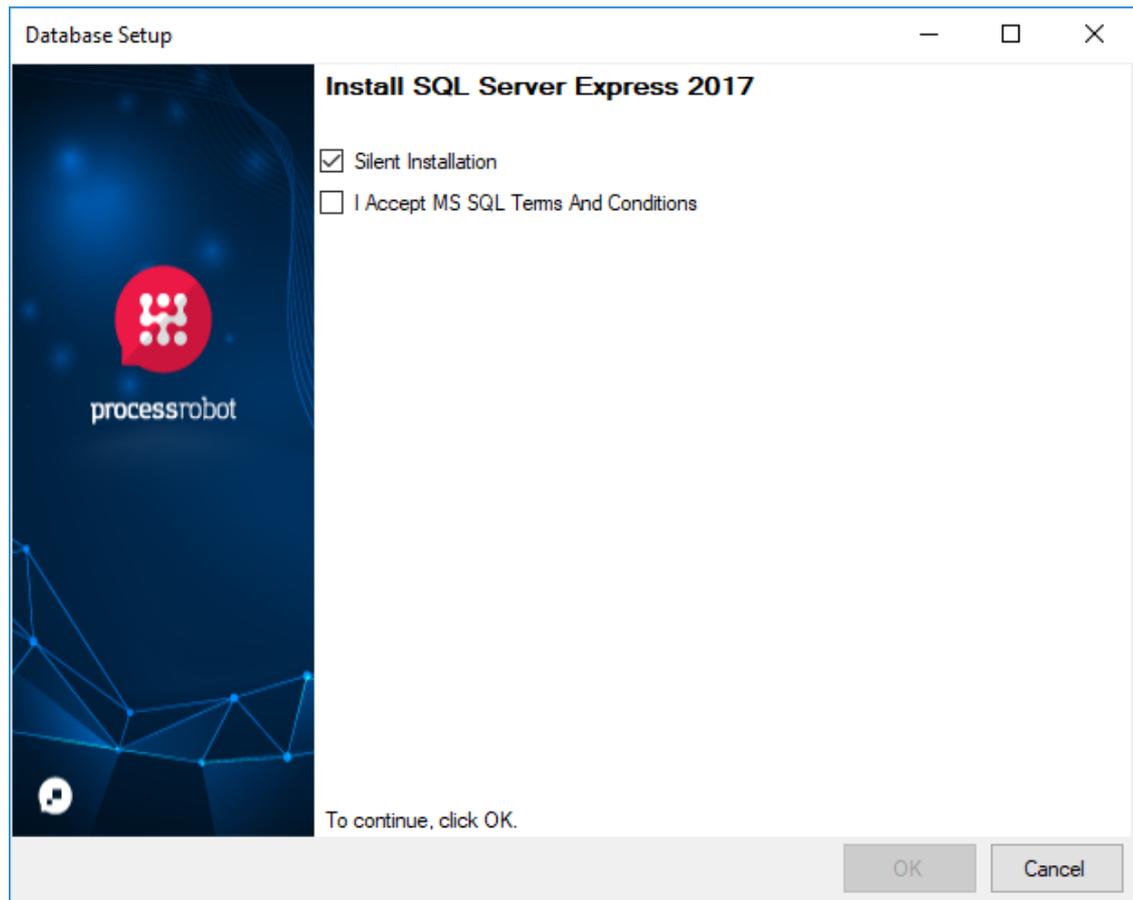
Installing SQL Server Express through the ProcessRobot installer

The new ProcessRobot installer includes the capability to install Microsoft SQL Server Express directly through the installer. To install SQL Server and SQL Server Management Studio and set up the ProcessRobot database without using the ProcessRobot installer, read these articles: [Installing Microsoft SQL Express 2016](#)^[79], [Installing SQL Server Management Studio](#)^[83], and [Setting up the ProcessRobot Database](#)^[84].

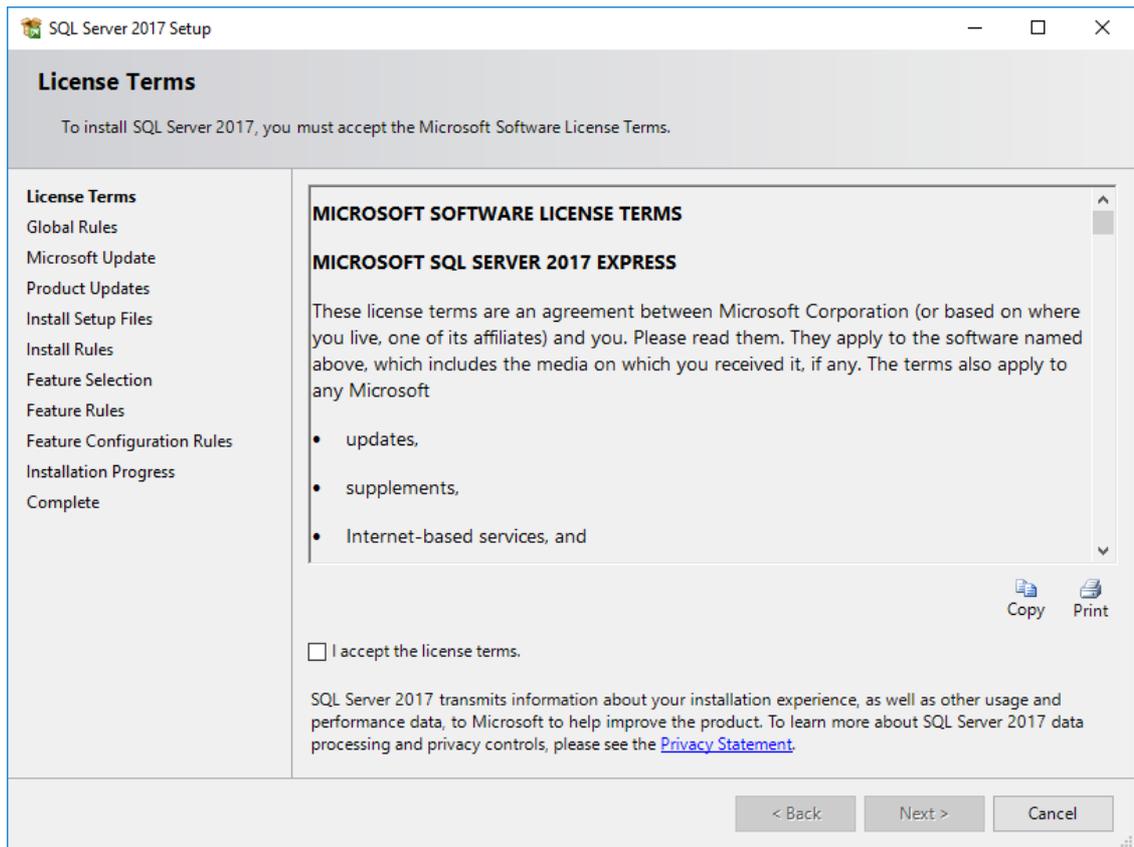


Having chosen to install SQL Server Express, next choose whether to enable Silent Installation.

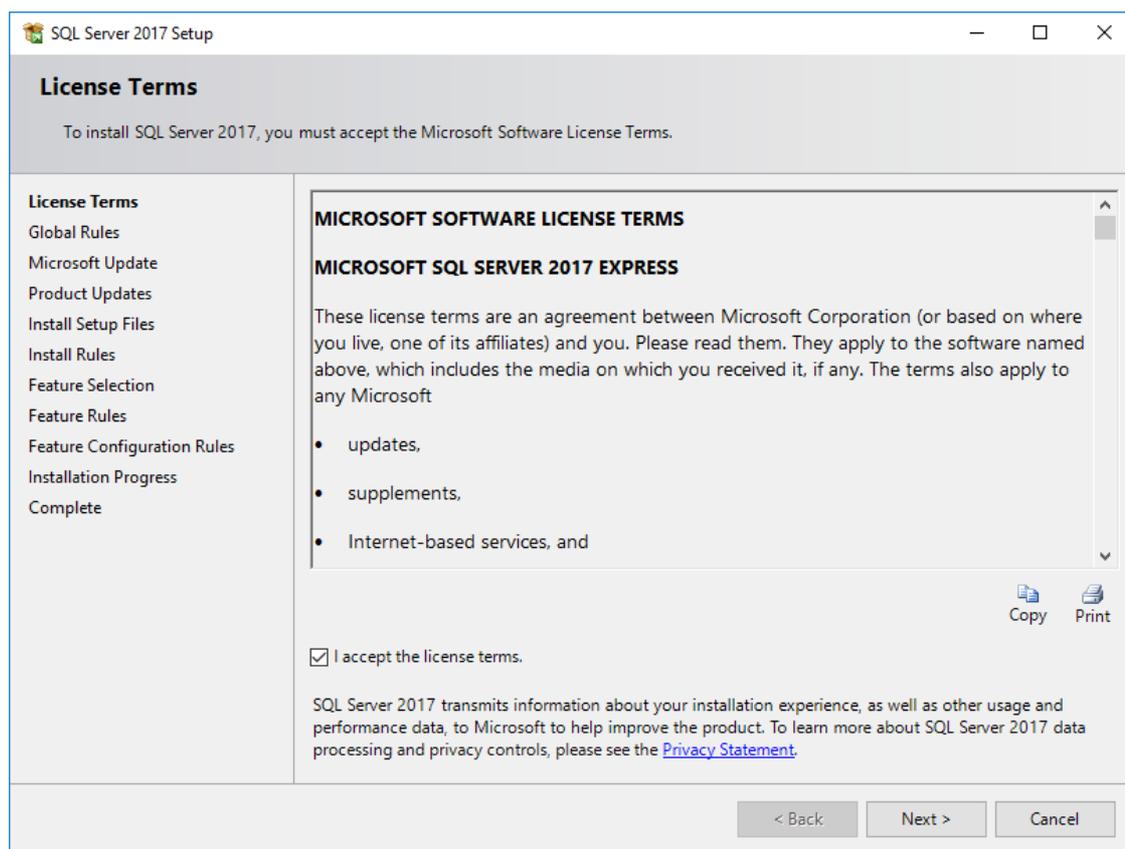




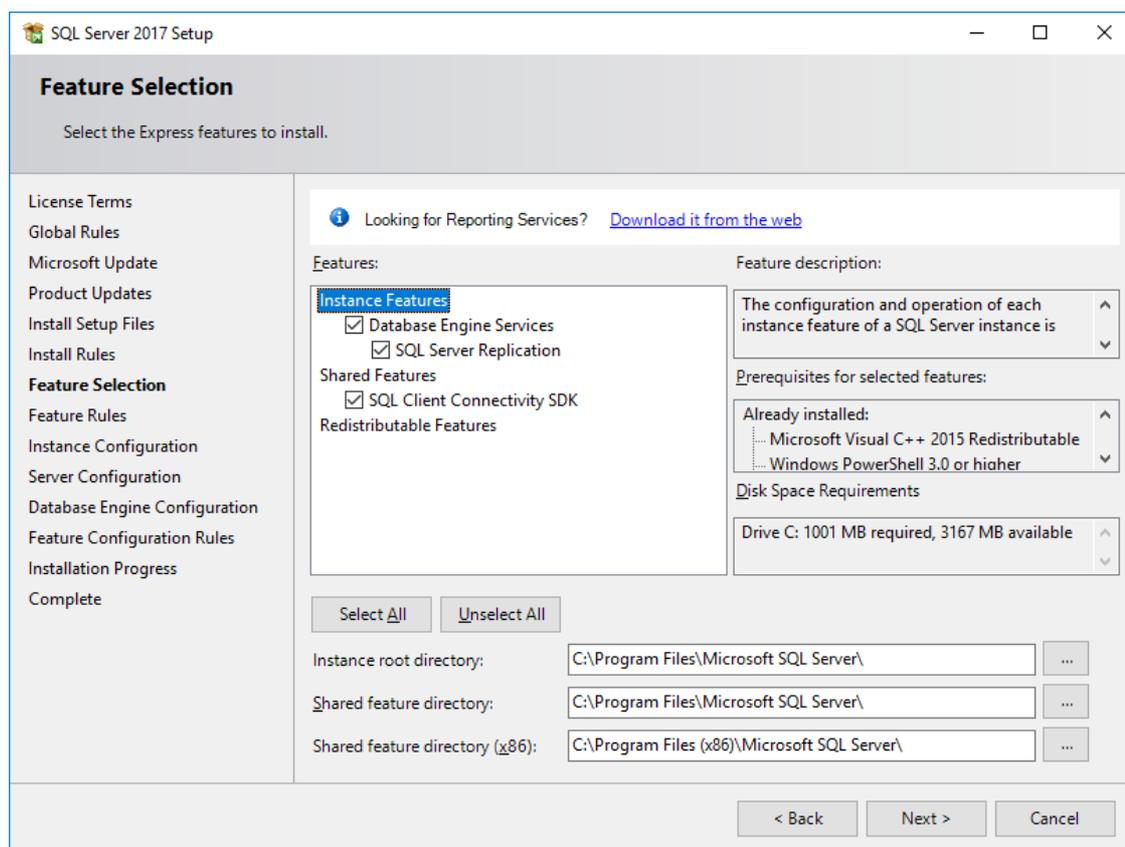
After this choice has been submitted in the ProcessRobot installer, the SQL Server installer will launch. Once this process has been completed, the ProcessRobot installation will continue.



Accept the license terms is required to proceed with the installation.



The installer will continue through some stages, such as “Global Rules”, without requiring user input. If no issues are found the installer will continue to “Feature Selection”. At this stage, the user can select which features to install.



At the “Instance Configuration” stage, it is recommended to select “Named instance”, and select how the SQL Server Express instance will be named. If no other instance of the application exists on the machine, the default instance name is “SQLEXPRESS”.

SQL Server 2017 Setup

Instance Configuration

Specify the name and instance ID for the instance of SQL Server. Instance ID becomes part of the installation path.

License Terms
Global Rules
Microsoft Update
Product Updates
Install Setup Files
Install Rules
Feature Selection
Feature Rules
Instance Configuration
Server Configuration
Database Engine Configuration
Feature Configuration Rules
Installation Progress
Complete

Default instance
 Named instance:

Instance ID:

SQL Server directory: C:\Program Files\Microsoft SQL Server\MSSQL14.SQLEXPRESS

Installed instances:

Instance Name	Instance ID	Features	Edition	Version
---------------	-------------	----------	---------	---------

< Back Next > Cancel

At the "Server Configuration" stage, determine the Server Service Accounts and the database Collation type. The default collation type is Latin1_General_CI_AS.

SQL Server 2017 Setup

Server Configuration

Specify the service accounts and collation configuration.

License Terms
Global Rules
Microsoft Update
Product Updates
Install Setup Files
Install Rules
Feature Selection
Feature Rules
Instance Configuration
Server Configuration
Database Engine Configuration
Feature Configuration Rules
Installation Progress
Complete

Service Accounts Collation

Microsoft recommends that you use a separate account for each SQL Server service.

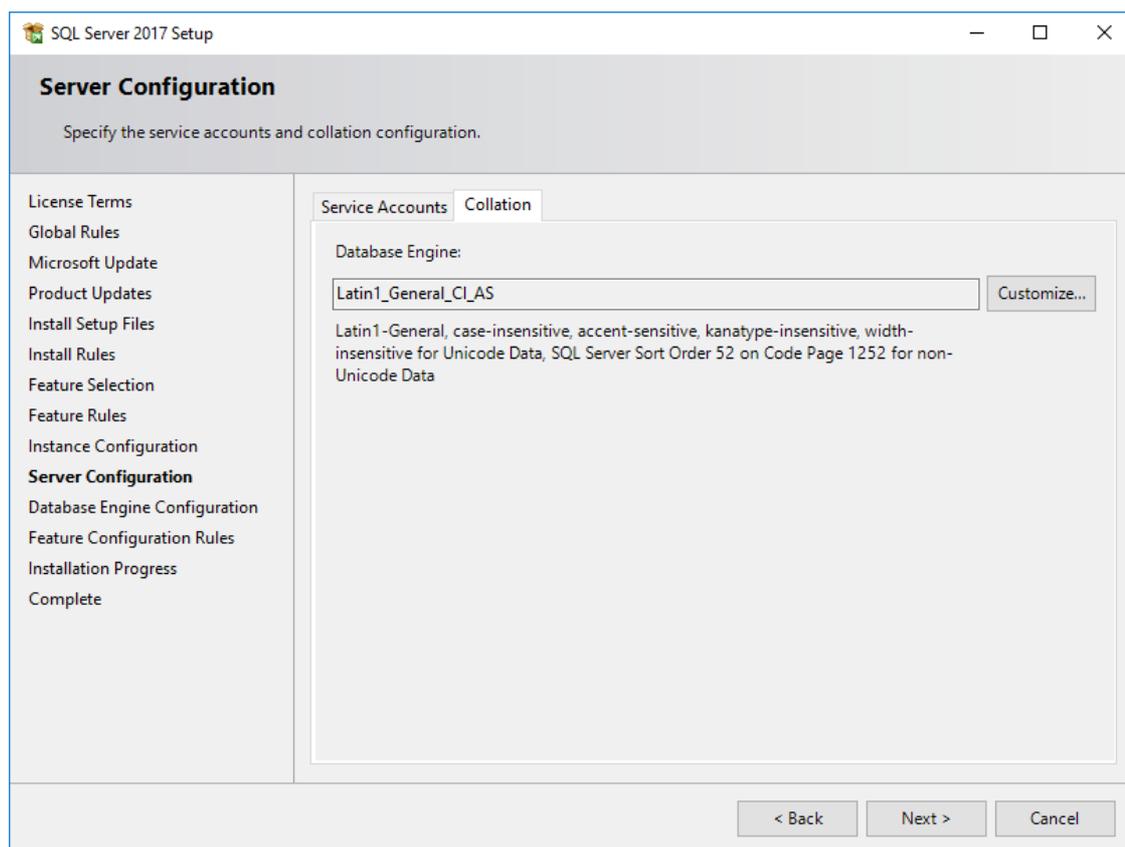
Service	Account Name	Password	Startup Type
SQL Server Database Engine	NT Service\MSSQL\$SQL...		Automatic
SQL Server Browser	NT AUTHORITY\LOCAL ...		Disabled

Grant Perform Volume Maintenance Task privilege to SQL Server Database Engine Service

This privilege enables instant file initialization by avoiding zeroing of data pages. This may lead to information disclosure by allowing deleted content to be accessed.

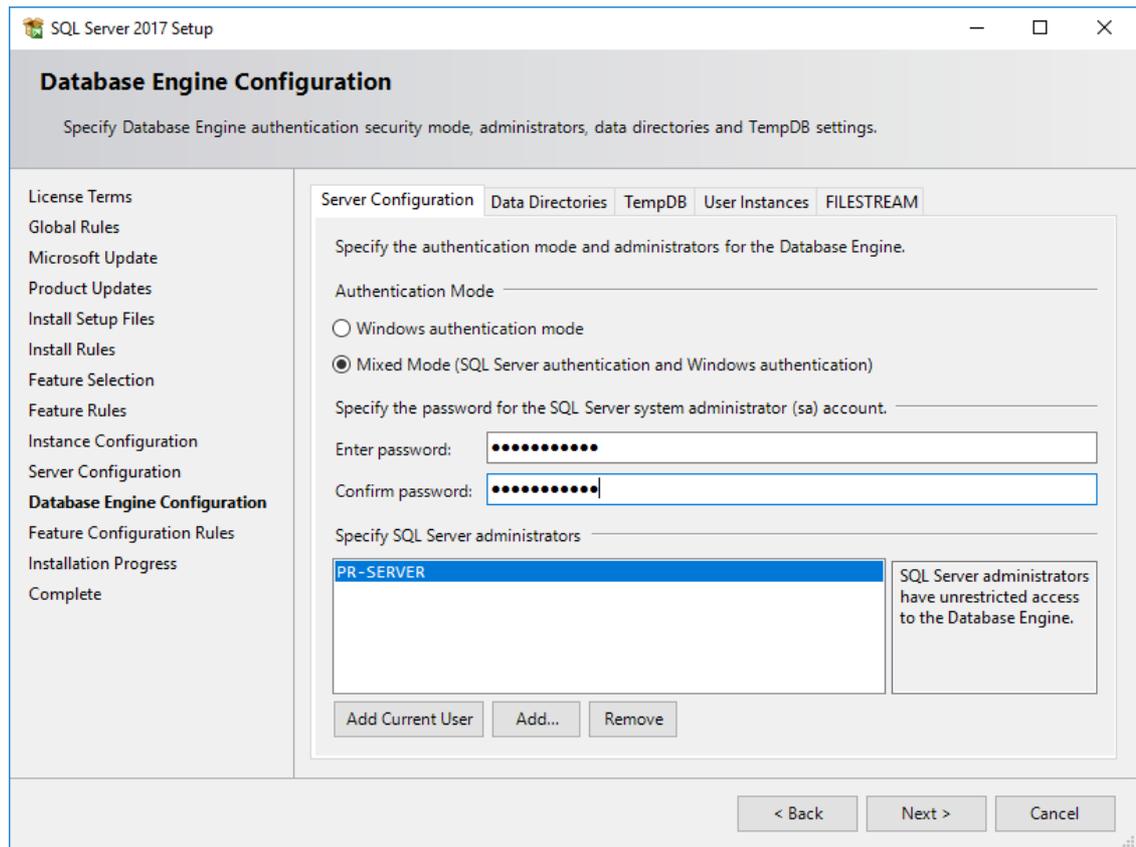
[Click here for details](#)

< Back Next > Cancel

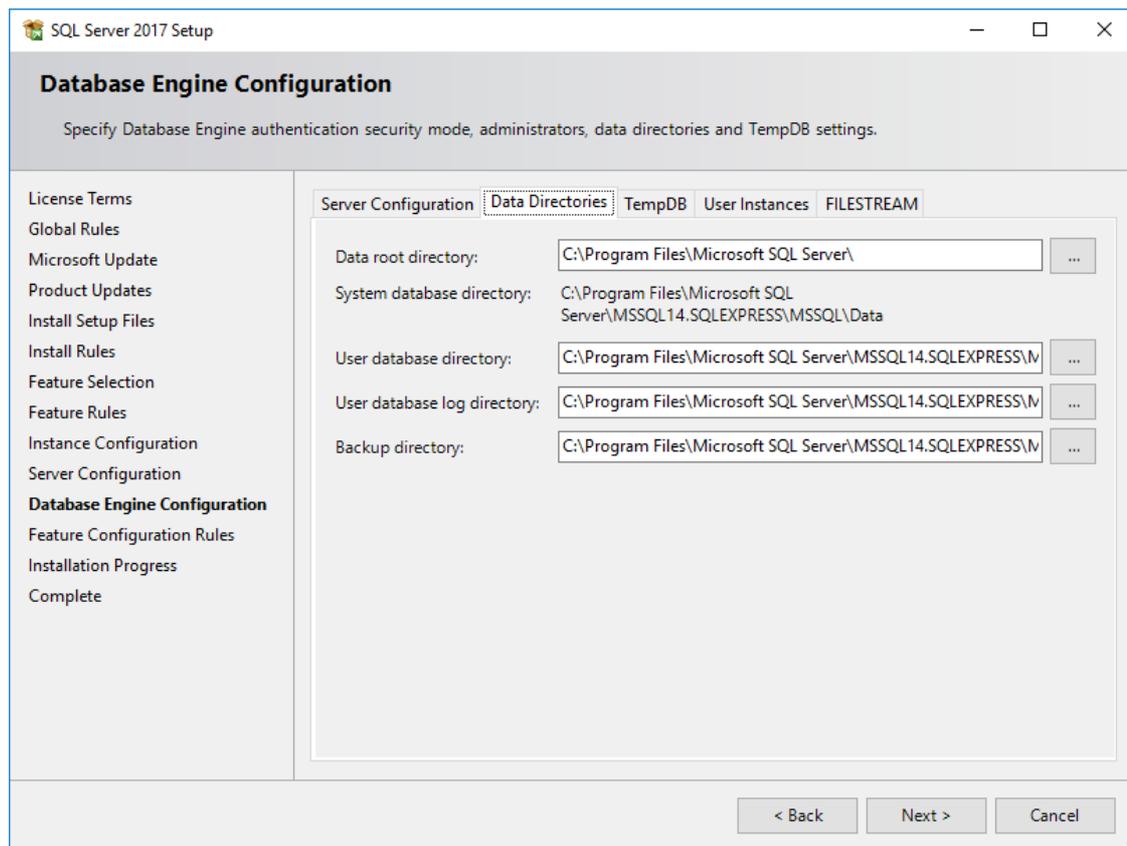


At the “Database Engine Configuration” stage, determine settings related to Server Configuration, Data Directories, TempDB settings, User Instances and the Filestream.

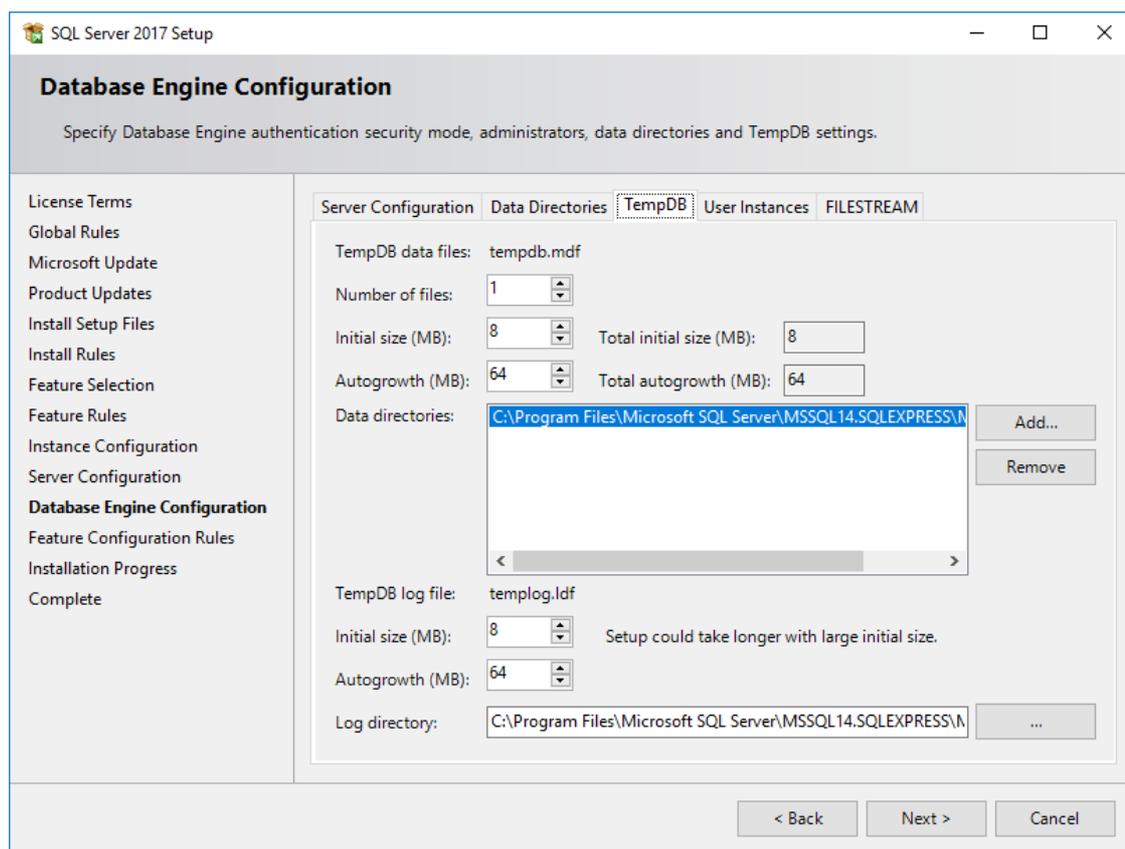
Under “Server Configuration”, determine whether SQL Server Express should use only Windows authentication or both Windows and SQL Server authentication. Optionally specify a password for the system administrator (sa) account.



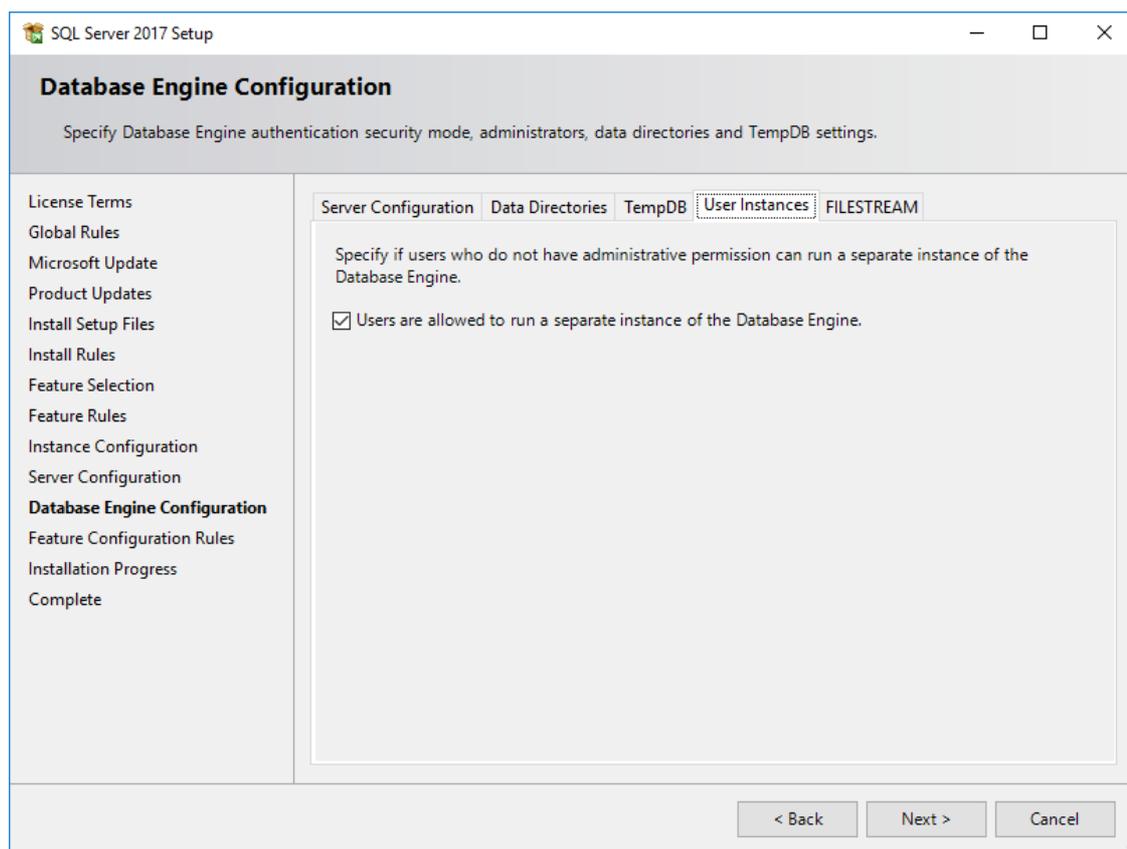
Under "Data Directories", determine the directories to be used for the System, User and other SQL Server databases.



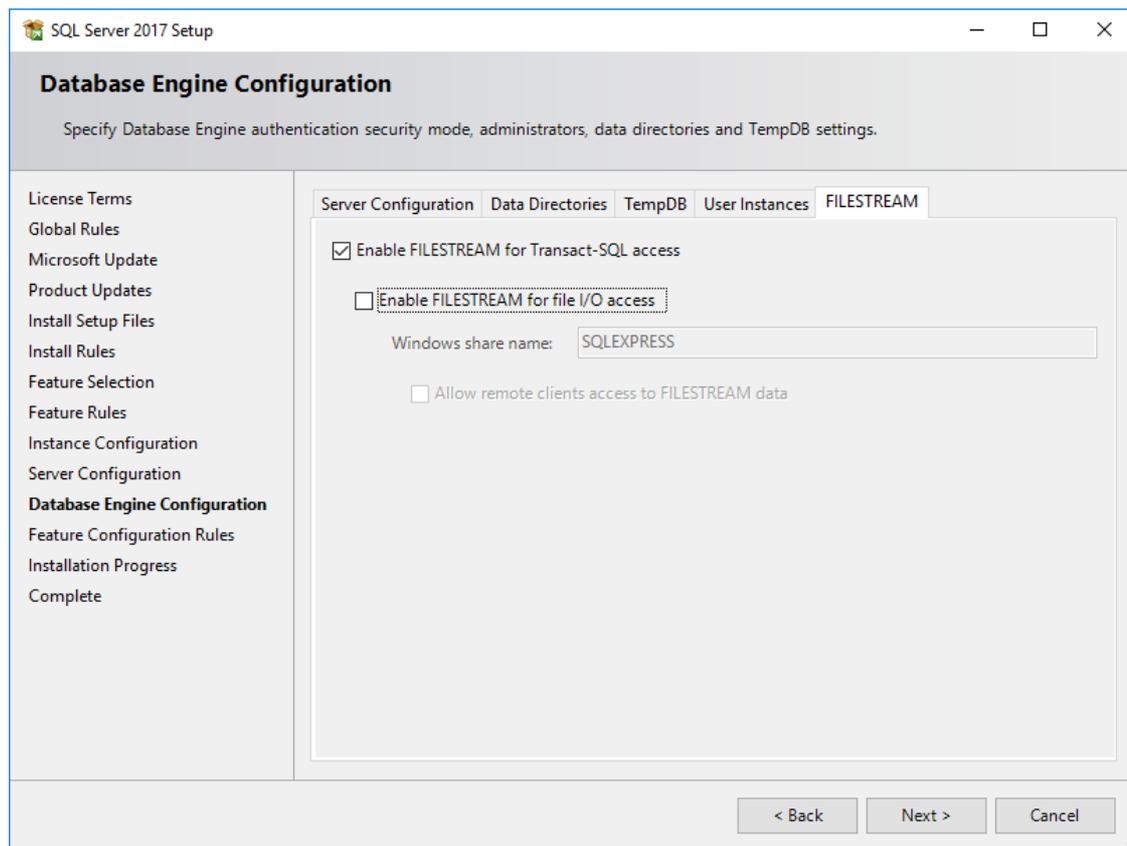
Under "TempDB", specify the directory be used and other settings for temporary databases.



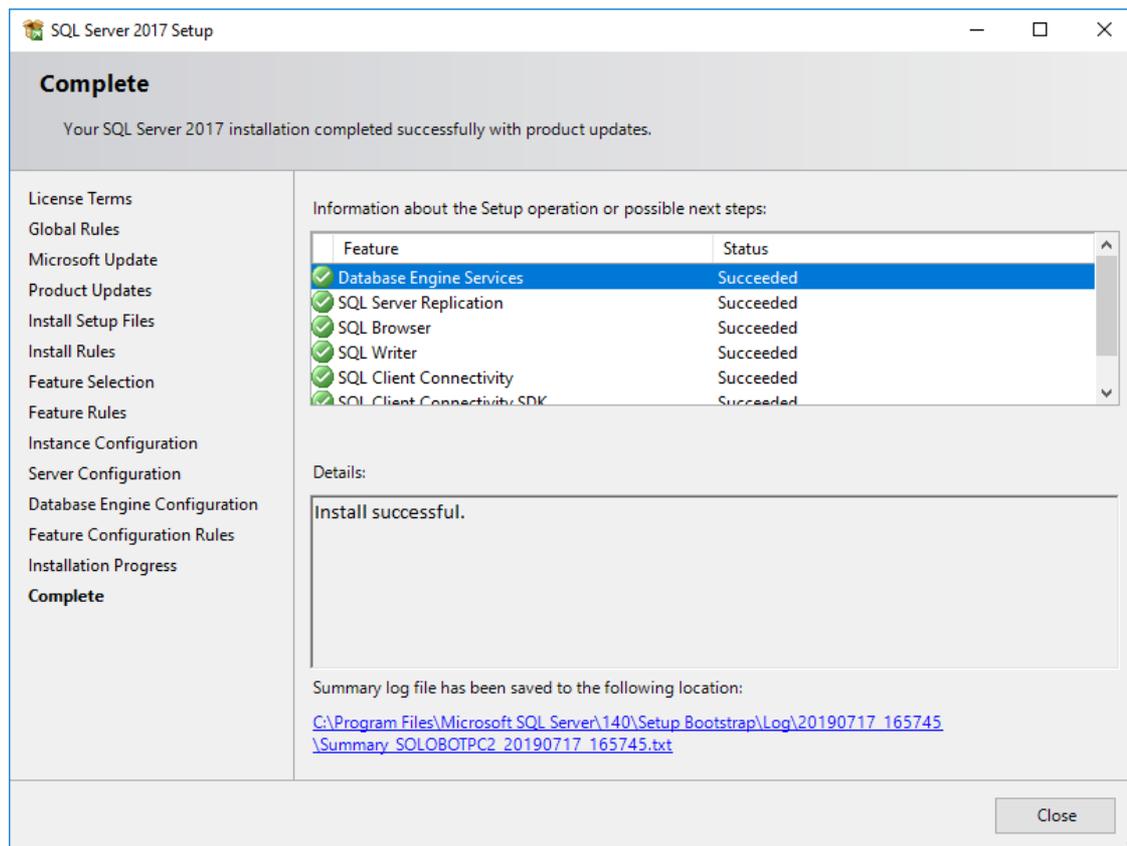
Under "User Instances", specify if users can run multiple instances of the Database Engine.



Under "FILESTREAM", select to enable the Filestream for Transact-SQL and file I/O access.



Following these stages, if no issues occur, the SQL Server installation will complete.



Once the SQL Server installation has completed, the user can create a SQL Server Database and User directly from the ProcessRobot installer.

Database Setup

Configure SQL Server Express 2017

Database Host:

Database User:

Database Password:

Database Options:

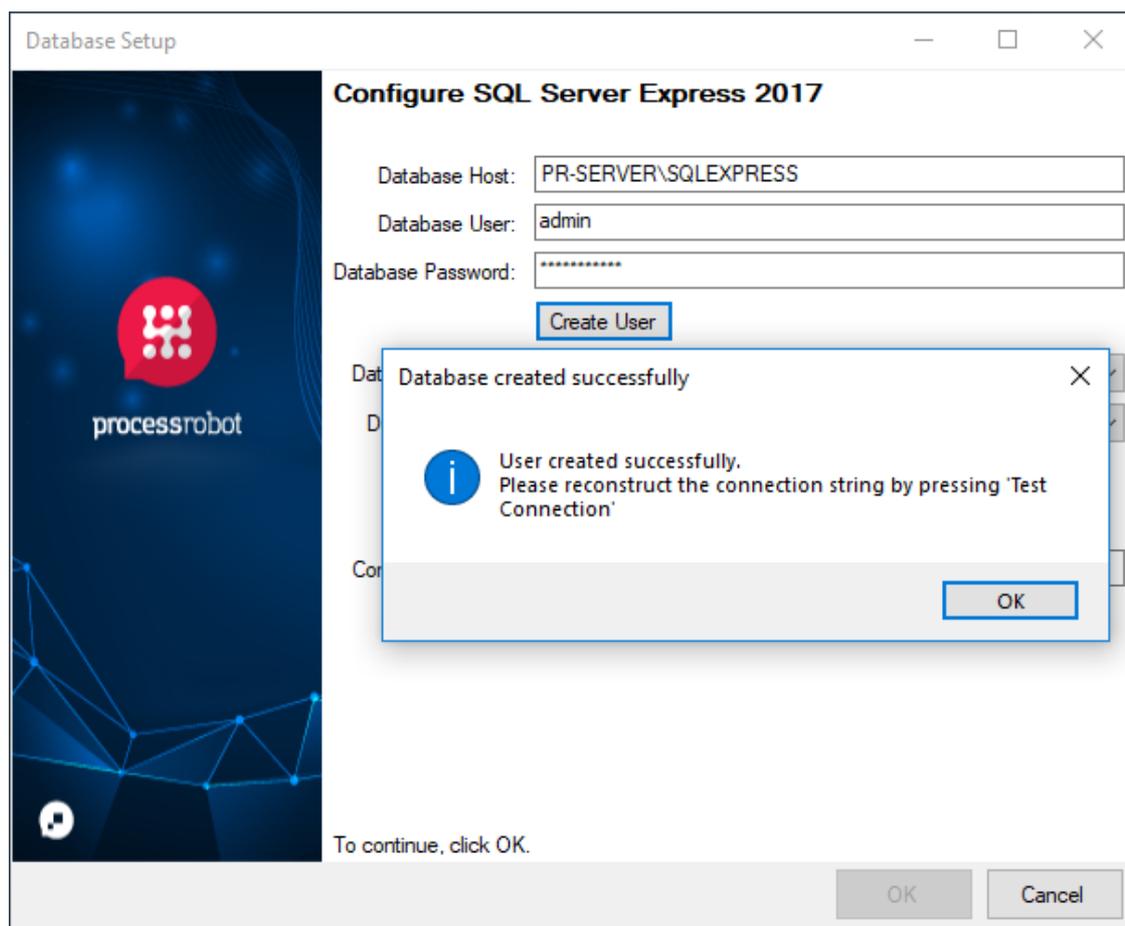
Database Name:

Use different DB for logs

Connection String:

To continue, click OK.

To create the Database and User, the Database Host is required. Enter this in the format **<machine name> \ <SQL Server instance name>**, e.g. PR-SERVER\SQLEXPRESS. Then enter a username and password combination for the Database User and click "Create User".



Next, assuming that SQL Server had not been previously installed on the machine, proceed to Create a New Database to be used for ProcessRobot.

Database Setup

Configure SQL Server Express 2017

Database Host: PR-SERVER\SQLEXPRESS

Database User: admin

Database Password: *****

Create User

Database Options: Use Existing Database

Database Name: Use Existing Database
Create New Database

Use different DB for logs

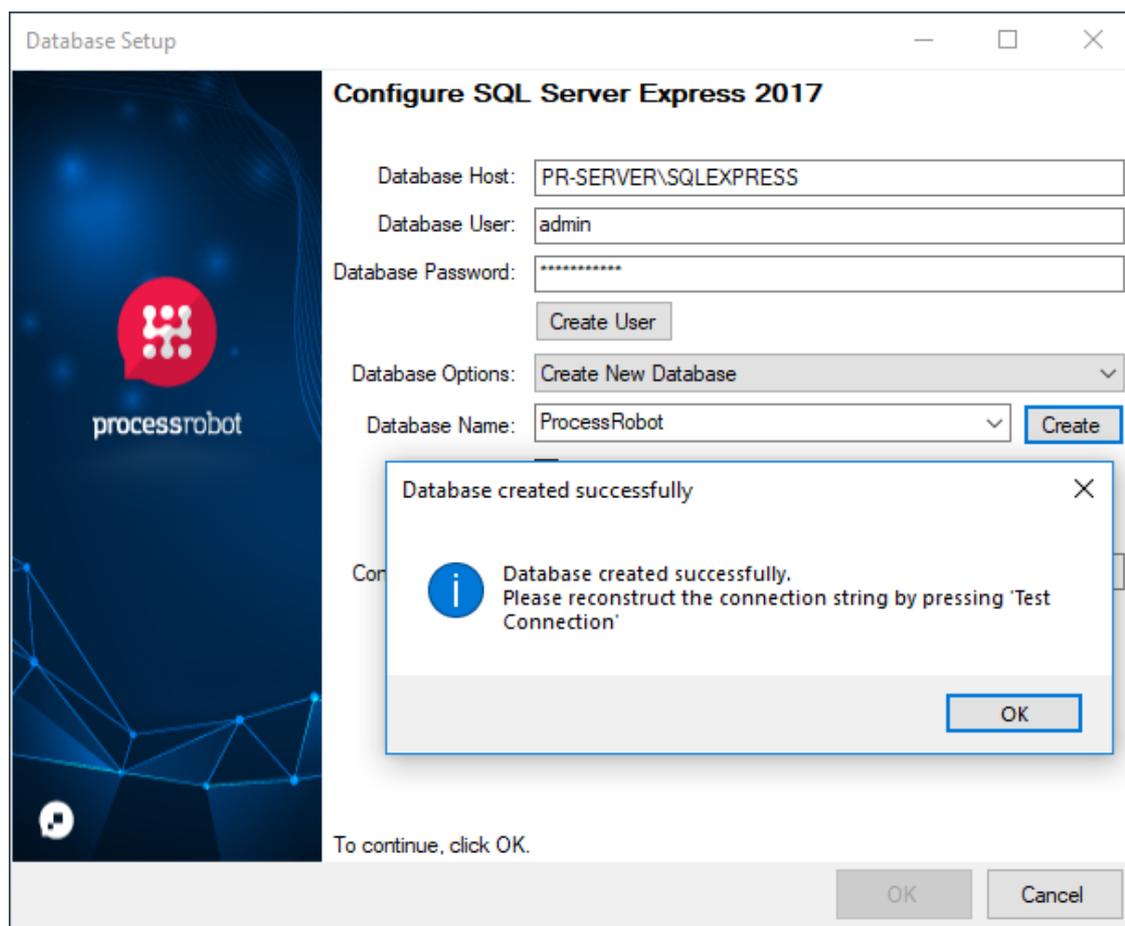
Test Connection...

Connection String:

To continue, click OK.

OK Cancel

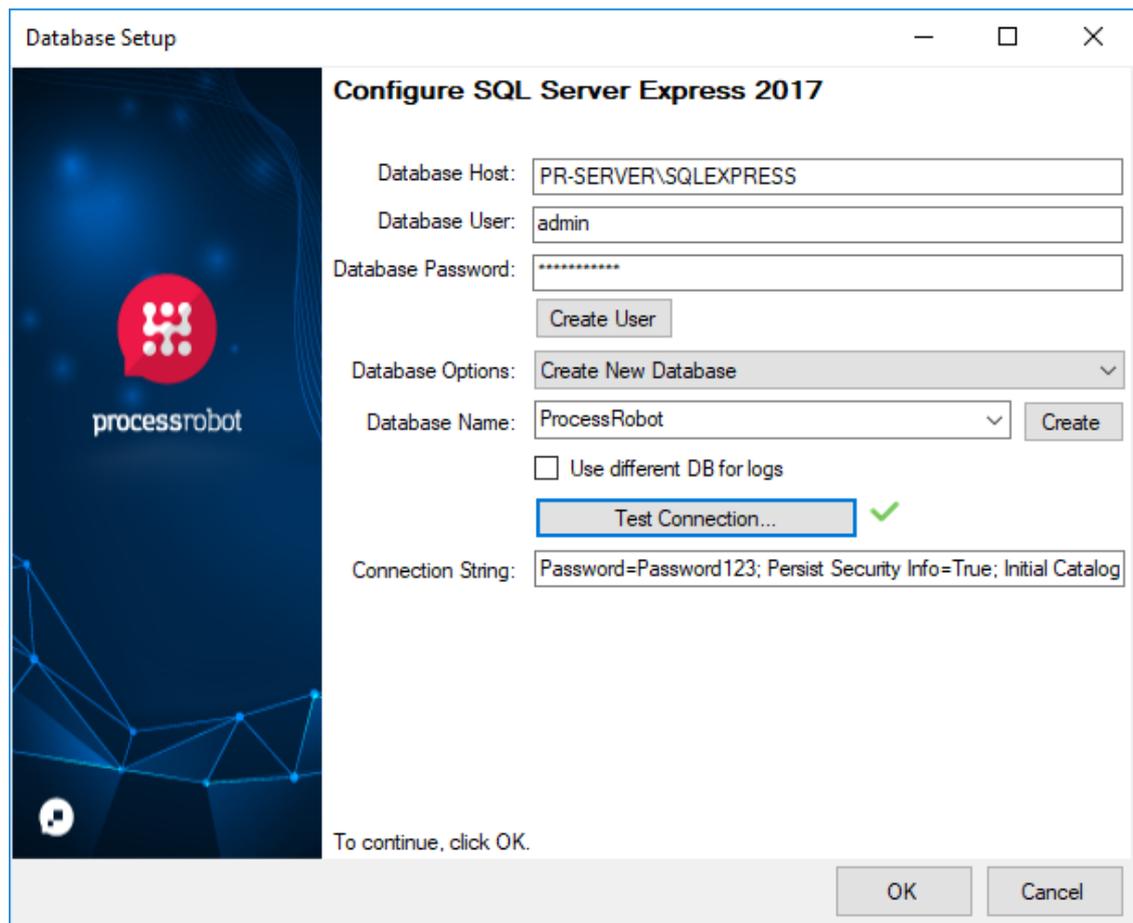




Finally, to test that ProcessRobot can successfully connect to the newly created Database with the new User, enter the Connection String in the format:

Password= <password>; Persist Security Info=True; User ID= <username>; Initial Catalog= <database name>; Data Source= <Server machine name>\<SQL Express instance name>;

Then click "Test Connection".

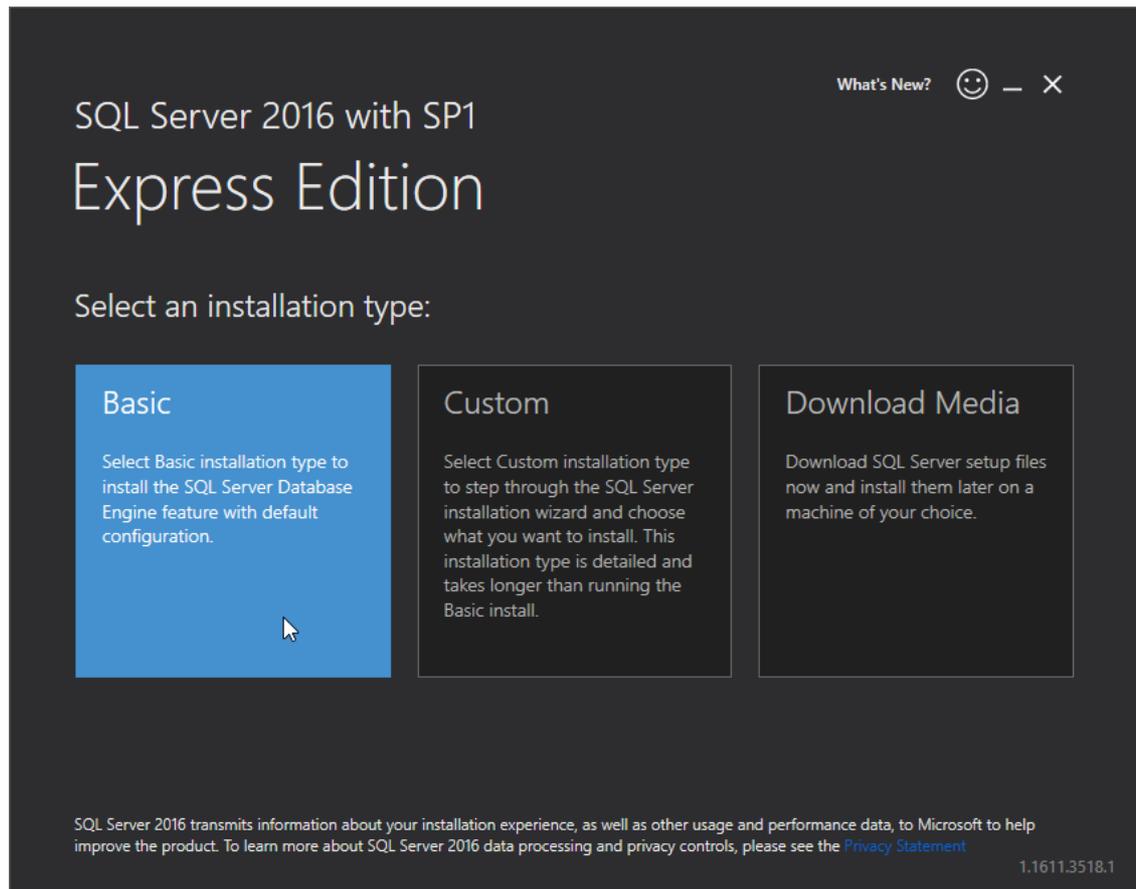


1.5.5 Installing Microsoft SQL Express 2016

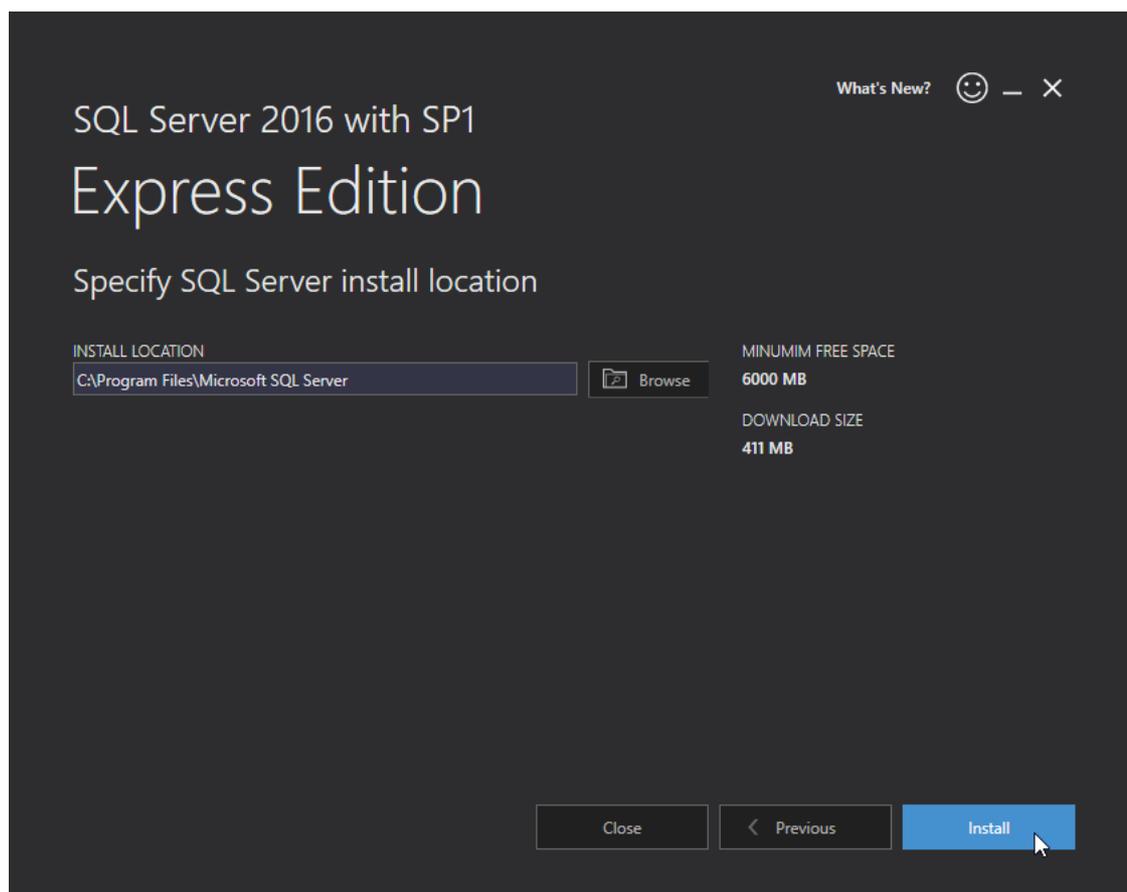
Get the Microsoft SQL Express 2016 by visiting the following URL: [Get the Microsoft SQL Express 2016 by visiting the following URL:](#) or downloading it directly from this [link](#).

The SQL installation is a straight forward process:

1. Select the basic installation type:



2. Accept the License agreement, choose the installation directory and press install.



3. When the installation is complete, choose the option to install the SQL Server Management Studio (this will take you to Microsoft's SSMS product page) or click [here](#) to download it directly from Microsoft's official product page.

... > SQL Server Management Tools (includin... > Download SQL Server Management Stu... ▾

Download SQL Server Management Studio (SSMS)

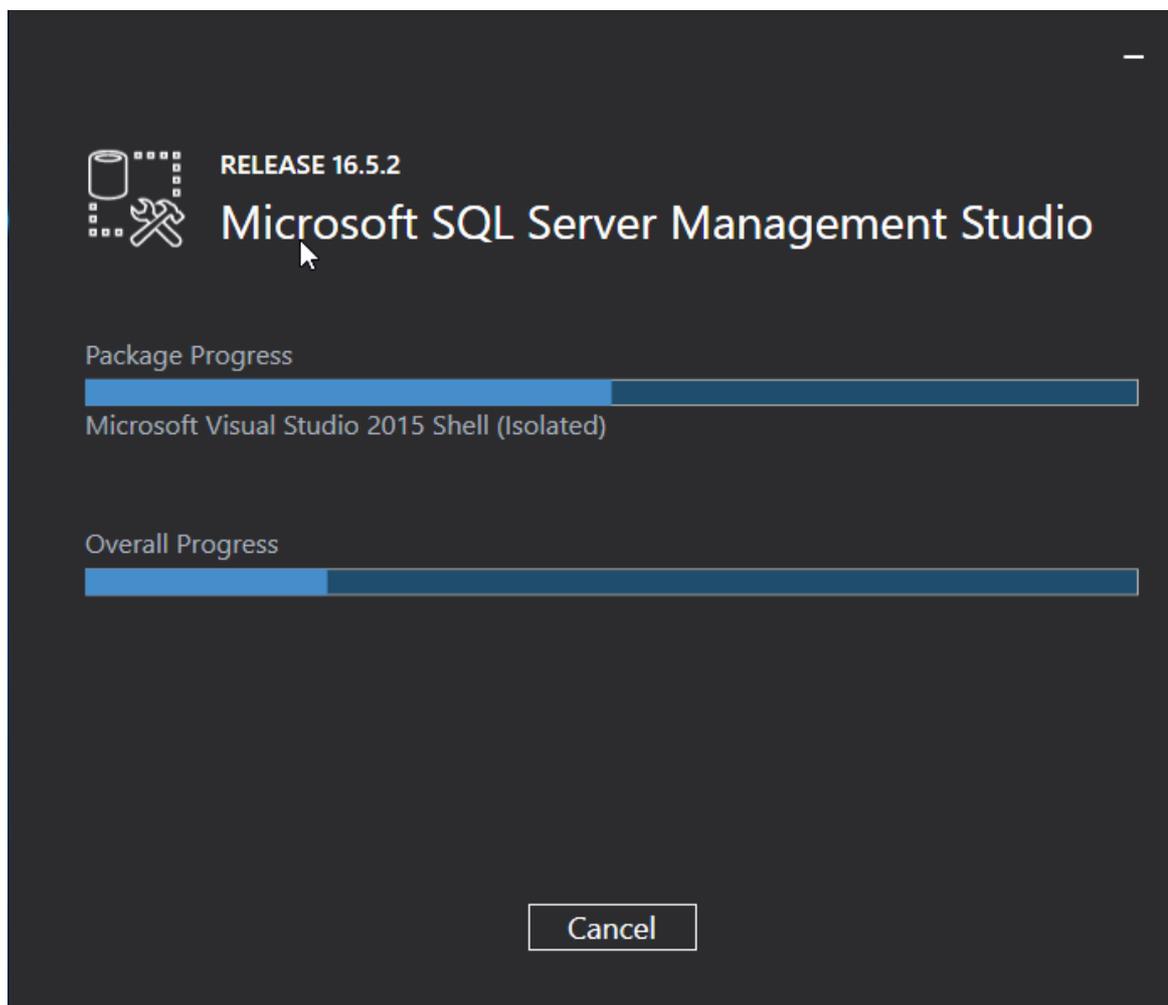
Updated: January 18, 2017

SQL Server Management Studio (SSMS) is an integrated environment for accessing, configuring, managing, administering, and developing all components of SQL Server. SSMS combines a broad group of graphical tools with a number of rich script editors to provide developers and administrators of all skill levels access to SQL Server. This release features improved compatibility with previous versions of SQL Server, a stand-alone web installer, and toast notifications within SSMS when new releases become available.

 Download SQL Server Management Studio (SSMS)	
Download SQL Server Management Studio (16.5.2) 	Current release for production use.
Download SQL Server Management Studio - Release Candidate	Includes support for SQL Server vNext CTP1, and works side-by-side with 16.x, but not recommended for production use.

1.5.6 Installing SQL Server Management Studio

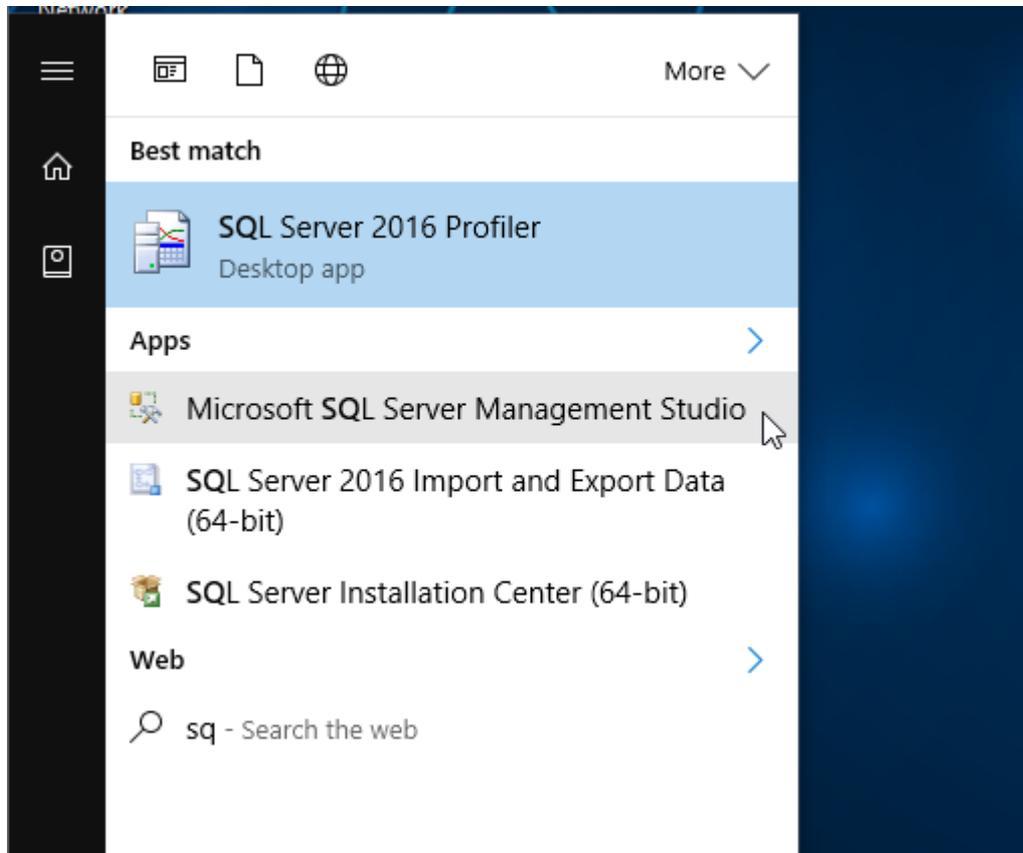
After it finishes downloading run the program and press install.



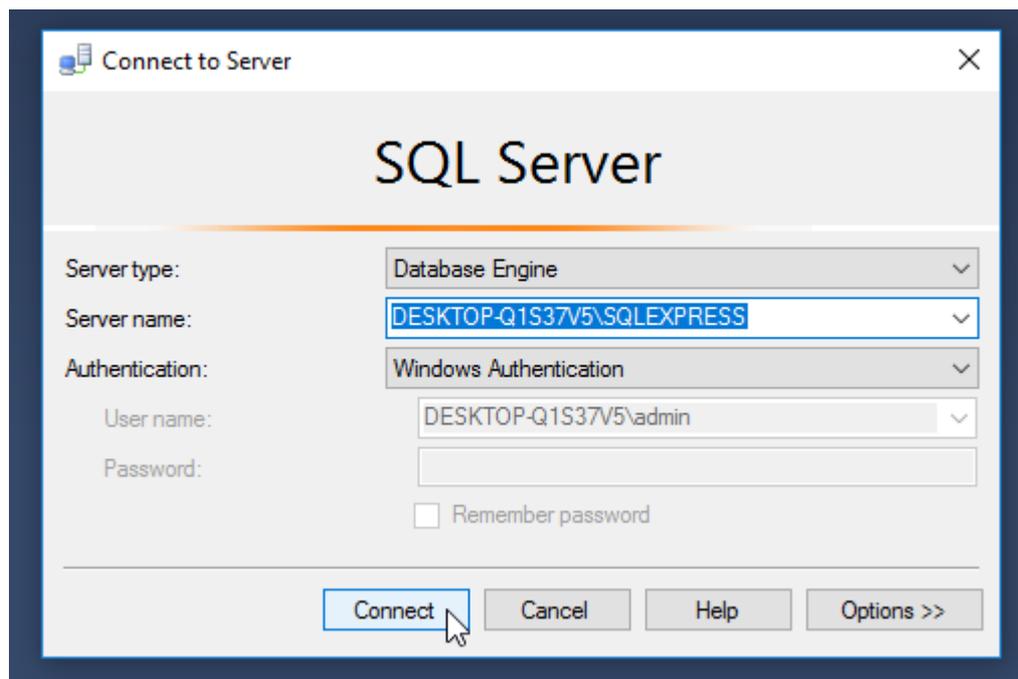
Next we're ready to setup the ProcessRobot database in the SQL Express 2016 server.

1.5.7 Setting up the ProcessRobot Database

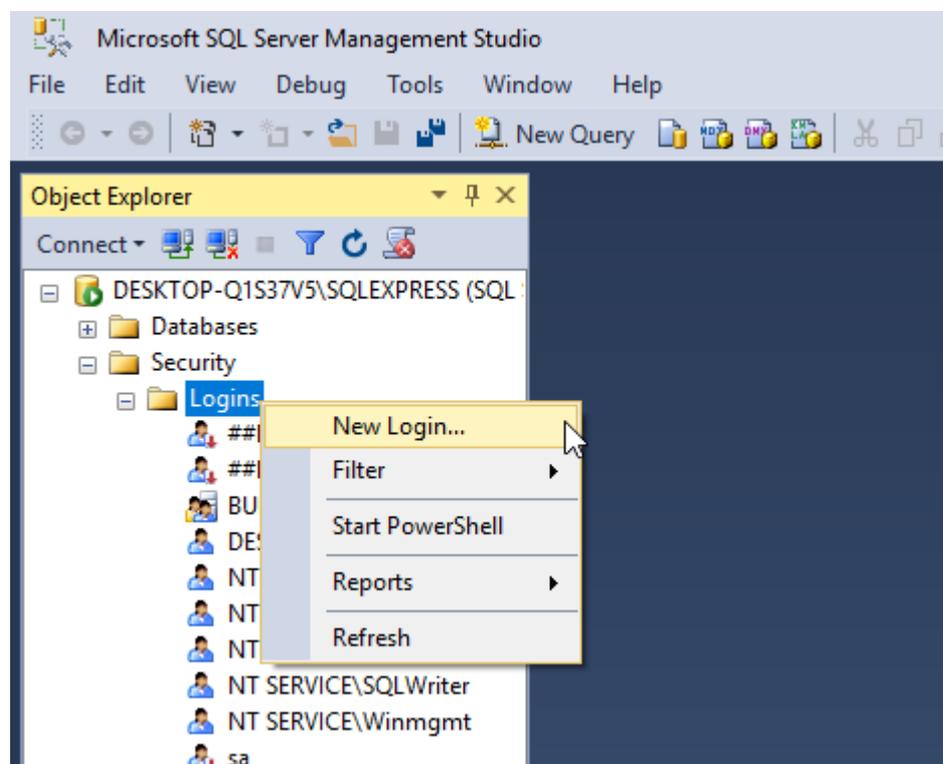
1. Run the SQL Server Management Studio.



2. Connect using the default settings (Windows Authentication).



3. From the object explorer go to Security -> Logins to create a new user for the ProcessRobot server.



4. Create a new user using SQL Server Authentication (a). (For demonstration purposes, we can uncheck the Enforce Password Expiration (b)).

Login - New

Select a page

- General
- Server Roles
- User Mapping
- Securables
- Status

Script Help

Login name: ProcessRobot Search...

Windows authentication

a SQL Server authentication

Password:

Confirm password:

Specify old password

Old password:

Enforce password policy

b Enforce password expiration

User must change password at next login

Mapped to certificate

Mapped to asymmetric key

Map to Credential

Mapped Credentials

Credential	Provider
------------	----------

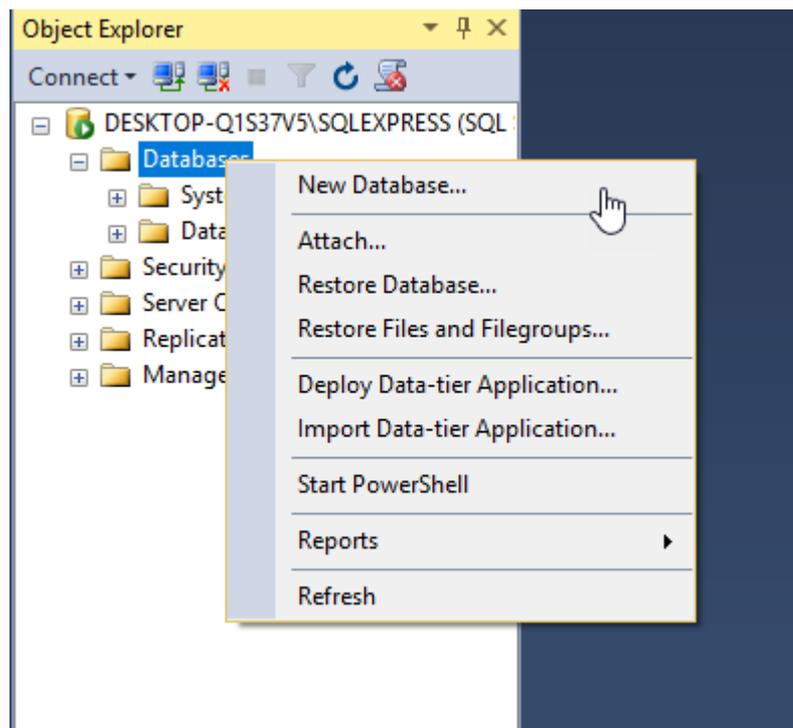
Add Remove

Default database: master

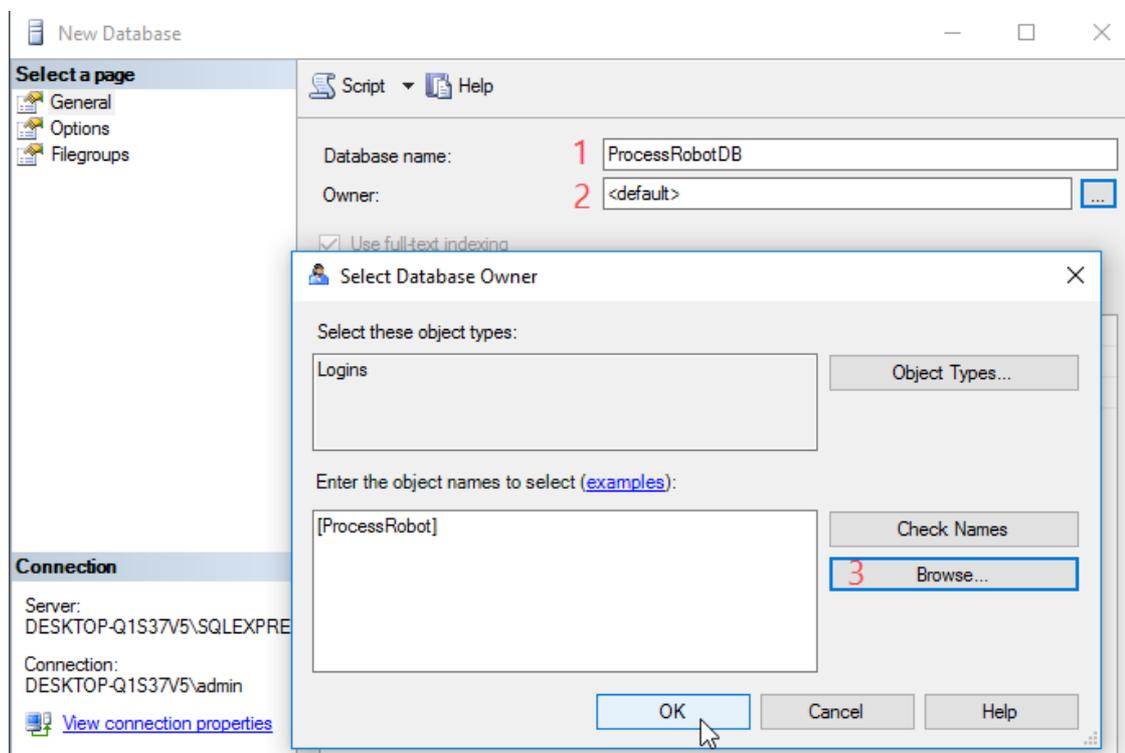
Default language: <default>

OK Cancel

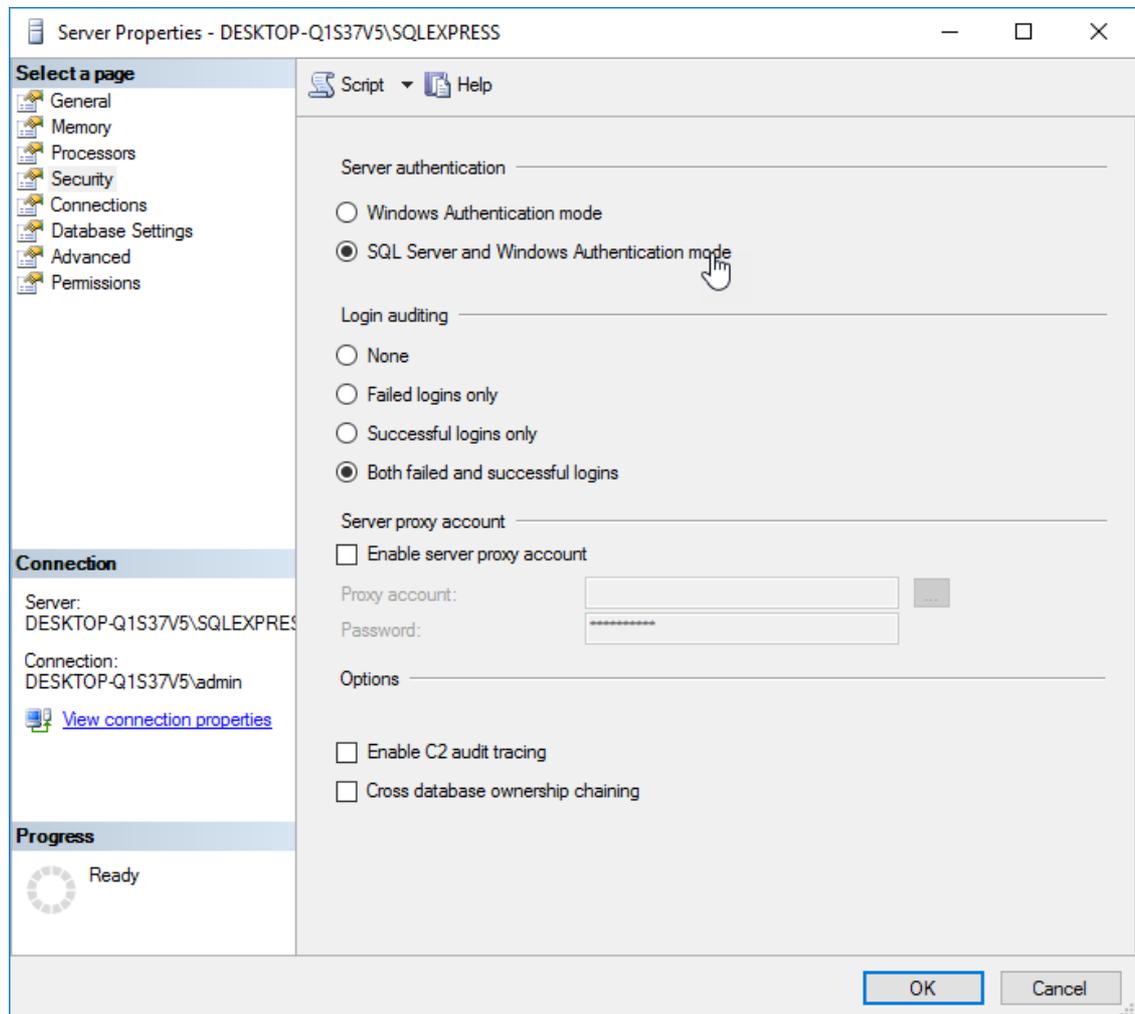
5. Next we will create the database for the ProcessRobot Server. On the object explorer right click on Databases and create a new one.



6. Give a name to the new database and choose as the owner the user we created in the previous steps and press ok.



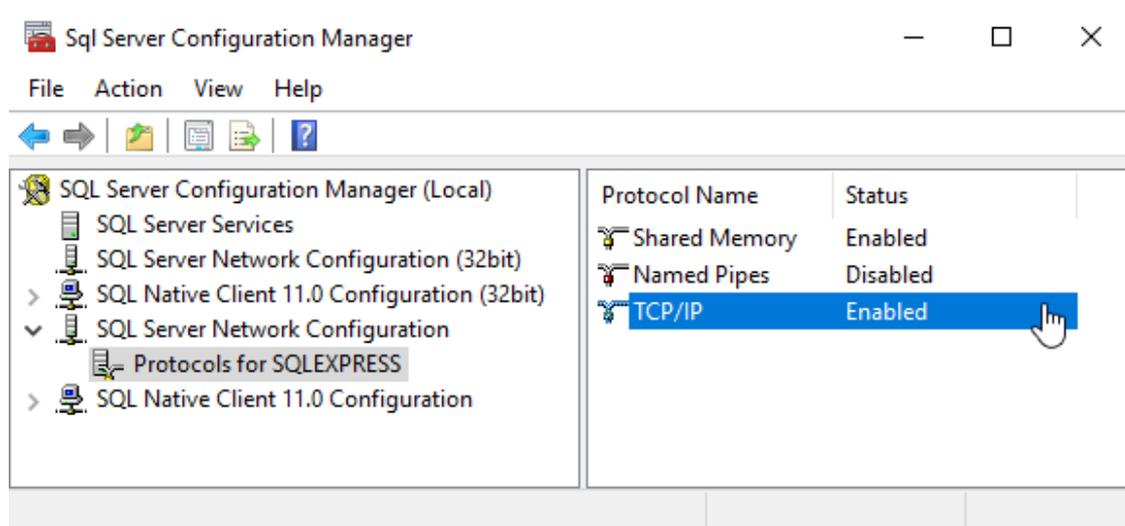
7. We need to make sure our SQL Server allows both Windows and SQL authentication. We do so by right clicking on the Server -> Properties -> Security and selecting "SQL Server and Windows Authentication mode".



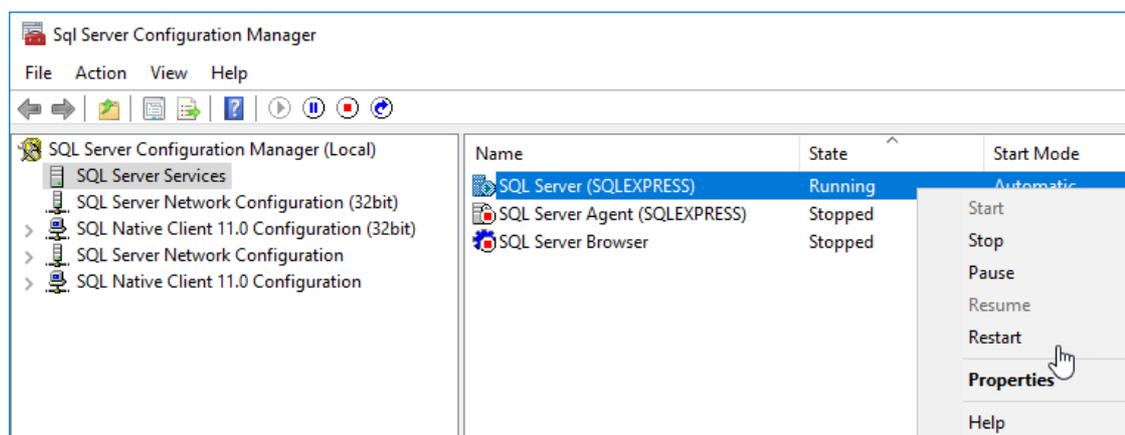
8. We now need to restart the SQL Server service to make sure our changes take effect.

Shell Hardware Detection	Provides no...	Running	Automatic	Local System...
Smart Card	Manages ac...		Disabled	Local Service
Smart Card Device Enumera...	Creates soft...		Manual (Trig...	Local System...
Smart Card Removal Policy	Allows the s...		Manual	Local System...
SNMP Trap	Receives tra...		Manual	Local Service
Software Protection	Enables the ...		Automatic (D...	Network S...
Spot Verifier	Verifies pote...		Manual (Trig...	Local System...
SQL Server (SQLEXPRESS)	Provides sto...	Running	Automatic	Local System...
SQL Server Agent (SQLEXP...	Executes jo...		Disabled	Local Service
SQL Server Browser	Provides SQ...		Disabled	Local Service
SQL Server CEIP service (SQ...	CEIP service...	Running	Automatic	Local Service
SQL Server VSS Writer	Provides th...	Running	Automatic	Local Service
SSDP Discovery	Discovers n...	Running	Manual	Local Service
State Repository Service	Provides re...	Running	Manual	Local Service

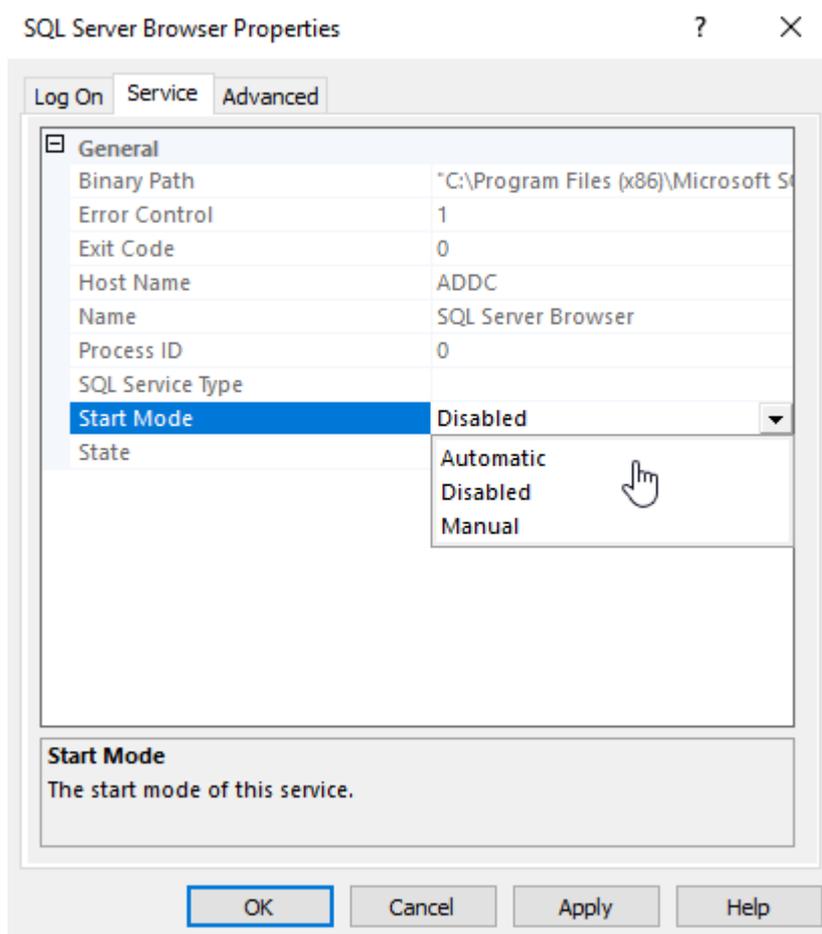
9. Now we just need to allow the TCP protocol for SQLEXPRESS in the SQL Server Network Configuration. To do so navigate to `C:\Windows\SysWOW64` and run `SQLServerManager13.msc`.



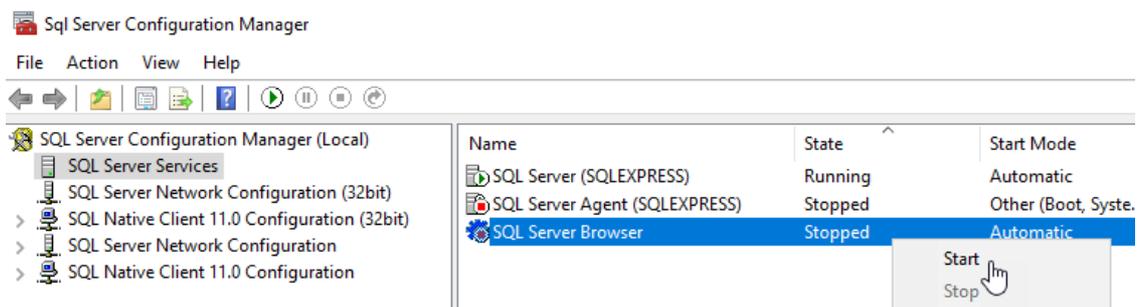
10. After enabling TCP/IP go to SQL Server Services and restart the SQL Server service.



11. Lastly, let's configure the SQL Browser service to automatically start. Right click on the SQL Server Browser service and select properties. Then from the Service tab select the "Start Mode" and select Automatic start. When you finish press OK.



12. Then right click on the service and start it:



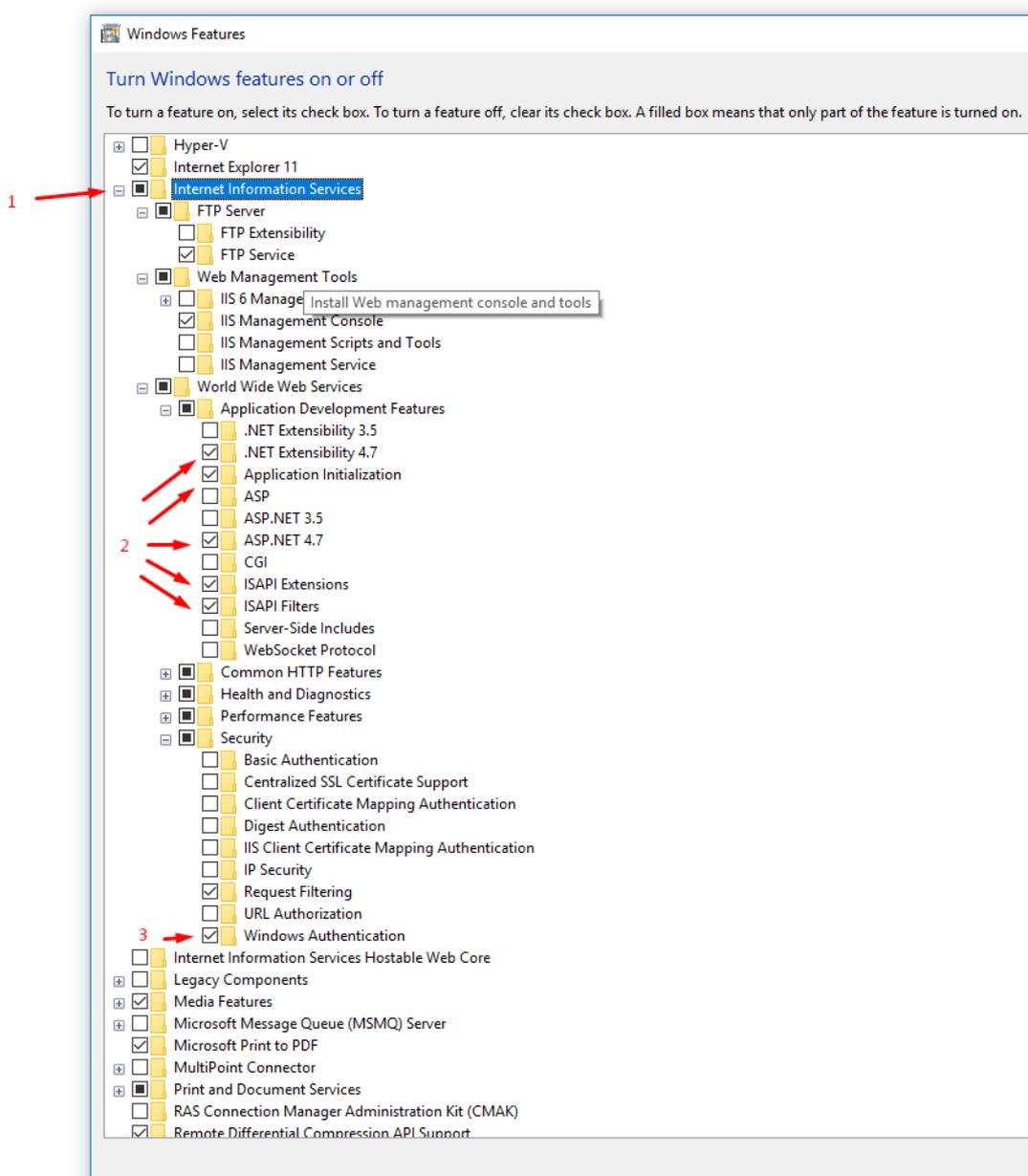
We are now ready to install the ProcessRobot Server. ?

1.5.8 Installing the Web Console

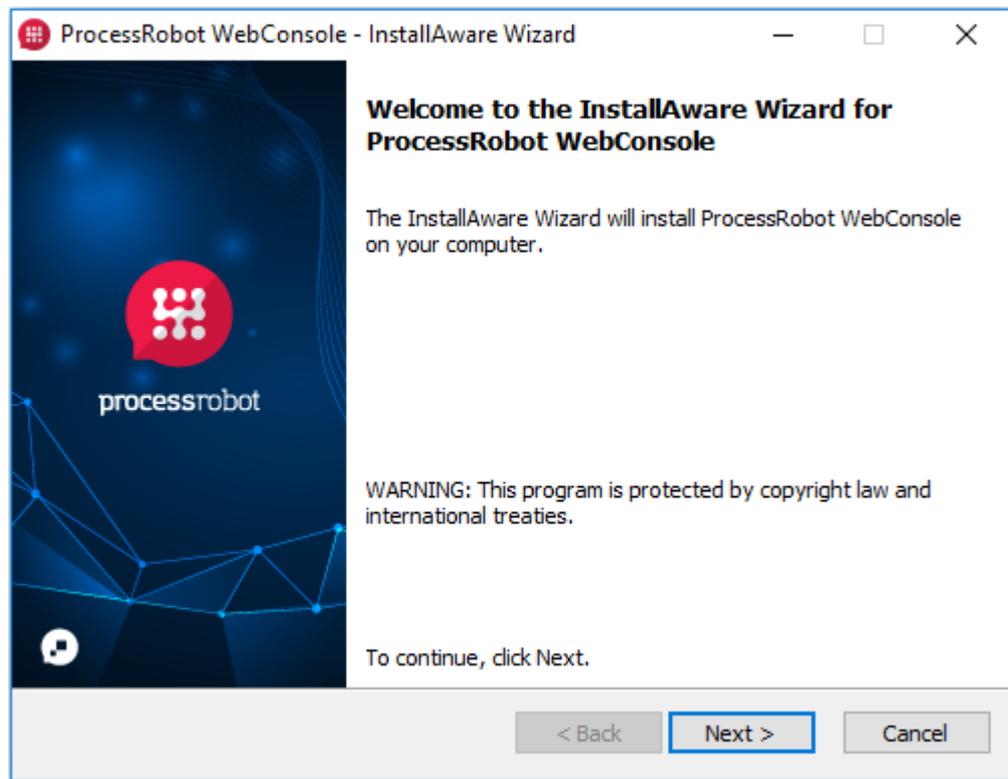
Now that we have installed Sidebots and Solobots we are ready to setup the Web Console. The Web Console is essentially a simplified version of the Control Desk that is accessible through any browser.

Pre-requisite:

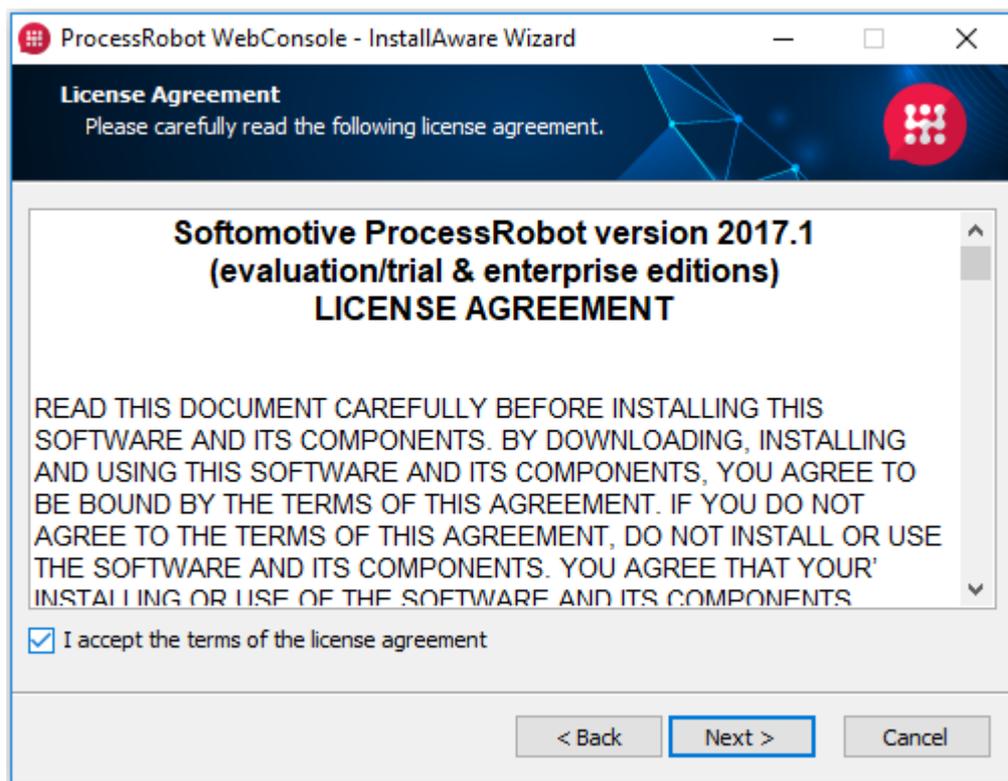
-If the machine is a Windows Server, then in the "Windows Features" window, go to "Add Roles and Features Wizard" ->"Features" and enable the options below:



1. Run the ProceRobot WebConsole Installer to begin the installation:



2. Read carefully the license agreement:



3. Enter the ProcessRobot Server information. You can enter either it's Host Name, FDQN, or IP address:

ProcessRobot WebConsole - InstallAware Wizard

ProcessRobot Server Address
Give default server address

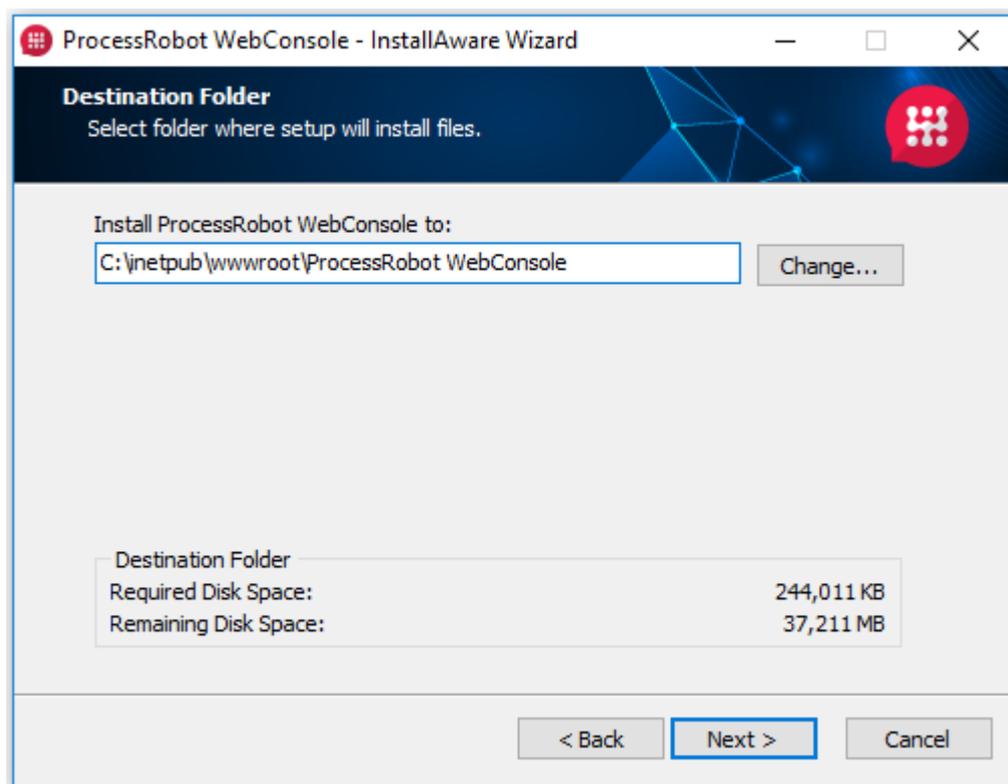
Setup will create a configuration file containing the server's IP address and port

Address

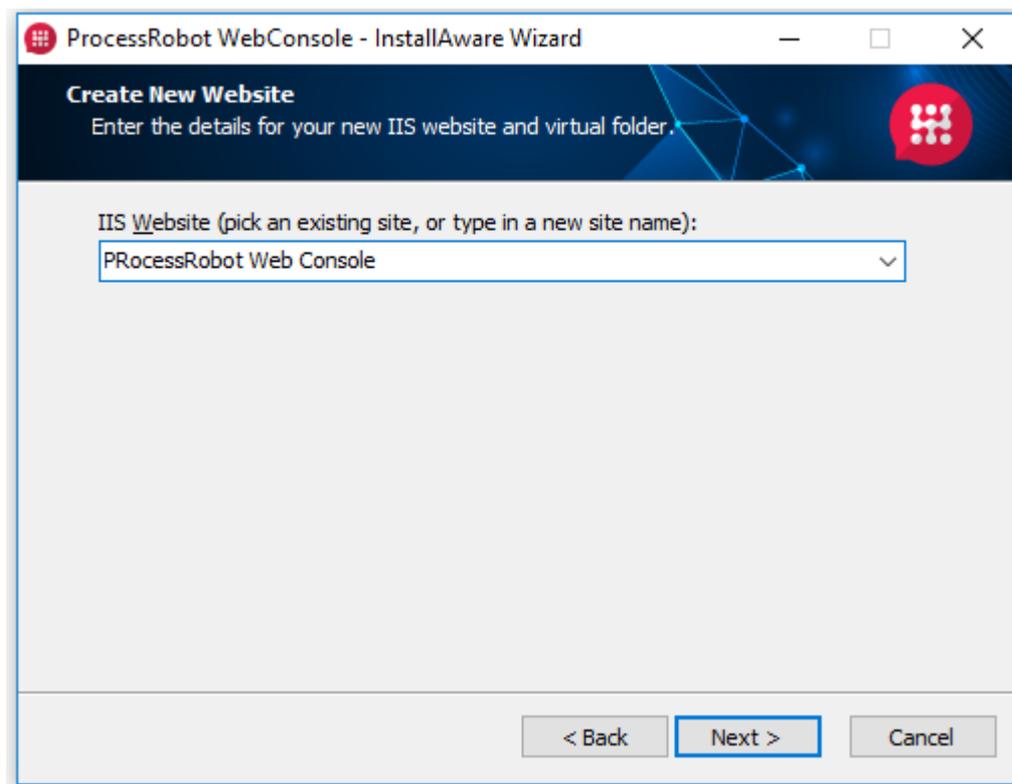
Port

< Back Next > Cancel

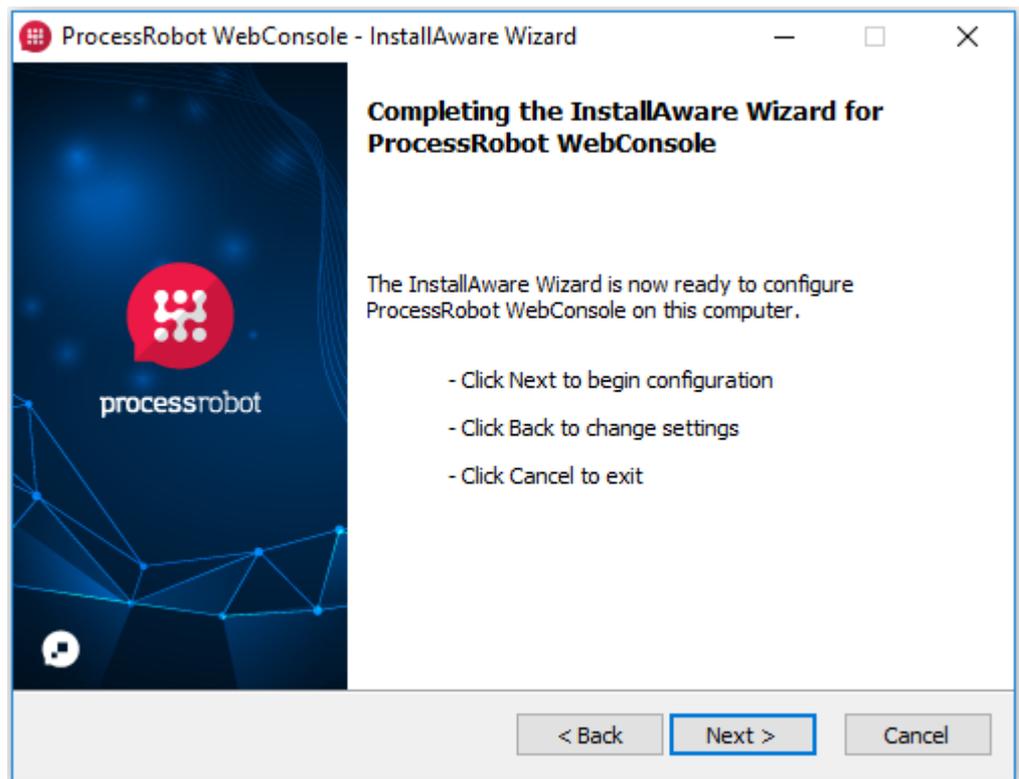
4. Choose the installation destination:



5. Select the IIS Website:



6. Press next to begin the installation:



7. If you have purchased the Web Console Module you will need to assign access rights to users. You can do so by going to "Settings" -> "License" -> "Manage Web Console Users" like so:

The screenshot shows the ProcessRobot Control Desk interface. The main content area displays the License Overview section, which includes the following information:

- Licensed To: Softomotive
- Company: Softomotive
- Control Desks: 5
- Process Studios: 5
- SideBots: 5
- SoloBots: 5
- Dashboard Viewers: 5
- Dashboard Designers: 5
- Web Consoles: 5

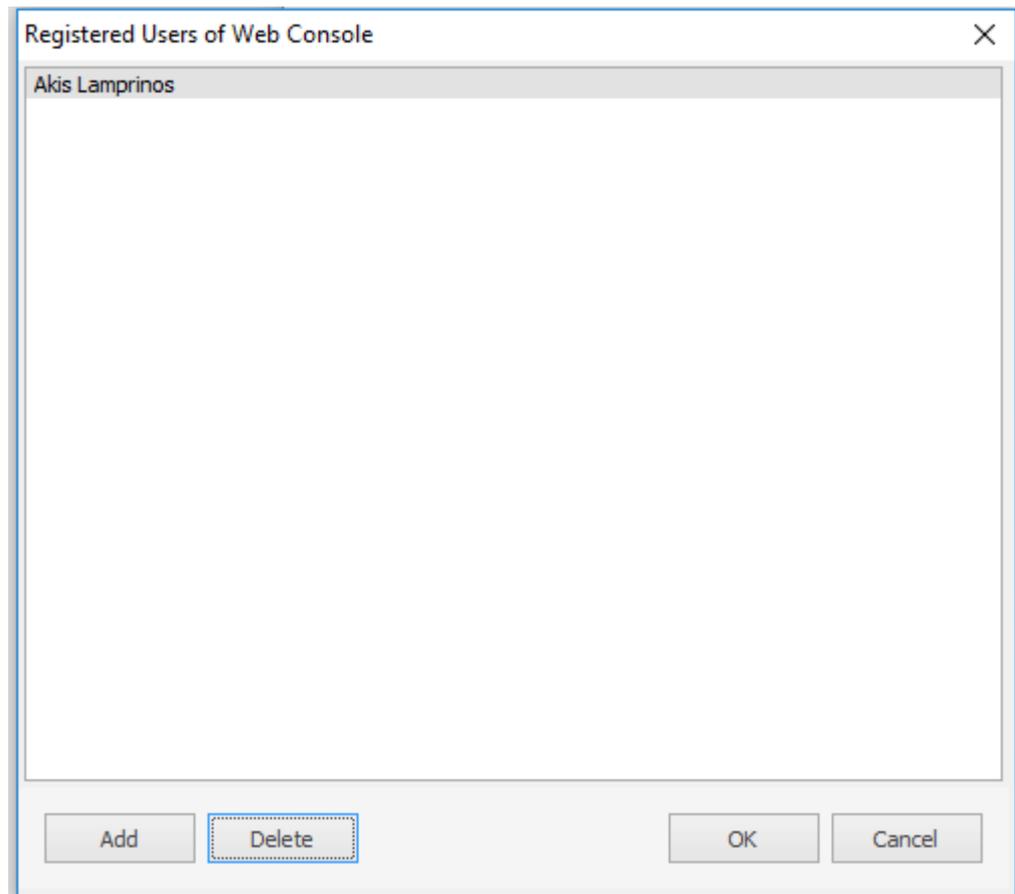
Below this information is a table with the following data:

Component Type	Quantity	Expiration Date
Control Desk	5	11-Aug-18
Process Studio	5	11-Aug-18
SideBot	5	11-Aug-18
SoloBot	5	11-Aug-18
Dashboard Viewer	5	11-Aug-18
Dashboard Designer	5	11-Aug-18
Web Console	5	11-Aug-18

At the bottom of the license overview section, there are two buttons: "Install License File" and "Manage Web Console Users".

The user name "User: Akis Lamprinos" is visible in the bottom right corner of the interface.

8. Add the users that will have access to Web Console:



1.5.9 Configuring Elastic Search

The process of configuring Elastic Search to work with ProcessRobot is described below

Step 1: Open ProcessRobot.Server.exe.config from the installation folder, using a text editor.

Step 2: Remove the comments in the sections marked below.

```
<log4net debug="false">
  <root>
    <level value="INFO" />
    <appender-ref ref="sqlServer" />
    <!-- uncomment to enable uploading logs to ElasticSearch -->
    <!--<appender-ref ref="LogElasticSearchAppender"/>-->
  </root>
  <logger name="Quartz" additivity="false" />
  <!-- Uncommnet to enable uploading kpi to ElasticSearch -->
  <!--<logger name="KPILogger" additivity="false">
    <level value="INFO" />
    <appender-ref ref="KPIElasticSearchAppender" />
  </-->
</log4net>
```

```

        </logger>-->
    <!-- Wiki of log4net.ElasticSearch
https://github.com/jptoto/log4net.ElasticSearch/wiki -->
    <!-- set bufferSize to 0 to upload to ElasticSearch immediately -->
    <!-- Appender to Log to ElasticSearch -->

        <!--<appender name="LogElasticSearchAppender"
type="log4net.ElasticSearch.ElasticSearchAppender, log4net.ElasticSearch.Net40">
            <connectionString
value
="Scheme=https;User=username;Pwd=password;Server=localhost;Index=prlog;Port=9200;ro
lling=false"/>
                <eventType value="log"/>
                <lossy value="false" />
                <evaluator type="log4net.Core.LevelEvaluator">
                    <threshold value="ERROR" />
                </evaluator>

                <bufferSize value="100" />
            </appender>-->
    <!-- Appender to send KPI Entries to ElasticSearch -->
    <!--<appender name="KPIElasticSearchAppender"
type="log4net.ElasticSearch.ElasticSearchAppender, log4net.ElasticSearch.Net40">
            <connectionString
value
="Scheme=https;User=username;Pwd=password;Server=localhost;Index=prkpi;Port=9200;ro
lling=false"/>
                <eventType value="kpi"/>
                <lossy value="false" />
                <evaluator type="log4net.Core.LevelEvaluator">
                    <threshold value="ERROR" />
                </evaluator>

                <bufferSize value="100" />
            </appender>-->

```

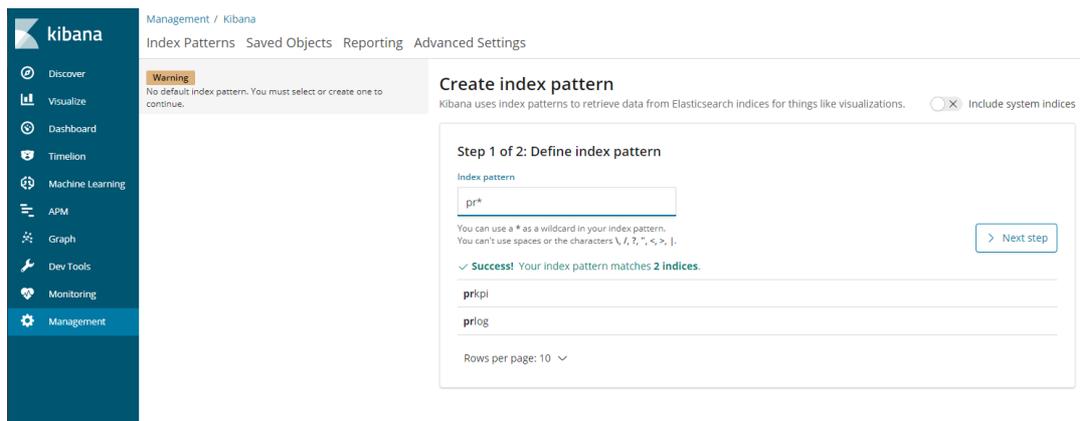
Step 3: The document should now look like the section above. Now replace the highlighted values with the ones that apply for your ElasticSearch installation.

- **Scheme:** http or https depending on the server the installation is on
- **User:** ElasticSearch Username.
- **Pwd:** ElasticSearch Password.
- **Server:** ElasticSearch Server Address (When on cluster do not use IP).
- **Index:** The indices used to categorize records to Elastic and Kibana.
- **Port:** ElasticSearch Port.
- **bufferSize:** The amount of records that gets buffered before sent to Elastic (Set 0 for instant upload).

Step 4: After changing the above values, save the .config file as Administrator and restart ProcessRobot.Server Service.

Step 5: On Kibana's Monitoring Page, turn on Monitoring.

Step 6: Use ProcessRobot to create logs or KPIs, which will automatically upload to Elastic, and navigate on the Kibana's Management Page to create the Index Patterns matching the prkpi and prlog indices respectively.



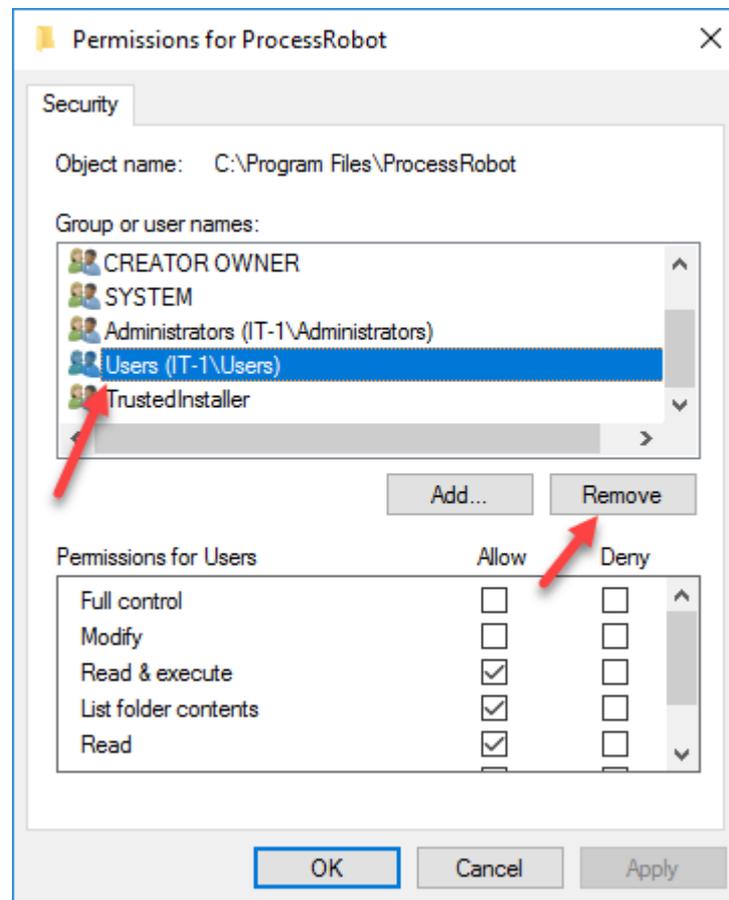
1.5.10 Optional: Securing ProcessRobot Directory from unauthorized access

Please note that using wrong NTFS settings could affect your ability to access folders and the ability of software or the OS to operate properly. Proceed with your own risk. For more information about NTFS permissions consult Microsoft's website.

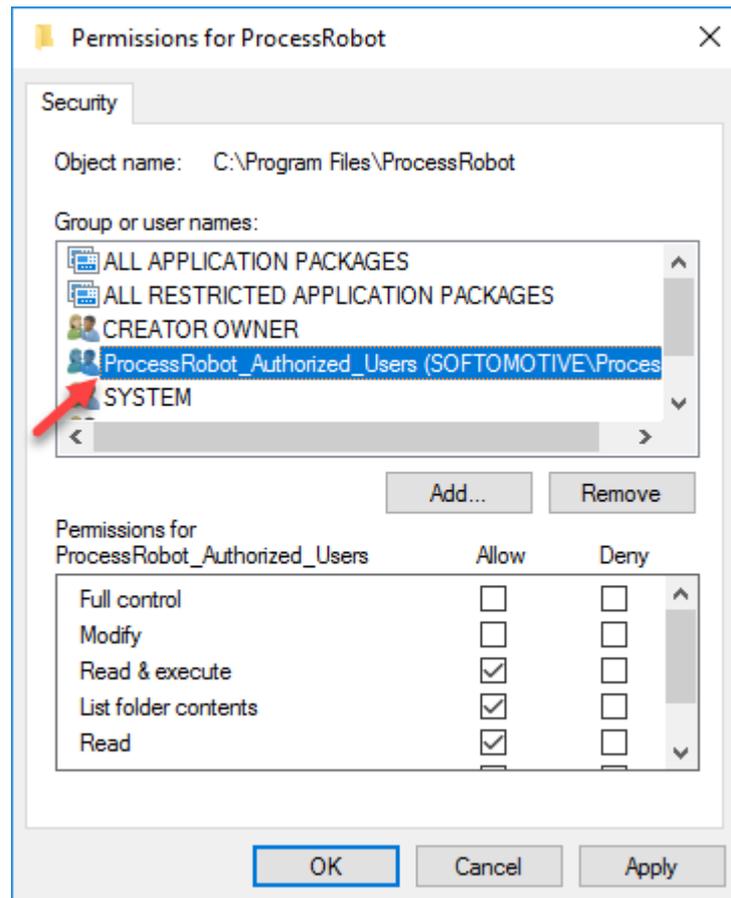
As an optional step we could secure the ProcessRobot Directory from unauthorized users using NTFS permissions.

1. Go to C:\Program Files and locate the ProcessRobot directory
2. Right click on the folder, and choose Properties

3. Select the Security tab and press Edit
4. Remove the Users group (or any other group you don't wish to have access to it)



5. Add the user group you wish to have access



1.5.11 FAQ

1. Why is Active Directory required?

Active directory is used for user & computer authentication, as well as resolving host names (via DNS).

2. Do I need to create any users or groups in Active Directory?

ProcessRobot uses Active Directory to provide user authentication as well as attach software permissions to its various components. For example, for Sidebot we need to define the Active Directory user that the Sidebot will work along with. For Solobot we need to define an active directory computer in which we allow it to start as well as user credentials that it will use if we require it to auto login. We can then define additional AD users who will be able to perform specific tasks in ProcessRobot, like creating processes, viewing information from Dashboard etc.

3. What are the ProcessRobot Components?

Component Name	Type	Multiple Instances	Dependencies
ProcessRobot Server	Service	No	AD, Microsoft SQL Server
Control Desk	GUI - User Application	Yes – Each on its own machine	AD, ProcessRobot Server
Process Studio	GUI – User Application	Yes – Each on its own machine	AD, ProcessRobot Server
Dashboard Viewer	GUI – User Application	Yes – Each on its own machine	AD, ProcessRobot Server, MS SQL
Dashboard Designer	GUI – User Application	Yes – Each on its own machine	AD, ProcessRobot Server, MS SQL
Sidebot	GUI - User Application	Yes – Each on its own machine	AD, ProcessRobot Server
Solobot	Service	Yes – Each on its own machine	AD, ProcessRobot Server

4. On which operating systems can ProcessRobot Server be installed on?

ProcessRobot Server can be installed on Windows 2008 or later. All the other components (Control Desk, Process Studio, Dashboard Viewer, Dashboard Designer, Sidebot and Solobot) can be installed on Windows 7 or later.

5. What are the minimum hardware requirements and what is the least amount of machines / vms for a demo installation?

The absolute minimum setup scenario is 2 Virtual Machines. One will need to run Active Directory, MS SQL Server Express, ProcessRobot Server, Control Desk, Process Studio, Dashboard Viewer, Dashboard Designer and Sidebot. For this virtual machine we recommend allocating at least 4GB of RAM and 40GB of storage (in total). The other virtual machine will just hold Solobot. For this virtual machine we recommend allocating at least 2GB of RAM and 30GB of storage (in total). So theoretically we can have a working demo in any machine that can allocate 6GB of memory and 70GB of hard drive space to 2 Virtual Machines. Please keep in mind that this will not work in production as each component will most probably need its own machine for control and permissions delegation. Also, depending on the amount and type of workload the system requirements can change dramatically. You can find additional info in the [Requirements for Components](#)^[22] topic.

6. Which user can create / remove users from Control Desk

Only users with the Administrator role can add or remove users access from control desk

7. How are user credentials stored in the Database?

User credentials are stored in the database using AES-256 bit encryption.

1.5.12 Install MODI

To make the OCR Actions usable in runtime, please, install MODI for Microsoft Office 2007+.

This document provides methods you can follow to install MODI (Microsoft Office Document Imaging) on your computer. It also describes the alternative methods that you can use to regain the functionalities of certain MODI features.

To install Microsoft Office Document Imaging (MODI) for use with Microsoft Office 2010, use one of the following methods.

Method 1: Download and install MDI to TIFF File Converter

To download and install MDI to TIFF File Converter, go to the following Microsoft website:
<http://www.microsoft.com/en-us/download/details.aspx?id=30328>

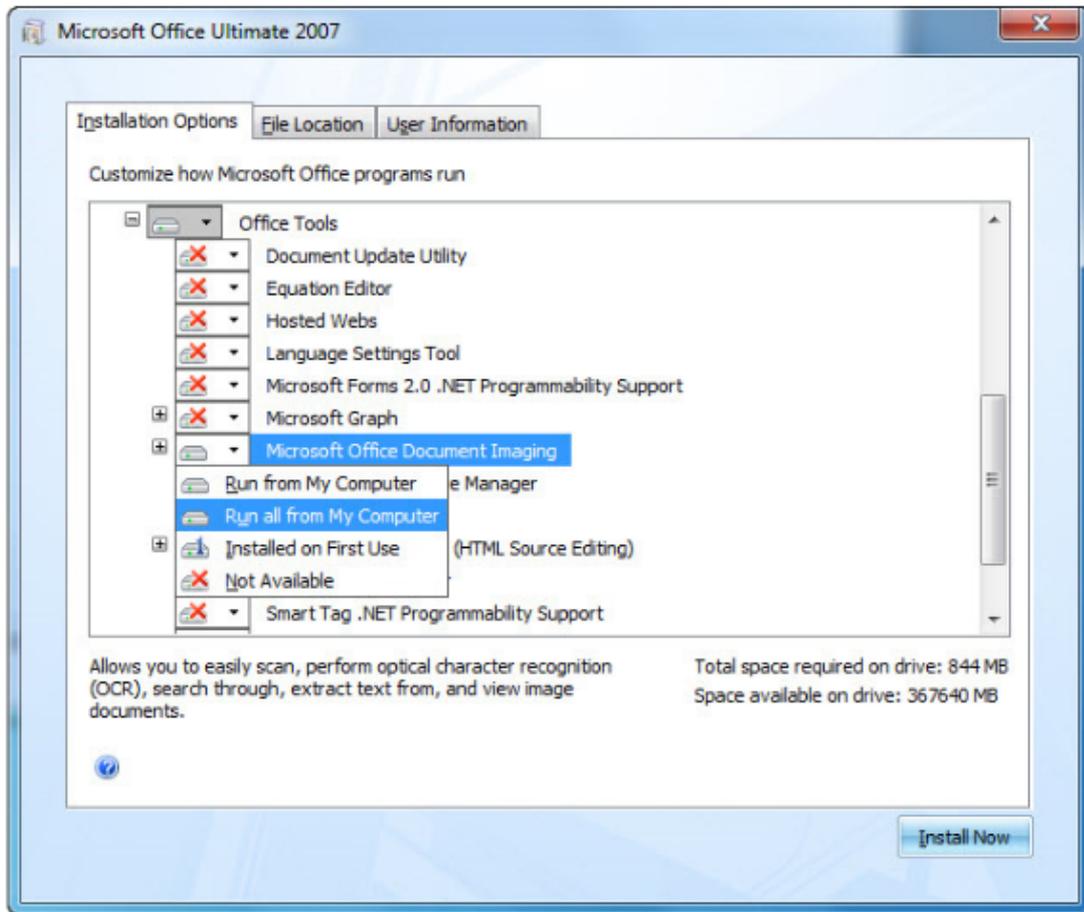
To start Microsoft Office Document Imaging, follow these steps:

1. Click **Start**, and then select **All Programs**.
2. Click **Microsoft Office**, and then select **Microsoft Office Tools**.
3. Click **Microsoft Office Document Imaging**.

Method 2: Install MODI with the 2007 Office System media

To use MODI in the 2007 Office System together with Office 2010, follow these steps:

1. Start the 2007 Office System setup process.
2. Review, and then accept the software license.
3. Click **Customize** to open the installation options.
4. On the **Installation Options tab**, click the down arrow and select **Not Available** for all sections.
5. Expand **Office Tools**.
6. Click the down arrow for the **Microsoft Office Document Imaging** option, and then select **Run all from my computer**.
7. Click **Install Now**, and then click **Close**.



Screenshot displaying the location of MODI during the 2007 Office system installation.

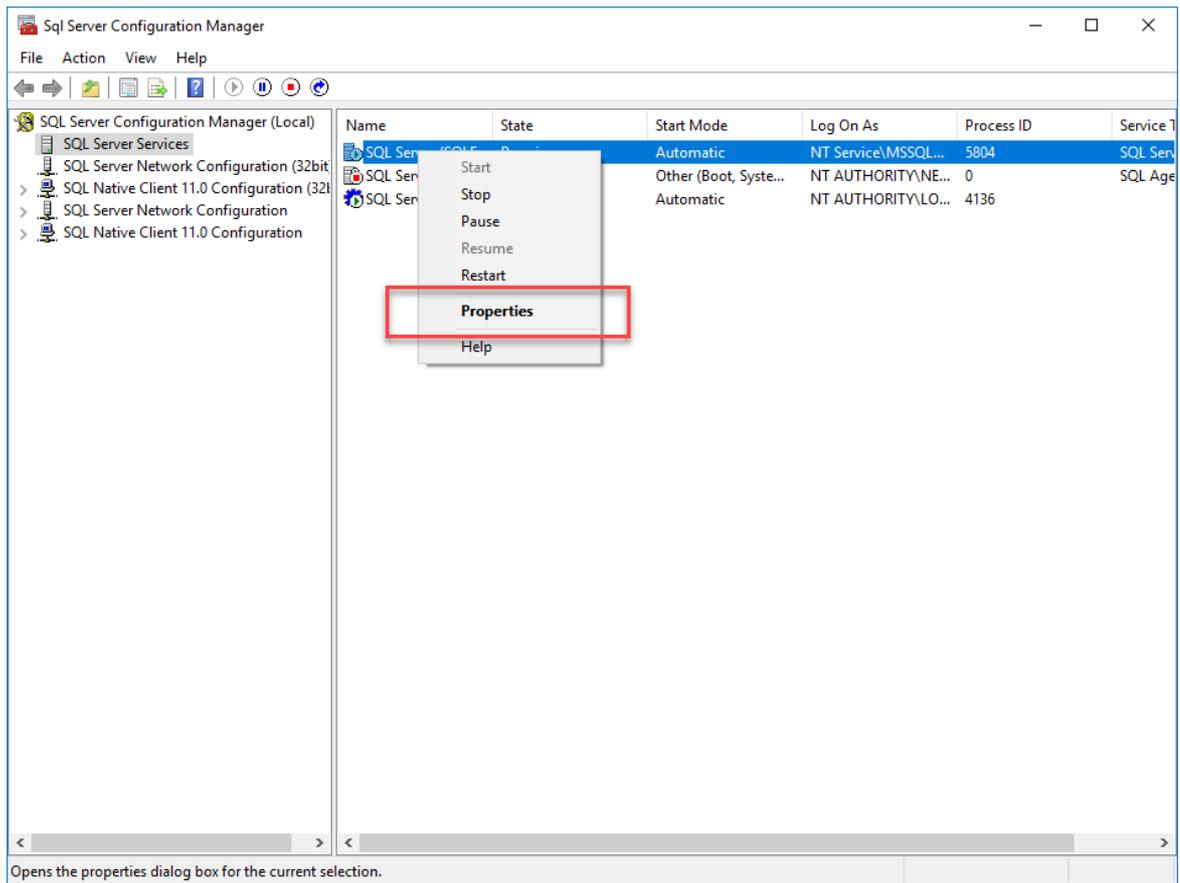
To start Microsoft Office Document Imaging, follow these steps:

1. Click **Start**, and then select **All Programs**.
2. Click **Microsoft Office**, and then select **Microsoft Office Tools**.
3. Click **Microsoft Office Document Imaging**.

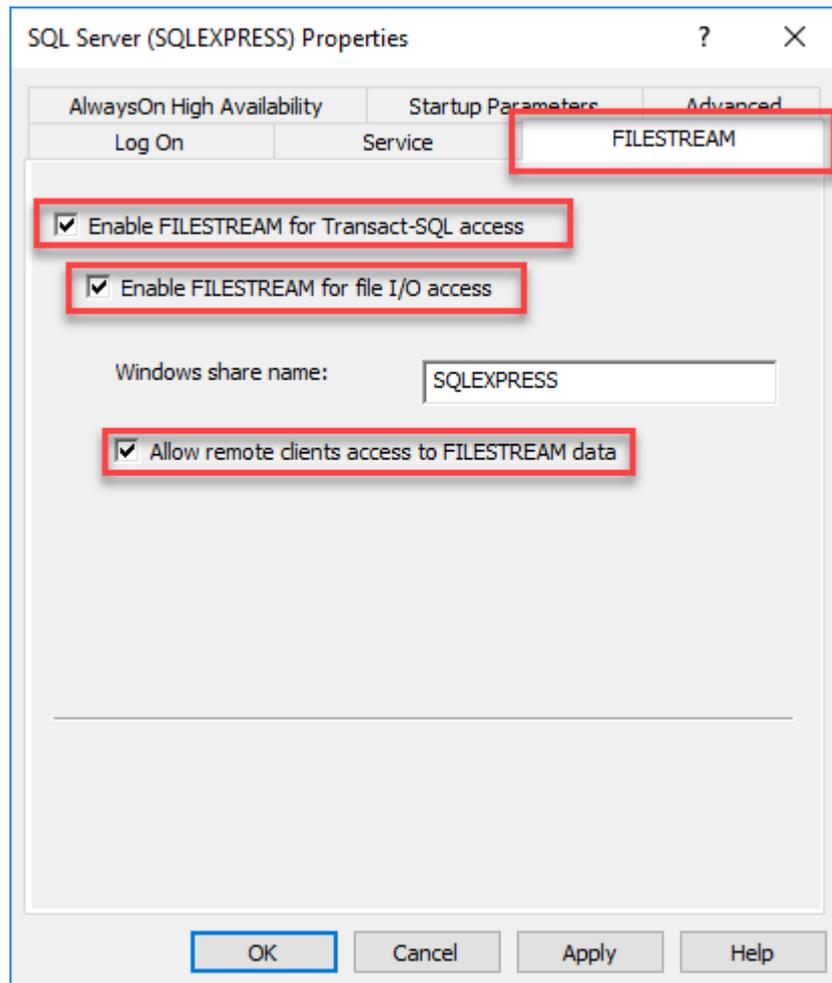
1.5.13 Enabling the Filestream

To enable the Filestream, please follow the procedure described below.

1. Navigate to `C:\Windows\SysWOW64` and run `SQLServerManager13.msc`.
2. Right Click on the SQL Server which is currently running, and open the service properties.



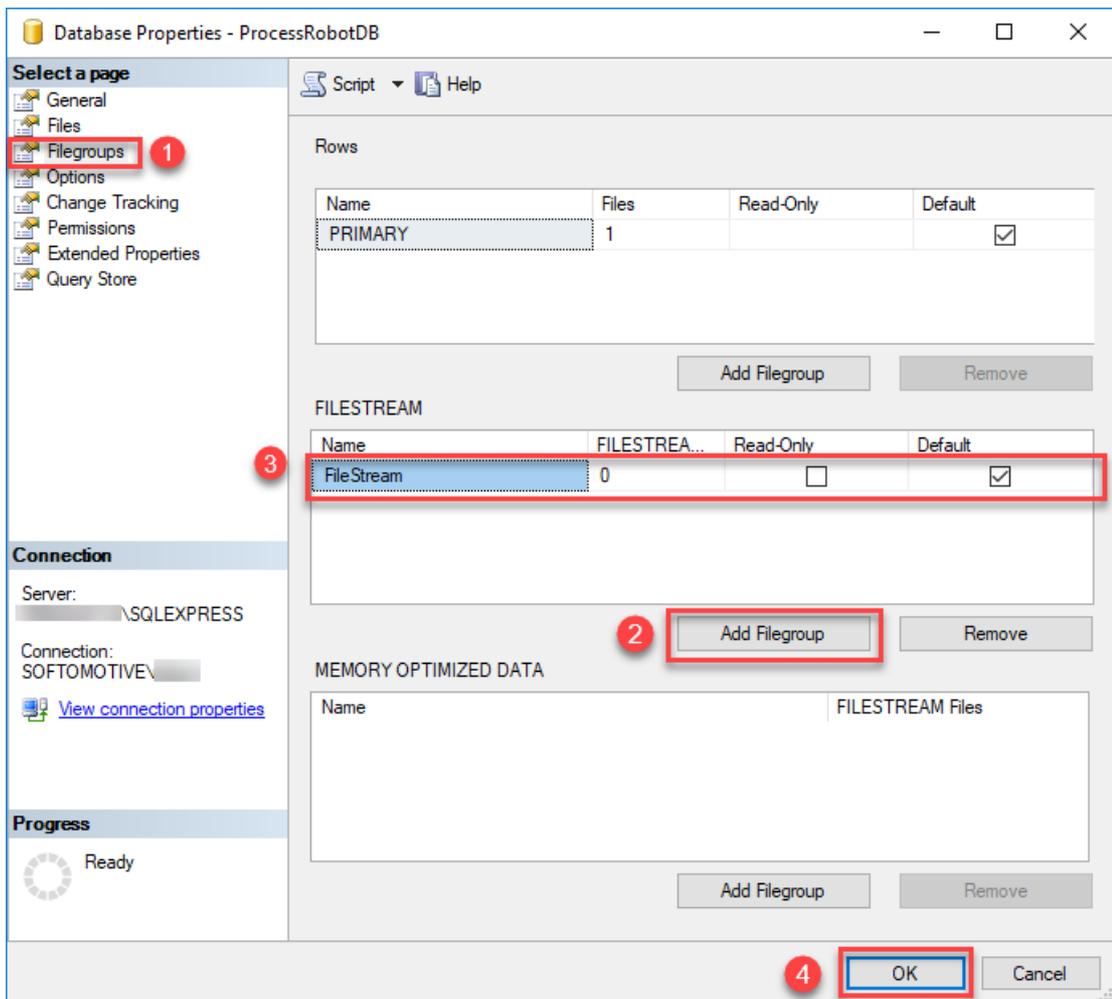
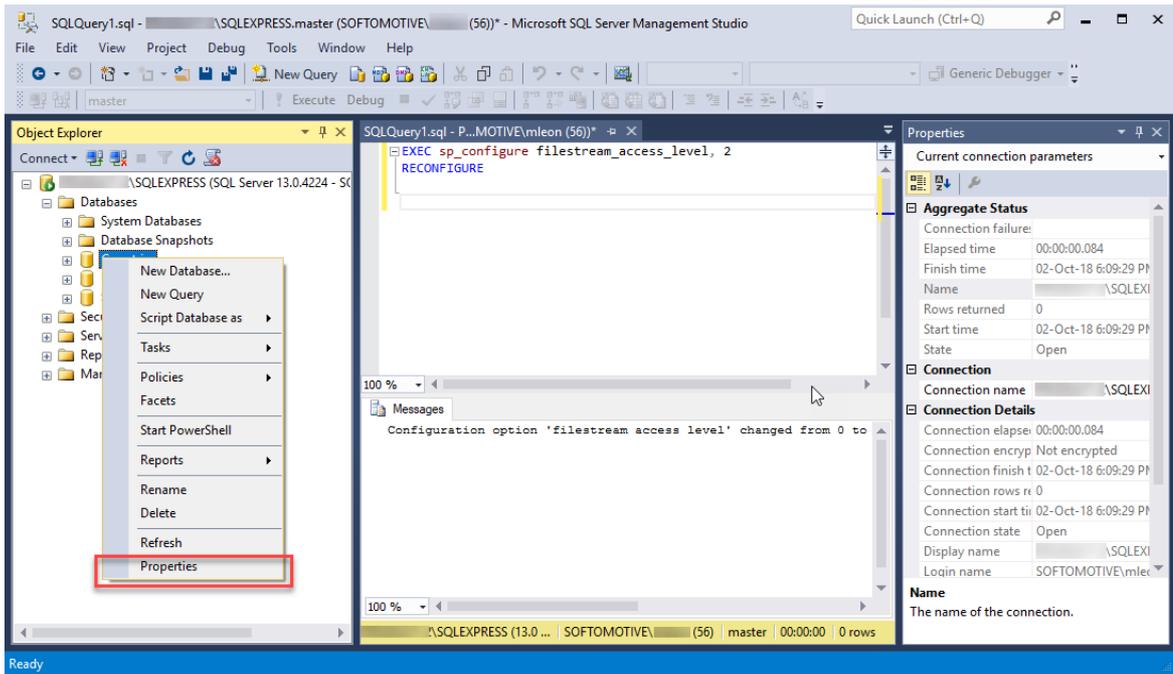
3. In the Properties page, select the "FILESTREAM" tab and enable the FILESTREAM for Transact-SQL access, I/O access and allow remote clients to access the FILESTREAM data.



- Next, navigate to the SQL Server Management Studio and execute the following query.

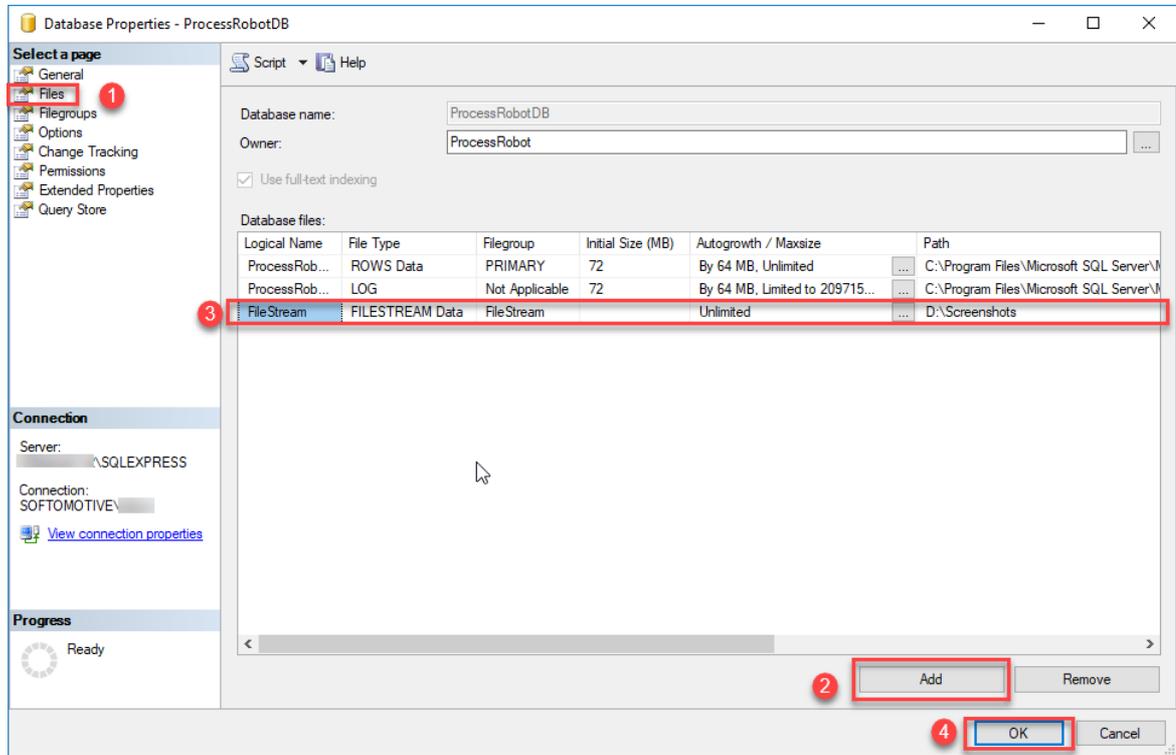
```
EXEC sp_configure filestream_access_level, 2  
RECONFIGURE
```

- Once the query completes successfully, restart the SQL Server Service and then navigate to the SQL Server Management Studio and open the properties of the Database.



In the Properties, navigate to the "Filegroups" page, click on the "Add FileGroup" option under "FILESTREAM" and set the Name of your preference, tick the default option and click the OK button.

6. Next, navigate to the Properties "files" page, click on the "Add" Button. In the new entry that has been created, set the name of your preference, the File Type as "FILESTREAM Data" and under the "Path" column, set the directory under which the screenshots should be saved.



7. As a final step, execute the query below (The Query can be also found [here](#)).

```
sp_rename 'logs_screenshots.screenshot_bytes',
'screenshot_bytes2' , 'COLUMN'
GO
```

```
ALTER TABLE logs_screenshots ALTER COLUMN id ADD ROWGUIDCOL
GO
```

```
ALTER TABLE logs_screenshots ADD screenshot_bytes varbinary(max)
FILESTREAM NULL
GO
```

```
UPDATE logs_screenshots
SET screenshot_bytes = screenshot_bytes2
GO
```

```
ALTER TABLE logs_screenshots
DROP COLUMN screenshot_bytes2
```

GO

Note: To disable the FileStream, run the below query (The Query can be also found [here](#)).

```
sp_rename 'logs_screenshots.screenshot_bytes' ,
'screenshot_bytes2' , 'COLUMN'
GO
```

```
ALTER TABLE logs_screenshots ADD screenshot_bytes varbinary(max)
NULL
GO
```

```
UPDATE logs_screenshots
SET screenshot_bytes = screenshot_bytes2
GO
```

```
ALTER TABLE logs_screenshots
DROP COLUMN screenshot_bytes2
GO
```

```
ALTER TABLE logs_screenshots ALTER COLUMN id DROP ROWGUIDCOL
GO
```

1.5.14 BotStack

Multi-SoloBot hosting on one Server

ProcessRobot now supports the ability for multiple Robots to execute Processes on a single Windows Server.

In environments where a single Robot is installed, the flow of execution is the following:

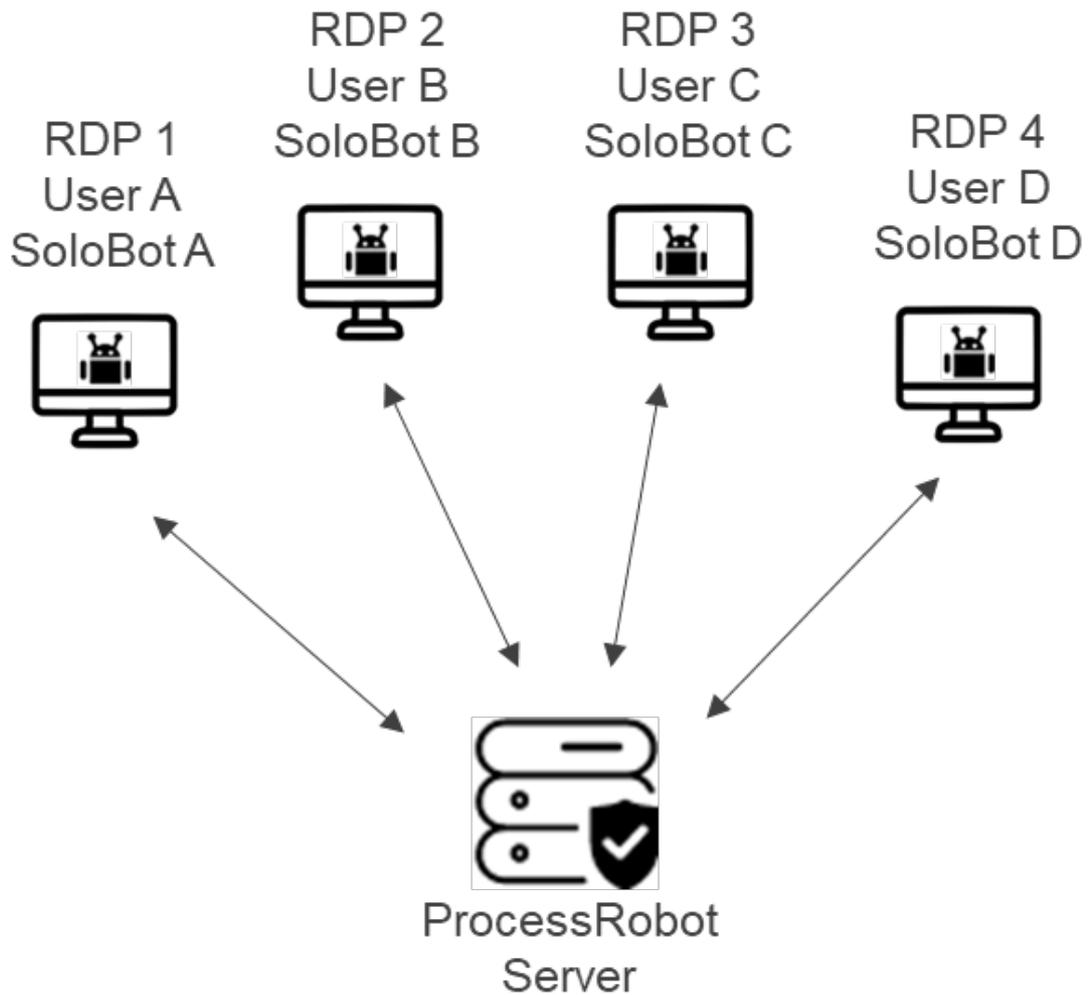
On machine A, Solobot A logs in and executes Processes using User's A credentials:

SoloBot A > Machine A > User A (Autologin)

In a Multi-SoloBot environment, it is possible to assign multiple SoloBots to a single Windows Server machine using different User credentials. Each User will launch an RDP session on the Server, as per the following scheme:

SoloBot A > Machine A > Remote Desktop session from User A
SoloBot B > Machine A > Remote Desktop session from User B
SoloBot C > Machine A > Remote Desktop session from User C

...and so on.

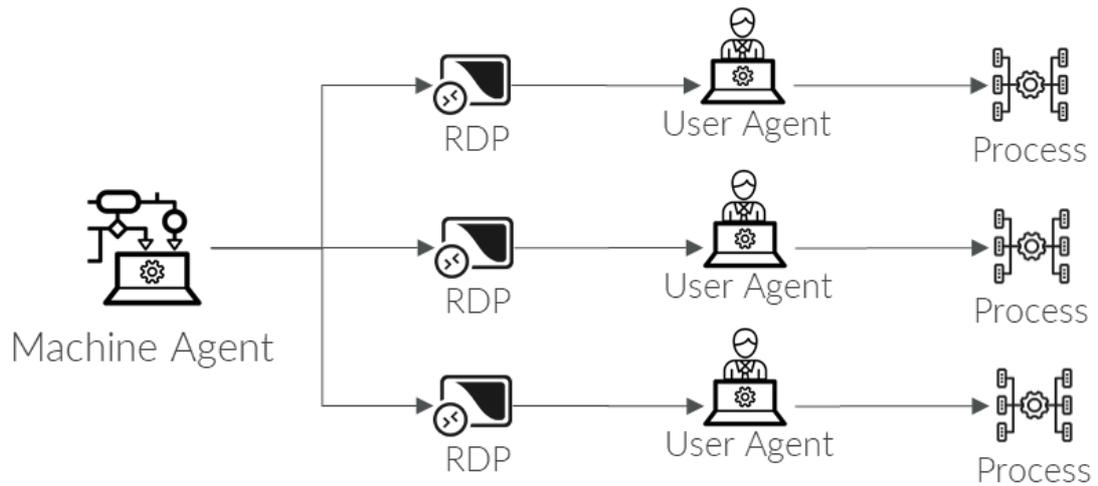


In this regard, ProcessRobot will allow multiple SoloBots on the same Server to be connected under different Users; in the same way as when one initiates a remote desktop session on a server.

The instances will be spawned from a single installation per server. A single installation of the SoloBot software is required on the workstation. The multiple Robots and Users will then be declared in the Control Desk.

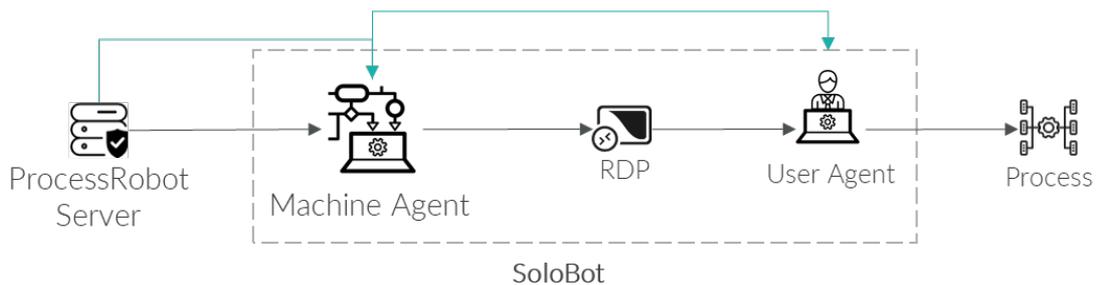
This feature will utilize the machine in the best manner, in order to run multiple Processes on multiple SoloBots, taking maximum advantage of the available hardware resources.

Overview



A single Machine Agent on the Windows Server SoloBot machine will be able to establish multiple RDP connections to itself, one connection per registered SoloBot. Login in each of these connections will be achieved by using the credentials of the Active Directory User account assigned to each SoloBot.

Process Flow Overview



The Process Flow of initiating the execution of a Process on a SoloBot may follow one of two possible paths, depending on whether the Active Directory User is already logged in.

If the User is logged in:

The ProcessRobot Server sends a signal to the Machine Agent, which in turn sends a signal to the User Agent, which runs the Process.

If the User is NOT logged in:

The ProcessRobot Server sends a signal to the Machine Agent, which initiates a RDP session (to machine it resides), logs in with the User's credentials, starts the User Agent, which then runs the Process.

Note:

1. High density scenarios are supported only for different User instances.
2. In cases where Users are logged in, ProcessRobot supports the use of SideBots as well

1.5.15 BotStack Requirements

Software Requirements

The recommended OS for enabling the BotStack feature is Windows Server 2016 or later. By default, Windows Server supports up to two simultaneous User sessions. For additional sessions, the appropriate Remote Desktop Services Client Access Licenses must be in place.

Use of Credentials

The technology used to establish the RDP connection is the .NET Microsoft Library: **IMsRdpClient**.

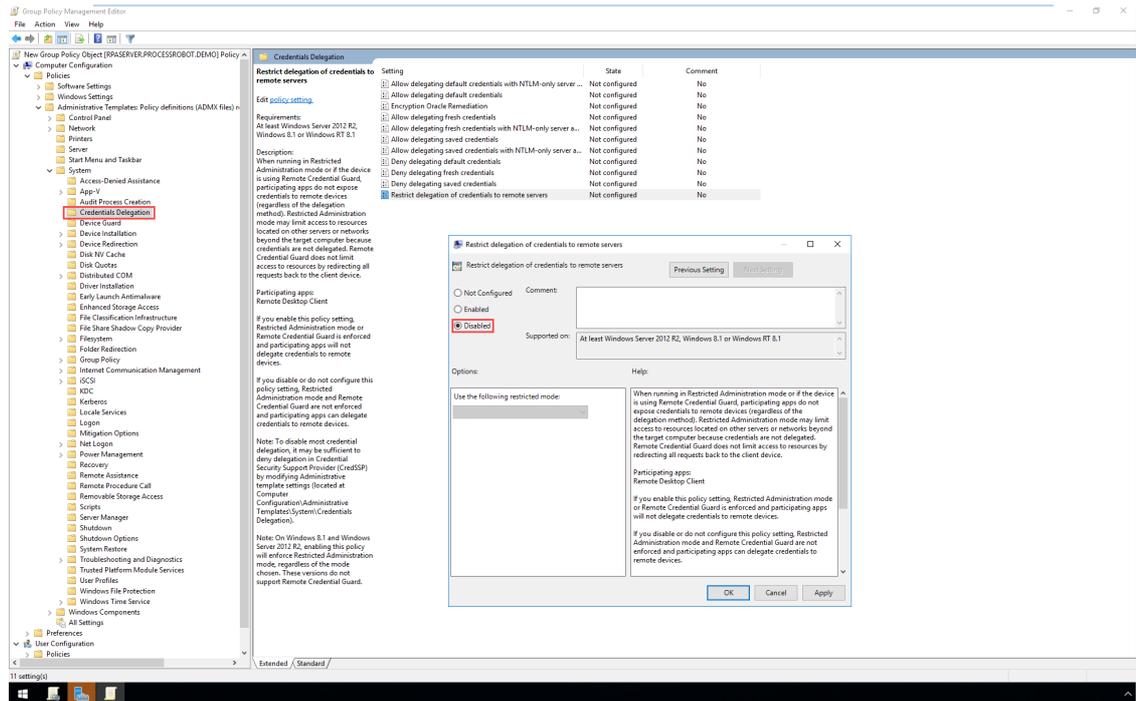
ProcessRobot does not store the Active Directory Users' credentials, neither username nor password via the Credential Manager. Instead, the credentials are supplied programmatically to the RDP library. Consequently, no credential file is created, as ProcessRobot does not store these credentials anywhere.

Group Policy Requirements

Enabling the multi-tenant approach requires changes to the following two Group Policies:

Policy 1 - "Restrict delegation of credentials to remote server"

The "Restrict delegation of credentials to remote server" policy must be disabled since the credentials are sent from the Machine Agent on the Window Server machine to the .NET Microsoft RDP library on the same machine in order to initialize RDP connections. In essence, credentials are delegated to the same machine that sends them.

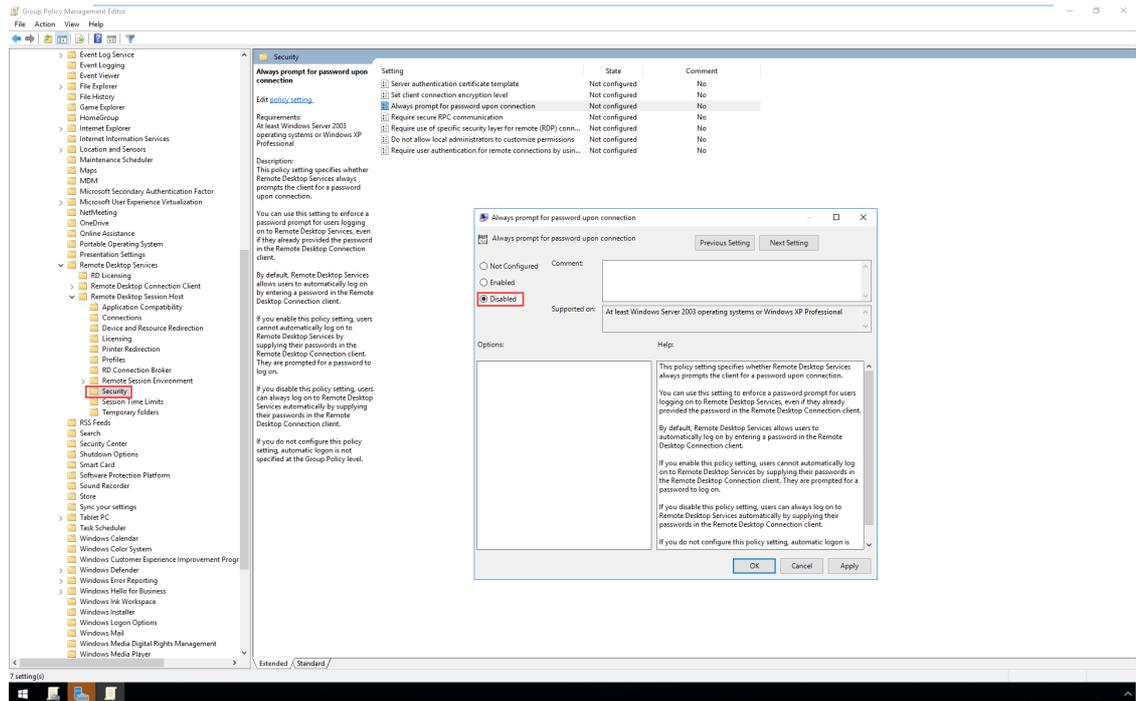


The path to disable the above policy is:

Gpedit.msc > Computer Configuration > Administrative Templates > System > Credentials Delegation > Restrict delegation of credentials to remote servers: **Disabled**

Policy 2 - "Always prompt for password upon connection"

The "Always prompt for password upon connection" policy must be disabled. Otherwise, this policy will interfere with the SoloBots' Auto Login feature. As mentioned previously, the credentials are passed programmatically to the RDP library. If this policy is enabled, the system will again require the User's password, thus preventing the Auto Login.



The path to disable the above policy is:

Gpedit.msc > Computer Configuration > Administrative Templates > System > Windows Components > Remote Desktop Services > Remote Desktop Session Host > Security > Always prompt for password upon connection: **Disabled**

2. Control Desk Components

2 Control Desk Components

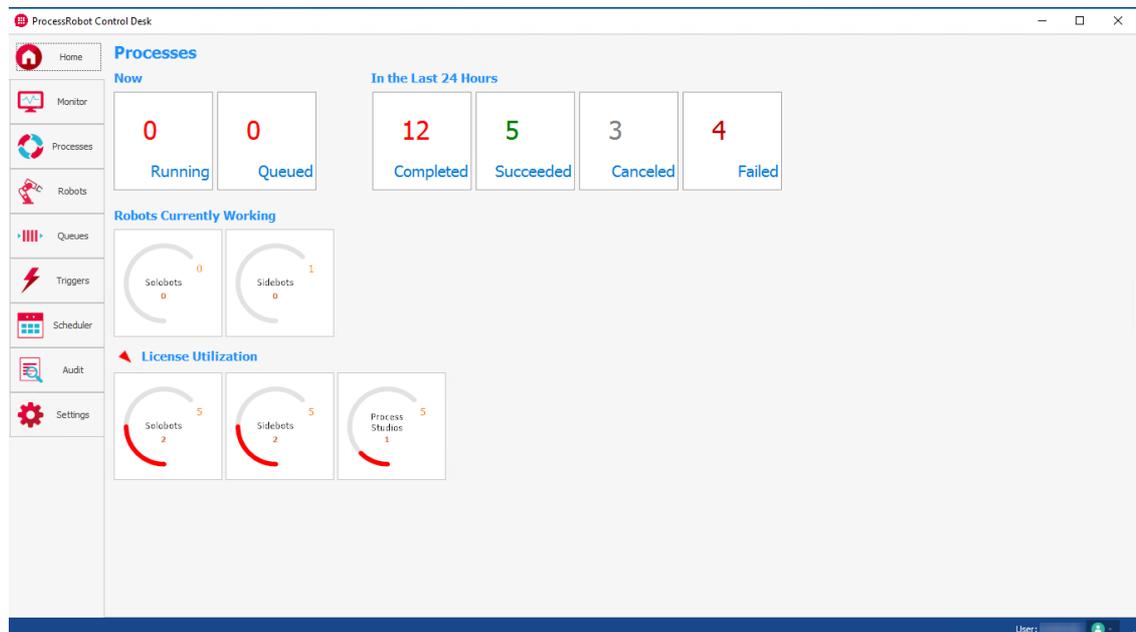
2.1 Control Desk

2.1.1 Control Desk Introduction

The Control Desk is the component through which you can have the general Control of your ProcessRobot.

What the Control Desk does?

- Automates distribution of Processes and Robot workload across the enterprise.
- Set up and administer the operating environment with the creation of transactional queues, triggers and scheduling.
- Select from a rich list of trigger options, including Event Log, Email Monitor, Ping, Hotkey and File Monitor.
- Auditing, log monitoring and governance support.
- Monitoring of operations, with notifications for error handling and fallbacks.
- User adding, Roles, Environments, Global variables, Robot Pools and Queues setting and handling.

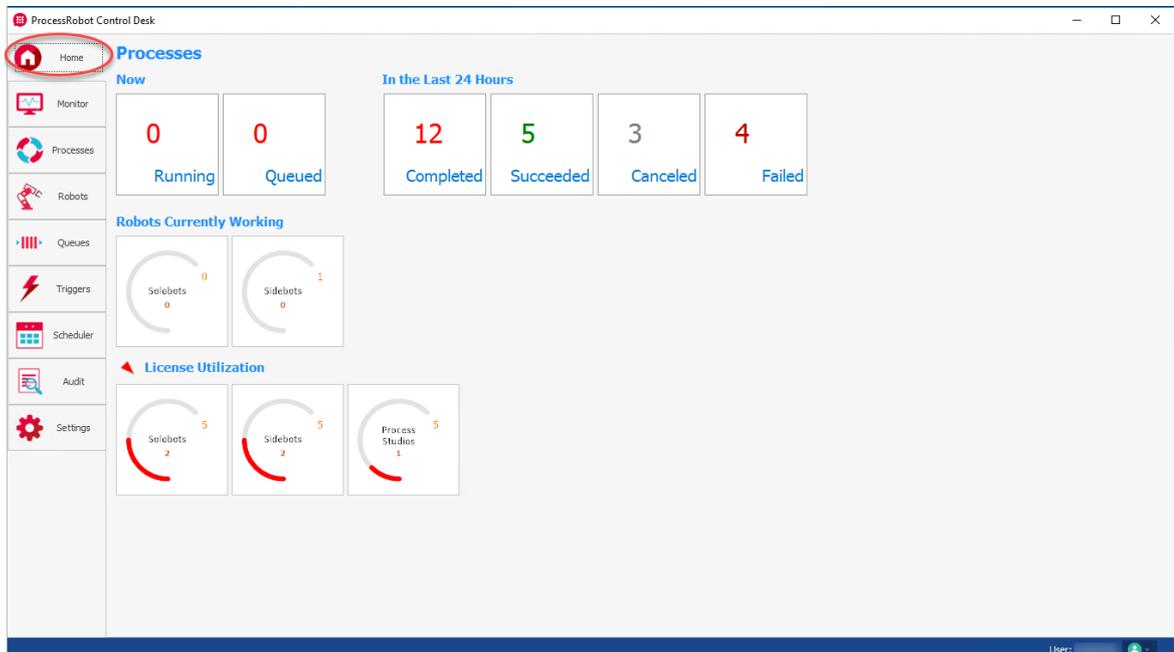


Control Desk

2.1.2 Control Desk Tabs

2.1.2.1 Home

Once you open the Control Desk, the Home Tab is the default tab that you will see.



The Home Tab

In the Home Tab you will be provided with information regarding the Processes, the Robots that are currently working and the License Utilization.

Processes: The Number of Processes that are running or queued are displayed here. There are also indications about the Processes that were executed, completed, canceled, failed in the last day (24 hours).

Robots Currently Working: Right below, you will be able to see the Solobots and Sidebots information that are currently working.

License Utilization: At the bottom of the Home tab the License Utilization is displayed, so that you can have an overall look at the Robots and Process Studios currently utilized according to your purchased licenses.

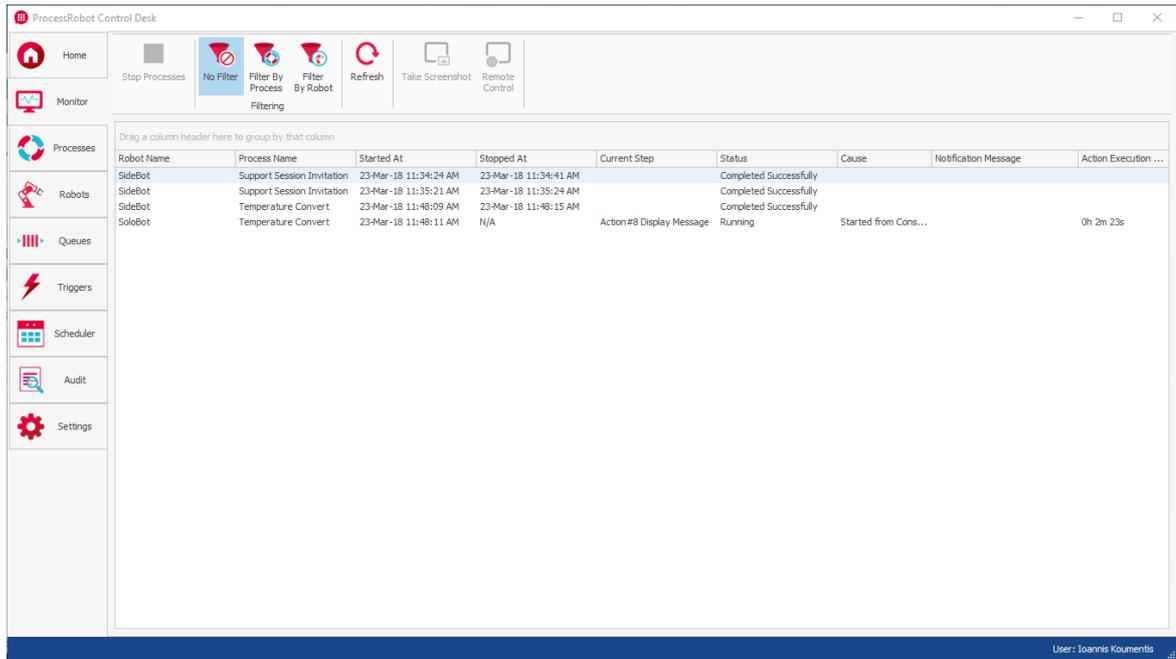
The overall design helps you get a very good idea of what is happening in your ProcessRobot just by taking a look at the Home tab, due to the neat graphic display.

2.1.2.2 Monitor

The next tab in the Control Desk is the Monitor tab, in which you are able to Monitor all the running Processes and the Robots that they are being executed on at the moment.

Here, you have the following options to apply:

- No Filter at all, which means that all the running processes on all Robots will be listed



ProcessRobot Control Desk

Home | Stop Processes | No Filter | Filter By Process | Filter By Robot | Refresh | Take Screenshot | Remote Control

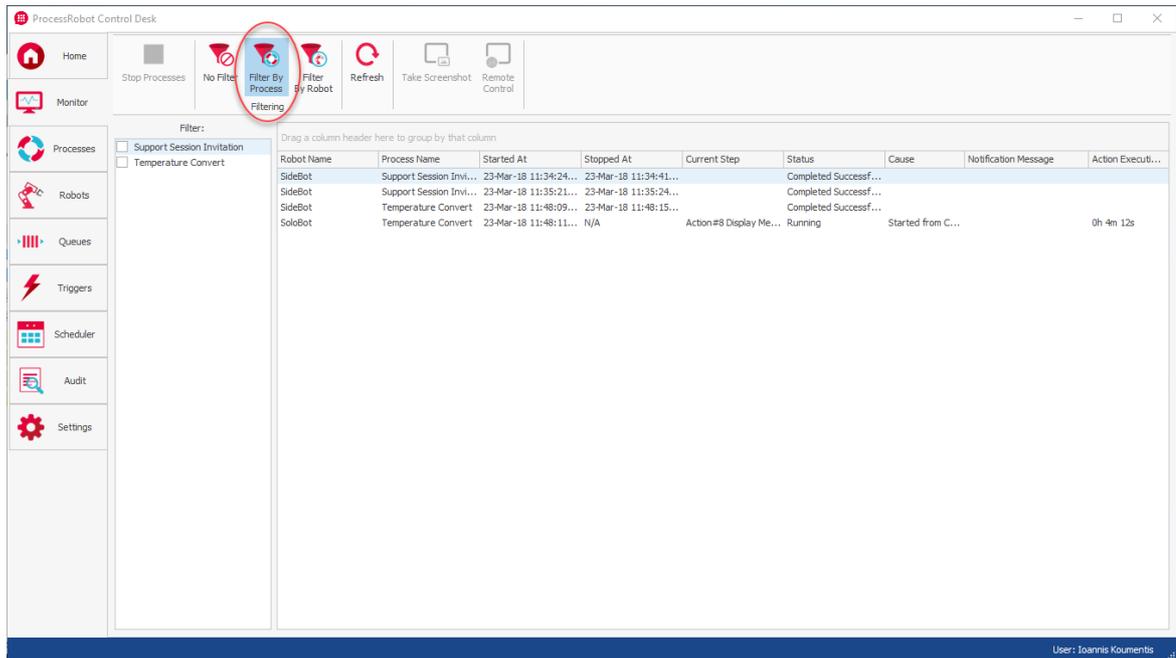
Drag a column header here to group by that column

Robot Name	Process Name	Started At	Stopped At	Current Step	Status	Cause	Notification Message	Action Execution ...
SideBot	Support Session Invitation	23-Mar-18 11:34:24 AM	23-Mar-18 11:34:41 AM		Completed Successfully			
SideBot	Support Session Invitation	23-Mar-18 11:35:21 AM	23-Mar-18 11:35:24 AM		Completed Successfully			
SideBot	Temperature Convert	23-Mar-18 11:48:09 AM	23-Mar-18 11:48:15 AM		Completed Successfully			
SoloBot	Temperature Convert	23-Mar-18 11:48:11 AM	N/A	Action#8 Display Message	Running	Started from Cons...		0h 2m 23s

User: Ioannis Koumentis

Monitor-No Filter

- Filter by Process, which means only the Robots that run a specific Process will be listed. You also have the option to narrow down the displayed results by checking the corresponding check-box of the Process in the Filter pane.



ProcessRobot Control Desk

Home | Stop Processes | No Filter | Filter By Process | Filter By Robot | Refresh | Take Screenshot | Remote Control

Filter:

Support Session Invitation

Temperature Convert

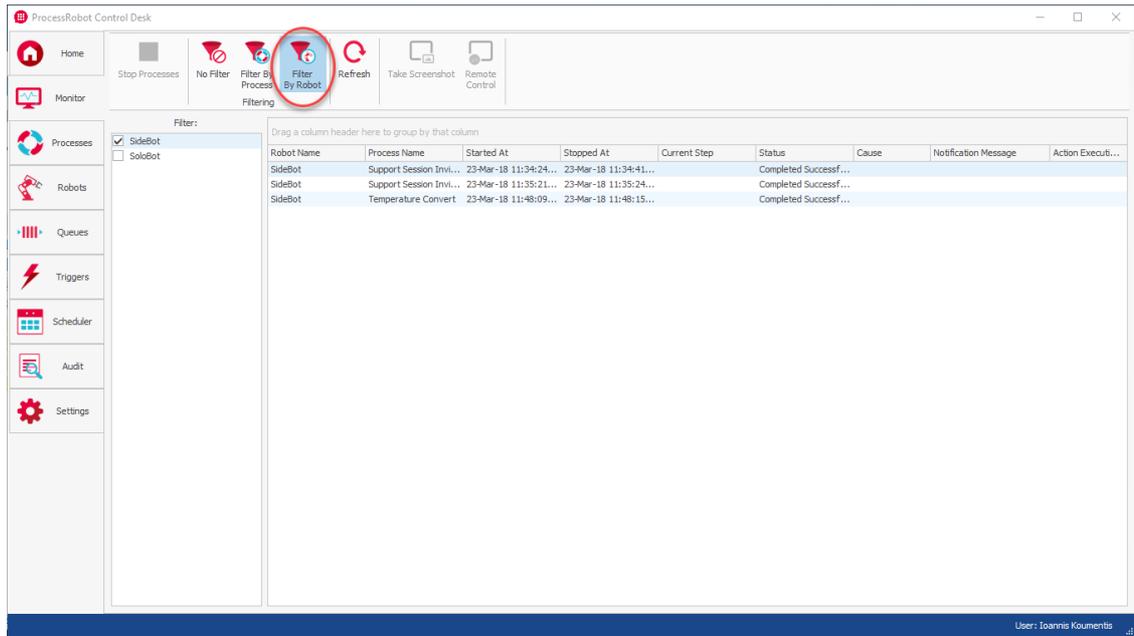
Drag a column header here to group by that column

Robot Name	Process Name	Started At	Stopped At	Current Step	Status	Cause	Notification Message	Action Execution ...
SideBot	Support Session Invitation	23-Mar-18 11:34:24...	23-Mar-18 11:34:41...		Completed Successf...			
SideBot	Support Session Invi...	23-Mar-18 11:35:21...	23-Mar-18 11:35:24...		Completed Successf...			
SideBot	Temperature Convert	23-Mar-18 11:48:09...	23-Mar-18 11:48:15...		Completed Successf...			
SoloBot	Temperature Convert	23-Mar-18 11:48:11...	N/A	Action#8 Display Me...	Running	Started from C...		0h 4m 12s

User: Ioannis Koumentis

Monitor - Filter by Process

- Filter by Robot, which means that only the Processes that run on a specific Robot will be listed. You also have the option to narrow down the displayed results by checking the corresponding check-box of the Robot in the Filter pane.

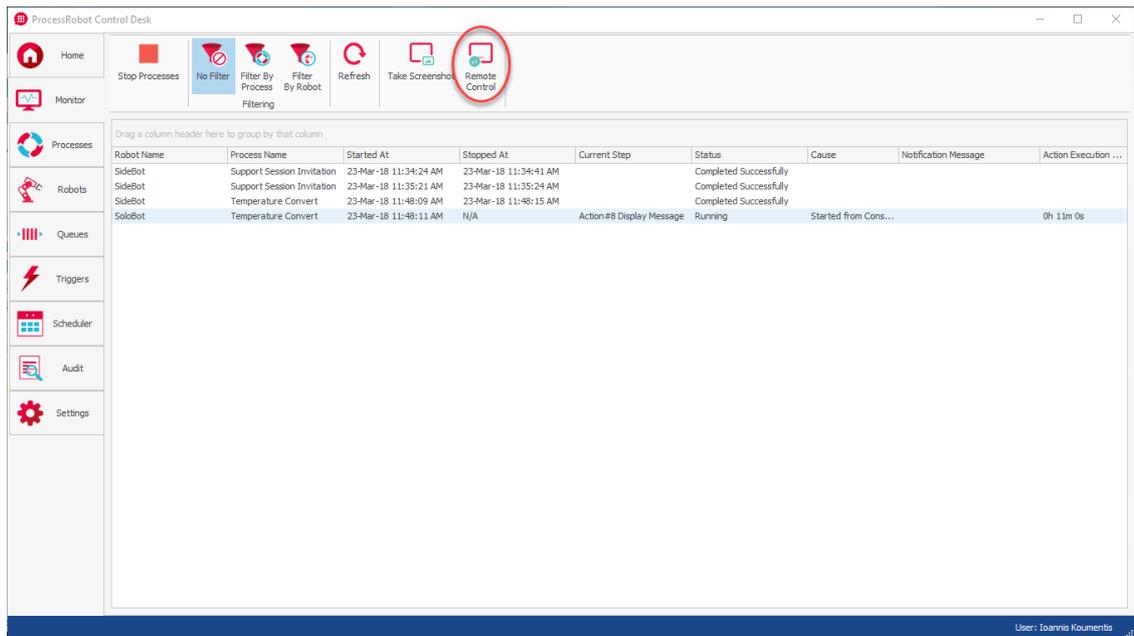


Monitor-Filter by Robot

The Refresh button will allow you to refresh the information being displayed.

The Take Screenshot button will allow you to easily take a screen shot for future reference purposes.

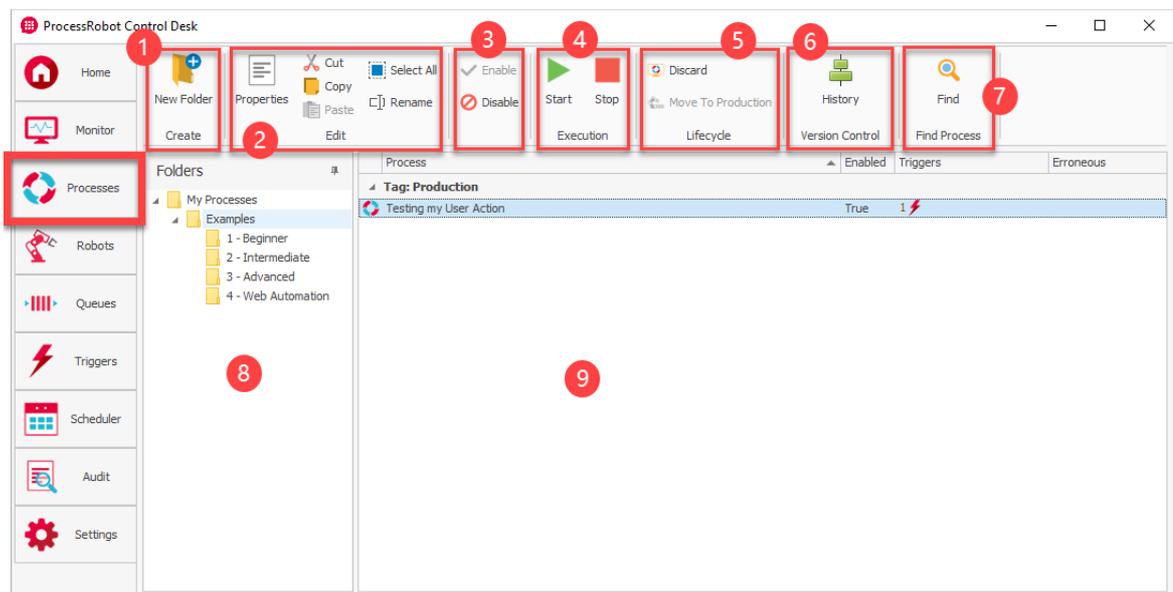
The [Remote Control](#) button will be enabled while a Process is being executed on a Solobot machine. Upon pressing on that button the Remote Control window opens, that gives you the ability to access the monitor(s) of the aforementioned machine and review Process' execution. Please note that a Control Desk User will have access to either view only, or view and control (by using mouse and keyboard) the Solobot monitor(s), based on her/his User's Role and Permissions.



Monitor-Remote Control

2.1.2.3 Processes

The Processes tab consists of the following menus:



Process tab

1. Create:

In the Create group you can [create a new folder](#)^[28] to save your Processes in the Control Desk.

2. Edit:

In the Edit group you can view the [Properties](#) of a Process, Select all Processes in the folder that you are currently in, Rename/Cut/Copy/Paste a Process and [Enable or Disable](#) it.

3. Enable/Disable

In this group you can [Enable or Disable](#) a Process.

4. Execution:

Through the Execution group you can [Run-Start](#) a Process on a specific Robot or Robot Pool and then [Stop it](#) at any time before its completion.

5. Lifecycle:

In this group you can move a Process to another step of its [Lifecycle](#). If a Process is in the stage of Production, then, it can ONLY be Discarded and be transferred back to Development. If a Process is in Review, you can either Discard it OR move it to Production. *You can also Discard a folder.*

The Lifecycle of a Process consists of three stages:

Development: When a Process is being created and it is under Development, then it is visible only in the Process Studio in its corresponding folder. Once the development of a Process has been completed, then it is ready to move to the Review stage, via the corresponding button in Process Studio "Move to Review". In case that a Process has already been in Production, but it is discarded for development again due to an error, then it is available both in the Control Desk and the Process Studio.

Review: When a Process is under Review then it is available in the Control Desk and it is listed under the "Review" Tag. Being under Review means that this Process can be Started for testing purposes from an Administrator, but it cannot be used in sophisticated features of ProcessRobot, like Schedules or Triggers.

Production: After testing is completed, the Process can move to Production. In this stage, the Process is available to all features of Control Desk.

Processes are available to different components according to the Lifecycle stage they are currently in.

Before you consider your self ready to build an enterprise grade RPA solution with PR please make sure that you feel comfortable with [Version Control](#).

6. Version control:

In this pane you are able to view the Version History of a Process. Through this button you can see information about the Process's versions and the [Lifecycle](#) stage each version is or has been in the past.

7. Find Process or Folder:

An invaluable facility that can help you locate the process or folder of your interest in case you are working with a complex treeview (or you simply can not remember where a favorite process is).

8. The Processes Folders List pane:

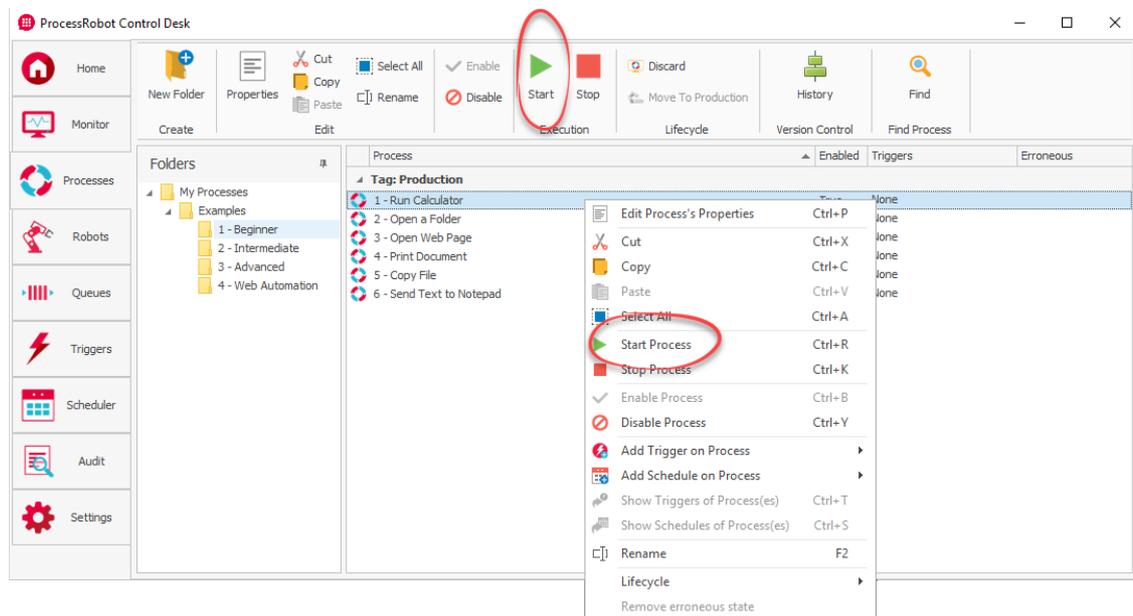
In this pane you are able to view your saved Processes folder structure.

9. The Processes List pane:

In this pane you can see all the Processes that are saved in a specific folder. You can click on a folder in the Processes' Folder pane and all its Processes will be listed in the Processes' pane. The Columns in the Process List pane can be rearranged should you click and drag them according to your desirable display.

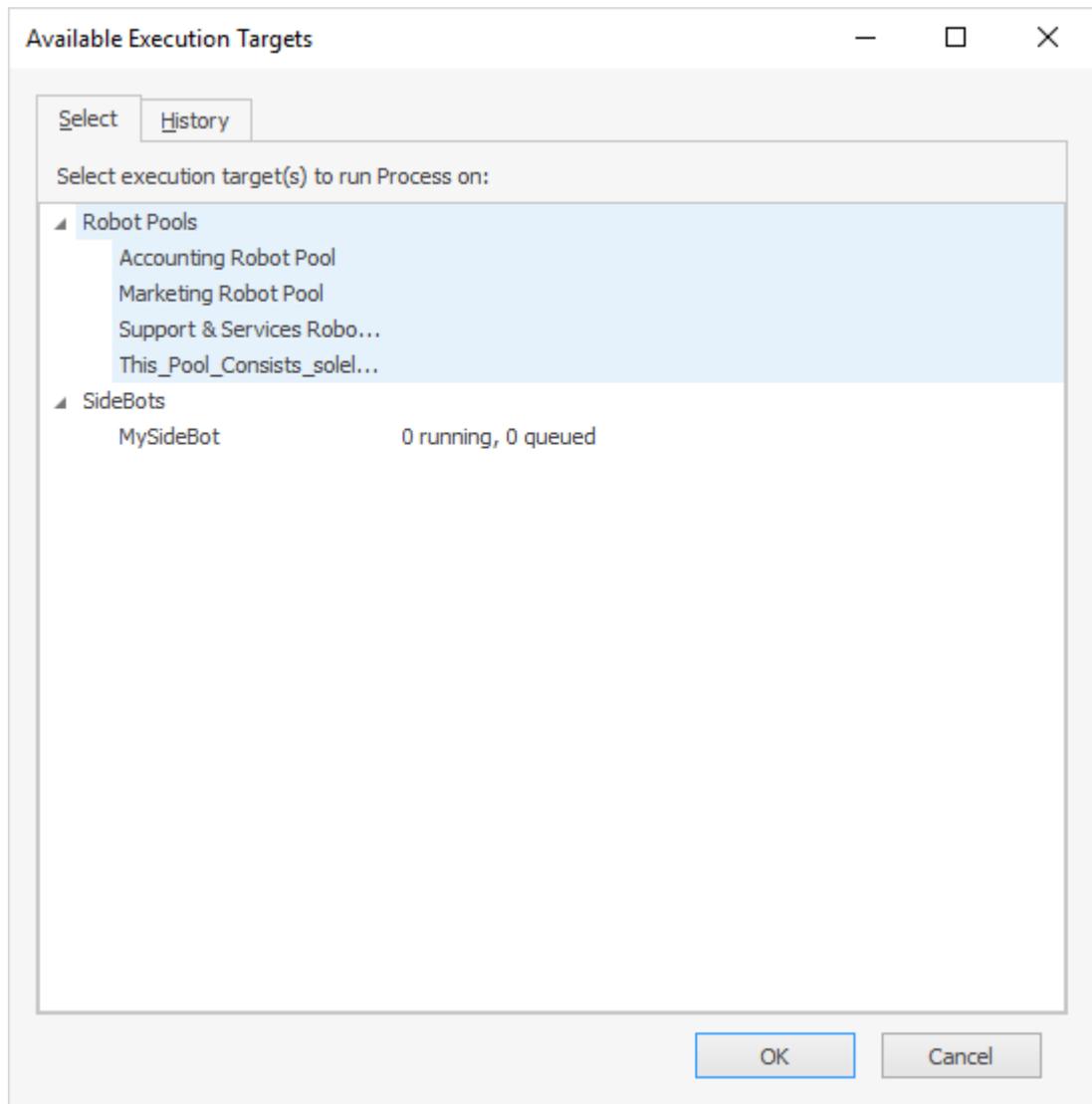
2.1.2.3,1 Run/Stop a Process

You can start a Process manually by clicking on the green arrow icon "[Start](#)", or by right-clicking on the Process and choosing Run Process. The Process you want to start must be highlighted. If no Process is highlighted, the Start "▶" and Stop "■" options are not available.



Start a Process

Upon selecting a process to run, you get a pop-up window "Available Execution Targets" that will prompt you to select the Pool(s) or Robot(s) to run the Process on:



Select the execution target on which the Process will run on

Select the Execution Target on which the Process will run on and press "OK".

You can stop a running Process by clicking on the Stop Icon in the Processes tab, or by right-clicking on the Process and choosing Stop Process.

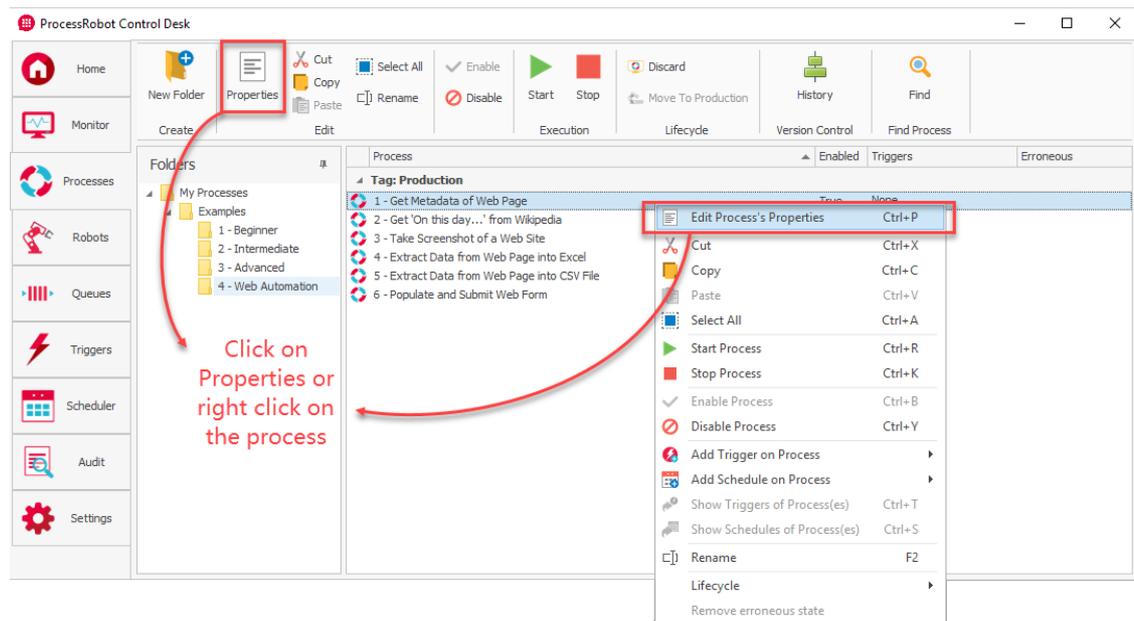
You can also stop one instance of a running Process by clicking on the "Stop Processes" link found on the [Notification Popup Window](#)²⁸⁴, although this will close only that instance of the Process. Bare in mind that only the Robot that the process is running on can do this. So in case that you are starting a Process from the Control Desk, then you are only able stop it from the Stop button.

Details on how to deploy a Process to a Pool (which includes a number of Sidebots and/or Solobots) you can find on the "[Deploy a Process to a Sidebot](#)"¹⁰⁹¹ topic.

2.1.2.3,2 Process Properties

You can open the Process Properties Dialog for a certain Process:

- by clicking on the Properties button  on the Processes tab in the Edit group menu or
- by right-clicking on the Process and choosing "Edit Process's Properties".



This brings up the Process Properties Dialog, on the first tab - [General](#)¹³⁰.

Enable/Disable a Process

The default setting "Enabled" means that a Process can run. If you Disable a Process, it cannot run, neither manually nor by a Trigger. This is useful for making sure that a Process that is not supposed to run, for any reason, and that it will not start accidentally. You can however edit a disabled Process and run it through the Designer.

You can enable or disable a Process by selecting it and choosing [Enable Process](#) or [Disable Process](#) on the Processes Tab.

If you disable a Process, the text will appear in a lighter, grayed-out color:

The screenshot shows the ProcessRobot Control Desk interface. The main window displays a list of processes under the 'Tag: Production' category. The processes are listed in a table with columns for 'Process', 'Status', and 'Life'. The processes are:

Process	Status	Life
1 - Get Metadata of Web Page	True	None
2 - Get 'On this day...' from Wikipedia	True	None
3 - Take Screenshot of a Web Site	True	None
4 - Extract Data from Web Page into Excel	False	None
5 - Extract Data from Web Page into CSV File	False	None
6 - Populate and Submit Web Form	False	None

Red callouts highlight the 'Enabled Processes' (rows 1-3) and 'Disabled Processes' (rows 4-6). The interface includes a sidebar with navigation options like Home, Monitor, Processes, Robots, Queues, Triggers, Scheduler, Audit, and Settings. The top toolbar contains actions like New Folder, Properties, Copy, Paste, Edit, Select All, Rename, Enable, Disable, Start, Stop, Move To Production, and History. The bottom status bar shows 'User: Peter Feleskouras'.

Enable or Disable Process

General

The first tab in the Processes Properties is the "General" tab.

Process Properties of '1 - Get Metadata of Web Page'

General Deployment Execution Maker-Checker Concurrency Policy Error t

Name: '1 - Get Metadata of Web Page'

Description:

Automatic Logging

Use Secure Screen

Max allowed running time 1 minute(s)

Process expiration time after robot disconnect 1 minute(s)

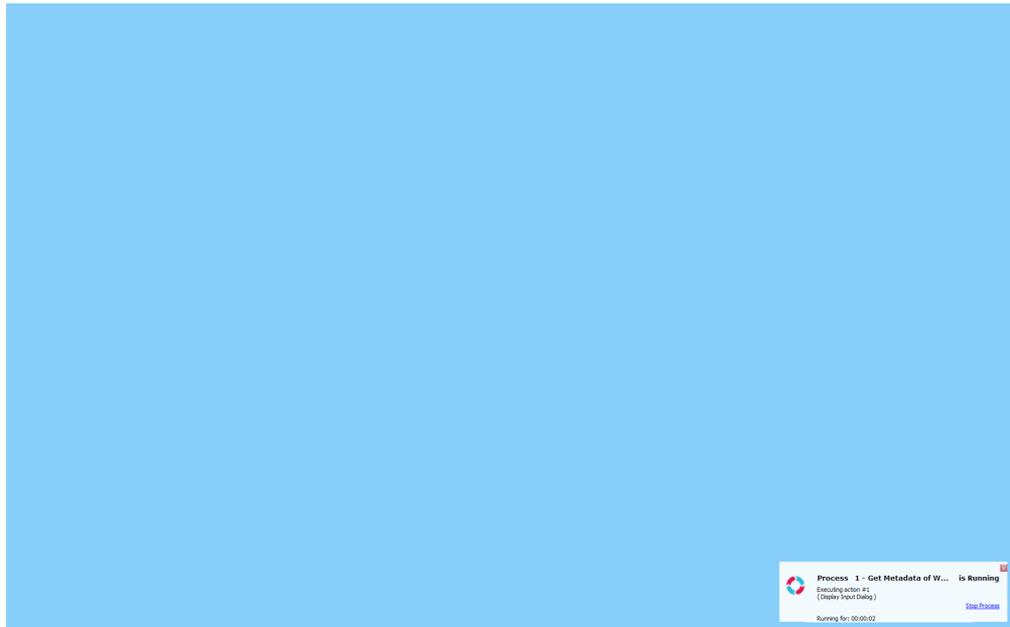
Help OK Cancel

Process Properties - General Tab

1. Description: Enter a description and your notes, which will be mostly used for documentation purposes. If you need to remember something about the Process, or tell someone else about it, write it here.

2. Automatic Logging: If this option is checked the Logs/Audit will contain all the info for every step/action executed in the Process.

3. Use a Secure Screen: If you check this option, the screen will turn blue upon running the Process and no one will be able to see what the Process does. The only visible window will be the Notification Window.



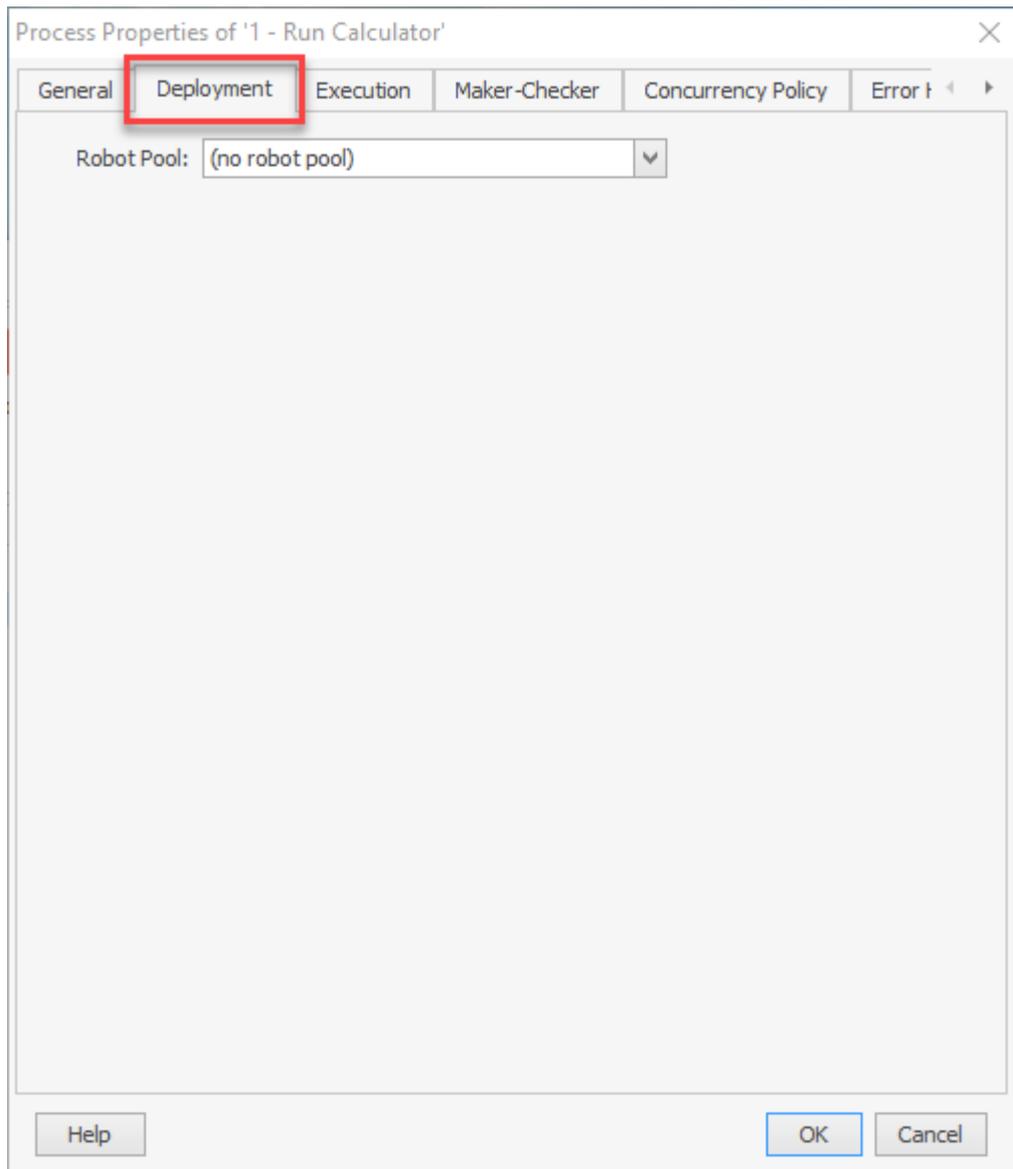
Running with Secure Screen

4. Max allowed running time: A maximum running time (in minutes) can be set in this option, in order to stop the Process after that time has passed. This helps you stop Processes that you are certain they should not be running after a predefined period of time.

5. Process expiration time after robot disconnect: Set the maximum allowed time (in minutes) to elapse after the Robot which is executing the Process disconnects before canceling/stopping the Process execution.

Deployment

In the Deployment Tab you can specify the Processes you want to be available on a specific Sidebot without their user having access on any other ProcessRobot component (Control Desk, Process Studio, etc):



After specifying the Robot Pool on which the Sidebot belongs to, you will be presented with a number of additional settings grouped together under the name Deployment Data (if this Sidebot does not belong to any Pool then create one that will contain it via Settings > [Robot Pools](#)^[217]):

Process Properties of '3 - Open Web Page'

General Deployment Execution Maker-Checker Concurrency Policy Error t

Robot Pool: This_Pool_Consists_solely_of_My_SideBot

Deployment Data

Process Name:

Folder:

Hotkey: Control + Shift + Alt +

Help Text:

Help OK Cancel

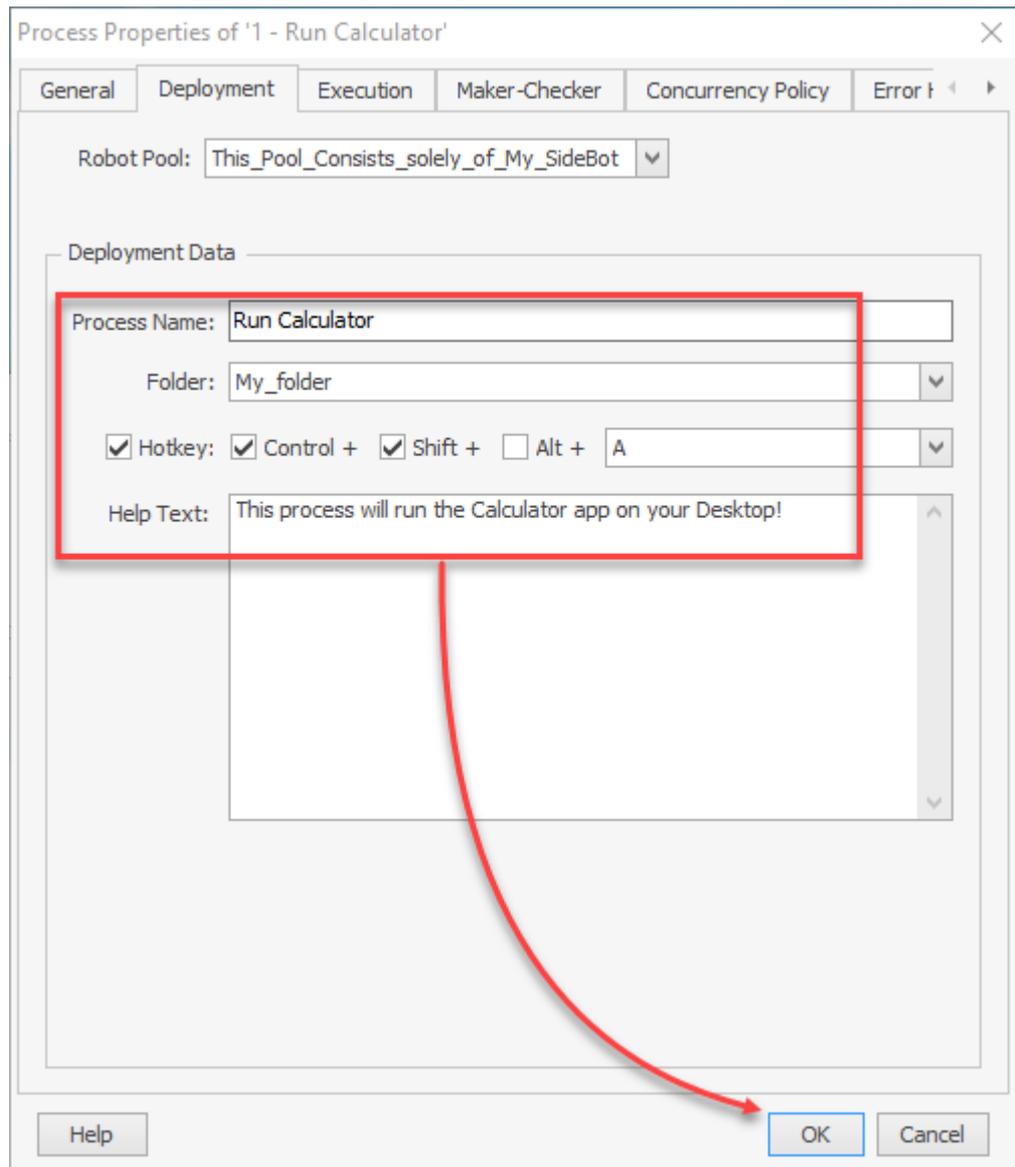
Through the Deployment Data you can specify:

The Process Name: The name with which the Process will appear in the Sidebot's interface.

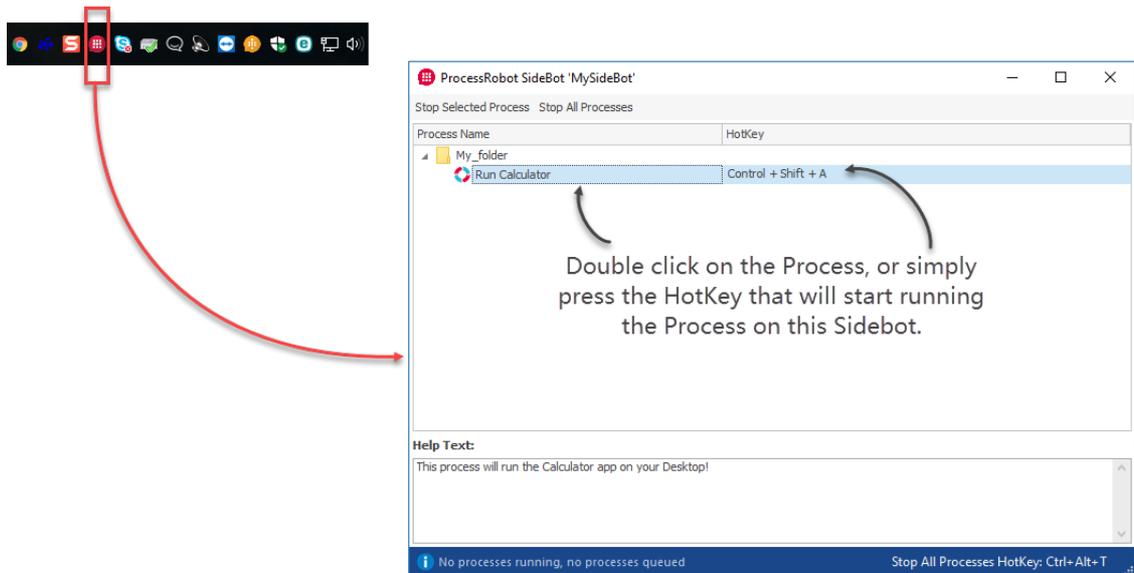
Folder: Here you can specify the folder path that will be shown in the Sidebot, which will contain the Process (for additional details please refer to the ["Deploy a Process to a Sidebot"](#) topic).

Hotkey: Assign a hot-key to the Process, so when this hot-key is pressed from the Sidebot user, the Process will fire immediately on the target Robot.

Help Text: A description that you can give to the Process in regards to what is supposed to do.



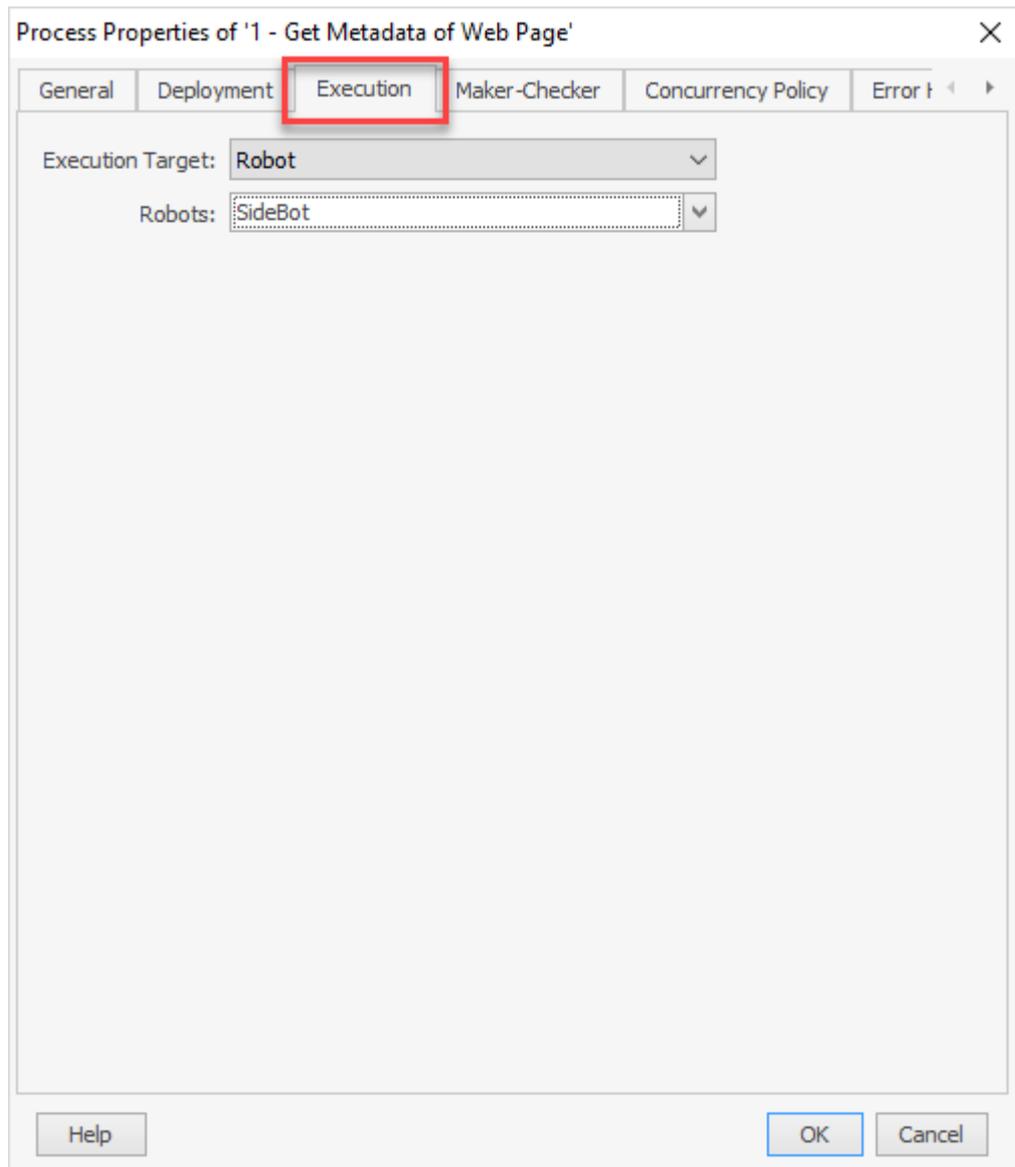
After pressing OK you can check the impact of your actions by opening the Sidebot affected (double-click on the ProcessRobot icon in the System Tray - the System Tray is that group of icons at the right of the Windows Taskbar):



You can Stop All Processes running from this Sidebot at anytime, using the Stop All Processes Hotkey (Ctrl+Alt+T).

Execution

Execution has to do with which robots exactly we want to run a process on (ie the definition of our Execution Target), when we run it manually from the Control Desk by pressing the "Start" button. If the Execution tab is not configured then, the user will always be asked on which Robot or Robots Pool he wishes to execute the process:



This is why we do not have any additional option settings when we declare the Execution Target of the Process.

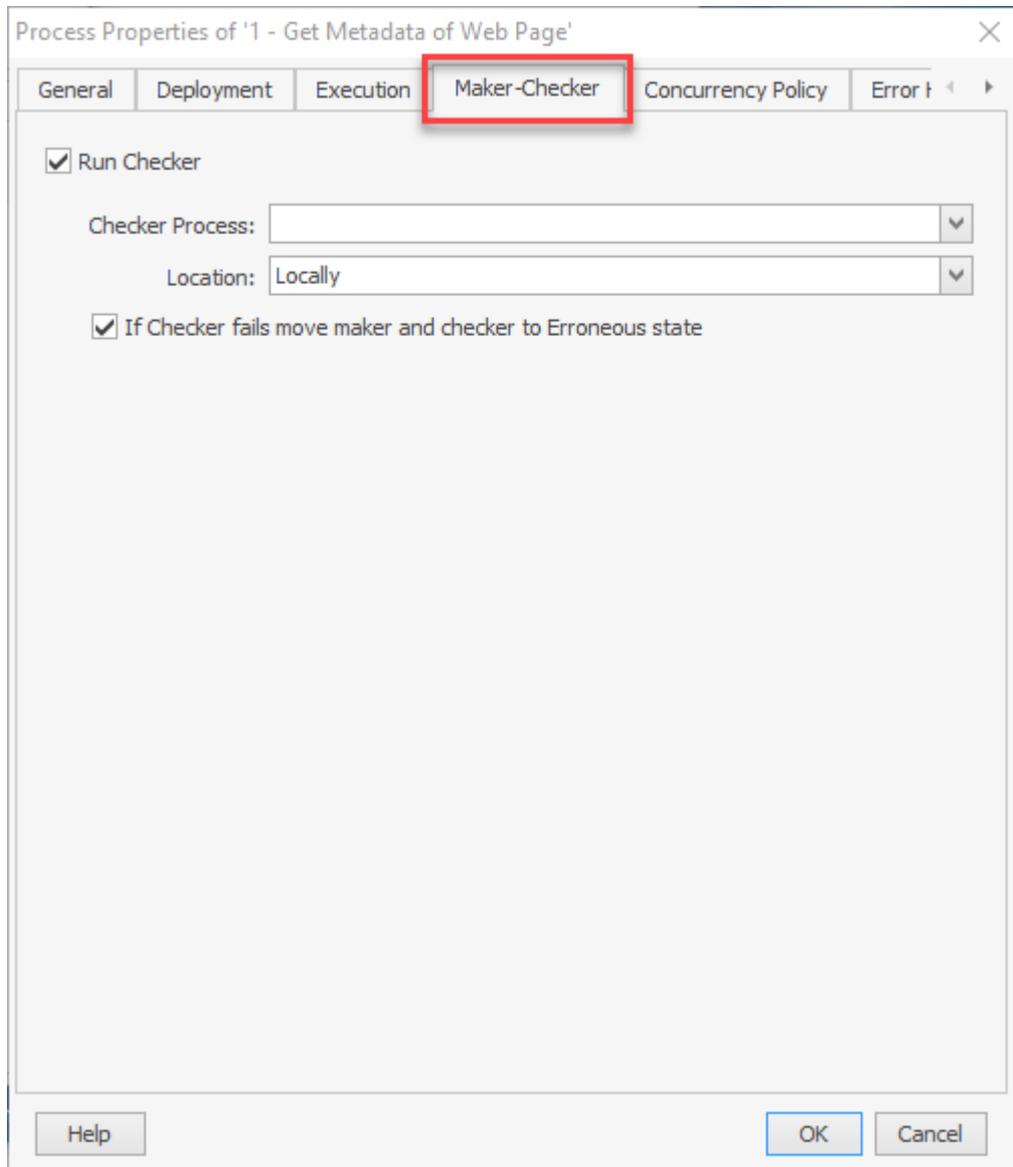
Maker-Checker

The third tab in the Process' Properties is the "Maker-Checker" tab.

As **Maker** we define the Process that has just run on a Robot or [Robot Pool](#)²¹⁷.

As **Checker** we define **another** Process that will run right after the Maker is completed, in order to make sure that the Maker did its job correctly and double check that everything went smoothly.

A very simple example; lets say that you wish to enter a new contact in your database. The Maker will update the database with the appropriate info, so the Checker could be a Process that would query the database to see if the contact was successfully created.



Robots Properties - Maker-Checker Tab

Run Checker: If this box is checked then the Checker will run right after the completion of the Maker.

Checker Process: Here you will specify the path of the Process that will be the Checker for this Maker.

Location: Select whether you wish the Checker to run Locally or on another Pool.

If Checker fails move Maker and Checker to Erroneous state: If this box is checked then the maker as well as the checker will go to an Erroneous State, so that you can later on review what went wrong.

Concurrency Policy

The fourth tab in the Process' Properties is the "Concurrency Policy" tab, where you can:

1. Limit the number of Process instances that can run at the same time by setting the maximum number of instances.
2. Choose what should happen if that limit is reached (Queue or don't start the Process).
3. Set the minutes for a Process instance to wait before being rejected via a Timeout.

Process Properties of '1 - Get Metadata of Web Page'

General Deployment Execution Maker-Checker **Concurrency Policy** Error Handling

Limit the number of Process Instances that can run concurrently per Robot

1 Max processes limit: 1

If this limit is reached: Queue Process **2**

Set Timeout

3 Wait for max 1 minutes before being rejected

Help OK Cancel

Process Properties - Concurrency Policy Tab

Error Handling

You can choose to use the same default policies you set in "Settings Tab -> Error Handling" in the Control Desk, or select new ones for this specific Process here. If you wish to choose a new specific policy for this Process, choose one or more from the following which will replace the [Error Handling Tab](#) general settings (of course you would have to choose to "Override Default Options"):

1. **Send an email**, possibly with multiple recipients (separated by semi-colons), with details set in the [Email/SMTP tab of the Option tab](#)^[248].
2. **Run another Process** from your Processes database.
3. **Write (append) Event to a text file** whose directory you can specify. The Event will be written at the end, so you can keep a running log of Process failures.
4. **Record the event to the Windows Applications Event Log**. You can view the Windows Event Logs through Control Panel -> Administration -> View Event Log.
5. **Set the Process to an Erroneous State**.
6. **Add Screenshot to Logs** so as to produce a screenshot at failure that will be available for viewing and saving through the [Audit Tab](#)^[179] (last column at the right without title).
7. **Capture**. You can record the erroneous state of your Process running on a Solobot machine. The recorded video can be accessed through the corresponding option on the [Audit Tab](#)^[179].

The screenshot shows the 'Process Properties of Testing' dialog box with the 'Error Handling' tab selected. The dialog has a title bar with a close button (X) and a tab bar with 'Deployment', 'Execution', 'Maker-Checker', 'Concurrency Policy', and 'Error Handling'. The 'Error Handling' tab is highlighted with a red box. Below the tab bar, the text reads 'If the Process fails, perform the following actions:'. There are two radio buttons: 'Use Default Options (as specified in Options -> Error Handling)' and 'Override Default Options', with the latter selected. Below this, there are three sections: 1. 'Send Email' (checked), containing 'Sender Address' (john@softomotive.com), 'Recipient Address' (peter@sofotmotive.com), and 'Attach Screenshot' (checked). 2. 'Run Another Process' (unchecked), containing a 'Process to Run' dropdown menu. 3. 'Write Event to Text File' (unchecked), containing a 'Text File Location' text box with a browse button (...). At the bottom, there are four checked checkboxes: 'Record event to windows event log', 'Set Process as erroneous', 'Add Screenshot to Logs', and 'Capture Video Log'. The 'Capture Video Log' checkbox has a spinner box set to '2' and the text 'The last: 2 minutes.'. At the very bottom, there are 'Help', 'OK', and 'Cancel' buttons.

Robots Properties - Error Handling Tab

Credentials

If you check the "Override Credentials" check box, additional fields will appear on the Tab, inviting you to enter Credentials that belong to a third person:

Process Properties of '1 - Run Calculator'

Execution Maker-Checker Concurrency Policy Error Handling Credentials

Override Credentials

Password mode: Enter password directly

User: John Anastasopoulos Select User

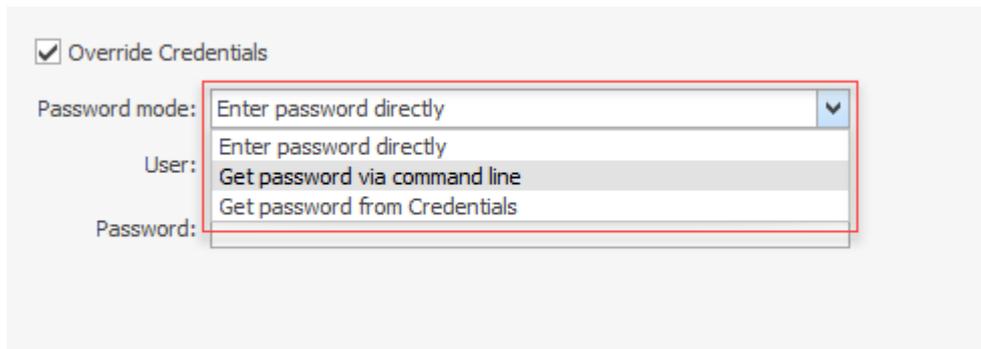
Password: *****

Confirm Password: *|

Help OK Cancel

This interface allows you to run a process as someone that is not you.

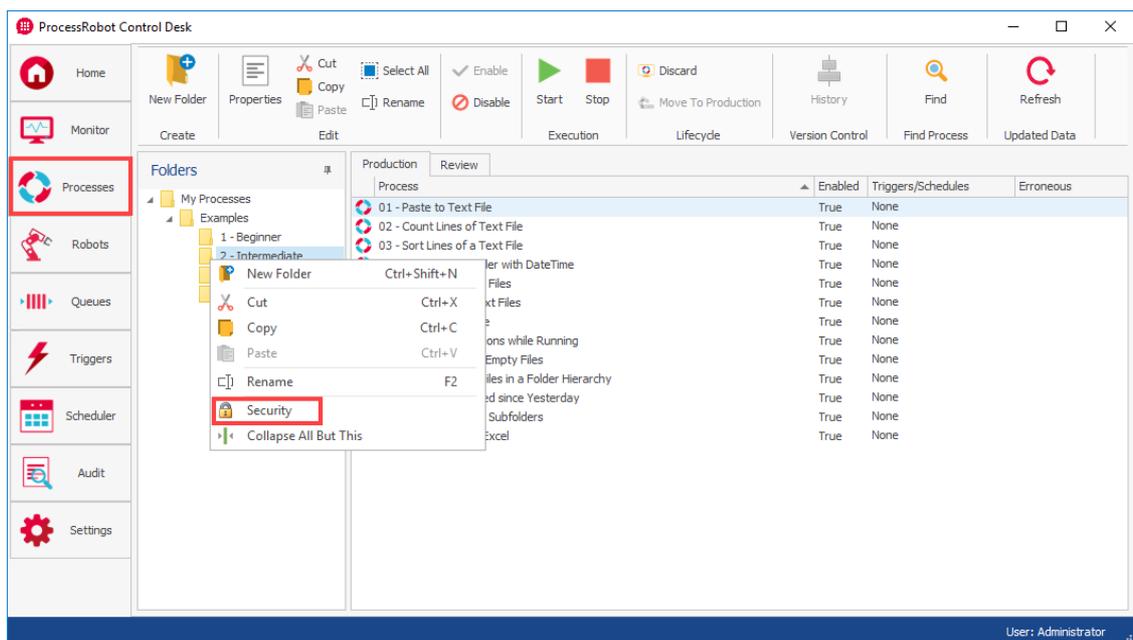
You can find that user through the Select User facility and you can access their encrypted password according to the option you have selected in Password mode:



Password mode allows you to Enter the password directly, Get password via command line or Get password from [Credentials](#)^[241] (if available). The 'Enter the password directly' option entails that the person you want to run the process as them, has to come and enter their password in person at your machine. Their password will not be visible on screen as it will be immediately encrypted through the [Master Key](#)^[191].

2.1.2.3,3 Processes Security

In the Processes tab of the Control Desk, right click on any folder in the Folders pane and select "Security" to add Security options for that folder.



The "Security: Roles and Permissions" window will appear.

Security: Roles and Permissions

Object: Folder 'Examples'

Roles:

Add Delete

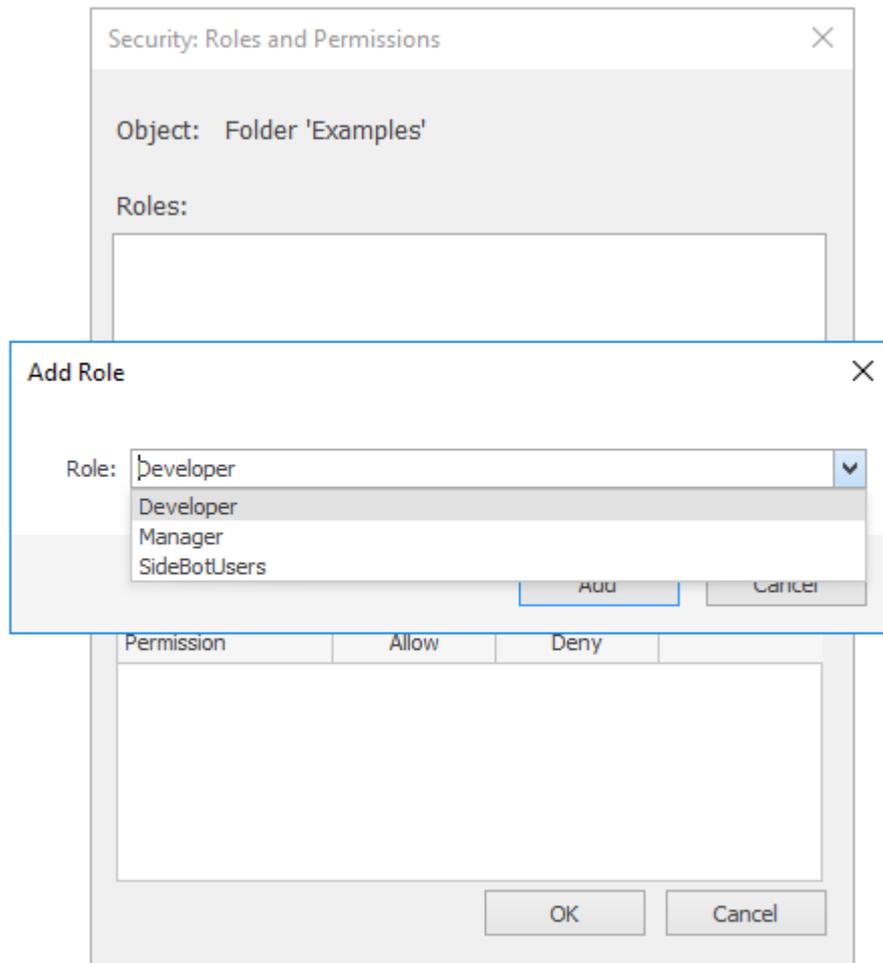
Permissions:

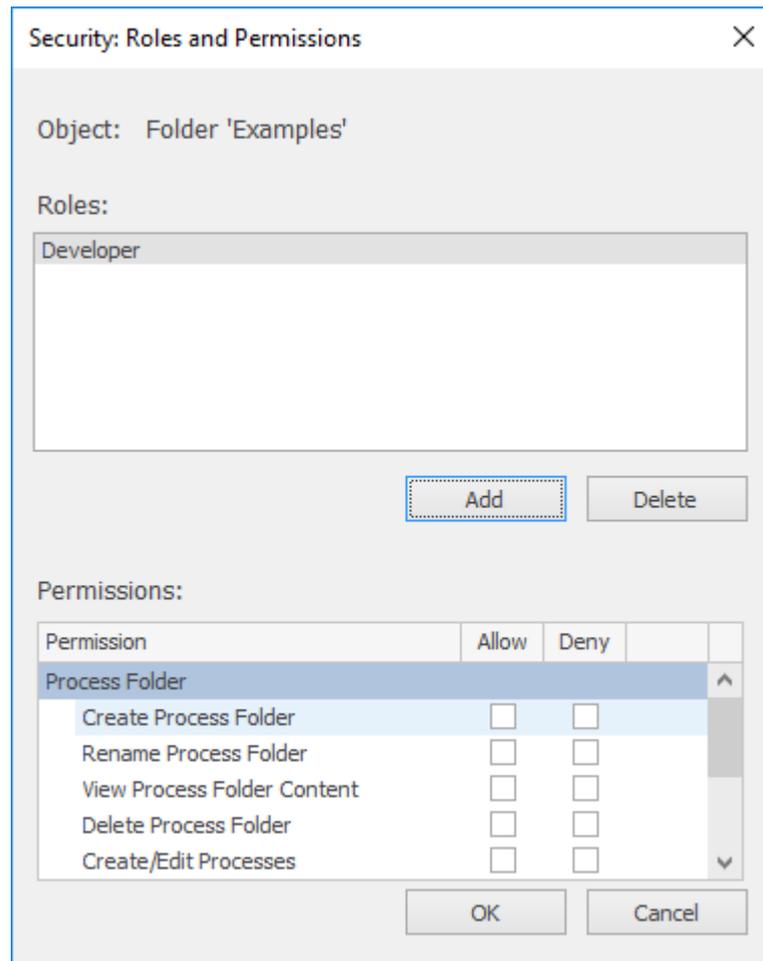
Permission	Allow	Deny
------------	-------	------

OK Cancel

The Object field will specify for which Folder the Roles and Permissions are being set.

To add a Role, click the "Add" button and select from one of the existing Process Robot Roles, created in the Control Desk's Settings > Roles tab.



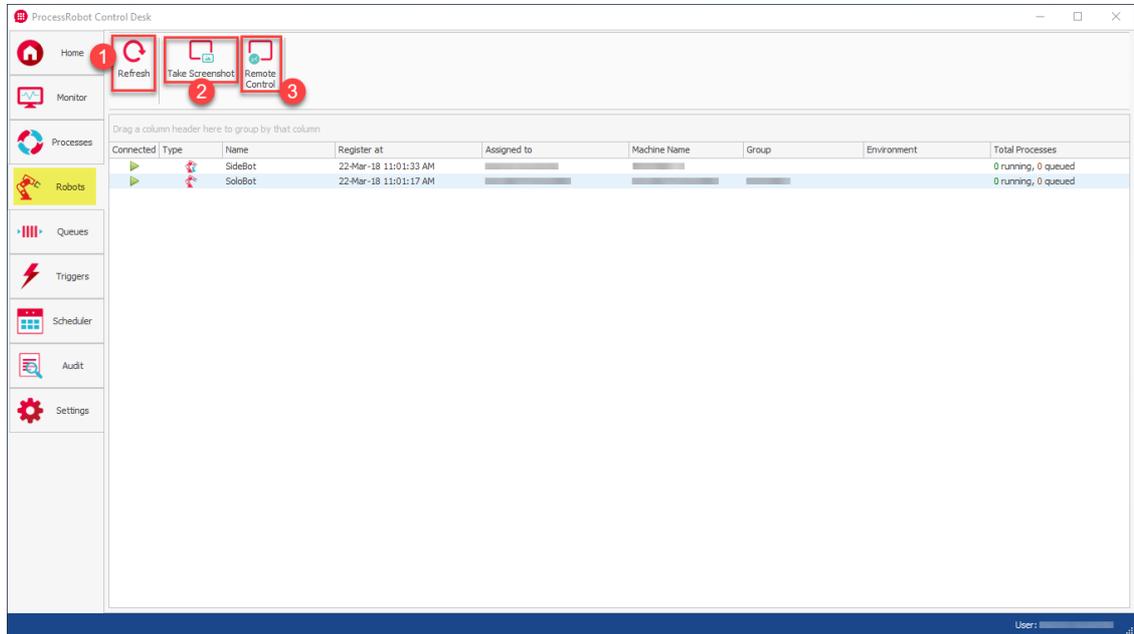


To assign Permissions to the selected Role, check whether to allow or deny the Role to:

- Create Process Folder
- Rename Process Folder
- View Process Folder Content
- Delete Process Folder
- Create/Edit Processes
- Execute Processes
- Import/Export Processes
- Set Processes For Review
- Move Processes To Production
- Discard Processes

2.1.2.4 Robots

In the Robots tab you can see all the Robots (Sidebots and Solobots) that you have added along further details about them.



Robots Tab

If the Robot is connected there will be a green arrow in the "Connected" column, otherwise there will be a red cycle which is indicative of the Robots disconnection.

The type will have the Sidebot " " or Solobot " " icon for you to distinguish what kind of Robot is it.

You will also get information about where it is registered at, who it is assigned to, in which Group it belongs to, in which Environment, what is the current Running Process, and what is the total number of Processes Running and Queued.

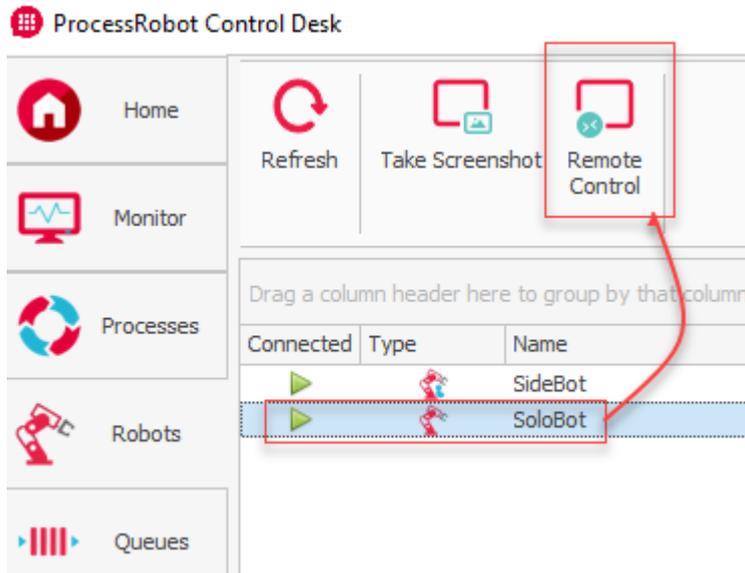
The Toolbar of this Tab allows you to either Refresh the screen [1] or to Take a Screenshot [2] for future reference in the press of a button.

If the selected Robot is an already connected Solobot machine, then the Remote Control option [3] will be enabled, that allows you to open the Remote Control window, and access the monitor(s) of the aforementioned machine. Please note that a Control Desk User will have access to either view only, or view and control (by using mouse and keyboard) the Solobot monitor(s), based on her/his User's Role and Permissions.

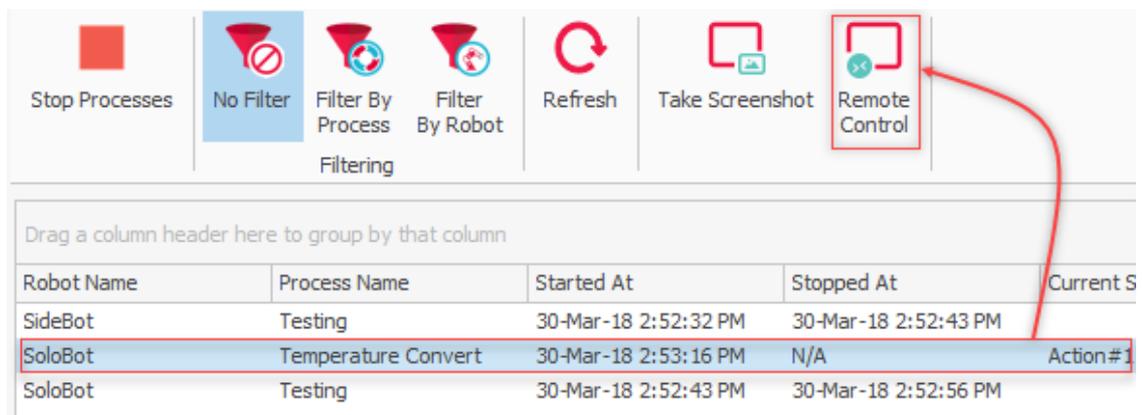
2.1.2.4,1 Remote Control

The Remote Control offers you the ability to access the Desktop of a connected Solobot. You can open that window either:

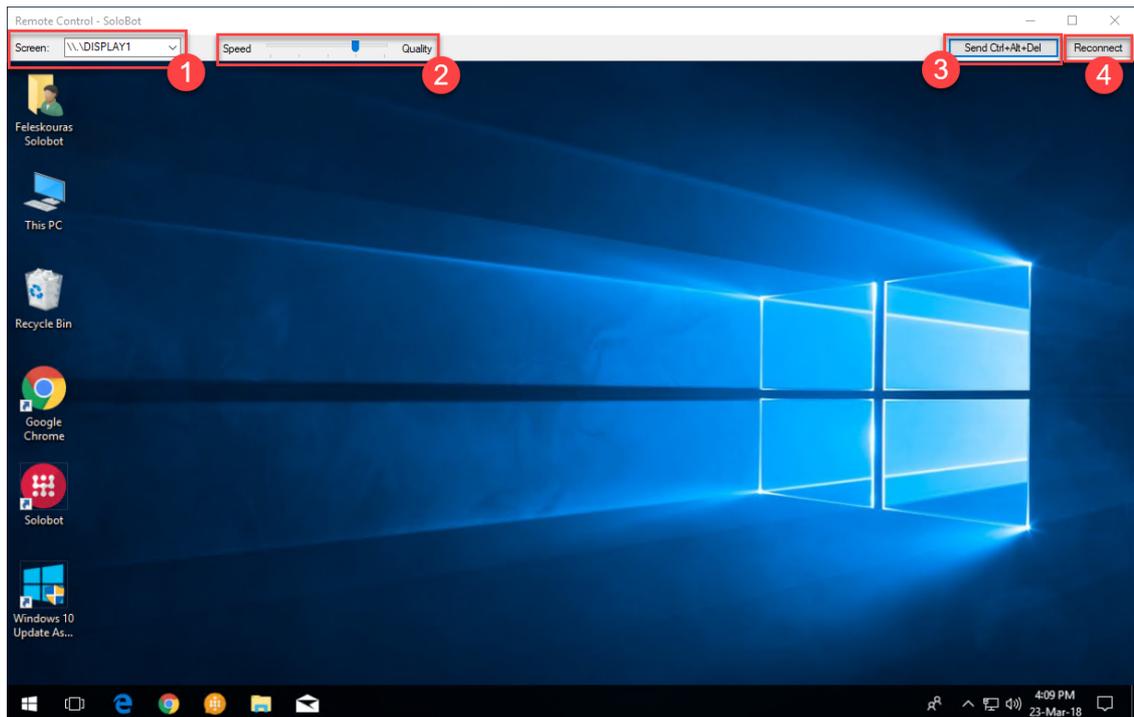
- through the [Robots](#)^[147] tab or,



- in case a Process is executing on a Solobot machine, through the [Monitor](#)^[121] tab of the ProcessRobot Control Desk.



Generally, the Remote Control window represents a Remote Desktop Protocol (RDP) session integrated inside the Control Desk, that the Control Desk Users with the corresponding permissions can use to access the machine where a Solobot is installed and check if a Process is being executed correctly in unattended mode.



The Remote Control window

The menu on the top of the Remote Control window consists of the following options:

1. **Screen.** In case the machine has multiple screens, you may choose the desired one.
2. **Speed - Quality** bar. You can adjust the bar so as to minimize the window's response time, or improve its resolution, according to your needs.
3. **Send Ctrl+Alt+Del** button, so as to unlock the remote machine.
4. **Reconnect** to the remote machine, in case the connection to the remote desktop is lost.

The Control Desk Administrator can determine whether a particular Control desk User can View only or View and Control the Robot Screen via mouse and keyboard, according to its [Role's Permissions](#)¹⁹³

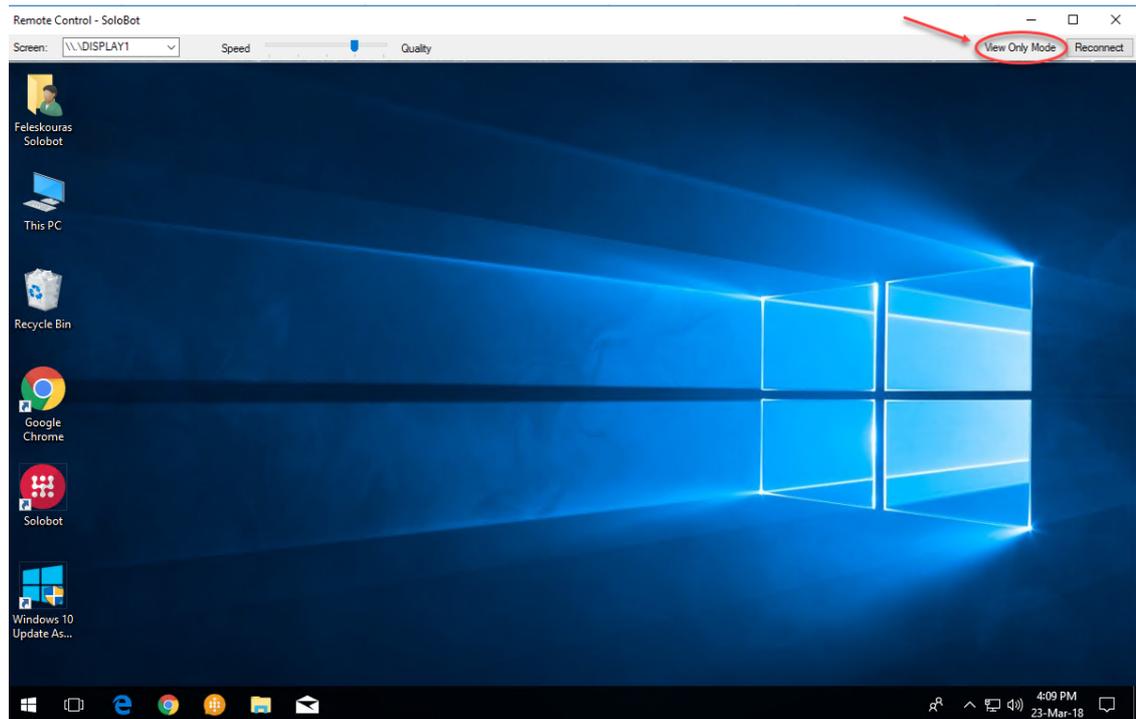
Edit Role [X]

Role Name:

Permissions:

Permission	Allow
Component	<input type="checkbox"/>
Access to Control Desk	<input type="checkbox"/>
Access to Process Studio	<input type="checkbox"/>
Access to Dashboard Designer	<input type="checkbox"/>
Access to Dashboard Viewer	<input type="checkbox"/>
Control Desk Tabs	<input checked="" type="checkbox"/>
Process Studio Tabs	<input type="checkbox"/>
Process	<input type="checkbox"/>
User Libraries	<input type="checkbox"/>
Process Folder	<input type="checkbox"/>
User Library Folder	<input type="checkbox"/>
Trigger	<input type="checkbox"/>
Schedule	<input type="checkbox"/>
Robot	<input checked="" type="checkbox"/>
Execute Process on Robot	<input type="checkbox"/>
Take Screenshot of Robot	<input type="checkbox"/>
View Robot Screen	<input checked="" type="checkbox"/>
View and Control Robot Screen	<input type="checkbox"/>
Robot Pool	<input type="checkbox"/>

The Remote Control window of a User that is allowed to only view the Solobot machine should look like:



Remote Control - View Only Mode

Please note that the '**Send Ctrl+Alt+Del**' button has been replaced with the '**View Only Mode**' notification, as long as that User is not allowed to control the screen.

2.1.2.5 Queues

Queues can be considered as a basket of items. A process can "[Add Queue Item in Queue](#)"¹⁰¹⁰ filling it up, and then another process can "[Get Queue Item from Queue](#)"¹⁰¹² in order to process it.

ProcessRobot can offer two ways of Queue Items Handling:

1. Automatic distribution of Queue Items:

In this case as long as the Queue has items for processing, a dedicated Process is automatically handling them non stop.

2. Manual Queue Items consuming.

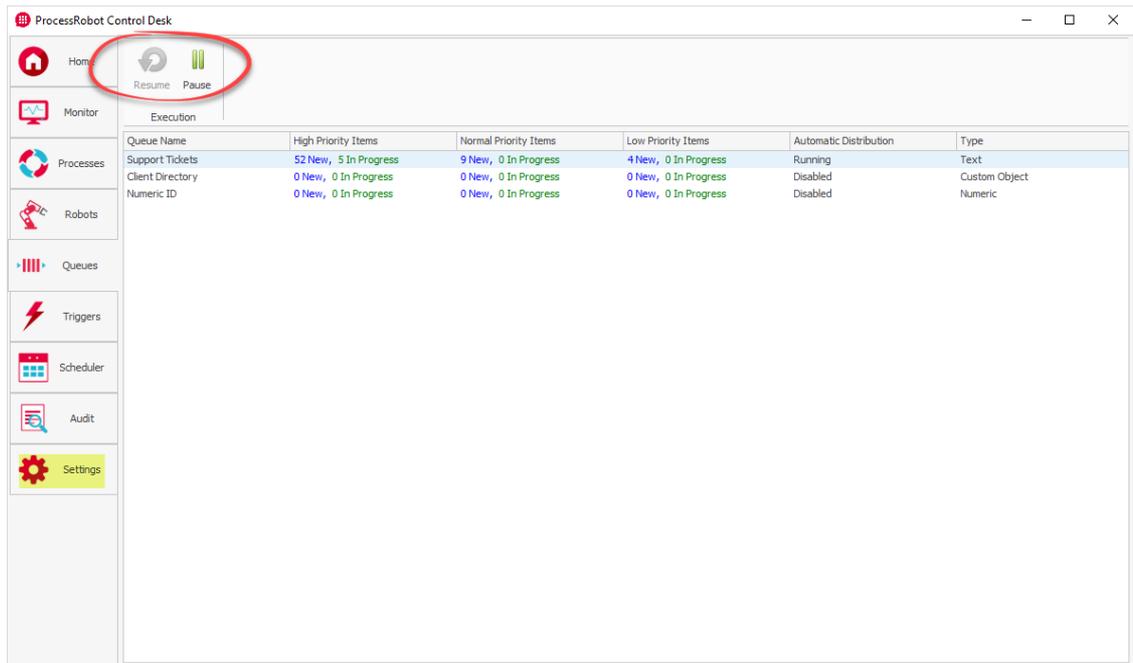
In this case the Queue is filled with items, which may be consumed on demand at any point from any process, with no automatic distribution.

The above is also the reason why the "[Get Queue Item from Queue](#)"¹⁰¹² action in the Process Designer has two options a) **First Queue Item Of Queue** and b) **Queue Item that Initiated this Process**. You may find more information in the topic "[How Queues Work](#)"²²⁴

Back in the Queue Tab, you can see how many Items your Queues have, the "Queue Name" the "Type" of the Items and whether the "Automatic Distribution" is active or not.

The "Automatic Distribution" will run the Process that is specified in the "Process to Run" option in the [Settings > Queues](#) when you created the Queue.

At any point, you can "Resume" or "Pause" a certain Queue. If you Pause a Queue it will no longer fire the dedicated Process regardless of the fact that there are Work Items in the Queue. It will continue running the Process and getting Items from the Queue only once you hit the "Resume" button.



The Queues Tab

2.1.2.6 Triggers

2.1.2.6.1 What Triggers Are

Triggers, help you to run a ProcessRobot Process **automatically**, without the user's interference. For example, if you want something to happen when:

- a file is created,
- a service is starting/stopping,
- a process is starting/stopping,
- an email arrives in an email account with the subject, To, From, Body containing a text,
- an event occurs in the Event log,
- a host does not respond to pinging,

- a certain operation takes place in a database,
- the performance of a workstation meets certain criteria, or
- the machine is idle for a certain time.

Triggers work whenever something happens, even if there is no user logged into the computer at that time (obviously you need to have already created a Solobot to enable this kind of functionality).

Triggers can be disabled universally from the Triggers tab in the Control Desk after selecting "Select All" and "Disable Triggers". This means that no Triggers will work until you change this setting. If you want to disable the Triggers for one Process without deleting the Triggers, right-click on the Process and choose Disable Process. Now this Process will not run, manually or through Triggers, until you Enable that Process again.

2.1.2.6,2 Trigger Variables

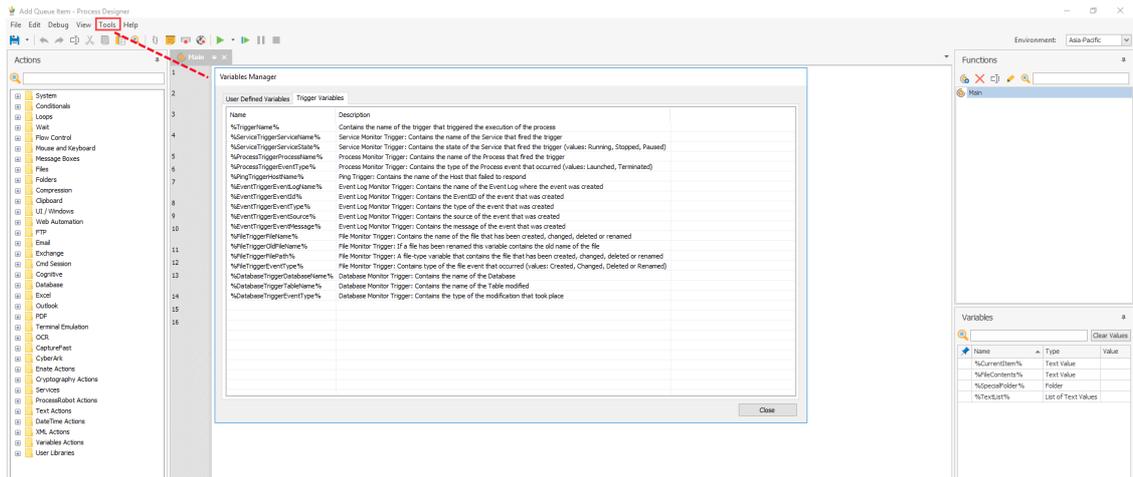
All Processes have several predefined built-in Variables that contain information according to what Triggered them.

For example, all Triggers will populate %TriggerName% with the name of the Trigger that fired it.

If a Process was executed manually (whether or not it has Triggers attached), the contents of this Variable will be an empty text.

In the same spirit, a Process Trigger will populate %TriggerName%, %ProcessTriggerProcessName%, and %ProcessTriggerEventType% with the name of the Trigger that fired the Process (a Process Trigger), the name of the Process, and the Event that fired it (Launched or Terminated). These Variables will be filled at the beginning of the Process, so you can access them throughout the Process to notify the user.

You can find all Trigger Variables and their descriptions through [Tools -> Variables Manager](#) in the Process Designer Window in the Process Studio.

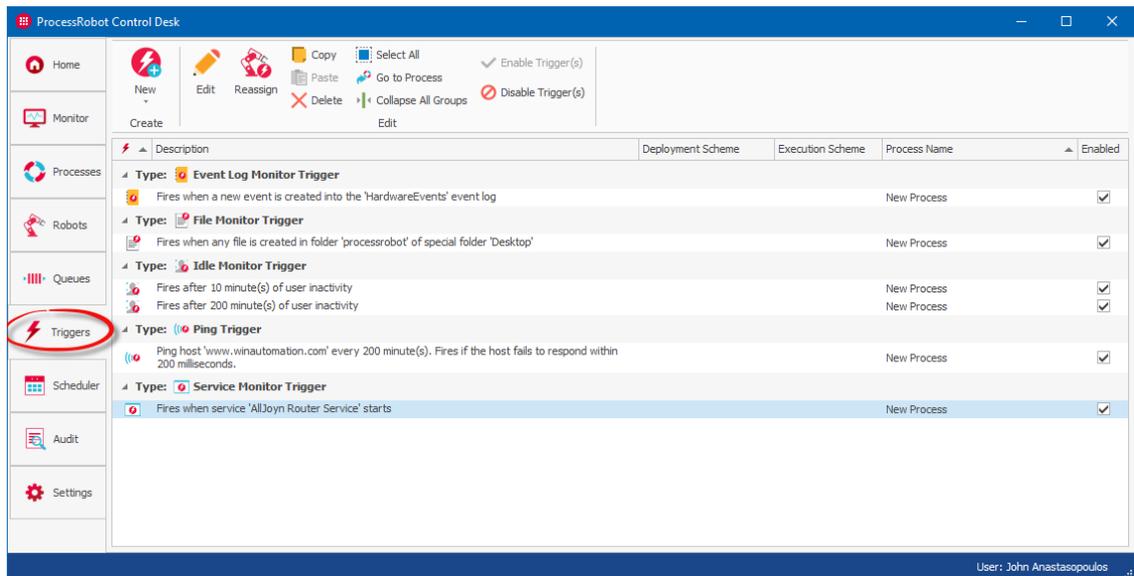


Variables Manager Window

2.1.2.6,3 Triggers tab

In the Triggers tab on the ProcessRobot Control Desk you are able to create and assign Triggers to your Processes.

You can [assign a Trigger to a Process](#) ¹⁵⁵ by clicking on the "New" option in the Create menu group.



Control Desk > Triggers Tab

In the Edit menu group you can:

Edit the trigger of a Process: Select the trigger that you want to edit from the Triggers list in the pane below, click on edit and modify it.

Reassign a Trigger to a Process: Select a trigger from the Trigger list pane and then click on the "Reassign" option. This will pop-up a dialog, so that you will choose the Process on which you want to reassign the Trigger.

Copy/Paste/Delete Trigger: Select a trigger from the Trigger list pane and copy/paste it. Once you select the paste option the "Assign Trigger to Process Window" will pop-up, prompting you to select the Process to which you want to Paste the Trigger to. To Delete a Trigger simply select it from the Triggers list pane and click on the delete option.

Select All: This option allows you to select all the triggers in the Trigger list pane.

Go to Process: Clicking on this option, after selecting a Trigger from the Trigger list pane, will take you directly to the Processes tab highlighting the Process.

Collapse All Groups: It will collapse the details of all the Trigger groups in the Trigger list pane.

Enable/Disable Trigger: Click on a Trigger in the Trigger list pane and choose whether you want to Enable or Disable it. By Disabling a Trigger it will no longer fire the Process until you Enable it again. You can also disable a Trigger for a Process if you right click on it and select "Disable Process".

In the Trigger list pane you can see the type of the Triggers, the Description, the Deployment Target, the Execution Target, The Process Name that they are assigned to and if they are Enabled or not.

The Deployment and Execution Targets are explained in the following [topic](#)^[155].

2.1.2.6,4 How to set a Trigger to a Process

Triggers are attached to a Process outside of the Action steps and so in the Control Desk, not in the Process Designer.

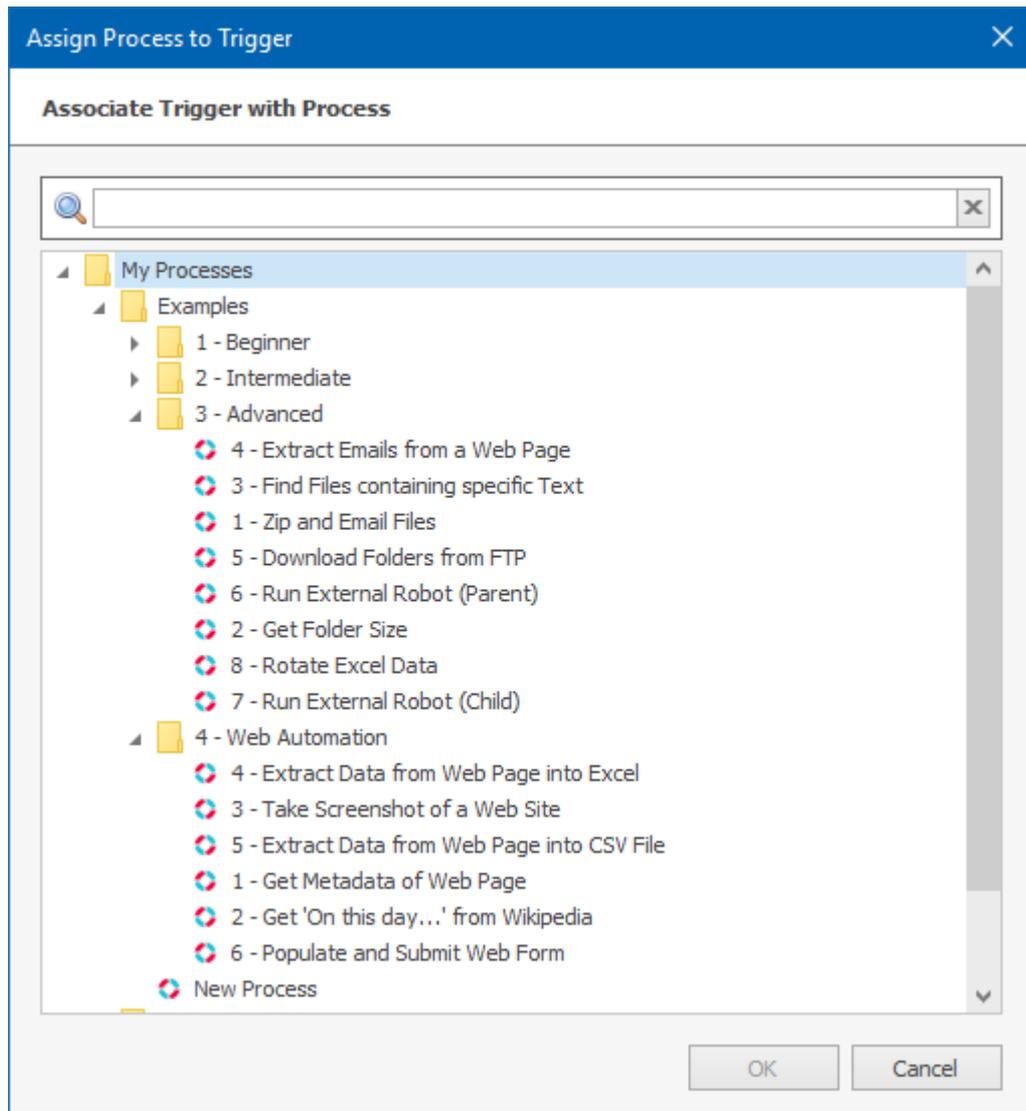
To create a Trigger and attach it to an existing Process, click on the "Triggers" tab and in the Create menu group select "New". In the menu that pops up will have to choose the type of the Trigger that you want to attach to a Process.

All the Trigger Types are described in detail in the "Trigger Types" topic:

- [Email Monitor Trigger](#)^[159]

- [Exchange Email Monitor Trigger](#) 
- [Event Log Monitor Trigger](#) 
- [File Monitor Trigger](#) 
- [Idle Monitor Trigger](#) 
- [Ping Trigger](#) 
- [Service Monitor Trigger](#) 
- [Process Monitor Trigger](#) 
- [Database Monitor Trigger](#) 
- [Performance Counter Trigger](#) 

After selecting and configuring the Trigger you wish, a Dialog pops up where you are prompted to select the Process to which you will attach the trigger.



Assigning Process to a Trigger window

Once you have created the Trigger, then you will see it in the Triggers tab.

Example:

Let say that you wish to monitor the Desktop for any .txt file creation on one Sidebot and run a Process on all Robots included in the Marketing Robot Pool. The Trigger should look like the following:

File Monitor Trigger Properties

This trigger monitors files in a Folder (and subfolders, if chosen), and fires if a file is created, modified, changed, or deleted within that folder.

Custom Description:

*** This trigger will monitor the Desktop for any .txt file creation ***

Folder to Monitor: Special Folder Desktop

Include Subfolders

File Events to Monitor

Created Changed Deleted Renamed

Files to Monitor: *.txt

Files to Exclude:

Enter wildcard filters for files you want to monitor or exclude.
You may enter more than one filters separated by ';' e.g.: *.txt;*.csv

Help OK Cancel

Desktop monitor Trigger for .txt files

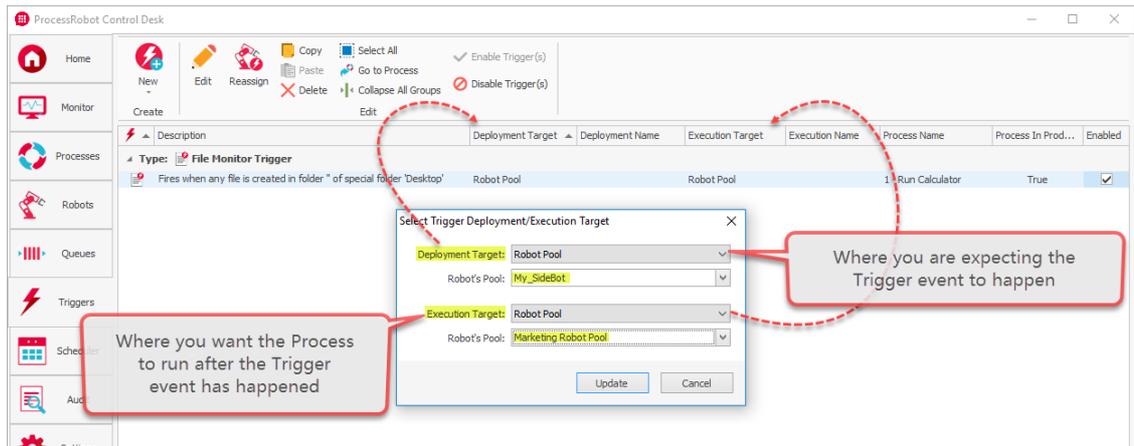
There are two kind of Targets in ProcessRobot as far as Triggers are concerned. The **Deployment** Target and the **Execution** Target.

As a Deployment Target, you can indicate where you want the trigger event to take place, or in our example on which machine(s) you are monitoring the Desktop for the creation of a *.txt file.

The Execution Target is where you want the Process to run, after the trigger event has occurred.

So, the trigger event will happen on the Deployment Target but the Process will run on the Execution Target.

Therefore, if you want to monitor the desktop of a specific Robot, i.e a Sidebot, then you should create a Target that will include this Sidebot and set it as Deployment Target. Having done so, you should then specify the Execution Target, which will include all robots the Process will run on, once the trigger event has happened on the Sidebot (which is the Deployment Target).



Deployment and Execution Scheme of a Trigger

Important Note: *In order for Triggers to work, you MUST make sure that the Process on which you have assigned the Trigger is in the "Production" state of its lifecycle. Else, the Triggers will not fire.*

2.1.2.6,5 Trigger Types

Email Monitor Trigger

The Email Monitor Trigger periodically checks an email account and executes a Process when an email that meets the specified criteria arrives.

Email Monitor Trigger Properties

This trigger fires when specific emails arrive.

Custom Description:

Check Email Every Minute(s)

Email Account Settings:

Imap Server:

Port:

Use SSL

Username:

Password:

Check For Emails where ...

Status is:

Mail Folder is:

"From" Field Contains:

"To" Field Contains:

Subject Contains:

Body Contains:

Help OK Cancel

Email monitor Trigger

- **Custom Description**

Add a custom description to the trigger which will be shown in the Triggers tab under "Description".

- **Check Email Every**

In the trigger's configuration dialog you specify how often the trigger should check for new emails. The default value is every 5 minutes.

- Email Account Settings

For ProcessRobot, to monitor the email messages on an email server (as also to retrieve and process them), it must be able to access the server through the IMAP protocol. For more information you may check the corresponding session at the [Retrieve Emails](#)^[797] action.

Therefore, in order set a Email Monitor Trigger, it is necessary that you enter the IMAP Server Connection properties, consisting of:

- the *IMAP Server* property, where you enter the IMAP Server address (e.g. imap.gmail.com).
- the *Port*, where you specify the communication port, with the most commonly used Server Port for IMAP being 993. However some IMAP Servers may require to specify different Port in order to perform properly.
- whether *SSL* should be enabled, so that communication with the client could be performed over a secure connection (as required by some email servers).
- *User Name*, where you enter the username of the e-mail account you want to access.
- *Password*, where you enter you email account's password.

- Check for Emails Where...

In the next group of fields, you need to specify the values of the email properties that will fire the trigger. Leaving all the fields of this group empty will cause the trigger to fire and the Process will be executed for every email that arrives into the mailbox.

In order to restrict the type emails that will fire the trigger you should enter specific values in the appropriate fields defining that you want to check:

- all emails or only for unread (*Retrieve* property field)
- for emails that end up into a specific folder (*Mailbox Folder is:* property field)
- for emails coming from a specific sender (*"From" Field Contains:* property field)
- for emails being addressed to a specific recipient (*"To" Field Contains:* property field)
- for emails that contains specific keywords in the subject (*Subject Contains:* property field)
- for emails that contains specific keywords in the body (*Body Contains:* property field)

Of course you may enter your preferences for more than one of the above properties and combine trigger firing criteria.

Note that the Email Monitor Trigger will only check for the specified email types and not retrieve them from the email server (where they will consequently remain marked as "unread").

So, if in the Process that is executed by the trigger you need to process the email(s) that caused the Process to run, you should use the [Retrieve Emails](#)^[797] and the [Process Emails](#)^[806] actions.

Exchange Email Monitor Trigger

The Exchange Email Monitor Trigger periodically checks an Exchange email account and executes a Process when an email that meets the specified criteria arrives.

Exchange Email Monitor Trigger Properties [X]

This trigger fires when specific emails arrive.

Custom Description:

Check Email Every Minute(s)

Email Account Settings:

Exchange Version: Exchange 2010

Connection Type: Autodiscover

Email Address:

Credentials: Exchange Default

Mailbox Type: Personal

Check For Emails where ...

Status is: Any (All Emails)

Mail Folder is:

"From" Field Contains:

"To" Field Contains:

Subject Contains:

Body Contains:

Help OK Cancel

- **Custom Description**

Add a custom description to the trigger which will be shown in the Triggers tab under "Description".

- **Check Email Every**

In the trigger's configuration dialog you specify how often the trigger should check for new emails. The default value is every 5 minutes.

- Email Account Settings

For ProcessRobot, to monitor the email messages on an email server (as also to retrieve and process them), it must be able to access the Exchange server. For more information you may check the corresponding session at the [Connect To Exchange Server](#)^[812] action.

Therefore, in order set an Exchange Email Monitor Trigger, it is necessary that you enter the Exchange Server Connection properties, consisting of:

- the *Exchange Version* property, where you enter the Exchange Server Version.
- the *Connection Type*, where you specify whether Autodiscover will be used, or the Exchange Server Address will be provided. If Autodiscover is used, an Email Address will have to be provided; if not, then the Exchange Server Address is required.
- Whether the credentials to be used will be the default Exchange credentials, or user defined; in the latter case, the Domain, Username and Password will have to be provided - the Password may be provided directly or through the command line.
- Whether the mailbox is personal or shared.

- Check for Emails Where...

In the next group of fields, you need to specify the values of the email properties that will fire the trigger. Leaving all the fields of this group empty will cause the trigger to fire and the Process will be executed for every email that arrives into the mailbox.

In order to restrict the type emails that will fire the trigger you should enter specific values in the appropriate fields defining that you want to check:

- all emails or only for unread (*Retrieve* property field)
- for emails that end up into a specific folder (*Mailbox Folder is:* property field)
- for emails coming from a specific sender (*"From" Field Contains:* property field)
- for emails being addressed to a specific recipient (*"To" Field Contains:* property field)
- for emails that contains specific keywords in the subject (*Subject Contains:* property field)
- for emails that contains specific keywords in the body (*Body Contains:* property field)

Of course you may enter your preferences for more than one of the above properties and combine trigger firing criteria.

Note that the Exchange Email Monitor Trigger will only check for the specified email types and not retrieve them from the email server (where they will consequently remain marked as "unread").

So, if in the Process that is executed by the trigger you need to process the email(s) that caused the Process to run, you should use the [Retrieve Exchange Email Messages](#)^[816] and the [Process Exchange Email Messages](#)^[822] actions.

Event Log Monitor Trigger

The Windows Event Log is a Log that many programs write to when they start, stop, and/or fail. If you wish to monitor your Windows Event Log for some action, this is how you do it.

Custom Description:

A custom description that you want to set for this Trigger.

Event Log Name - Event Type - Event ID - Event Source:

You will be able to choose which Log you want to watch, and what type of Event to watch for (Information, Error, Warning, Audit Success, Audit Failure, or all of them). You can also choose the Event ID (an ID associated to a specific event) or the Event Source (a name chosen by the application, not necessarily its executable name - you will have to look through the Logs to find this possibly).

Event Message must (Not) contain:

Finally you can set the Trigger to fire only for events that contain a specific text within their message or alternatively to ignore events that contain a specific text.

EventLog Monitor Trigger

The Event Log Monitor Trigger, when fired, assigns values to five Trigger Variables that are made available to the Process:

%EventTriggerEventLogName% which will contain the name of the Log where the event was created,

%EventTriggerEventID% which will contain the Event ID of the event that was created,

%EventTriggerEventType% which will contain the type of the event that was created,

%EventTriggerEventSource% which will contain the source of the event that was created, and

%EventTriggerEventMessage% which will contain the message of the event that was created.

As you can have as many Triggers as you want attached to the same Process, this would allow you to know which Event fired the Process. You can access this information through Actions using those Variables.

File Monitor Trigger

This Trigger monitors files in a Folder (and subfolders, if chosen) and fires if a file is created, modified, changed, or deleted within the folder(s). Specify the event(s) to look for, and file types you wish to include or exclude; wildcards can also be used here. This allows monitoring of a folder to see if any file of the type .doc is created, for example, or any file other than an executable file is changed.

File Monitor Trigger

The File Monitor Trigger, when fired, assigns values to four Trigger Variables that are made available to the Process:

%FileTriggerFileName% which will contain the name of the file that fired the Trigger,

%FileTriggerOldFileName% which will contain the original name of the file if it has been renamed,

%FileTriggerFilePath% which will contain the complete path (along with the filename) of the file. Note that this value is a File object meaning that has the [properties](#)⁴⁴⁰ of every file object. For example you can retrieve the size of the file that triggered the Process by specifying: %FileTriggerFilePath.Size%

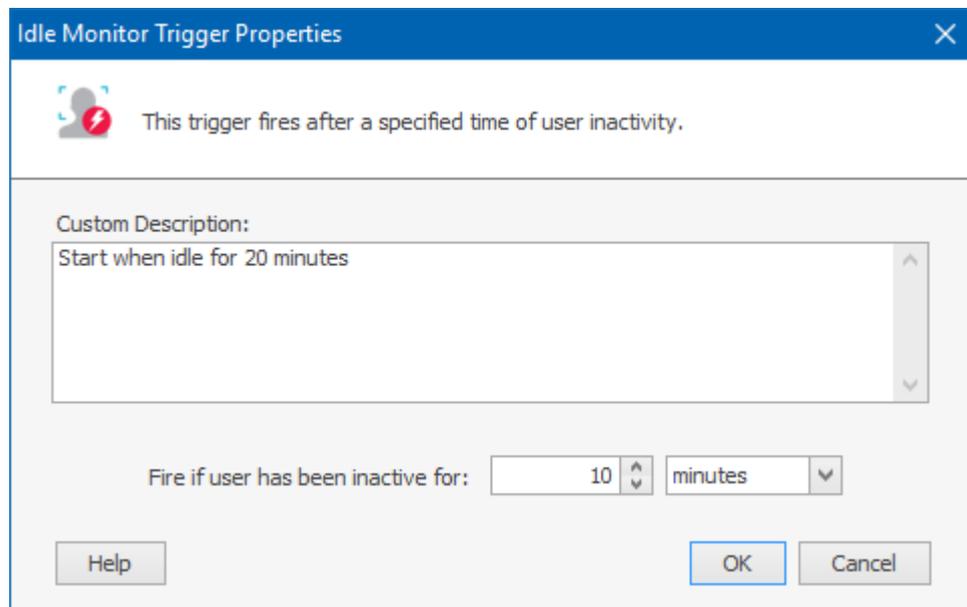
and

%FileTriggerEventType% which will contain 'Created' or 'Changed', 'Deleted' or 'Renamed'.

You can create Actions in the Process to show this information, for example you can create a Process Trigger that fires when an executable file in the Windows directory (with subfolders) is changed, and the Process then puts up a Message Box informing you of what happened.

Idle Monitor Trigger

This Trigger fires once the machine has had no input for a set amount of time that you choose in minutes, hours, or seconds. This can be useful for resource-intensive Processes that you only want to run when you are not using the computer, for example. This Trigger will not fire if no user is logged in.



Idle Monitor Trigger

Ping Trigger

This Trigger will 'Ping' a host (usually an IP address or a URL) at the interval you set. If the host **does not** respond within the Timeout you set, this registers as a failed Ping attempt, and the assigned Process is Triggered.

Once the host fails to respond and Triggers the Process, subsequent failures will not fire the Process again until a Ping attempt is successful. Thus, for example, if the host responds 5 times, then fails 3 times, then responds once and fails again, this Trigger will fire after the 6th attempt (first failure) and after the 10th attempt (first failure after a new success).

Ping Trigger Properties

This trigger pings a specified host and fires if the host fails to respond within a set timeout.

Custom Description:
Ping Softomotive

Host To Ping:

Timeout: Milliseconds

Ping Every: minutes

Help OK Cancel

Ping Trigger

The Ping Trigger, when fired, assigns values to a Trigger Variable that is made available to the Process:

%PingTriggerHostName%, which will be filled with the name of the host that failed to respond.

As you can have as many Triggers as you want attached to the same Process, this would allow you to know which Ping Trigger fired the Process, and thus which host went down. You can access this information through Actions using this Variable.

Service Monitor Trigger

The Service Monitor Trigger fires when the state of a specific Windows Service changes. Enter or choose a Service that exists on your computer, and select for it to Trigger when that Service starts, pauses, or terminates.

Service Monitor Trigger

The Service Trigger, when fired, assigns values to two Trigger Variables that are made available to the Process:

%ServiceTriggerServiceName%, which will be filled with the name of the Service that Triggered the Process, and

%ServiceTriggerServiceState%, which will contain 'Running', 'Paused', or 'Stopped'.

As you can have as many Triggers as you want attached to the same Process, this would allow you to know which Service Trigger fired the Process, and why.

Process Monitor Trigger

This Trigger monitors a certain process and fires when this process starts or terminates. Enter or choose a Process that exists on your computer, and select for it to Trigger when that Process starts or terminates.

Process Monitor Trigger

The Process Trigger, when fired, assigns values to two Trigger Variables that are made available to the Process:

%ProcessTriggerProcessName%, which will be filled with the name of the Process that Triggered the Process, and

%ProcessTriggerEventType%, which will contain either 'Launched' or 'Terminated'.

As you can have as many Triggers as you want attached to the same Process, this would allow you to know which Process Trigger fired the Process, and why. You can access this information through Actions using those Variables.

Database Monitor Trigger

The Database Monitor Trigger monitors a certain database and fires a Process when any of the selected operations is performed on it.

Database Monitor Trigger Properties [X]

This trigger monitors a database and fires when a selected operation happens.

Custom Description:

Database Type: **SQL Server**

Server:

Username:

Password:

Database Name:

Table:

Operations: Insert Update Alter
 Delete Drop

The File Monitor Trigger, when fired, assigns values to three Trigger Variables that are made available to the Process:

%DatabaseTriggerDatabaseName%, which contains the name of the database,

%DatabaseTriggerTableName%, which contains the name of the modified table, and

%DatabaseTriggerEventType%, which contains the operation that triggered the Process.

As you can have as many Triggers as you want attached to the same Process, this would allow you to know which Database Monitor Trigger fired the Process, and why.

Custom Description

Add a custom description to the trigger which will be shown in the Triggers tab under "Description".

Database Type

Choose between SQL Server and Oracle databases; in the latter case, the Port and Notification Port will also have to be entered.

Username

Enter the name of a User with access to the database.

Password

Enter the User's password.

Database Name

Specify the name of the database to be monitored.

Table

Enter the name of the specific table to be monitored.

Test Connection

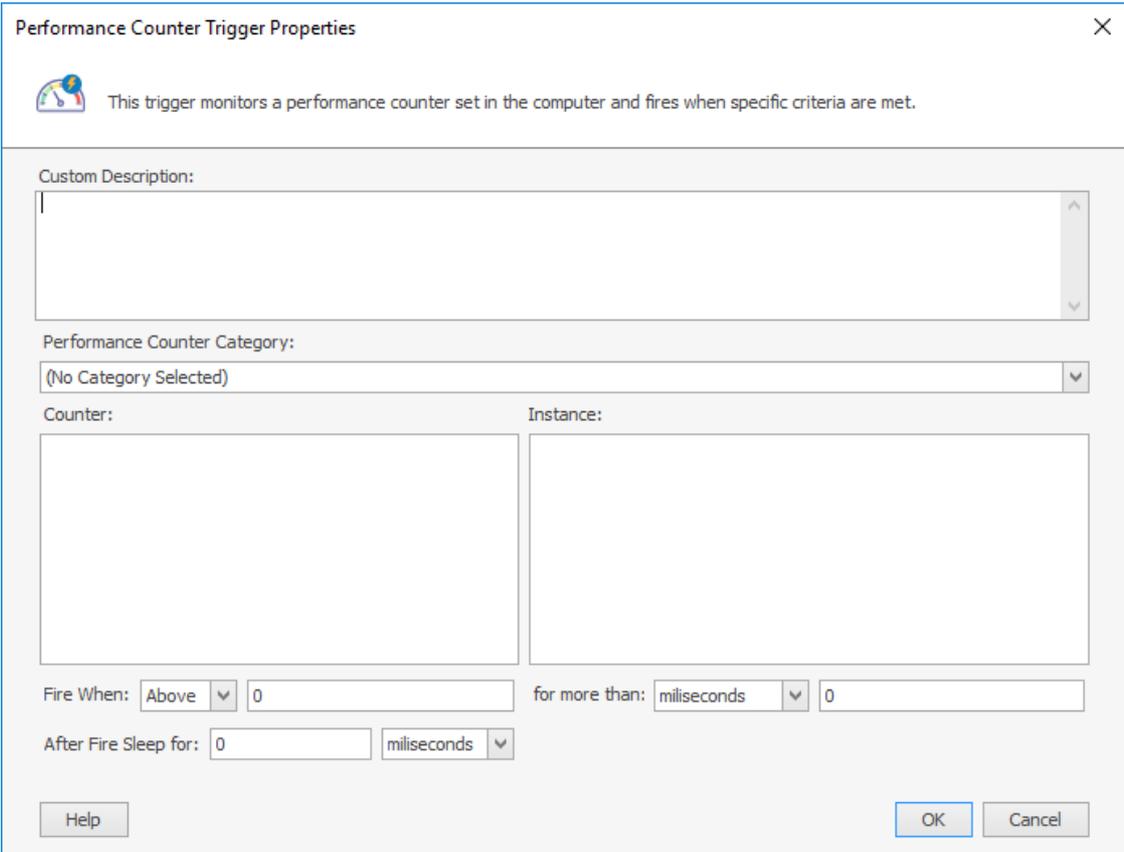
Use the Test Connection button to ensure that ProcessRobot is able to access the database using the entered information.

Operations

Select the types of operations which should be monitored.

Performance Counter Trigger

The Performance Counter Trigger monitors the workstation's performance and fires a Process when certain criteria are met.



The image shows a dialog box titled "Performance Counter Trigger Properties". At the top, there is a description: "This trigger monitors a performance counter set in the computer and fires when specific criteria are met." Below this, there is a "Custom Description:" field with a text area. Underneath is a "Performance Counter Category:" dropdown menu currently set to "(No Category Selected)". Below that are two empty text boxes labeled "Counter:" and "Instance:". At the bottom, there are three input fields: "Fire When:" with a dropdown set to "Above" and a value of "0"; "for more than:" with a dropdown set to "milliseconds" and a value of "0"; and "After Fire Sleep for:" with a value of "0" and a dropdown set to "milliseconds". There are "Help", "OK", and "Cancel" buttons at the bottom.

Custom Description

Add a custom description to the trigger which will be shown in the Triggers tab under "Description".

Performance Counter Category

Choose the category of the counter to be monitored.

Counter

Select the counter to monitor.

Instance

Select which instance of the selected counter will be monitored.

Fire when:

Configure the exact criteria to determine when the Trigger should fire: When the monitored value is Above or Below a certain threshold value, for more than a set number of time units.

After Fire Sleep for:

Specify a cooldown period for the Trigger.

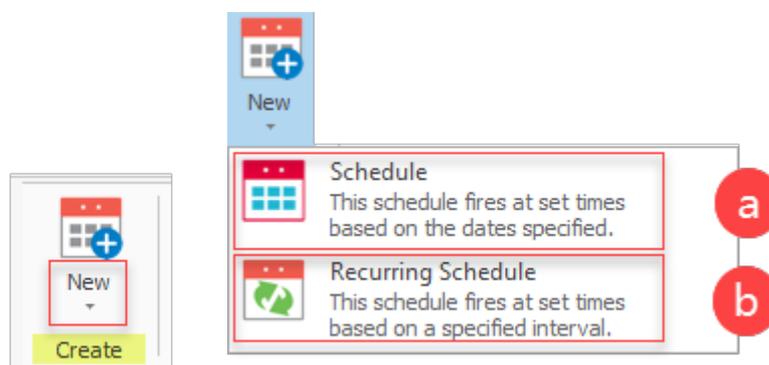
2.1.2.7 Scheduler

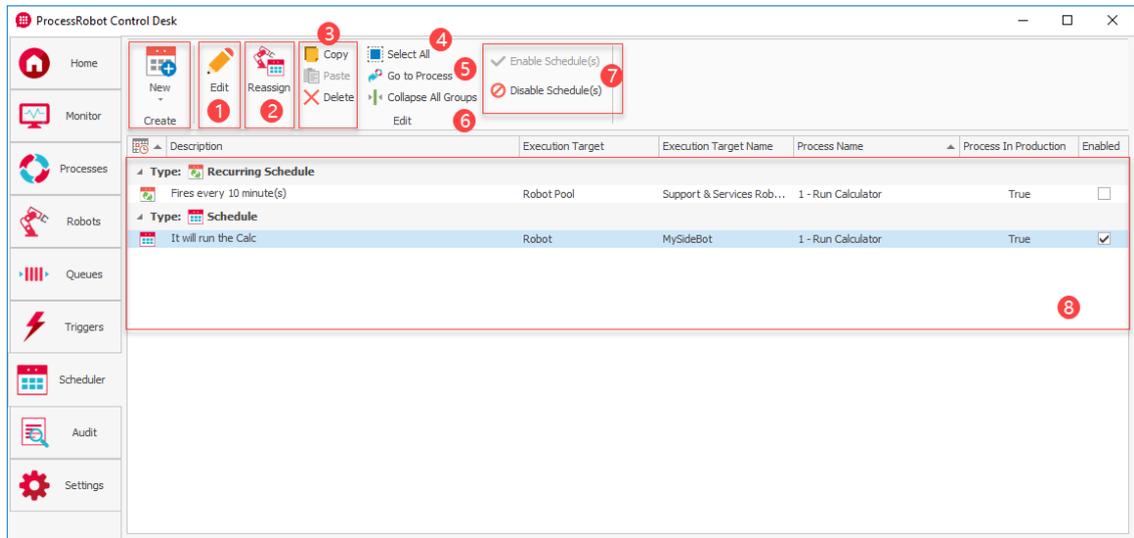
A Schedule can be set for a Process if we want it to run it at specific days/times, or keep running it in time intervals using a recurring Schedule.

Schedules, like Triggers, can be disabled universally from the Schedulers tab after selecting "Select All" and "Disable Schedule(s)". This means that no Process will run on its Schedule, until you change this setting. If you want to disable the Schedule for one Process without deleting it, right-click on the Process and choose Disable Process. Now this Process will not run, manually or on its Schedule, until you Enable it again.

2.1.2.7.1 Scheduler Tab

In the Scheduler tab of the Control Desk you can set Processes to run either on specific time through the creation of a [Schedule](#)^[174] [a] or in recurrent intervals through the creation of a [Recurring Schedule](#)^[176] [b].





The Edit menu group is the same as in the [Triggers tab](#) ¹⁵⁴

1. **"Edit"** the schedule (the one that is highlighted), should you wish to change its properties.
2. **"Reassign"** the schedule to a different Process.
3. **Copy, Paste, Delete** the selected schedule.
4. **"Select All"** schedules.
5. **"Go to Process"** will get you to the Processes tab highlighting the selected Process.
6. **"Collapse All Groups"**. It will collapse the "Recurring Schedule" and "Schedule" groups.
7. **Select the Schedule** (or Schedules if you hold down the ctrl button while clicking on the them) that you wish to **Disable** or **Enable**.

The main pane will show the Schedule type the description of the Schedule which is your the Custom Description that you have typed once you have set the Schedule, the Execution Target (whether it is a Pool or Robot), the Execution Target Name (the name of the Pool or Robot), the Process Name, and whether the Schedule is Enabled or not.

2.1.2.7,2 Schedule

Attaching a Schedule allows you to set the Process to run on a Daily, Weekly, Monthly basis or on Specific Dates.

Custom Description:

A custom description that you want to set for this schedule. This will just be your own words and it will have no effect on the Process.

Schedule Type:

Once you are on the Schedule Properties, you can choose to have the Process fire Daily, Weekly, Monthly or on Specific Dates. Each choice brings up several other choices, and those must be fully filled in to work. The last Working Day of Month is the last weekday, as a computer doesn't know the local holidays.

If Process is late:

The next choice is what to do if Process is late. You can choose to have the Process run immediately (usually on start-up, and this is a good way to have a Process fire once at the beginning of each day or whenever your computer is turned on), or to skip that firing of the Process.

Schedule Expires on:

Finally, if you choose to have the Schedule Expire, after the date and time you enter, the Schedule will still be there, but will not fire a Process when the Schedule comes up.

Schedule Properties

This schedule fires at set times based on the dates specified

Custom Description:

Schedule Type: Daily

At the Following Time(s):

12:35:57 PM

Add Time

Edit Time

Delete Time

If Process is late: Start Process immediately

Schedule Expires on 18-Apr-18 11:59:59 PM

Help OK Cancel

Schedule Properties

If a Schedule is attached to a Process then the date/time for the next run of the Process is displayed in the Next Scheduled Run column in the ProcessRobot console on the Processes Tab.

2.1.2.7.3 Recurring Schedule

A Recurring Schedule repeats at the given frequency. Once you have set this schedule, the Process will fire after that interval, so if you set a Recurring Schedule for 10 minutes, for example, it will fire for the first time 10 minutes after this schedule is set, and every 10 minutes after that until you change it.

Recurring Schedule Properties [X]

This schedule fires at set time intervals.

Custom Description:

Fire Every:

Schedule Starts on

Schedule Expires on

Custom Description:

A custom description that you want to set for this schedule.

Fire every:

Set the frequency at which the schedule should repeat.

Schedule Starts on:

Set the starting point of the recurring schedule.

Schedule Expires on:

Set the ending point of the recurring schedule.

If you choose to have the Schedule Expire, after its expiration the Schedule will still be there, but will not fire a Process.

If a Recurring Schedule is attached to a Process then the date/time for the next run of the Process is displayed in the Next Scheduled Run column in the WinAutomation Console in the Processes Tab.

2.1.2.8 Audit

The ability to audit the activity of your enterprise grade RPA is tremendously important for businesses. Enterprises often rely on external and internal auditing in order to ensure good order and compliance to standards. ProcessRobot is offering you a dedicated facility that keeps detailed logs of your automated activities, along with the capability to view additional details.

2.1.2.8.1 ProcessRobot Records (Logging)

In ProcessRobot, all the Processes that run through Robots are logged so that you can look at the history of any or all Processes.

Process completion are listed as successful, unsuccessful, or stopped by user.

If a Process is executed through the Process Designer in the Process Studio, it is considered to be debugging, not a run, so the event **is not going to be recorded in the Audit**.

When you run the Process through a Robot or due to a Trigger or Schedule, the Process will be logged.

To view the Logs for a Process you can either right click on it and select "Show Logs for Process(es)" or hit Ctrl+L, or you can select it in the Select Processes pane in the Logs tab. When you want to see older Logs, or more details, simply set the options as per the [Audit Tab](#)¹⁷⁹ description.

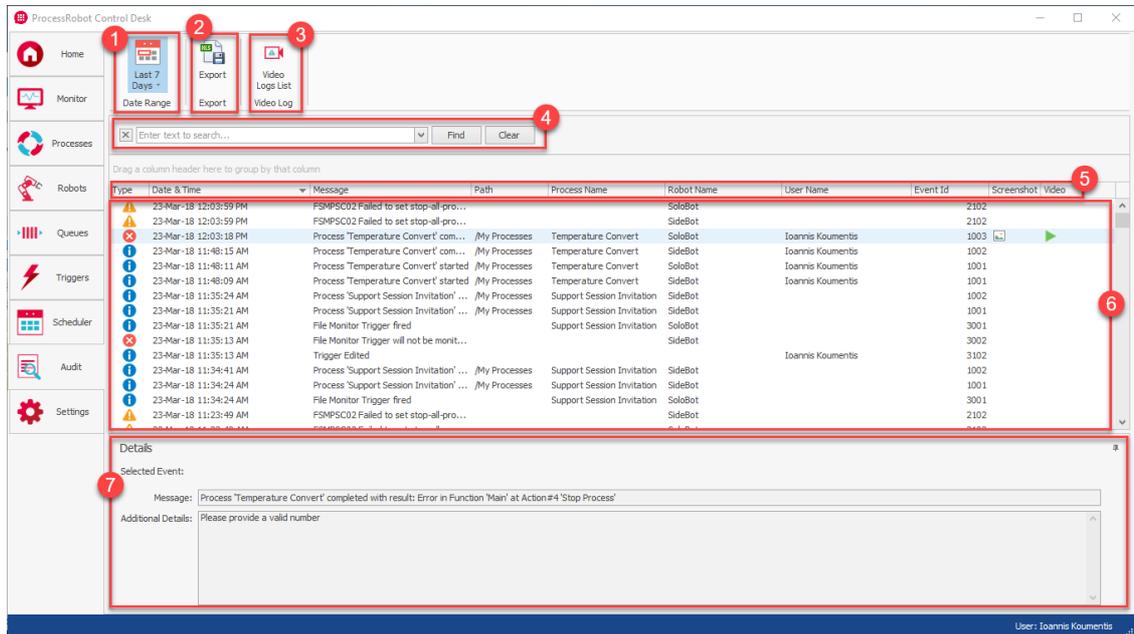
If the Process itself encounters an error or bug, it is also logged into the logs database as an "Internal Exception". Open the Audit tab and review the error Log. If you don't know what to do, ProcessRobot tech support can use this information to aid you quickly and efficiently.

The Export Option in the Audit Tab opens another Window that allows you to save the logs in an *.xls or *.xlsx or *.csv file. Please, note that only the logs being currently displayed, based on the filter that you may have applied, will be shown in the exported file. Therefore, make sure that you either have no filters if you wish to export all the logs, OR apply the filter you are interested in to get only the log entries that interest you.

A Process can record an additional custom log entry through the "[Log Message](#)"¹⁰⁰² action. These events are recorded both when the Process runs through a Robot and the Process Designer.

2.1.2.8,2 Audit Tab

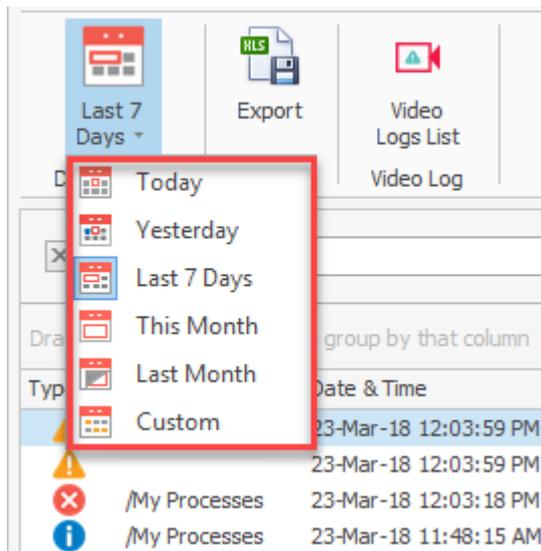
The highly customizable Softomotive Audit tab, consists of the following features:



What you need to understand in order to use the Audit Tab at its full potential, is that the key to unlock its filtering power is to **right-click** on the event Columns Bar [4]. For really sophisticated filtering options you should definitely have a look at **the Filter Editor** which is covered at the end this article.

1. Date Range Selection:

The Date Range option allows you to select the dates whose logs interests you, from a number of predefined ranges or a purely Customized range:



2. Export option:

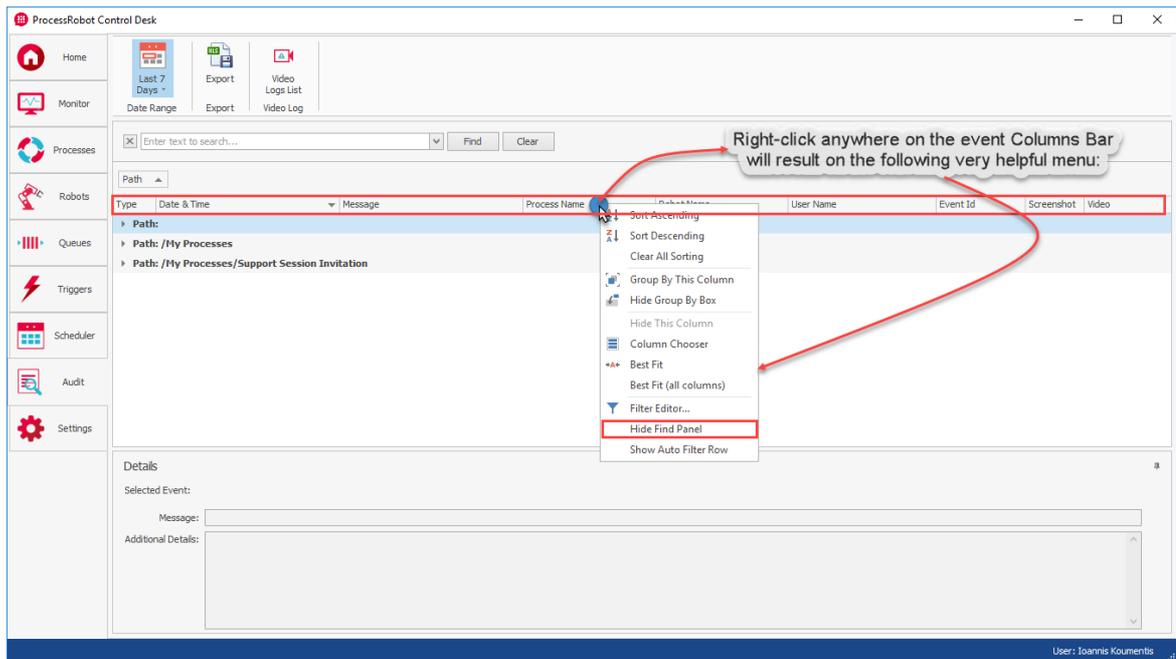
The Export option, opens a Window that allows you to save the displayed logs in a *.xls, *.xlsx, *.csv file in a directory that you will choose. Note, that only the currently displayed Logs will be passed on to the file. So in case you have filtered any of them out, make sure that you have everything you need to export.

3. [Video Log option](#) ¹⁸⁸:

Please feel free to check the corresponding Topic.

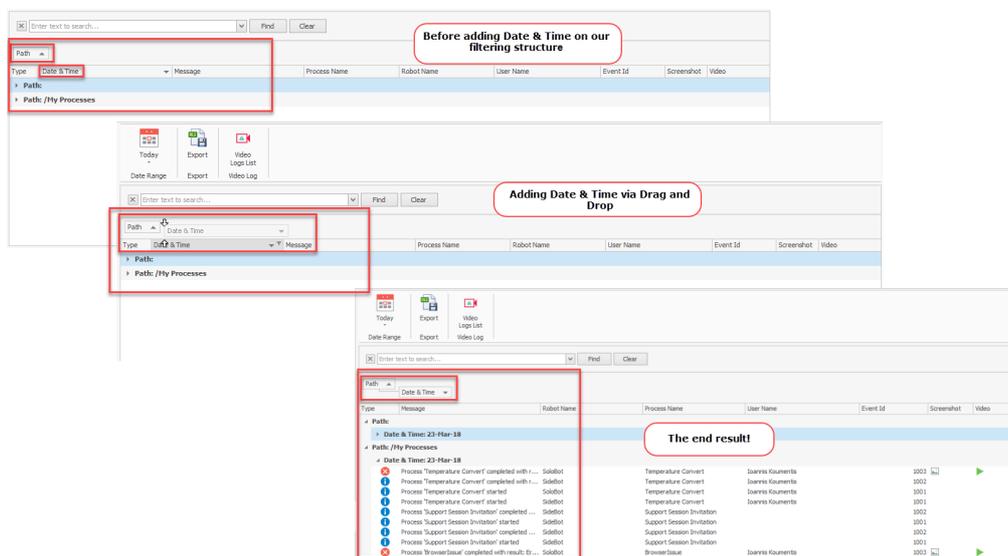
4. The "Search" bar:

You can search for any term that you expect to find in the logs. Please do keep in mind that the Search bar **may not be visible by default** so you would have to **right click on the event Columns Bar** to make it visible.



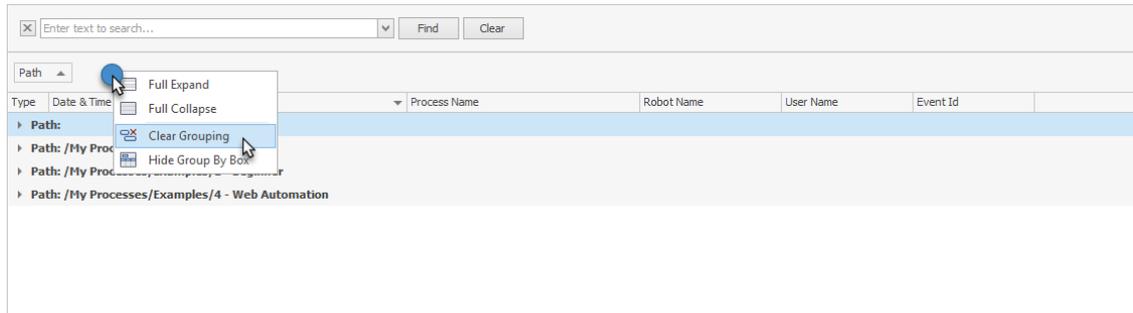
5. The event Columns Bar:

Unlike other panes' Head Bars *that do not allow interaction*, the reader should know that **this is a widely interactive element** that allows users to implement the filtering of their choice, by either drag and dropping *one or multiple* Column Headers to target the exact range of results they want to audit, OR by right-clicking on it and working through the resulting menu:



Filtering via dragging and dropping Columns

In order to return back to the default filtering settings (all) you have not but to right-click on the Drop area and select "Clear Grouping" from the resulting menu:



The columns in order of the default appearance are:

Type: Showing if the event is an Error, a Warning or an Info. These three are the event types that are being logged in ProcessRobot. **Errors** are happening mostly when a Process fails to run successfully and produces an error. **Warnings** are being displayed after the occurrence of expected problems, like trying to run a Process that has known errors. **Infos** simply let you know when a Process ran and completed or when a Trigger or Schedule was fired.

Date&Time: Date and time that the even happened.

Message: The descriptive message of the event.

Path: The path of the Process that generated the event.

Process Name: Self explanatory.

Robot Name: The Robot on which the process was executed.

User Name: The User Name of the Robot.

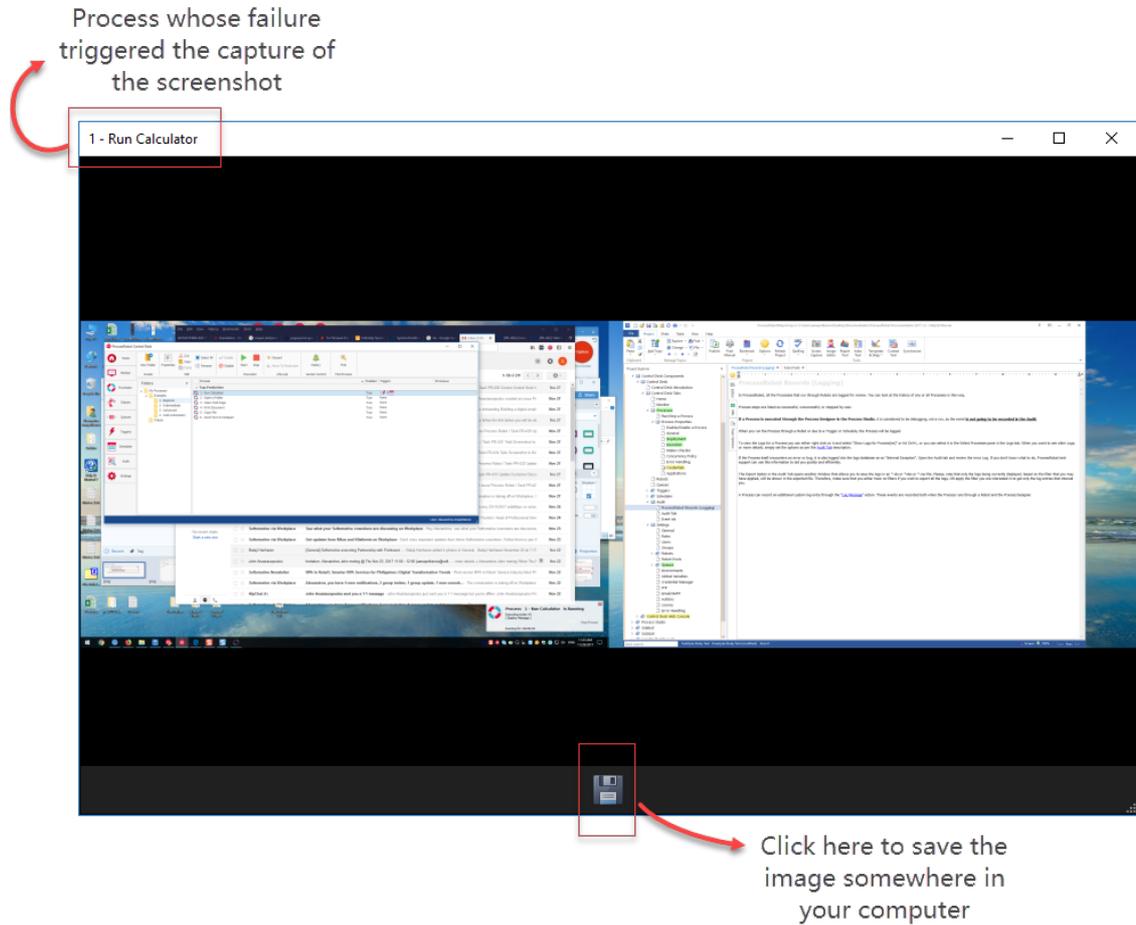
Event Id ¹⁸⁹: The Event Id from the list of Event Ids that are declared in ProcessRobot.

Screenshot (without Title): Exception / Error handling (through [Process Properties](#) ¹³⁹) and [Global Error Handling](#) ²⁴⁸) allows you to Add a Screenshot to Logs in case a Process fails. Ticking that check box will produce a Screenshot of your entire screen (even if you use two monitors) at the moment of failure and the image will be available to you in the event Columns pane, next to the Event Id column:

Drag a column header here to group by that column

Type	Date & Time	Message	Path	Process Name	Robot Name	User Name	Event Id
Info	11/28/2017 11:28:16 AM	Process 'OCR Testing' Edited	/My Processe...	OCR Testing			1102
Info	11/28/2017 11:28:08 AM	Process 'OCR Testing Z' Edited	/My Processe...	OCR Testing 2			1102
Info	11/28/2017 11:23:37 AM	Process '1 - Run Calculator' compl...	/My Processe...	1 - Run Calculator	MySideBot		1006
Info	11/28/2017 11:23:34 AM	Process '1 - Run Calculator' started	/My Processe...	1 - Run Calculator	MySideBot		1001
Info	11/28/2017 11:23:31 AM	Process '1 - Run Calculator' Prope...	/My Processe...	1 - Run Calculator			1104
Info	11/28/2017 11:23:12 AM	Process '1 - Run Calculator' Prope...	/My Processe...	1 - Run Calculator			1104
Info	11/28/2017 11:22:02 AM	Process '1 - Run Calculator' Prope...	/My Processe...	1 - Run Calculator			1104
Warning	11/28/2017 9:36:51 AM	FSMPSCO2 Failed to set stop-all-p...			MySideBot		2102
Warning	11/28/2017 9:35:27 AM	Auto-cleanup of Robot with no lif...					2004

Clicking that image icon will let you review the Screenshot immediately through a pop-up window (a built-in image viewer) of adjustable size that will also allow you to save the Image on your drive.



Video: Error handling (through [Process Properties](#)^[139] and [Global Error Handling](#)^[248]) allows you to open the [Video Logs](#)^[188] List and inspect the corresponding video:

Drag a column header here to group by that column

Type	Path	Date & Time	Message	Robot Name	Process Name	User Name	Event Id	Screens...	Video
⚠		23-Mar-18 12:03:59 PM	FSMPCO2 Failed to set stop-all-pro...	SoloBot			2102		
⚠		23-Mar-18 12:03:59 PM	FSMPCO2 Failed to set stop-all-pro...	SideBot			2102		
⚠	/My Processes	23-Mar-18 12:03:18 PM	Process 'Temperature Convert' com...	SoloBot	Temperature Convert	Ioannis Koumentis	1003	🖼	
ⓘ	/My Processes	23-Mar-18 11:48:15 AM	Process 'Temperature Convert' com...	SideBot	Temperature Convert	Ioannis Koumentis	1002		
ⓘ	/My Processes	23-Mar-18 11:48:11 AM	Process 'Temperature Convert' star...	SoloBot	Temperature Convert	Ioannis Koumentis	1001		
ⓘ	/My Processes	23-Mar-18 11:48:09 AM	Process 'Temperature Convert' star...	SideBot	Temperature Convert	Ioannis Koumentis	1001		
ⓘ	/My Processes	23-Mar-18 11:35:24 AM	Process 'Support Session Invitation' ...	SideBot	Support Session Invitation		1002		
ⓘ	/My Processes	23-Mar-18 11:35:21 AM	Process 'Support Session Invitation' ...	SideBot	Support Session Invitation		1001		
ⓘ		23-Mar-18 11:35:21 AM	File Monitor Trigger fired	SideBot	Support Session Invitation		3001		
ⓘ		23-Mar-18 11:35:13 AM	File Monitor Trigger will not be mont...	SideBot			3002		
ⓘ		23-Mar-18 11:35:13 AM	Trigger Edited			Ioannis Koumentis	3102		
ⓘ	/My Processes	23-Mar-18 11:34:41 AM	Process 'Support Session Invitation' ...	SideBot	Support Session Invitation		1002		
ⓘ	/My Processes	23-Mar-18 11:34:24 AM	Process 'Support Session Invitation' ...	SideBot	Support Session Invitation		1001		
ⓘ		23-Mar-18 11:34:24 AM	File Monitor Trigger fired	SoloBot	Support Session Invitation		3001		

6. The event Columns pane:

Here are listed all the events with all the info displayed in the columns.

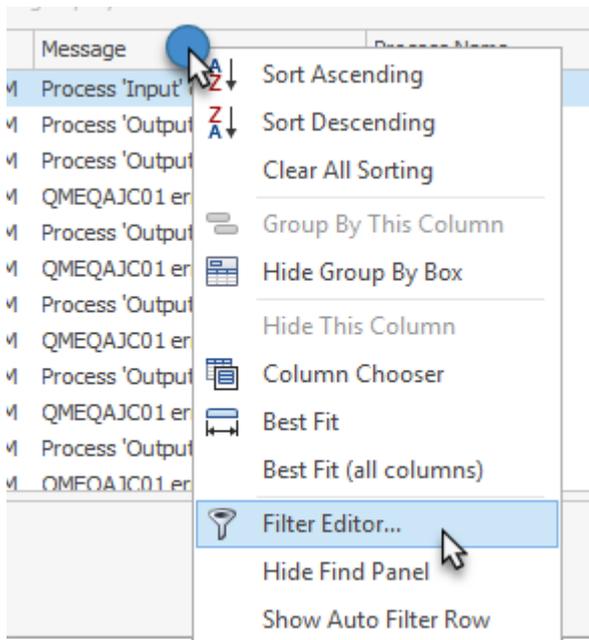
7. The Additional details of the error

Once you have specified all Types and the Date Range, the Processes that match your filter will be listed.

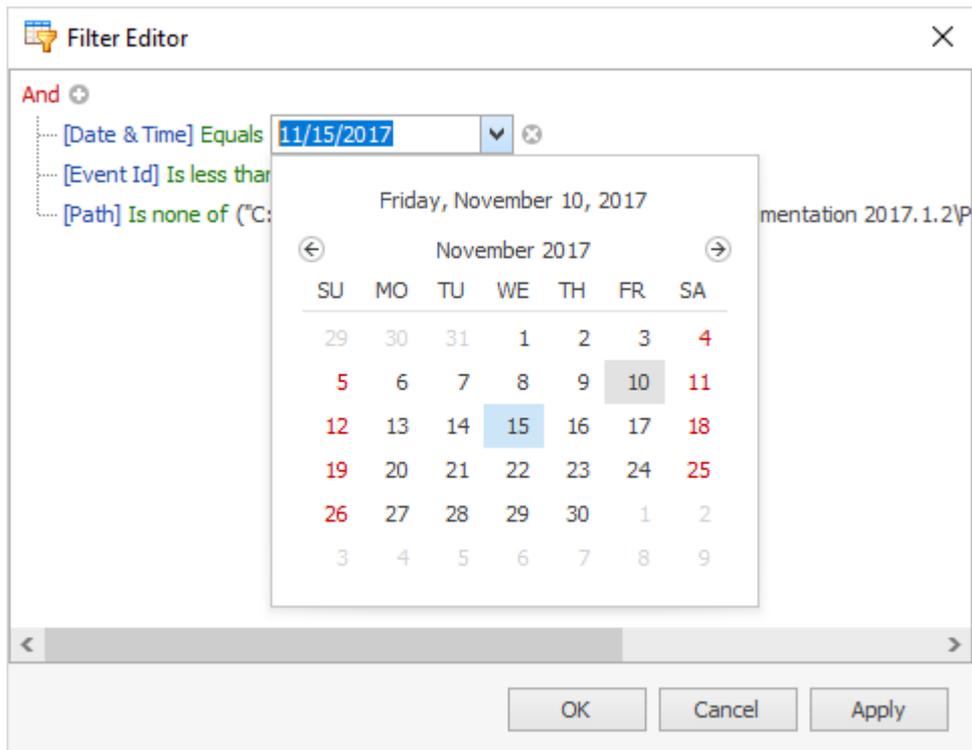
Selecting a Message will display the Details of that Log event. Most Info or Warning events will only repeat the Message here, as there is no more useful information. In the event of an Error, the Additional Details box will contain all appropriate information to help solve the problem. As this area is highly technical, use this information if you need to contact Technical Support.

The Filter Editor

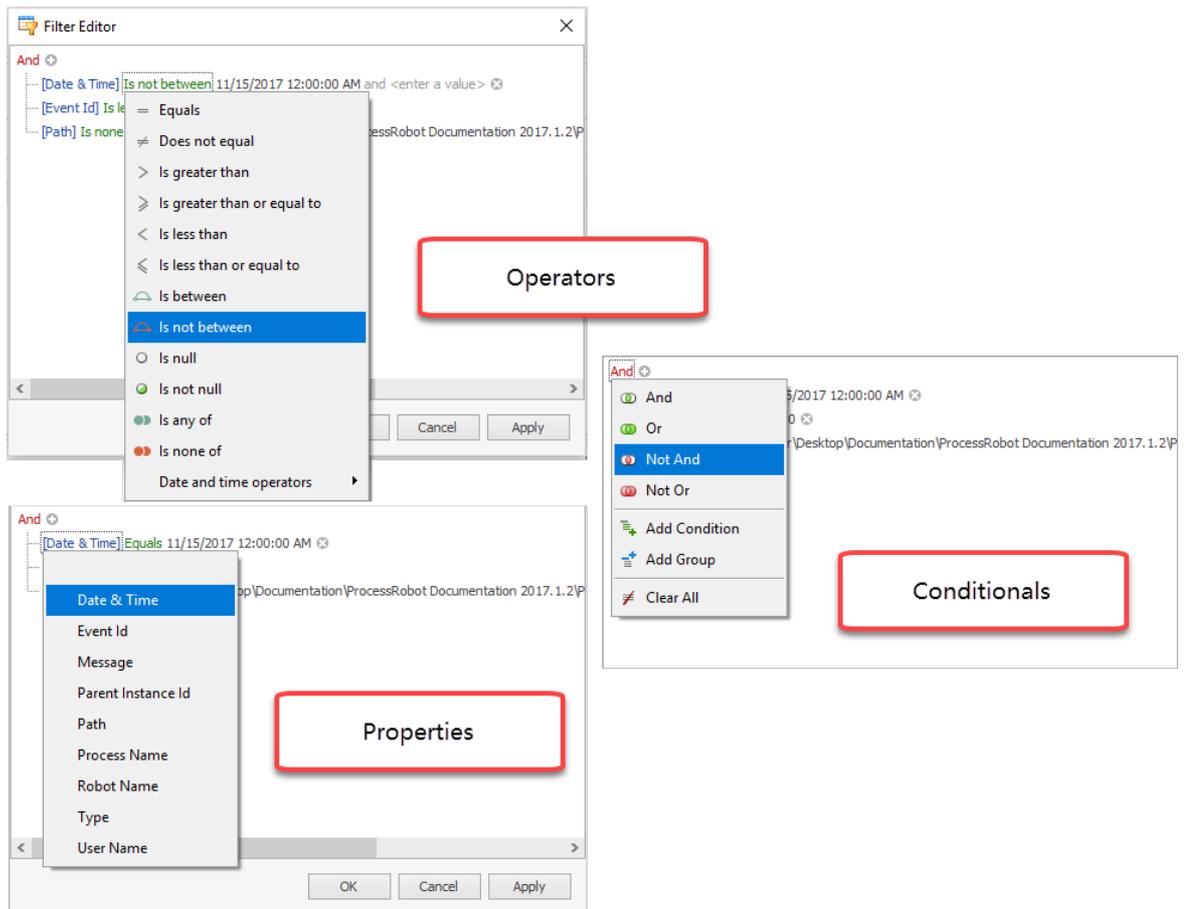
You can right click on the event Columns Bar in the Audit tab and you will get a number of options like this :



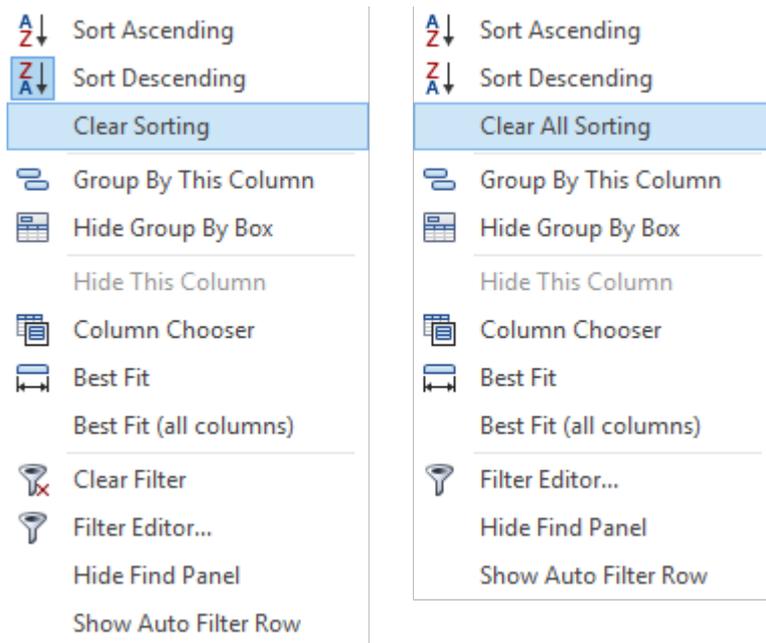
If you select the "Filter Editor" option you can create your own Filter with arguments applicable to any of the bar's Columns in a chained-query like fashion.



It is worth noting that all conditionals, properties and operators are highly customizable so to meet the needs of the most demanding of business projects.



Another important thing to note is that your records can be sorted through a unique combo of criteria at any time. If you want to start from a pure database all over again you will have to right-click on the events Columns Bar and choose the "Clear Sorting" or "Clear All Sorting" options (depending on whether you have one or multiple filters binded in a combo) in order to start afresh.



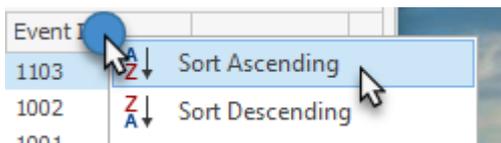
If you are wondering how you can bind multiple filters *without using the Filter Editor*, simply observe the following example:

Let's say that Lucy wants to sort her records on descending Date&Time order and then sort the resulting table according to ascending Event Id order. To do that, Lucy can do the following:

1) Click once upon the Type Column Header in order for the little arrow to show characteristically that the descending order has been activated.



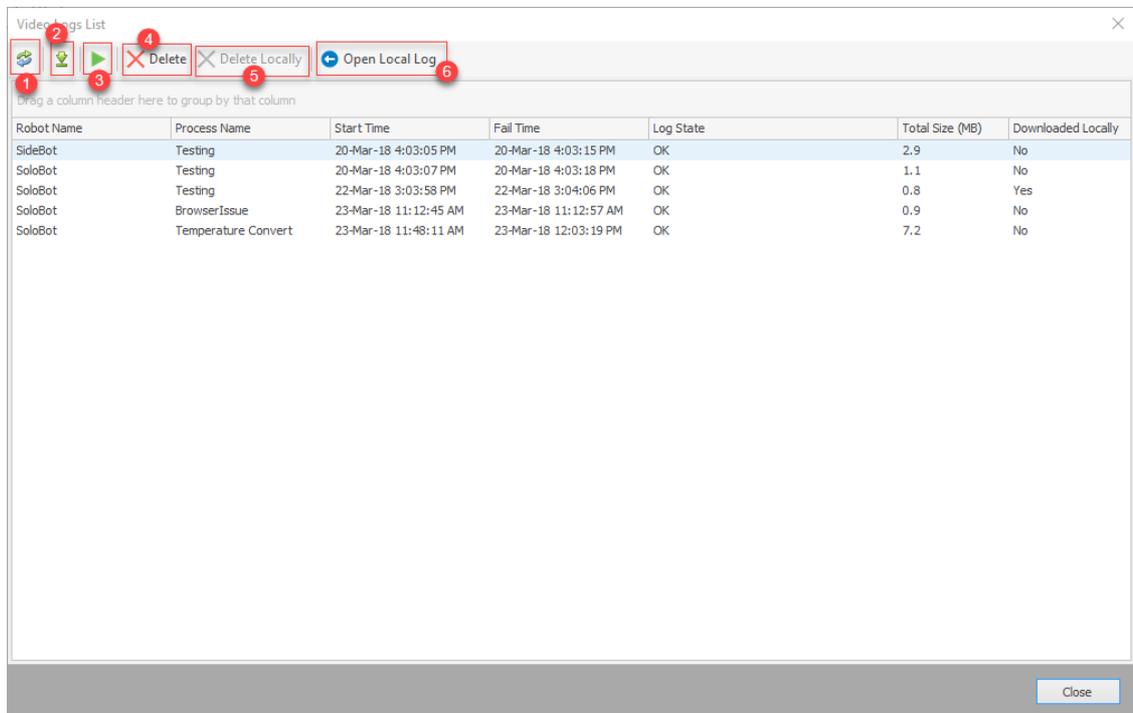
2) Right-click on the Event Id and then click on the Sort Ascending:



If Lucy desires to arrange her Logs on other criteria, she should not forget to click "Clear All Sorting" as we have seen above!

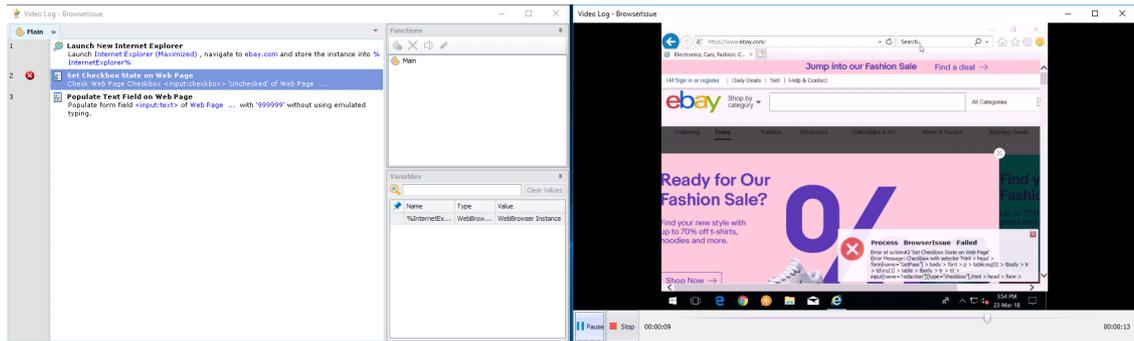
2.1.2.8.3 Error Video Log

The Video Logs option opens the Video Logs List Window. Here, you can access the captured Video Logs from the Processes that have been deployed on one or more Solobots and their execution resulted in an error. Please, note that the the Video Log will be captured only if the corresponding check-box is selected on the global [Error Handling](#)^[248] tab or the [Error Handling tab in Process Properties](#)^[139].



The Video Logs List window has the following options to apply:

- 1) **Refresh** the list so as to add the recently created Video Logs.
- 2) **Download Locally** the Video Log files to the machine that has the Control Desk installed. The default location of the Video Log files is the user's Videos folder.
- 3) **Play** a selected Video Log that has been downloaded locally. Upon pressing the Play button, two windows open side by side, the Video of the Process' execution on the Solobot machine along with the step by step execution representing its Process' Designer window. Hence, you are able to review how each action affects the execution of the Process and the values assigned to variables, as if you were running the Process from the Process Designer.



- 4) **Delete** a Video Log from the list. This option will remove the Video Log both from the local and the Solobot machine.
- 5) **Delete Locally** a Video Log that you have previously downloaded locally. The saved Video Log on the Solobot machine will not be affected.
- 6) **Open Local Log** option opens a window that allows you to access and view a Video Log, saved in a specific file-path.

2.1.2.8.4 Event Ids

In the Audit tab in the column "Event Id" you may see the following Ids. Their meaning can be found on the table below:

- 0 = Unitialized
- 1001 = ProcessStarted
- 1002 = ProcessCompleted
- 1003 = ProcessFailed
- 1004 = ProcessEvent
- 1005 = ProcessWarn
- 1006 = ProcessCanceled
- 1007 = ProcessQueued
- 1008 = ProcessSkipped
- 1009 = AutomaticLogging
- 1010 = ProcessWaitingValidation
- 1011 = ValidationFailed
- 1012 = StartProcessFailed
- 1013 = TryingToRunSameProcess
- 1101 = ProcessCreated
- 1102 = ProcessEdited
- 1103 = ProcessStateChanged
- 1104 = ProcessPropertiesChanged
- 1105 = ProcessDeleted
- 1106 = ProcessImportFailed
- 1107 = ProcessExported
- 2001 = ServerStarted
- 2002 = ServerStopped

2003 = ServerError
2004 = ServerWarning
2005 = ServerInfo
3001 = TriggerFired
3002 = TriggerError
3101 = TriggerCreated
3102 = TriggerEdited
3103 = TriggerDeleted
4001 = ScheduleFired
4002 = ScheduleError
4101 = ScheduleCreated
4102 = ScheduleEdited
4103 = ScheduleDeleted
5001 = FolderCreated
5002 = FolderRenamed
5003 = FolderDeleted
9001 = InternalError

2.1.2.9 Settings

ProcessRobot Options are accessed by clicking on the Option Tab in the console Menu.

The Options tab includes seven tabs.

1. [General](#)^[191]
2. [Roles](#)^[193]
3. [Users](#)^[195]
4. [Groups](#)^[199]
5. [Robots](#)^[200]
6. [Machines](#)^[210]
7. [Robot Pools](#)^[217]
8. [Queues](#)^[220]
9. [Dashboards](#)^[233]
10. [Environments](#)^[235]

11. [Global Variables](#)^[237]

12. [Credential Manager](#)^[241]

13. [Error Handling](#)^[248]

14. [License](#)^[250]

2.1.2.9.1 General

In the **Settings > General** tab you can specify:

1

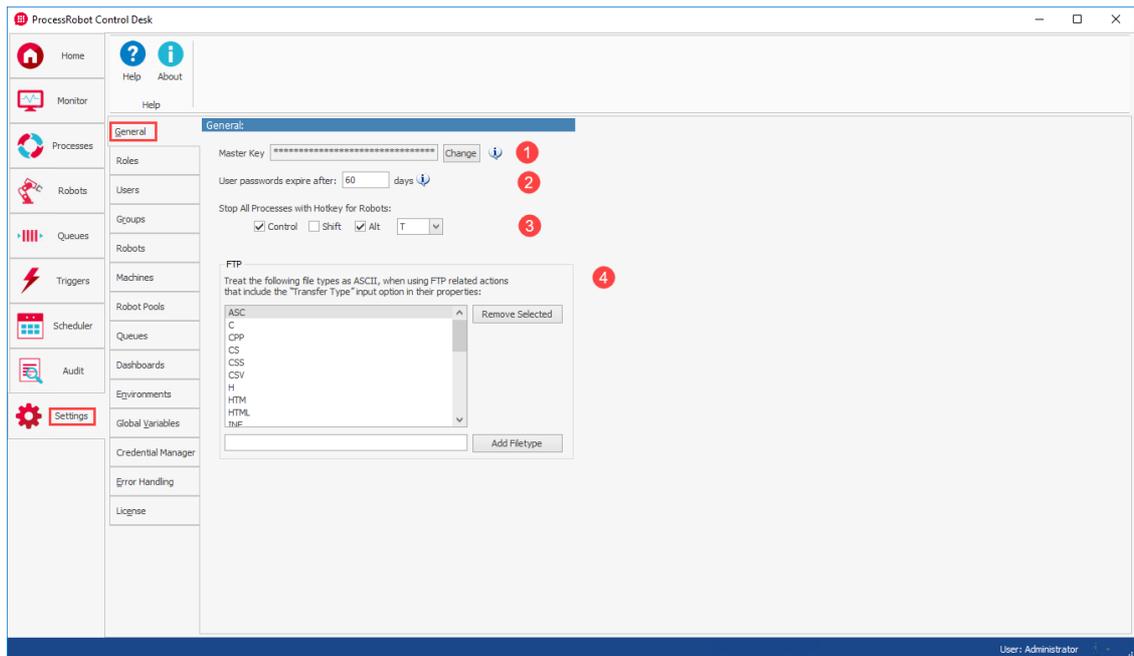
The Master Key for your Process Robot. This Master Key will be the one that will be used to encrypt all the provided user passwords required for the Autologin feature to work on Solobots, as well as the hidden values for the credentials. It is up to the administrator of the process Robot to Set or Change it. Please note that, the Master Key can only be reset if the user knows the old one.

2

The time period after which User passwords will expire. This applies only to Users added as the ProcessRobot type of User, not Active Directory. This expiration date can be overridden when creating or editing a specific User of that type in [Settings > Users](#)^[195].

3

A Hotkey that when pressed will stop all the Processes on all Robots.

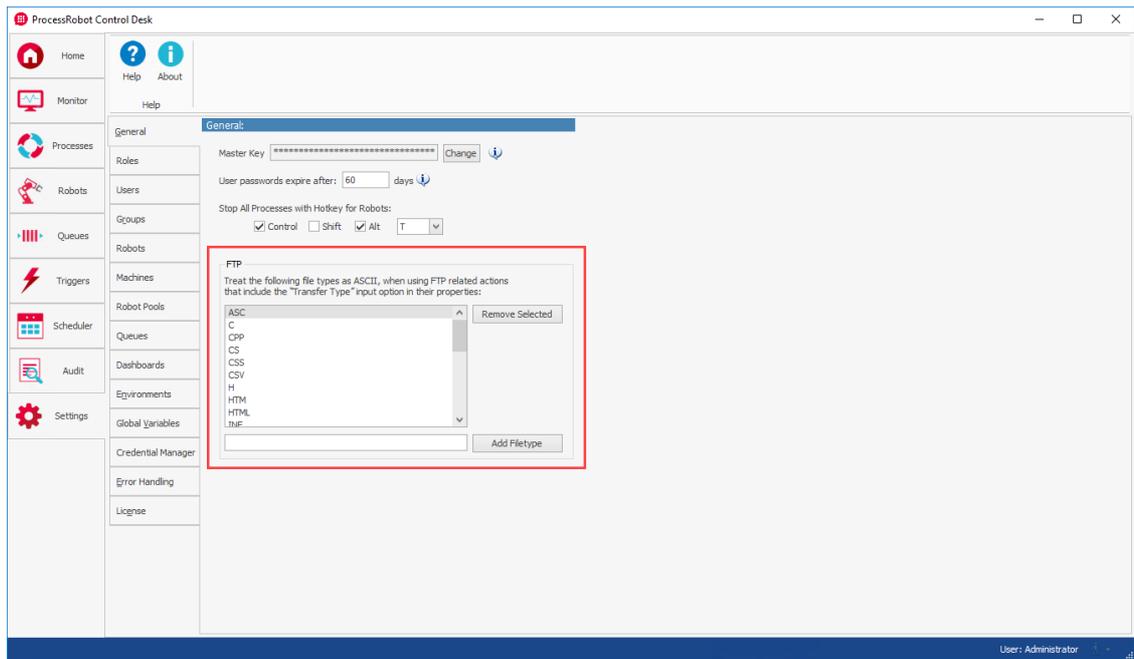


Control Desk Settings > General Tab

4

When working with FTP, downloads and uploads always come as ASCII or binary. Here you can select which files will be transferred as ASCII - the rest will default to binary. Just enter the file extension you wish to add and click Add Filetype, or Remove any File Type you don't wish to have on the list.

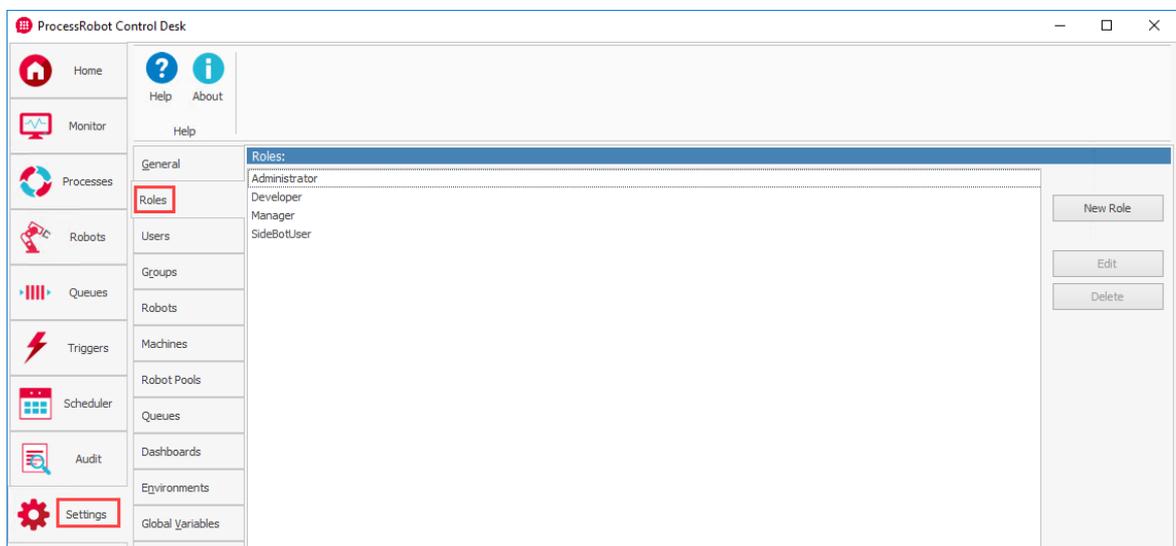
When downloading or uploading files (using the [Download File\(s\) from FTP](#)^[783] and [Upload File\(s\) to FTP](#)^[786] actions) you can specify the transfer type to be ASCII, BINARY or AUTO. If you select AUTO, ProcessRobot will use this list to decide whether the file will be transferred in ASCII or binary mode. This list is also used by the [Download Folder\(s\) from FTP](#)^[785] and [Upload Folder\(s\) to FTP](#)^[787] actions for deciding how to transfer the files contained into the folders.



Control Desk Settings > FTP Tab

2.1.2.9,2 Roles

In the Roles tab you can create roles and assign them to users. With the Roles you are able to pass permissions to your users as explained below:



Control Desk Settings > Roles Tab

The default Role that will be there when you open the Control Desk for the first time is the **Administrator**.

Please be aware: By default, the first user that will open the Control Desk after the installation is automatically the Administrator.

Then you can add new roles from the "New Role" button. Once you press this you will get a New Role pop-up window and you will be expected to enter the new role's name (developer, manager, guest, etc.) and then you are free to assign permissions to this role.

DO note that if nothing is checked, then the permission is by default "Denied".

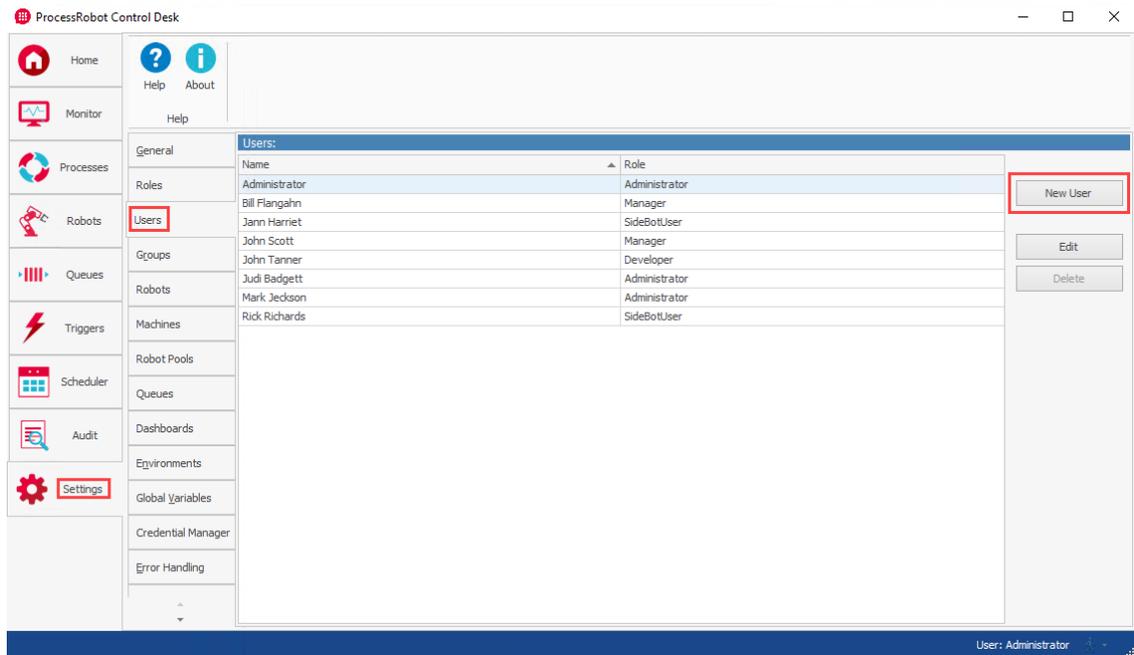
Having that in mind, please check the appropriate boxes in order to allow or deny permissions for specific Components, Processes, Folders, Triggers, Schedules , [User Libraries](#)³⁹⁴, Robots and Robot Pools.

Permission	Allow
All	<input type="checkbox"/>
Component	<input type="checkbox"/>
Access to Control Desk	<input checked="" type="checkbox"/>
Access to Process Studio	<input type="checkbox"/>
Access to Dashboard Designer	<input checked="" type="checkbox"/>
Access to Dashboard Viewer	<input type="checkbox"/>
Control Desk Tabs	<input checked="" type="checkbox"/>
Process Studio Tabs	<input checked="" type="checkbox"/>
Process	<input type="checkbox"/>
User Libraries	<input checked="" type="checkbox"/>
Process Folder	<input checked="" type="checkbox"/>
User Library Folder	<input checked="" type="checkbox"/>
Trigger	<input checked="" type="checkbox"/>
Schedule	<input checked="" type="checkbox"/>
Robot	<input checked="" type="checkbox"/>
Robot Pool	<input checked="" type="checkbox"/>
Dashboard	<input checked="" type="checkbox"/>
Credentials	<input checked="" type="checkbox"/>

You can of course edit or delete a role at any point from the relative buttons.

2.1.2.9,3 Users

Add new ProcessRobot Users and [assign Roles](#)¹⁹³ to them in the Users Tab. Click "New User" to create a new User.



Control Desk Settings > Users Tab

Doing so will launch the 'New User' window. Choose the Type and Role of the new User. The available User Types are Active Directory and ProcessRobot. The available Roles will be those that have been created in the Control Desk Settings > Roles Tab.

New User

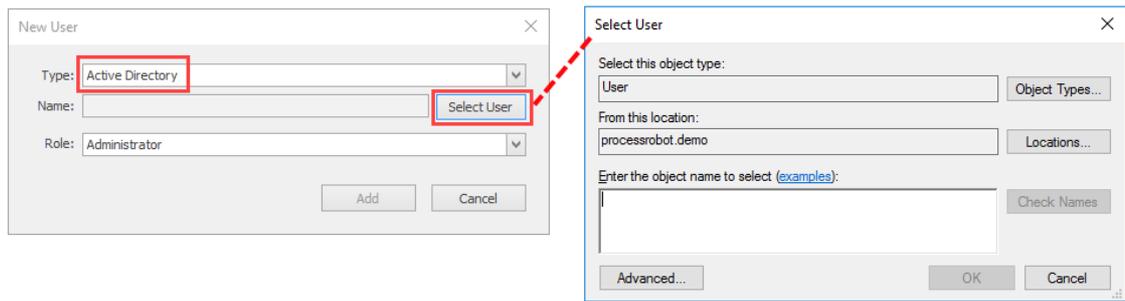
Type: Active Directory

Name: Active Directory
ProcessRobot

Role: Administrator

Add Cancel

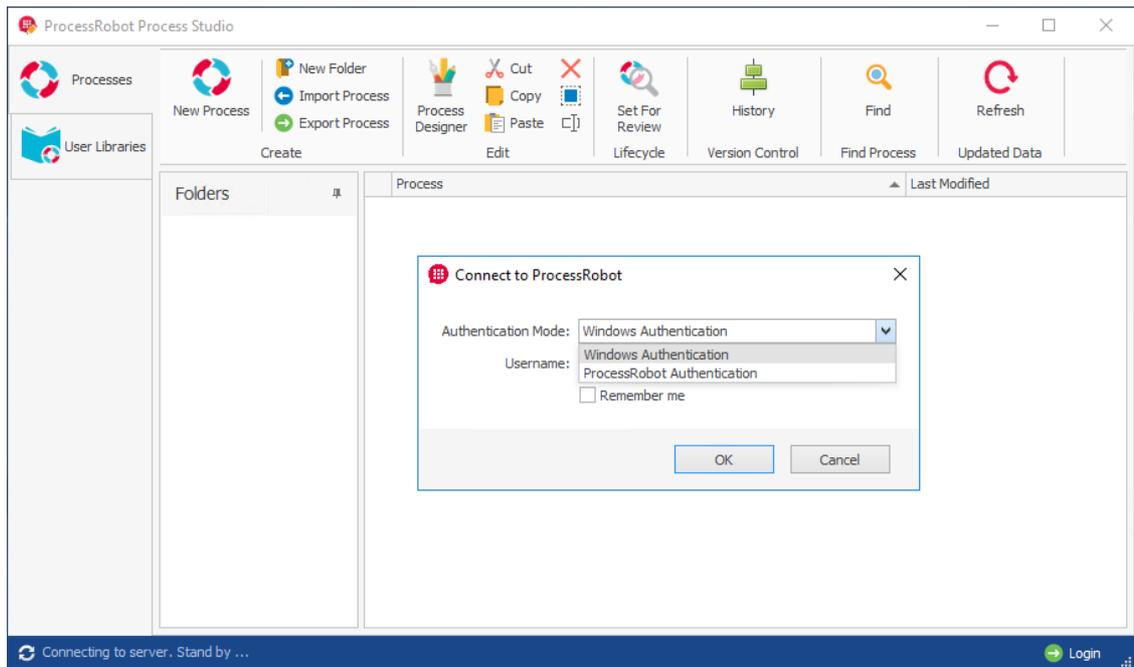
Select the Active Directory Type to add an existing User from an Active Directory domain to ProcessRobot. Note that the User must have already been created in the domain prior to adding them to ProcessRobot. Click 'Select User' to open the 'Select User' window. In this window, click 'Locations...' to choose the domain. Enter a User's name in the text box and click 'Check Names' to search for that User in the specified domain.



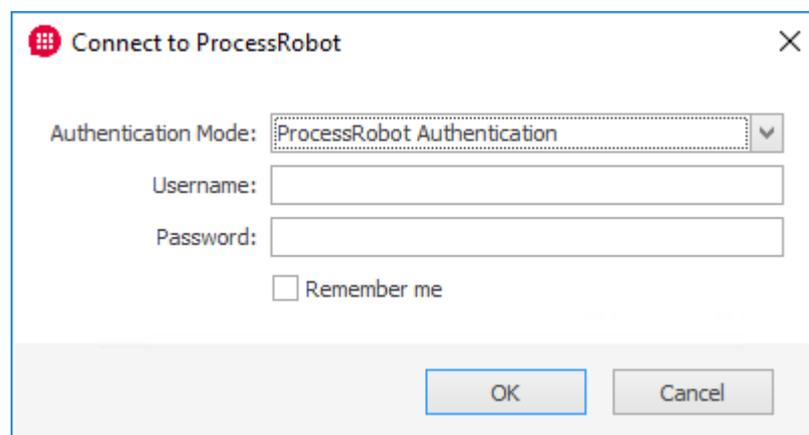
Important: only ProcessRobot Users that have been added with the Active Directory Type can be assigned as Web Console Users.

Alternatively, select the ProcessRobot User Type to create a ProcessRobot User that does not need to belong to an Active Directory domain. Specify the Username, Password, and Password Expiration Data for this new User, as well as their Role. Bear in mind that the default value in the 'Password Expires' field will be calculated based on the value in [Settings > General](#)¹⁹¹.

The difference between the two User Types affects user authentication when launching a ProcessRobot component. Users added from the Active Directory will be able to choose 'Windows Authentication' and proceed without being required to enter a password, assuming they have the necessary Permissions for the ProcessRobot component they are accessing.

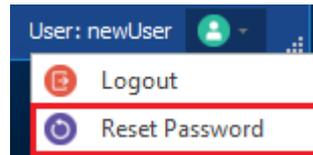


Users that have been added under the ProcessRobot User Type must select 'ProcessRobot Authentication'. These Users will then have to enter their Username and Password to proceed.



Custom Authentication - Reset Password

Custom Authentication Users who are already signed into a Client Tool, may click on their User name in the bottom right corner of the interface, to access the "Reset Password" option.

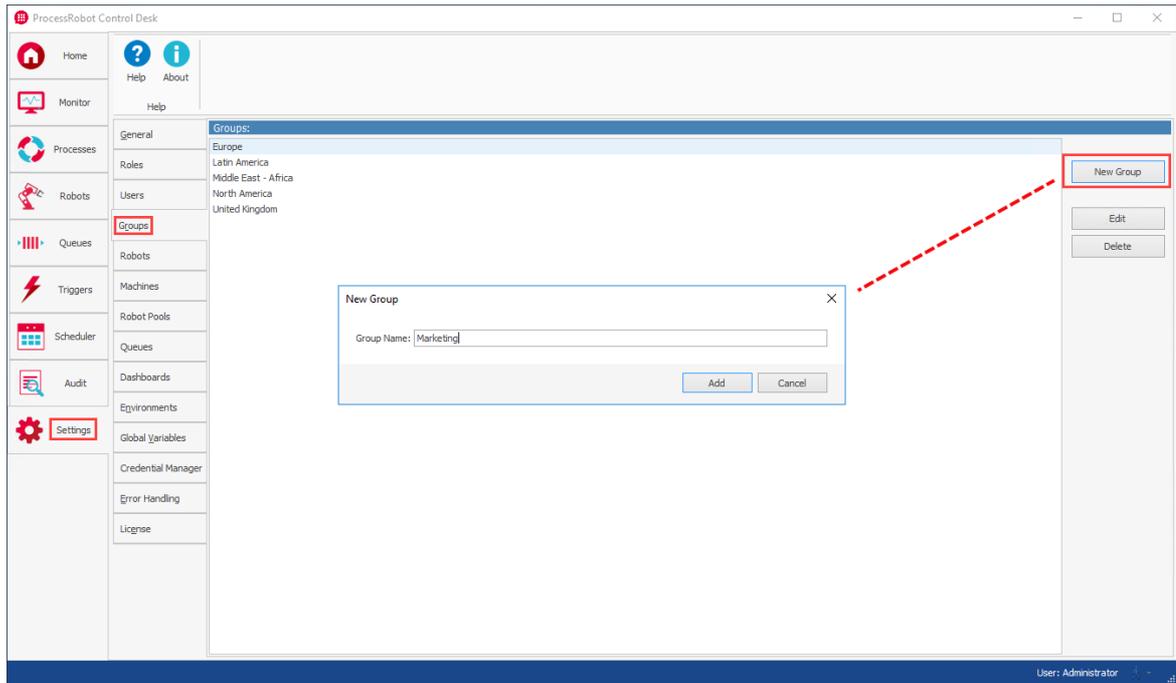


Click this option to open the "Reset Password" window, where the existing password must be entered before a new password can be specified.

A screenshot of a "Reset Password" dialog box. The dialog has a title bar with "Reset Password" and a close button (X). Below the title bar, it says "Enter a new login password". There are three input fields: "Old Password:", "Password:", and "Verify Password:". At the bottom right, there are two buttons: "OK" and "Cancel".

2.1.2.9,4 Groups

Groups make it really easy to handle your Robot Pools simply by allowing you to group Robots together. Below is the Groups Tab in the Control Desk 's Settings in which you are able to view the Groups that you have created, Create a new Group from the "New Group" option, "Edit" an existing one, or "Delete" it.



Control Desk Settings > Groups Tab

In order to understand how Groups work, first you need to note that in the "[Settings > Robots](#)"^[200] tab there is a Group column. When you create or edit a new Robot in the Settings, you are able to assign it to a specific Group.

For example let's say that for your Marketing Department you need to create 57 Robots. You will have to assign each Robot to the Marketing Group from the relative option every time you create one. At the end all your 57 Robots will be in the Marketing Group.

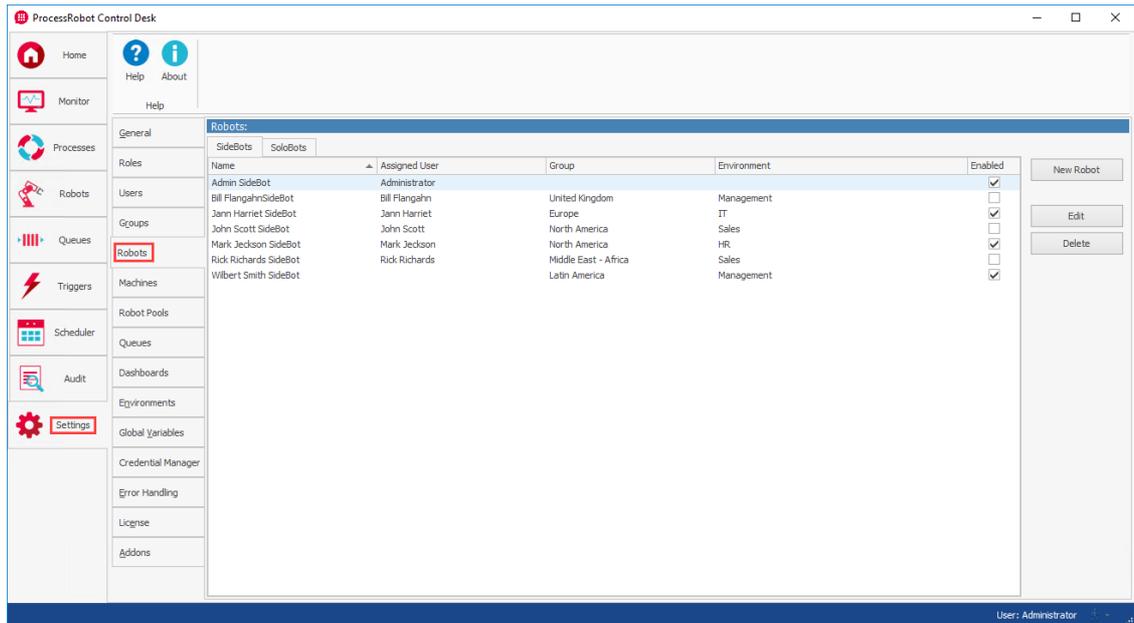
Now, when you wish to create a [Robot Pool](#)^[217] for your Marketing Robots, you have two options as far as adding Robots is concerned. You can either add the Robots to the Pool one by one, or add the whole Marketing Group in one move.

This will save you a lot of time as you will not have to add Robots **individually**, but add a Group **that contains them all**.

So instead of adding additional Robots to a Robot Pool, you can simply assign them to a Group when creating them.

2.1.2.9,5 Robots

In the Robots tab you will be able to see all the Robots that are installed and able to communicate with the Control Desk.



Control Desk Settings > Robots Tab

In each of the Sidebot or Solobot tabs you may find information about:

- The Name of the Robot,
- The User/Machine that the Robot is assigned to,
- The Group that the Robot belongs to,
- The Environment that the Robot is assigned to and
- Whether the robot is enabled or not. If the Robot is not enabled than it makes sense that it will not be able to run any Process until it is enabled again.

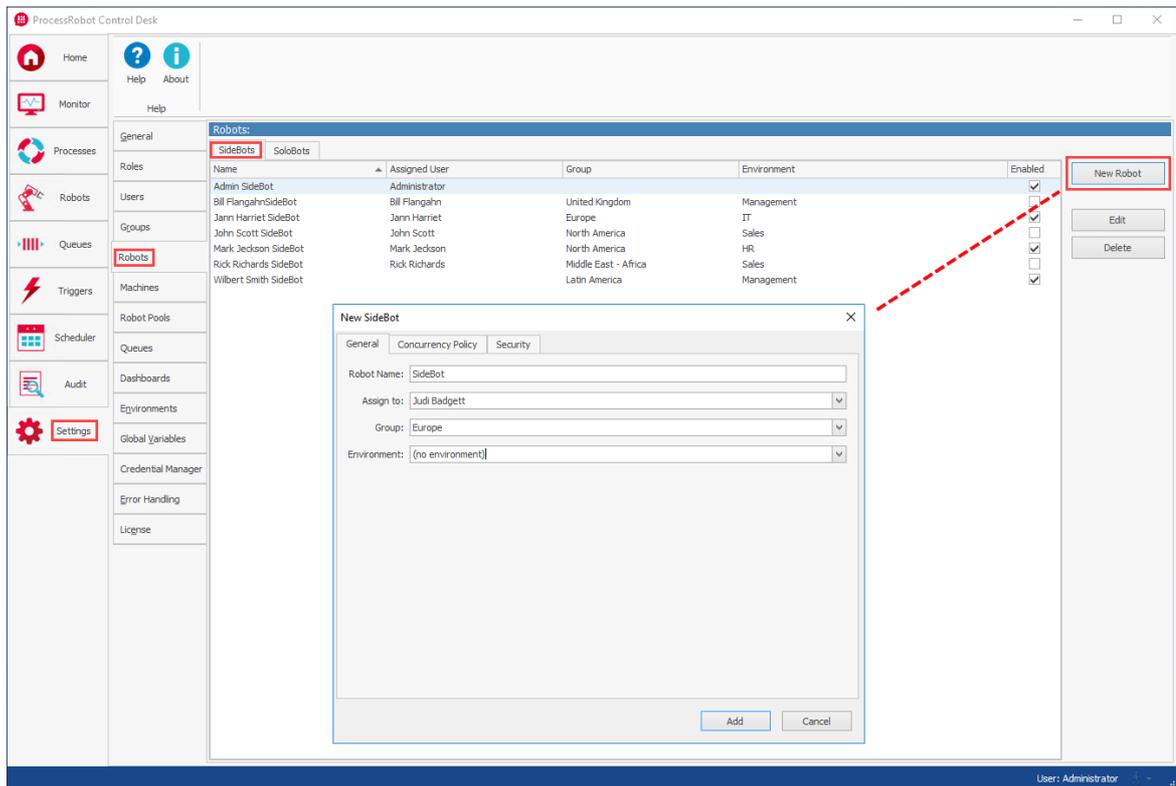
Add a new SideBot

In the "Settings > Robots" tab you can add new Sidebots and Solobots.

Lets see how you can add a Sidebot:

While in the Sidebot tab, click on the "New Robot" option and you will get the "New Sidebot" pop-up window. This window will have three tabs.

The first one is the "General" tab. You can give a Name to your Sidebot, assign it to a User from your Active directory, add it to a Group and to an Environment.



Add a new Sidebot > General Tab

The second tab is the "Concurrency Policy" tab which is exactly the same as explained in "[Process Properties > Concurrency Policy](#)"¹³⁸

The third tab is the "Security" tab in which you can add any Role to your new Sidebot that you have previously created in the "[Settings > Roles](#)"¹⁹³ tab. Also, for each of the Roles that you have added to the Sidebot, you are able to Allow or Deny the right to 'Execute Processes on Robot'. This means that any User assigned the particular Role will or will not be able to run Processes on this Robot, depending on how the Permissions have been configured in this window.

Additional Permissions can be configured in the Security Tab of the New Robot window, which apply to SoloBots. These are: 'Take Screenshot of Robot', 'View Robot Screen', and 'View and Control Robot Screen'.

New SideBot [Close]

General | Concurrency Policy | **Security**

Object: SideBot 'SideBot'

Roles:

- Developer
- Manager

[Add] [Delete]

Permissions:

Permission	Allow	Deny
Robot	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Execute Process on Robot	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Take Screenshot of Robot	<input type="checkbox"/>	<input type="checkbox"/>
View Robot Screen	<input type="checkbox"/>	<input type="checkbox"/>
View and Control Robot Screen	<input type="checkbox"/>	<input type="checkbox"/>

[Add] [Cancel]

Add a new Sidebot > Security Tab

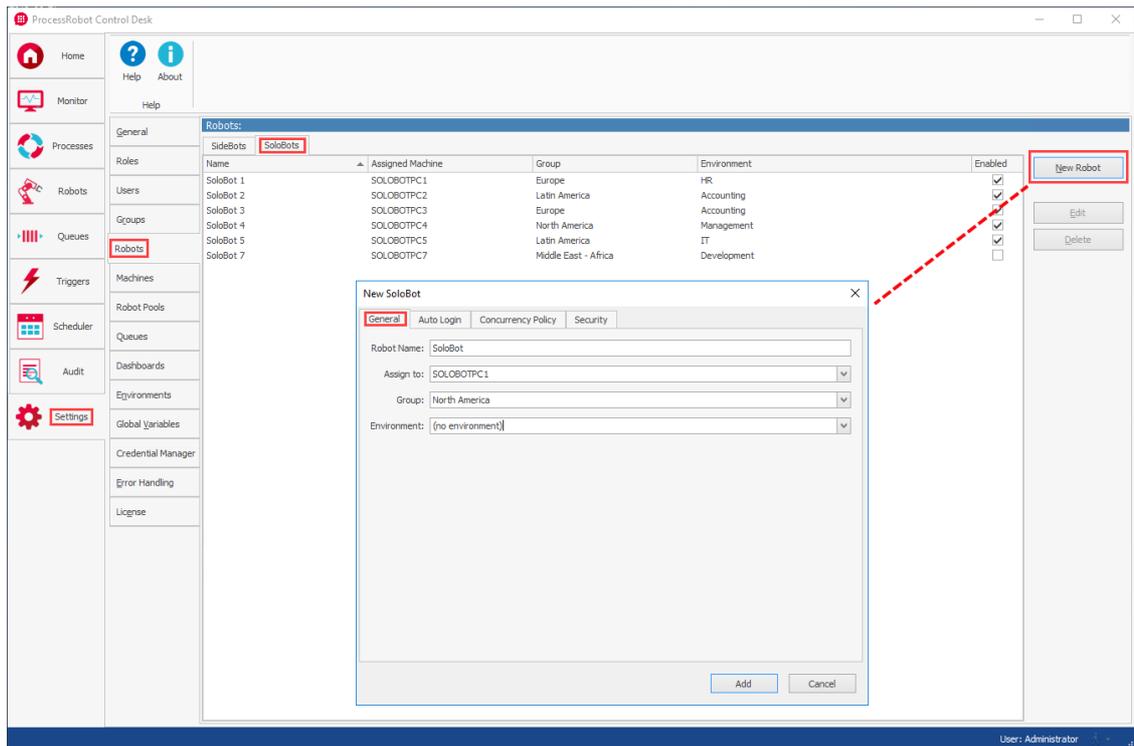
Add a new Solobot

Lets see how you can add a Solobot.

While in the Solobot tab, click on the "New Robot" option and you will get the "New Solobot" pop-up window. This window will have four tabs.

- General Tab

The first one is the "General" tab. In this tab, name the SoloBot and assign it to a machine. Optionally, you may also assign it to a Group and Environment.



Add a new Solobot > General Tab

- Autologin Tab

This Tab is included only in the 'New' or 'Edit SoloBot' windows, as SideBots do not support the Auto Login feature. Auto Login enables the user to configure the SoloBot so it can log into a machine using an Active Directory user's credentials.

The screenshot shows the 'New SoloBot' dialog box with the 'Auto Login' tab selected. The 'Provide Password' dropdown is set to 'directly'. The 'User' field contains 'solobot' and the 'Domain' field contains 'processrobot0'. The 'Password' and 'Confirm Password' fields are filled with asterisks. The 'Login Mode' dropdown is set to 'BotStack'. The 'Desktop Resolution' dropdown is open, showing 'Auto Login' and 'BotStack' options. A checkbox for 'Logout after all processes completed' is present and unchecked. The 'Add' and 'Cancel' buttons are at the bottom right.

Add a new Solobot > AutoLogin Tab

Please note that via the Password mode box, you have the option to Enter password directly, Get password via Command Line or Get password from [Credentials](#)^[24]

The screenshot shows the 'New SoloBot' dialog box with the 'Auto Login' tab selected. The 'Provide Password' dropdown is set to 'directly'. The 'User' dropdown is open, showing three options: 'directly', 'via command line', and 'from Credentials'. The 'Domain' field contains 'processrobot'. The 'Password' field is masked with asterisks. The 'Login Mode' is 'BotStack' and 'Desktop Resolution' is '1280x720'. There is an unchecked checkbox for 'Logout after all processes completed'. 'Add' and 'Cancel' buttons are at the bottom right.

Add a new Solobot > AutoLogin Tab: Exploring the password mode options

It is possible to set the SoloBot to use a local machine user's credentials for Auto Login purposes. When configuring the SoloBot this way, simply type in the local user's name into the 'User' field, ignoring the 'Select User' button. Leave the 'Domain' field blank. Enter the user's password in the relevant fields below.

The screenshot shows the 'Edit SoloBot' dialog box with the 'Auto Login' tab selected. The 'Provide Password' dropdown is set to 'directly'. The 'User' field contains 'newUser' and is highlighted with a red box, with a 'Select User' button to its right. The 'Domain' field is empty. The 'Password' and 'Confirm Password' fields contain masked text (asterisks). The 'Login Mode' dropdown is set to 'Auto Login'. There is an unchecked checkbox for 'Logout after all processes completed'. At the bottom right, there are 'Update' and 'Cancel' buttons.

Additionally, the Auto Login Tab includes the 'Login Mode' option, which relates to ProcessRobot's BotStack capability.

The screenshot shows the 'Edit SoloBot' dialog box with the 'Auto Login' tab selected. The 'Login Mode' dropdown is highlighted with a red box and set to 'BotStack'. The 'Desktop Resolution' dropdown is open, showing a list of resolutions: 800x600, 1024x768, 1280x720 (selected), 1366x768, 1440x900, 1680x1050, and 1920x1080. Other fields include 'Provide Password' (directly), 'User' (solobot), 'Domain' (processrobot0), and 'Password' (masked). Buttons for 'Update' and 'Cancel' are at the bottom right.

By selecting the 'BotStack' option, more than one SoloBot can be assigned to the same machine. See the [BotStack](#)^[113] and [BotStack Requirements](#)^[116] articles for more details. Having selected the 'BotStack' option, the user can select the desired Desktop Resolution that the specific SoloBot will use when active.

- Concurrency Policy Tab

The same as the rules found in [Process Properties > Concurrency Policy](#)^[138], except that in this case, the Concurrency policy rules will apply to the Robot, not a Process.

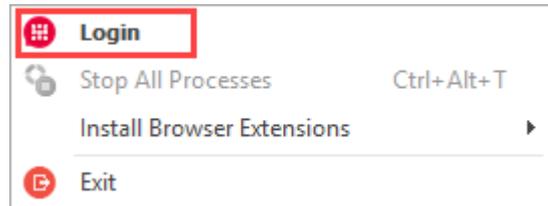
- Security Tab

Allows the user to set Permissions for operations that can be performed on the Robot for each user Role, exactly as described in the previous [SideBot](#)^[201] topic.

SoloBot Authentication

The first time a SoloBot launched, the user must manually login, so as to initiate the ProcessRobot SoloBot Service. After this initial step, the SoloBot will be able to Auto Login, as per the credentials entered in [Settings > Robots](#)^[200] for that particular Robot.

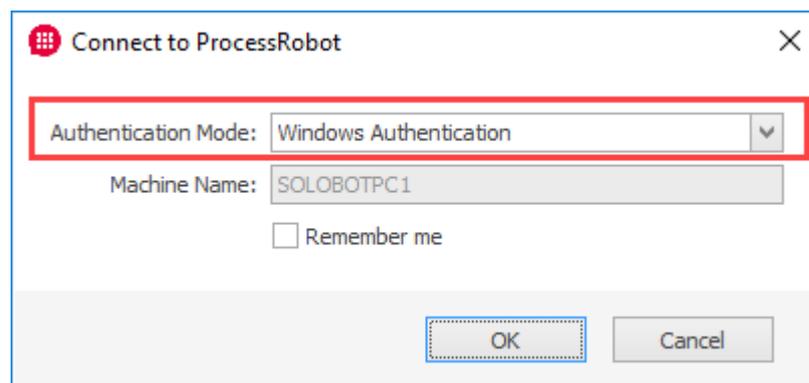
To initiate the first login, the user must right click on the SoloBot icon in the Windows system tray. Then click on 'Login'.



Having chosen to login, the 'Connect to ProcessRobot' window will appear. The user must select the 'Authentication Mode'. This can be either be 'Windows Authentication', or 'ProcessRobot'.

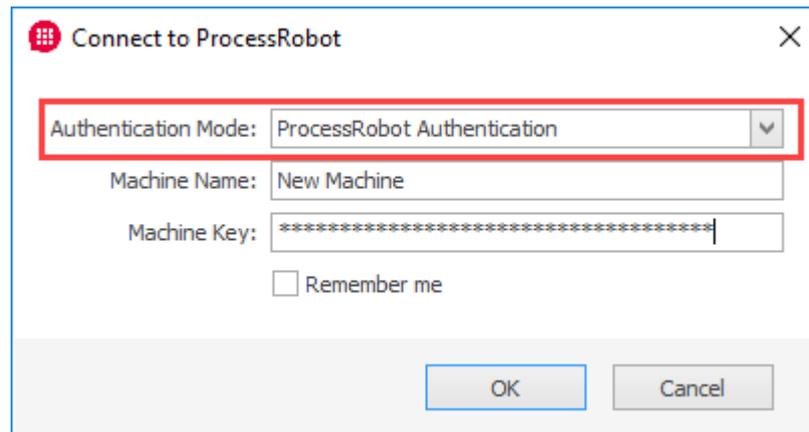
Windows Authentication:

When selecting 'Windows Authentication', ProcessRobot will attempt to connect to the SoloBot based on the Auto Login credentials entered for the specific Robot in [Settings > Robots](#)^[200].

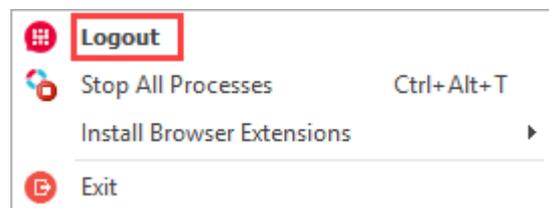


ProcessRobot Authentication:

When selecting 'Windows Authentication', the user must manually enter the machine name they created for this machine as well as the machine key they generated for it in [Settings > Machines](#)^[210].



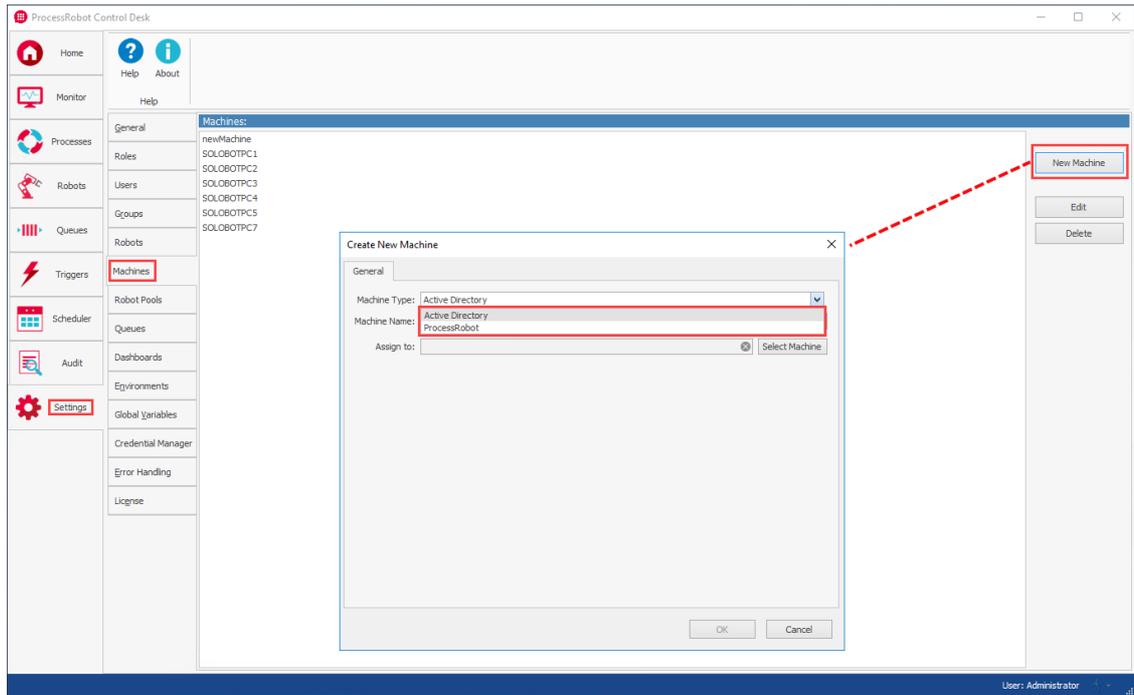
The user can manually logout from the SoloBot in a similar fashion to logging in.



2.1.2.9,6 Machines

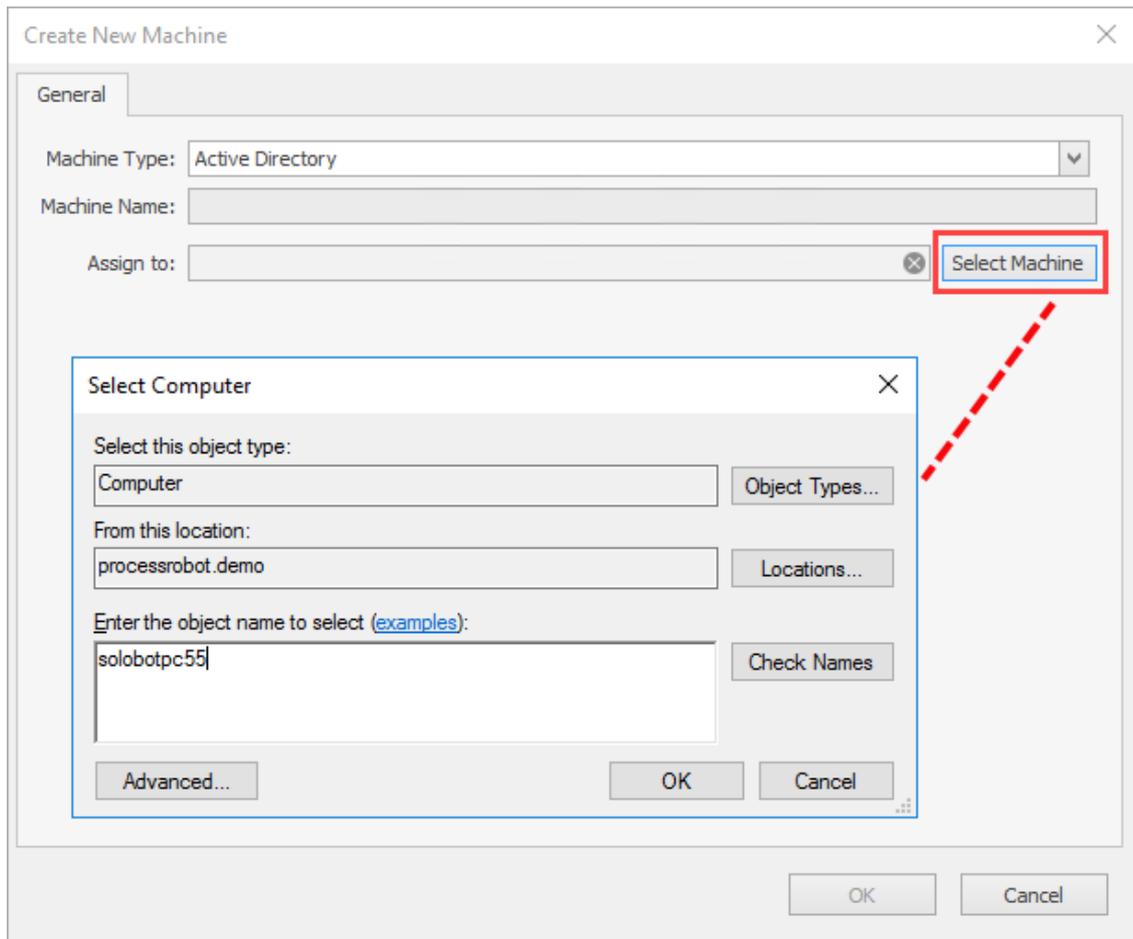
In Settings > Machines, designate machines to be used with ProcessRobot. These can be added either from the Active Directory, or as ProcessRobot machines. Both approaches are described below. Machines to be used as SoloBots must be added in this Tab before creating the SoloBot.

Click 'New Machine' to open the 'Create New Machine' window. In the 'Machine Type' field, select whether the new machine should be added from the Active Directory or as a ProcessRobot machine.

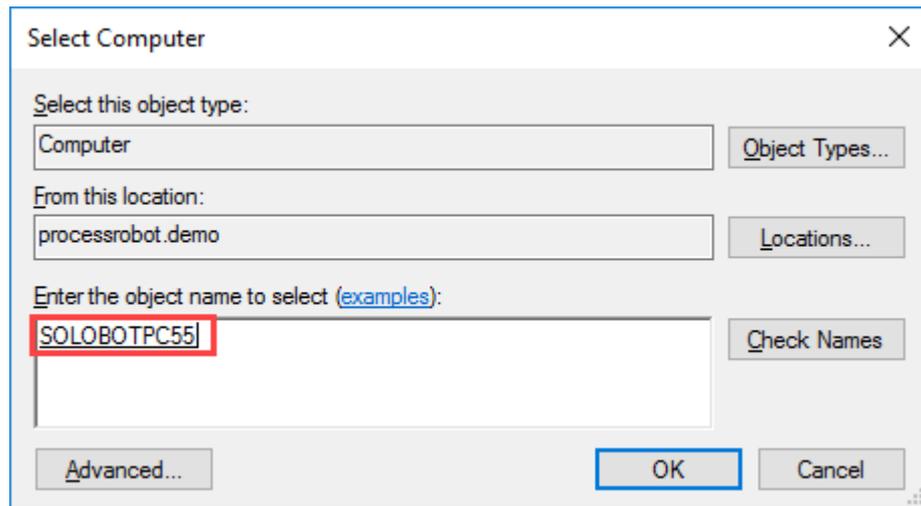


Active Directory machines:

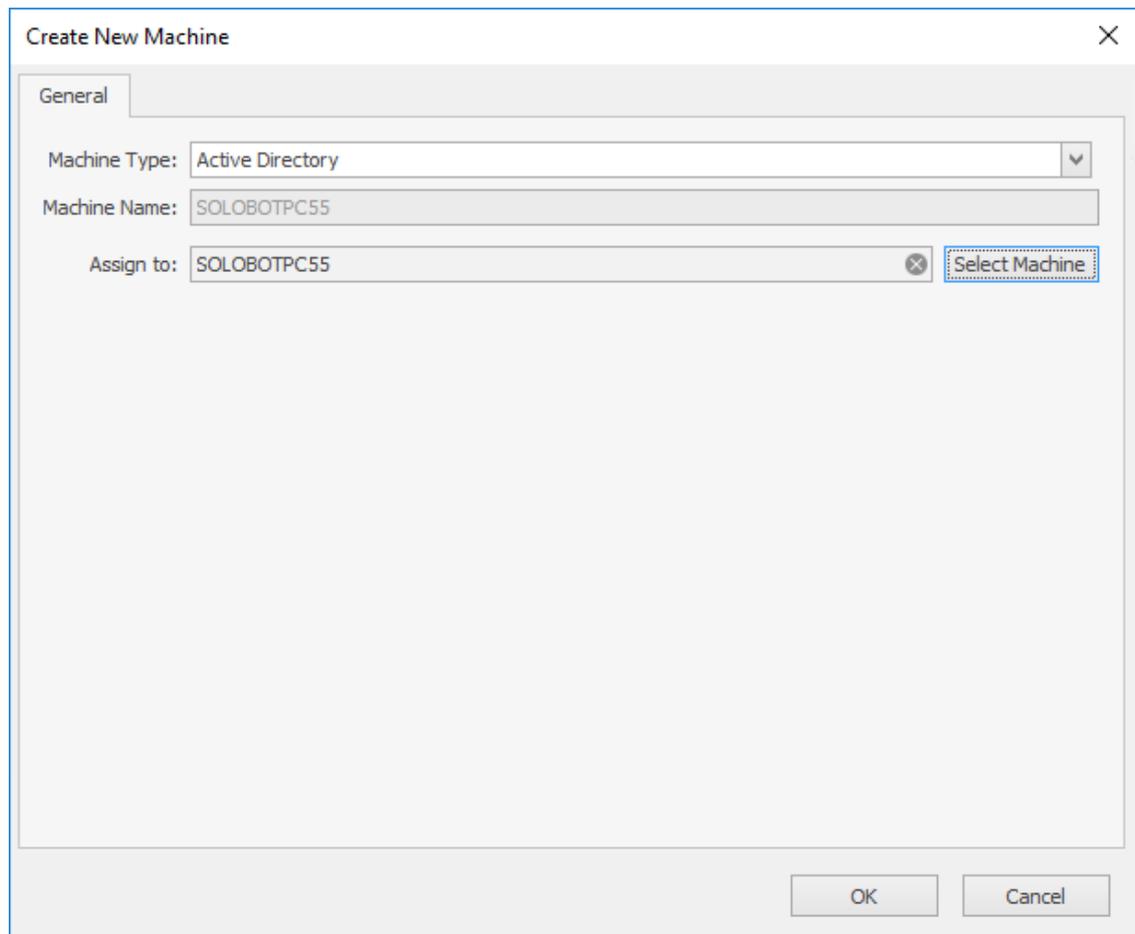
Having selected 'Active Directory' in 'Machine Type', the user must then click the 'Select Machine' button to the right of the 'Assign to' field. Doing so will open the 'Select Computer' window. In the 'Enter the object name to select' field, enter a machine name to search for in the Active Directory domain. Then click 'Check Names'.



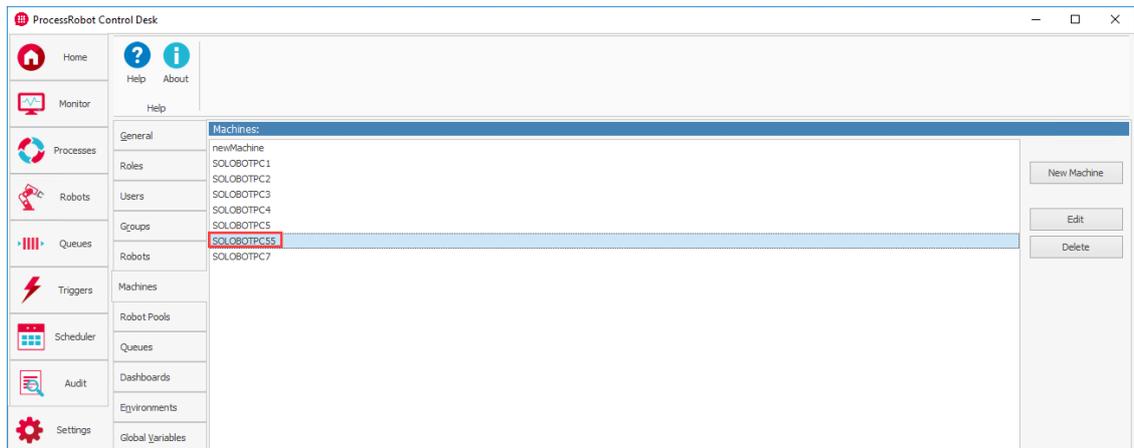
ProcessRobot will search the Active Directory for machine names that include the search term. If one or more relevant results are found, they will appear in the window underneath. Click 'OK' to select the machine.



The Active Directory machine will now appear in the 'Machine Name' and 'Assign to' fields of the 'Create New Machine' window.

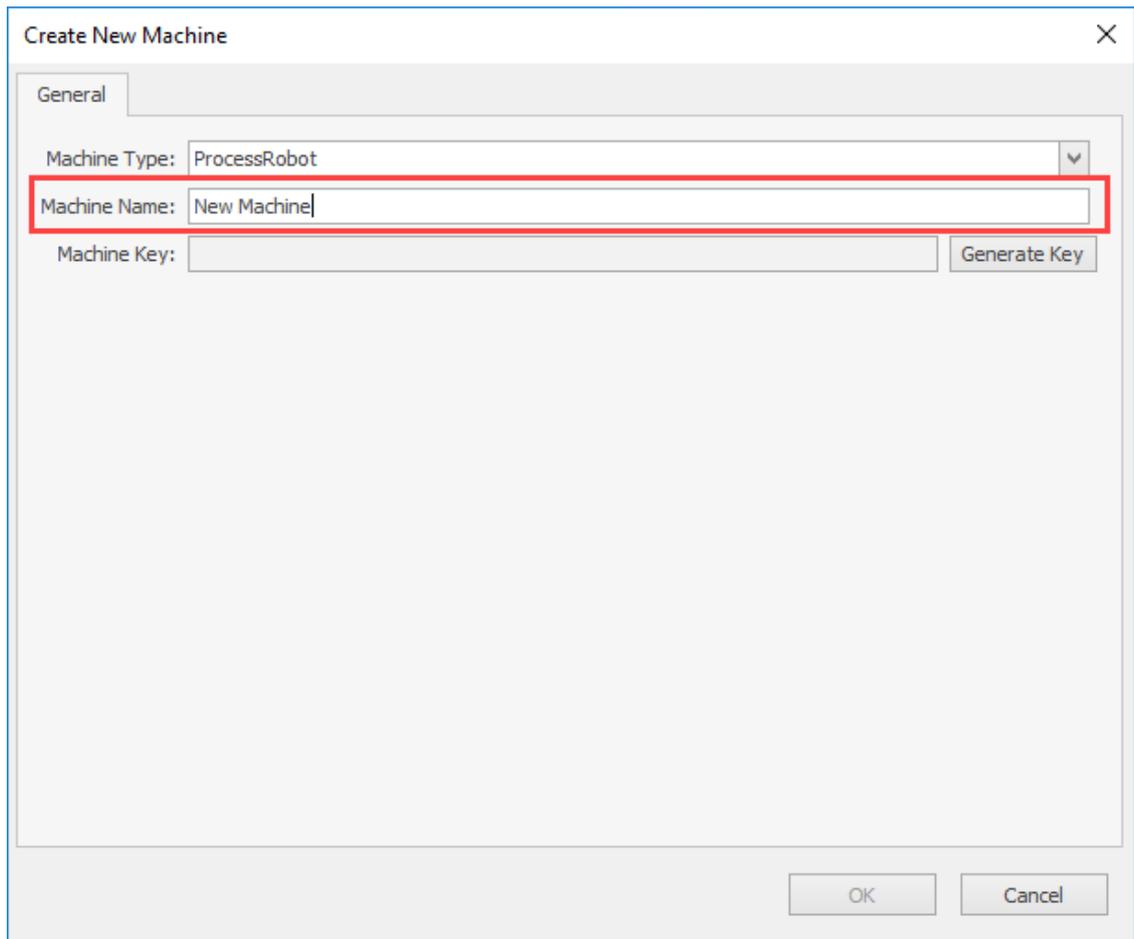


Click 'OK' to add the machine to ProcessRobot.



ProcessRobot machines:

Having selected 'ProcessRobot' in 'Machine Type', the user must enter a 'Machine Name'. This can be an arbitrary value. It serves only as an identifier for the machine within ProcessRobot. It does not need to correspond to the machine's name. The machine does not need to belong to an Active Directory domain.



The image shows a 'Create New Machine' dialog box with the following fields and controls:

- Machine Type:** A dropdown menu with 'ProcessRobot' selected.
- Machine Name:** A text input field containing 'New Machine', which is highlighted with a red rectangular border.
- Machine Key:** An empty text input field.
- Generate Key:** A button located to the right of the 'Machine Key' field.
- OK** and **Cancel** buttons are located at the bottom right of the dialog.

Having provided a Machine Name, the user must click the 'Generate Key' button to key a custom key for this machine.

Create New Machine ✕

General

Machine Type: ProcessRobot

Machine Name: New Machine

Machine Key: 590f09e1-cbc0-48f3-a3a1-29af2179302b Generate Key

Click 'OK' to add the machine to ProcessRobot.

ProcessRobot Control Desk - □ ✕

Home Help About

Monitor Help

Processes

Robots

Queues

Triggers

Scheduler

Audit

Settings

General

Roles

Users

Groups

Robots

Machines

Robot Pools

Queues

Dashboards

Environments

Global Variables

Machines:

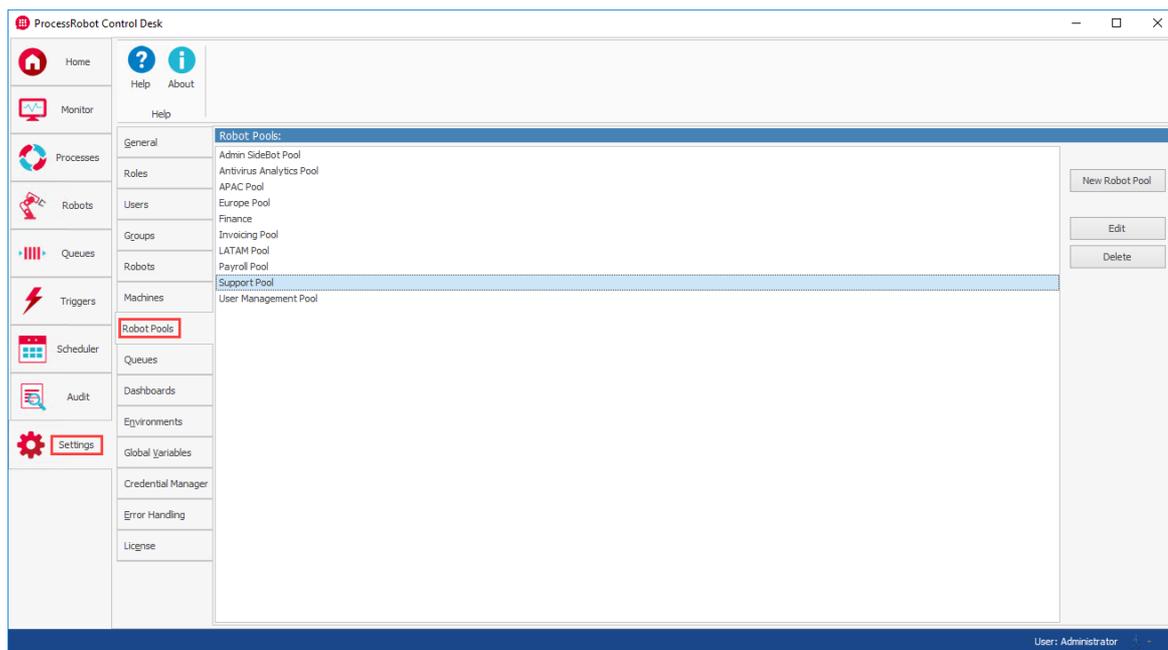
- New Machine
- SOLOBOTPC1
- SOLOBOTPC2
- SOLOBOTPC3
- SOLOBOTPC4
- SOLOBOTPC5
- SOLOBOTPC55
- SOLOBOTPC7

2.1.2.9,7 Robot Pools

When you are trying to Run a Process in ProcessRobot, the first thing that you are being asked, is the "Robot Pool" on which the Process will run.

Robot Pools define a group of one or more Robots. Therefore, you can create many Pools that may have one, two, three or any number of Robots you wish.

For example, you know that you have ten Robots for your Support processes. You can create a "Robot Pool" that will have all those 10 Robots and call it "Support Pool".



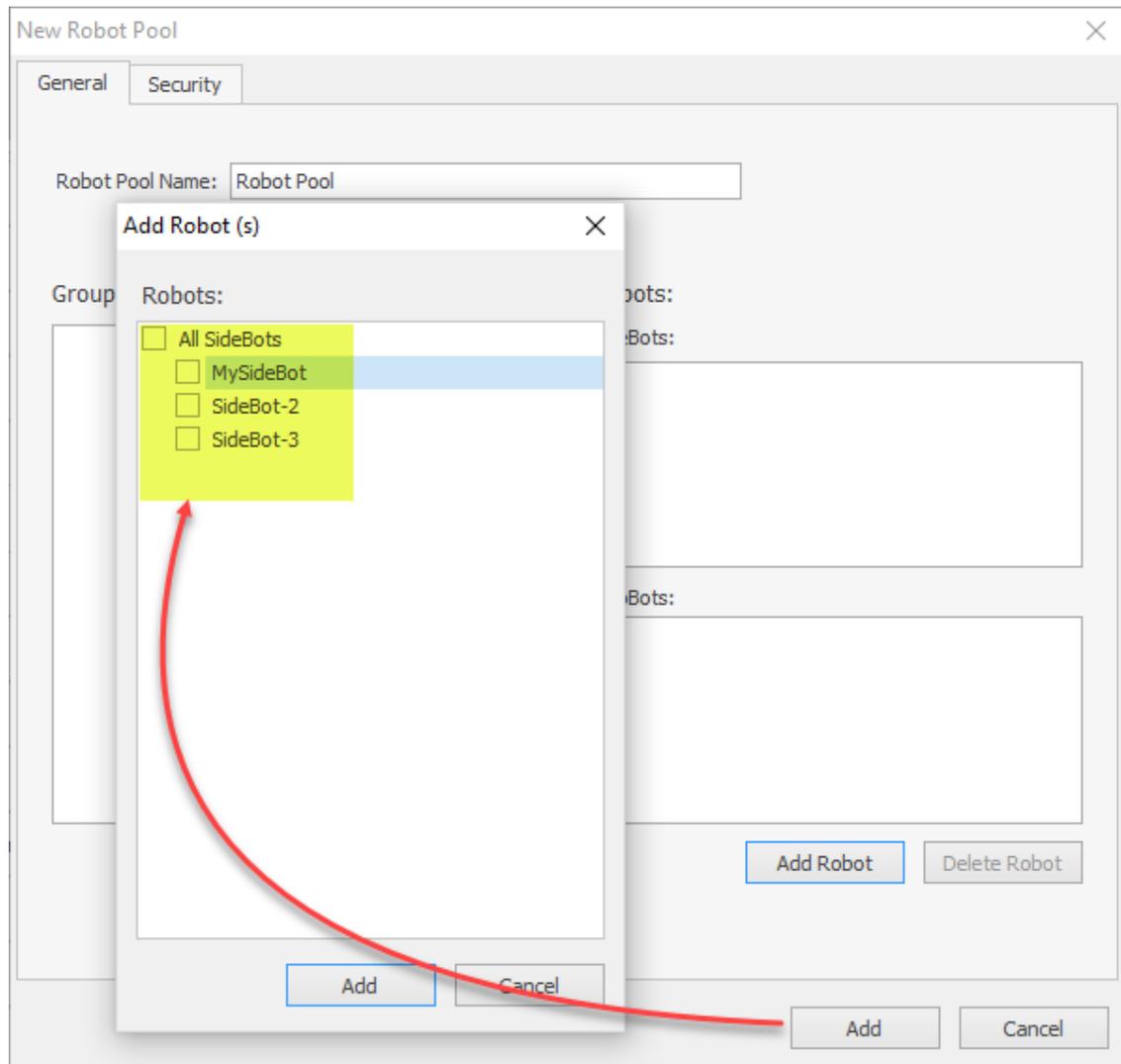
Now, let's see how you can create a New Robot Pool. Press on the "New Robot Pool" button and the following window will appear:

The screenshot shows a 'New Robot Pool' dialog box with two tabs: 'General' (selected) and 'Security'. In the 'General' tab, there is a text input field for 'Robot Pool Name' containing the text 'Robot Pool'. Below this, there are two main sections: 'Groups' and 'Robots'. The 'Groups' section has a large empty rectangular area and two buttons at the bottom: 'Add Group' and 'Delete Group'. The 'Robots' section has two sub-sections: 'SideBots' and 'SoloBots', each with a large empty rectangular area. Below these are two buttons: 'Add Robot' and 'Delete Robot'. At the bottom right of the dialog, there are two buttons: 'Add' and 'Cancel'.

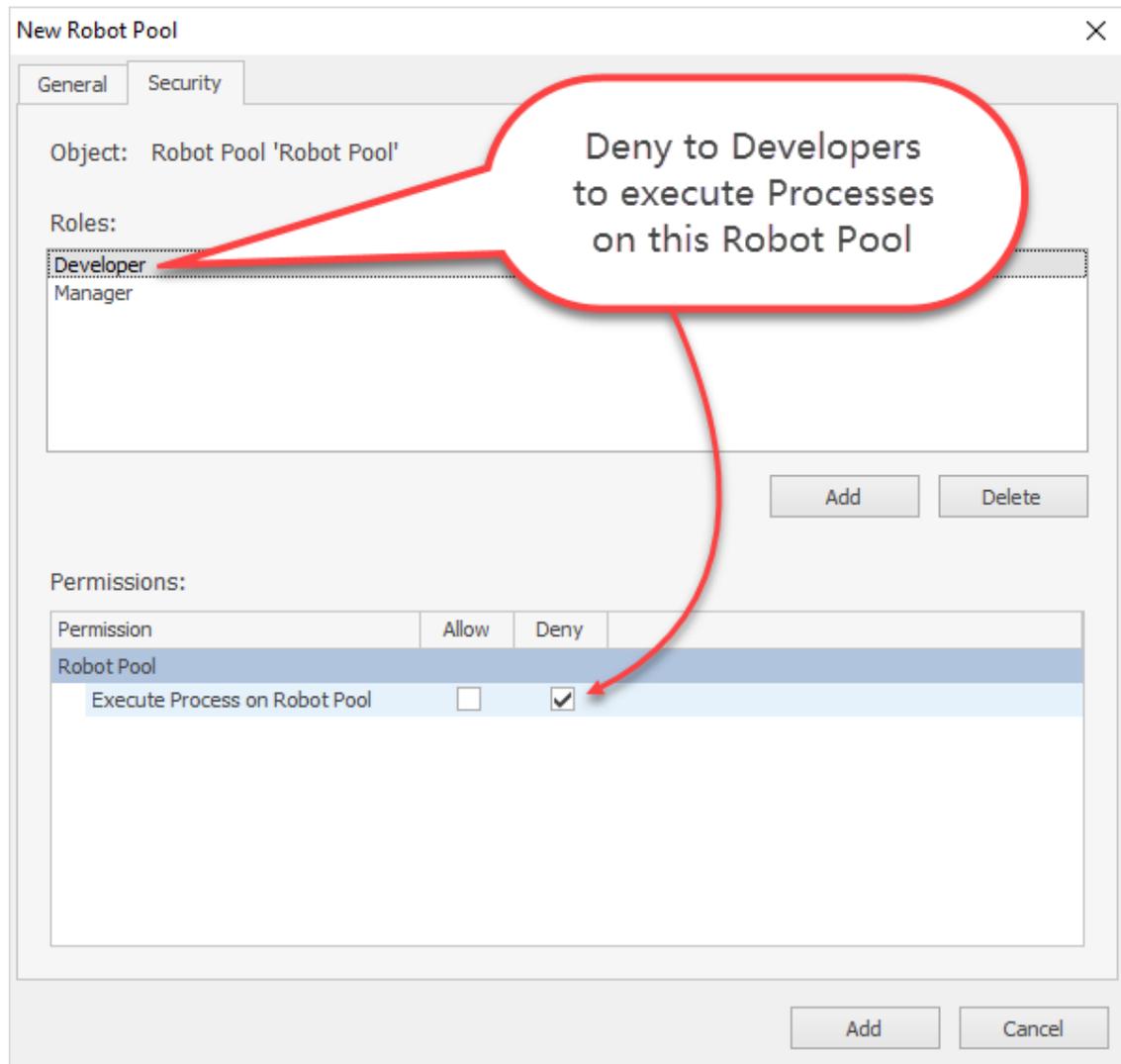
Create a New Robot Pool - General tab

Here you can give a name to your Pool and then you are able to add Groups to it from the available Groups that you have created in the ["Settings > Groups"](#)^[199] tab. You can add 0 or more groups to the same Pool.

Then you can add the Robots that you want your new Robot Pool to have. Click on the "Add Robot" option and check the Solobots or Sidebots that you want the Pool to contain.



Now the second tab is the "Security" tab. In this one you can Add Roles to this Pool and allow or deny users of a specific Role to execute processes. Let's say for example that you have a scheme which includes several robots. You want to allow any user that has the "Manager" Role to execute Processes, but deny the execution of Processes on this robot Pool to anyone who is under the "Developer" Role.



2.1.2.9,8 Queues

Queues can store multiple items. Queue items can be of three data types: text, numeric or custom object. The user can configure a Queue and the items it contains to be handled in multiple ways. The two main ways to configure are:

Sequential Queues - "Passive" use of Queues. Items are added in sequential order to the Queue. A Process can be executed manually to retrieve the first available item in the Queue and perform an action with it.

Transactional Queues - "Active" use of Queues. The Queue has been configured so that a Process is executed each time an item is added to the Queue. This configuration is achieved by checking the "Enable Automatic Distribution of Queue Items" box when and selecting the Process the items will trigger creating the Queue.

In the Settings > Queues Tab, click "New Queue" to create a new Queue. Click "Edit" to edit the existing Queues at any point.

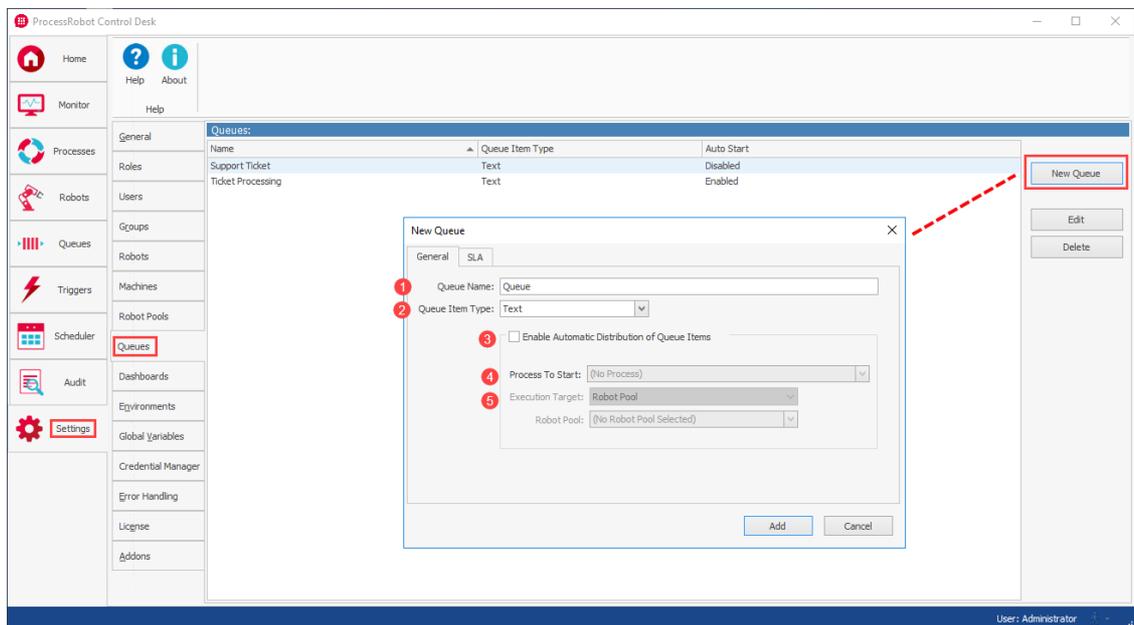
When creating or editing a Queue, the **Service Level Agreement settings (SLA)** can be configured. These define the way that Queue Items of different priority (Low, Normal, High) will be handled for the specific Queue.

After clicking on the "New Queue" button, the user can specify the following in the new window that appears:

1. The Name of the Queue.
2. The Queue Item Type. (Text, Numeric or Custom Object.)
3. Enable Automatic Distribution of Queue Items.

If this last option is checked [3], then every time an Item is added to the Queue, the "Process To Start" fires on the specified Execution Target ([Robot](#) [200] or [Robot Pool](#)) [217].

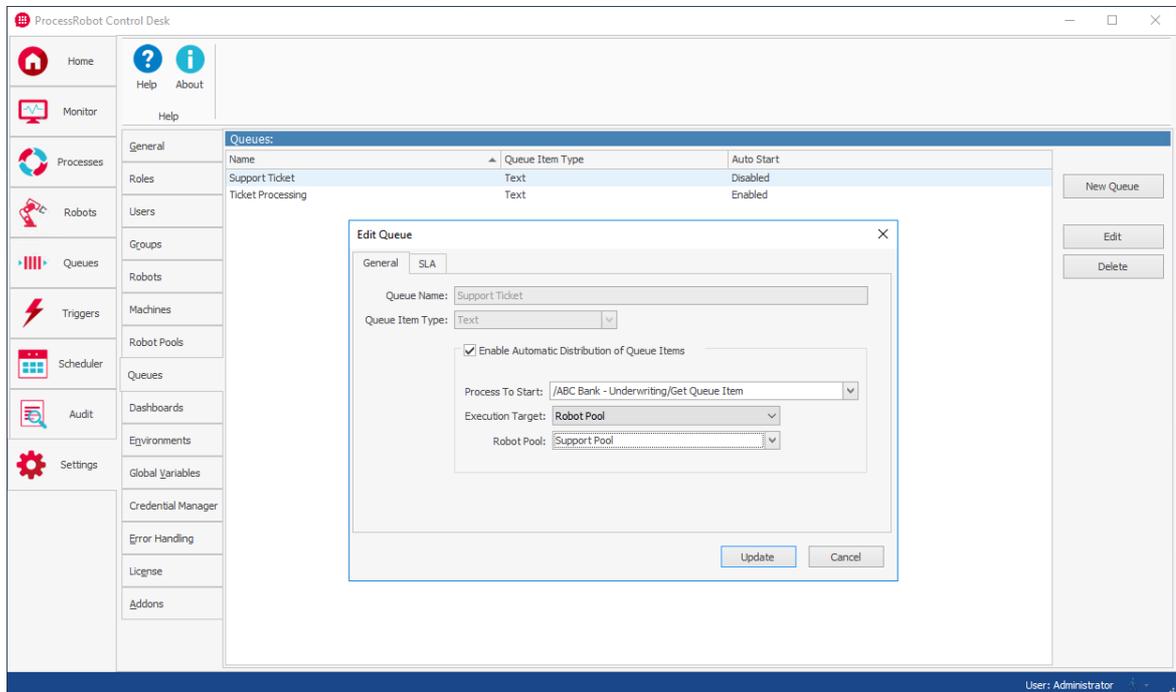
If this option is not checked then there will be no Process designated to run automatically once an Item is added. In this case, Users may manually run a Process to [get an Item from the Queue](#) [1012] and use the item to perform the desired actions.



Control Desk Settings > Queues

Once a Queue is created, when the user clicks "Edit", only the "Process to Start" [4] and Execution Target [5] can be edited. The Name [1] and Queue Item Type [2] cannot be changed, as they might be in use by Processes in Production.

PLEASE BEAR IN MIND THAT PROCESSROBOT DOES NOT ALLOW THE EDITING OF THESE PROPERTIES IN ORDER TO PREVENT ACCIDENTAL MISCONFIGURATION.



Edit Queue

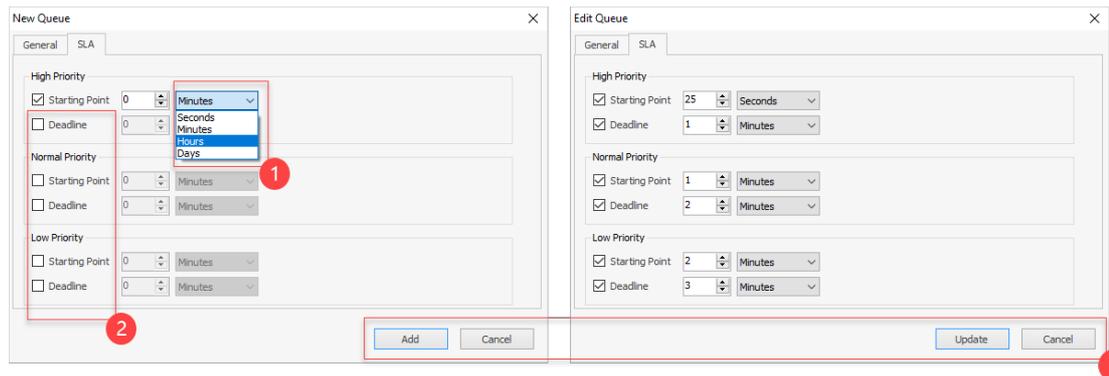
If it is absolutely necessary to change these properties of a Queue, it is recommended to create a new Queue with the new, desired configuration.

To Delete a Queue, select it from the list of existing Queues and click "Delete".

SLA Settings

Both the New Queue and Edit Queue Windows allow for the creation or configuration of SLA Settings that define the way that Queue Items of different priority status will be handled for the specific Queue. This is possible via the SLA Tab next to the General Tab of these Windows.

The interface of the SLA Settings is similar in both cases:



The only difference is that the New Queue Window allows the user to Add this New Queue [3] to the existing ones while the Edit Queue [3] Window allows them to Update the SLA Settings of an existing Queue.

The Settings allow the user to set a Starting Point for how long until the designated Process should execute after the Queue Item with a given priority has been added to that Queue. Each Starting Point can be accompanied by a Deadline. The Deadline marks the end of the time period the Queue will treat the Items according to their priority level.

If an Item is not processed during the specific time period, then it is marked on the Queue Items Window Status Column as **Expired**. Otherwise, a Queue item can be marked as:

- Completed, if the Process that is [Getting the Queue Items](#)^[1012] has completed successfully
- Failed, if the Process did not complete successfully
- New, if the Item has not yet been handled by the Process associated with the Queue
- In Progress, if the item is currently being handled

Let us for example consider a Queue that is being consumed by a specific Process that has Concurrency Limit 4 in its Properties and has executed on an Execution Target of a Robot Pool that consists of 5 Robots. Given these specs we understand that the maximum number of In Progress Items at any one time in this example is $4 \times 5 = 20$ if there is no concurrency limit imposed on a Robot level.

You can access the Queue Items Window by double clicking on a Queue at the central Queue Tab of the Control Desk:

The screenshot shows the ProcessRobot Control Desk interface. The main window displays a summary of the 'Support Tickets' queue with the following data:

Queue Name	High Priority Items	Normal Priority Items	Low Priority Items	Automatic Distribution	Type
Support Tickets	0 New, 0 In Progress	0 New, 0 In Progress	0 New, 0 In Progress	Running	Text
Client Directory	0 New, 0 In Progress	0 New, 0 In Progress	0 New, 0 In Progress	Disabled	Custom Object
Numeric ID	0 New, 0 In Progress	0 New, 0 In Progress	0 New, 0 In Progress	Disabled	Numeric

A red arrow points to the 'Support Tickets' row with the text: "Double-Click anywhere on the Queue row".

The detailed view of the 'Support Tickets' queue items is shown below:

Timestamp	Value	Added by Process	Added by Robot	Added by User	Priority	Status	Starting Point	Deadline	Processed by Robot	Processed by Process
11/30/2017 10:48:14 AM	volutpat	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	sapient	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	interdum.	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	Nulla	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	facilisi.	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	Aenean	Adding Queue Item	SideBot		Normal	Completed	11/30/2017 12:48:39...	11/30/2017 12:50:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	elementum	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	metus	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	id	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	ligula	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	aliquam	Adding Queue Item	SideBot		Normal	Completed	11/30/2017 12:48:39...	11/30/2017 12:50:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	elementum.	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	Quisque	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	tristique	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	turpis	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	in	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	auctor	Adding Queue Item	SideBot		Normal	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	sodales.	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	Prasent	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	solicitudin	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	vel	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	nisi	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	vehicula.	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	vulputate	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	vehicula.	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	Donec	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	porta	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	sanctus.	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item

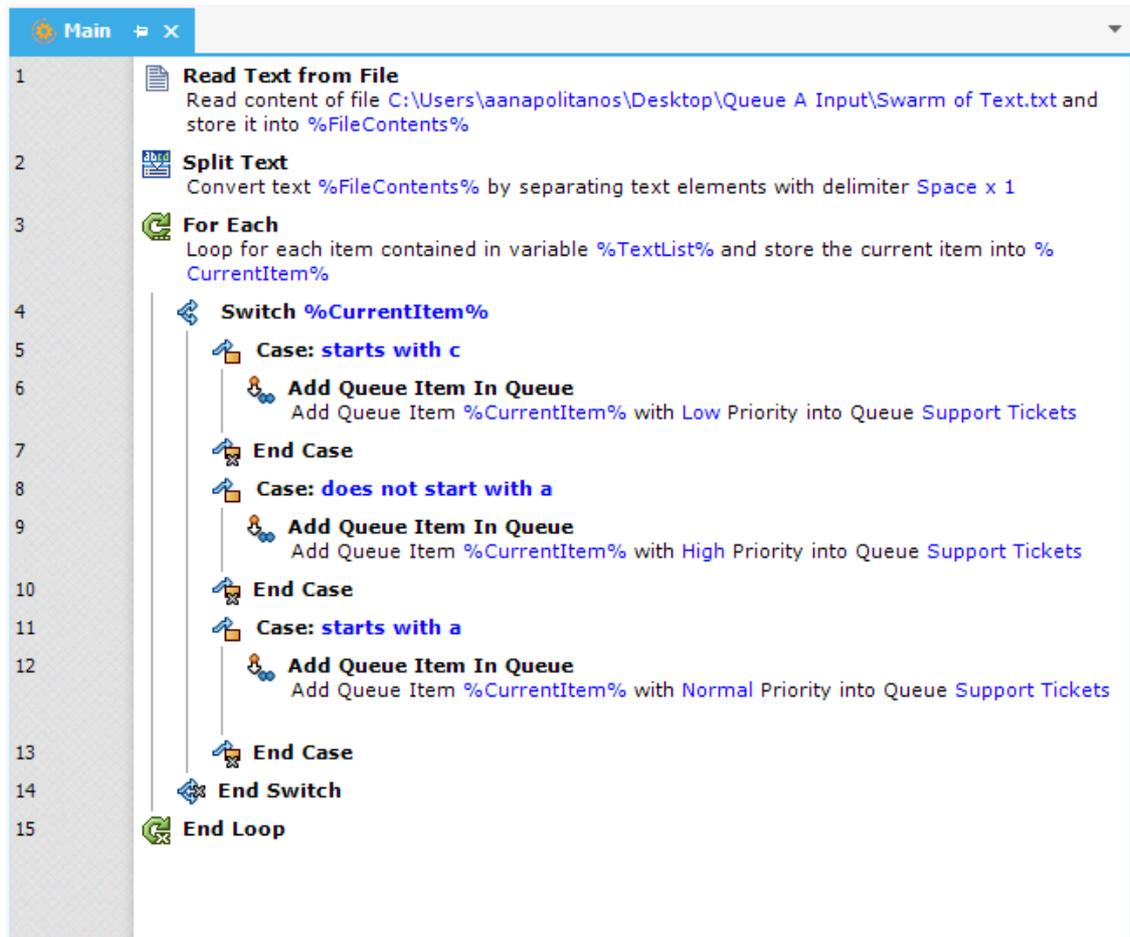
How Queues Work

Let us create a very simple Queue along with two Processes. One Process will add items of various Priority levels to a Queue, while the second Process will retrieve those items from the Queue.

We use process "Add Queue Item" to insert Queue Items inside the Queue "Support Tickets" and process "Consume Queue Item" to consume them through the automatic distribution feature.

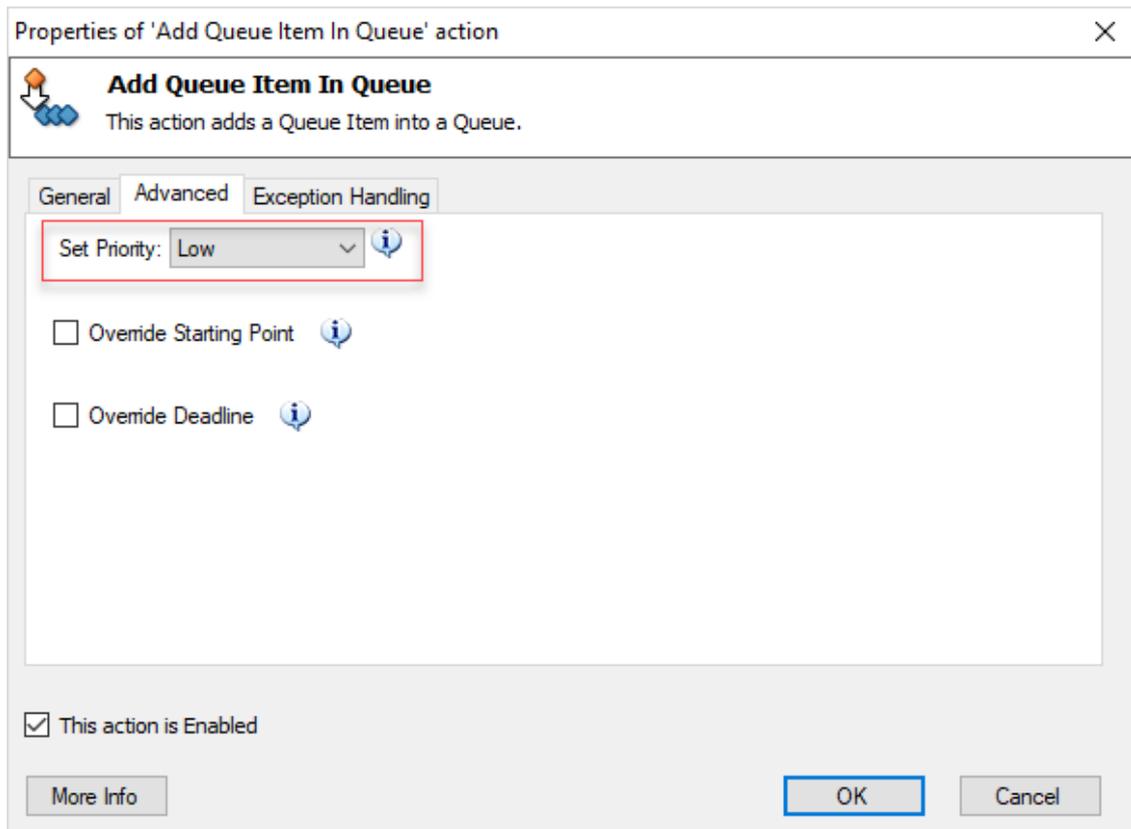
What the Add Queue Items does, is to Read some [Lorem Ipsum](#) text from a file, split this text into words and then loop through the produced [list](#)⁴³⁸ in order to add each word as a Queue Item (text) of varying priority status:

- If the word starts with 'c' then it is set as Low Priority status,
- If it does not start with 'a' it is set as of High Priority status (most words don't) and
- if it starts with 'a' it is set as of Normal Priority status



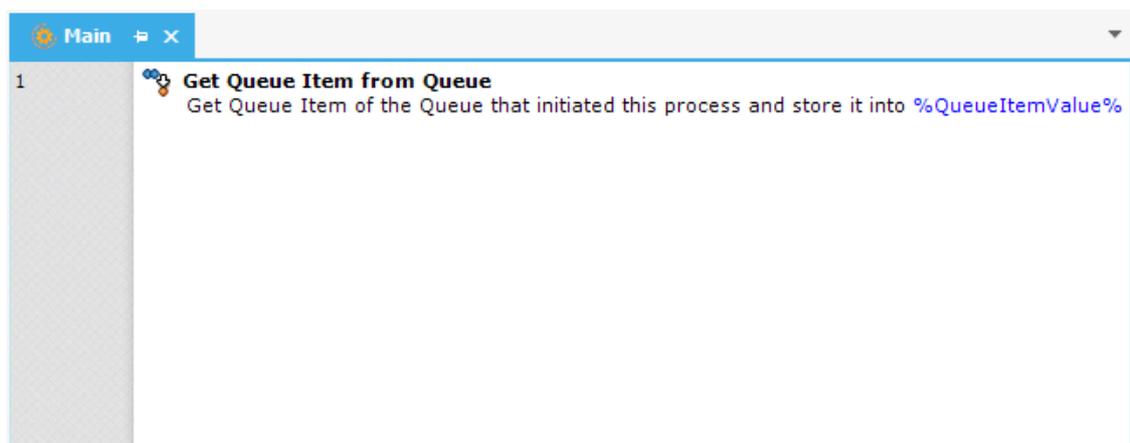
Please notice how eloquently we can describe this multiway branch conditional structure using a [Switch](#)⁵⁷⁸-[Case](#)⁵⁸⁰ combination.

We can set the Priority status we wish to an item via the properties of the [Add Queue Item in Queue](#)¹⁰¹⁰ > Advanced Tab:



Moving that process in production, we are ready to start working on the Consuming Process i.e. the "Consume Queue Item".

This has not to be complex:



A single action is enough to let us get going!

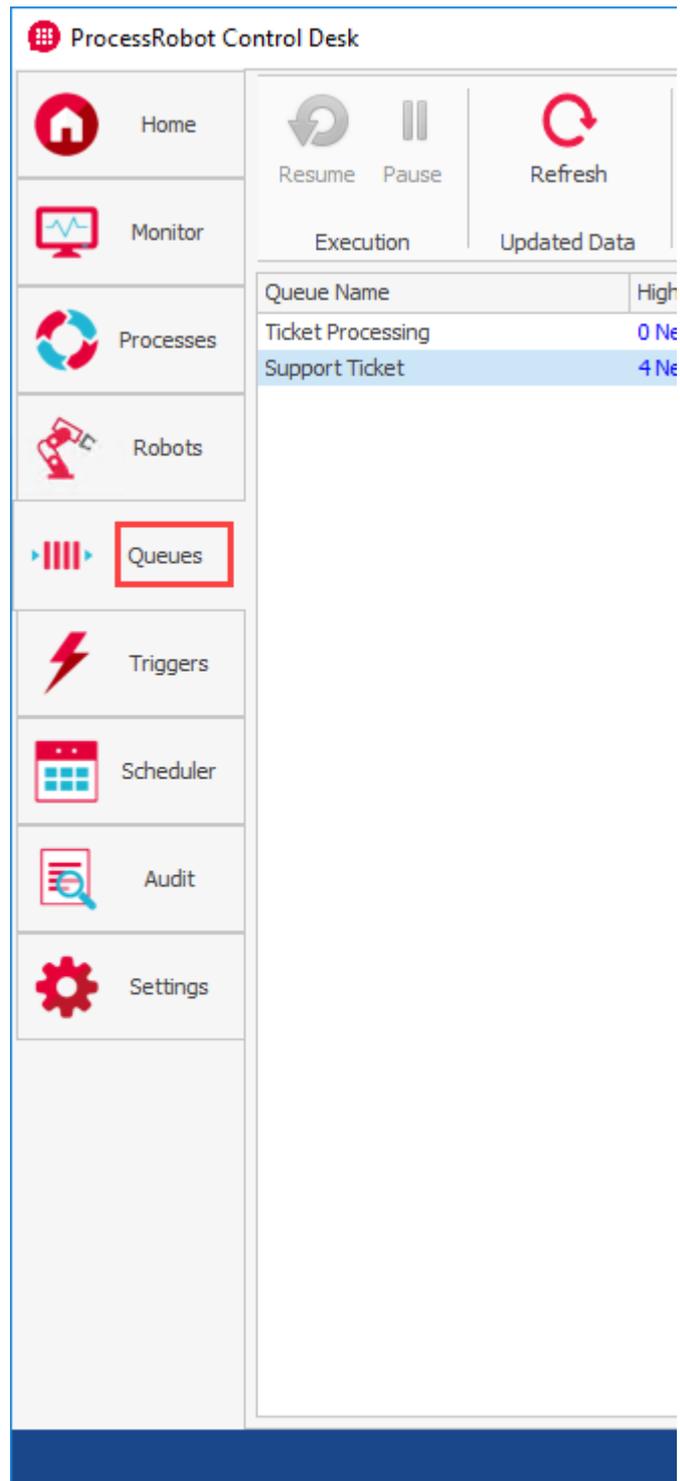
Since we want to consume Items via the automatic distribution feat we should not forget to specify Get: Queue Item that Initiated this Process in the Get Queue Item from Queue action.

The screenshot shows a dialog box titled "Properties of 'Get Queue Item from Queue' action". The dialog has a close button (X) in the top right corner. Below the title bar, there is a description: "Get Queue Item from Queue" followed by "This action retrieves the first Queue Item from a Queue or the Queue Item that initiates this process." Below this, there are two tabs: "General" (selected) and "Exception Handling". The "General" tab contains two sections: "Action Input" and "Action Output". In the "Action Input" section, there is a dropdown menu labeled "Get:" with the value "Queue Item That Initiated This Process" and an information icon (i). In the "Action Output" section, there is a text field labeled "Store Queue Item Into:" with the value "%QueueItemValue%" and an information icon (i). At the bottom of the dialog, there is a checkbox labeled "This action is Enabled" which is checked. There are three buttons at the bottom: "More Info", "OK", and "Cancel".

After doing this we are ready to move this process as well in production and start developing our Queue.

Queues are very simple and handy to use. Two things you need to understand however, before you are ready to thrive in Queue development is that Queues are controlled from the Control Desk: there we can find two different Tabs containing the word Queues:

The Main 'Queues' Tab serves Monitoring purposes:



ProcessRobot Control Desk

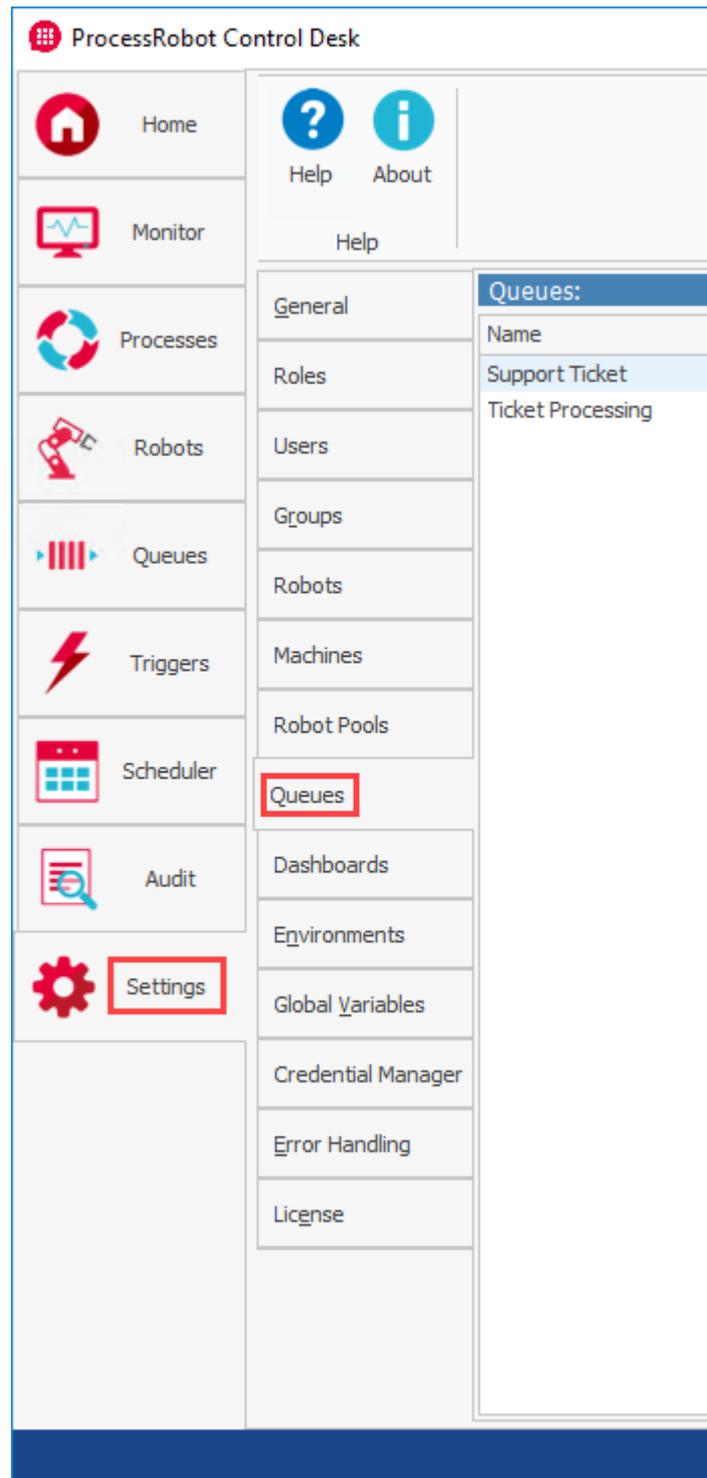
Home Monitor Processes Robots Queues Triggers Scheduler Audit Settings

Resume Pause Refresh

Execution Updated Data

Queue Name	High
Ticket Processing	0 New
Support Ticket	4 New

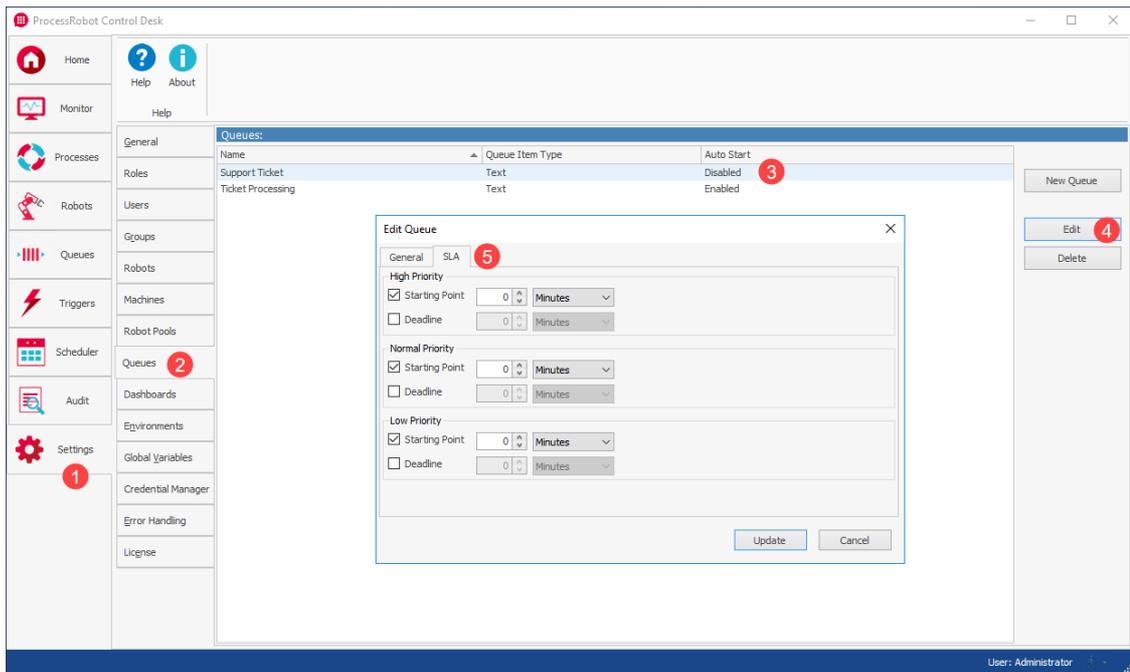
The other Queues Tab, accessible via the Settings Tab, offers the user the ability to create and edit Queues and their Settings.



So unlike Processes that are developed in Process Studio and Process Designer, Queues are created and edited in the [Settings > Queues](#) ²²⁰ Tab of the Control Desk. Through this Tab you can also set the default Service Level Agreement Settings (SLA) of the Queue, defining how Queue Items of different Priority status will be processed in time.

Whether the Queue you want to define its SLA Settings exists or not, the steps to follow are the same:

a) If the Queue already exists you need to go to the [Settings > Queues](#) Tab [1] -> [2], select that Queue from the main pane [3] and then press Edit [4]. Then you have not but to click on the SLA Tab [5] of the resultant Edit Queue Window:



b) If the Queue does not exist then you need to follow exactly the same process as in (a) but instead of clicking on the Edit button this time you will click on the New Queue and instead of selecting the SLA Tab from the Edit Queue, you will do so from the New Queue Window.

Now that we know the basics we are ready to return to the reality of our example. Creating the "Support Tickets" [1] we are careful of selecting "Text" [2] on Queue Item Type (since its Queue Item is nothing but a word from the Lorem Ipsum text), to check the Enable Automatic Distribution of Queue Items [3] and then select the already in production "Consume Queue Item" process [4] from the drop down and define the Execution Target [5] (i.e. where do we want the "Consume Queue Item" process to run). This can be done via the provision of either a Robot Pool [6] or a specific Robot.

Now let us move Tabs in order to appoint the SLA Settings:

The above arrangement will produce the following result:

a) **High Priority Items** that consist the majority of our Items due to the logic of our "Add Queue Item" process will start being processed 25 seconds after they are being added in the Queue. The Deadline for each of these Items in order to be successfully processed and gain the status Completed is a minute after they have been added in the Queue. That means that Items need to be processed by our resources within 35 seconds or else they will become expired.

b) **Normal Priority Items** will start being processed 1 minute after they are added within our Queue and will become expired if they have not been processed within another minute from their entry.

c) **Low Priority Items** will start being processed 2 minutes after they are added in our Queue and will become expired if they have not been processed within another minute from their entry.

--> Click Add and that's it!

To see your Queue in action, simply go to Processes Tab in the main Control Desk, run the "Add Queue Item" from there and click on the Main Queue Tab. You should be already looking at a non empty Support Tickets Queue. The real fireworks however will start 25 seconds later when High Priority Items will start being processed; in order to see that spectacle at its full glory double click upon the Queue Name and observe how the status of the Queue Items is transitioning over time:

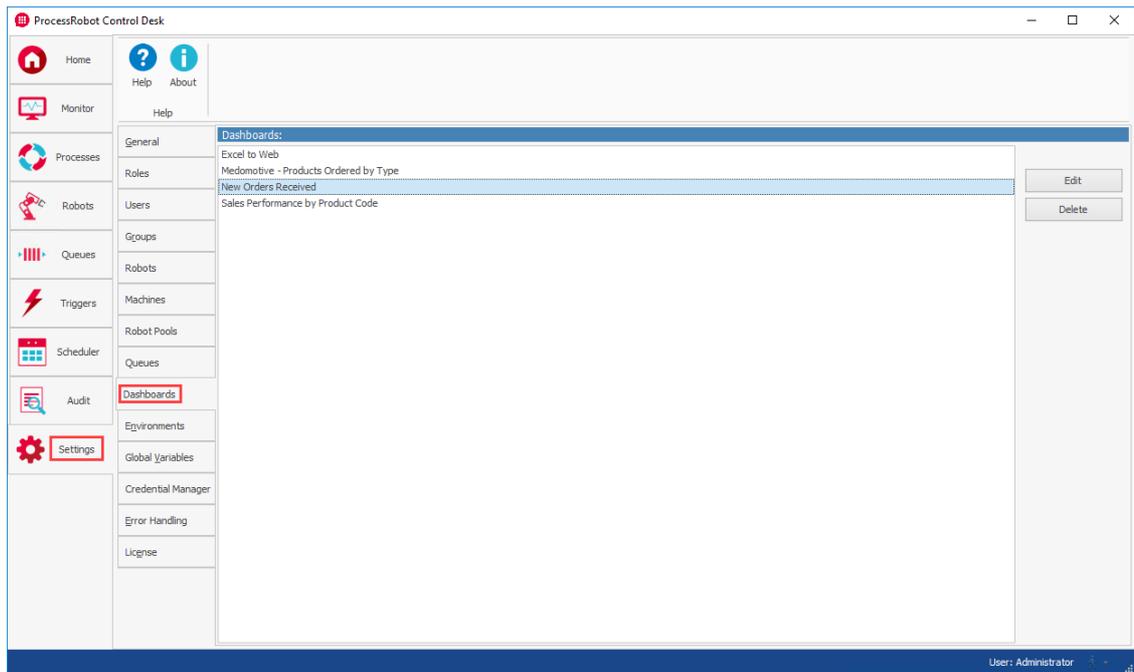
The screenshot shows the ProcessRobot Control Desk interface. The 'Processes' tab is active, displaying a table of queues. The 'Support Tickets' queue is highlighted, and a red arrow points to it with the text "Double-Click anywhere on the Queue row". Below this, a detailed view of the 'Support Tickets' queue is shown, displaying a list of items with columns for Timestamp, Value, Added by Process, Added by Robot, Added by User, Priority, Status, Starting Point, Deadline, Processed by Robot, and Processed by Process.

Timestamp	Value	Added by Process	Added by Robot	Added by User	Priority	Status	Starting Point	Deadline	Processed by Robot	Processed by Process
11/30/2017 10:48:14 AM	vulnbat	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	sapnem	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	interdum.	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	Nulla	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	facilisi.	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	Aenean	Adding Queue Item	SideBot		Normal	Completed	11/30/2017 12:49:14...	11/30/2017 12:50:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	elementum	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	metus	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	id	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	ligula	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	aliquam	Adding Queue Item	SideBot		Normal	Completed	11/30/2017 12:49:14...	11/30/2017 12:50:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	elementum.	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	Quisque	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	tristique	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	turpis	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	in	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	auctor	Adding Queue Item	SideBot		Normal	Completed	11/30/2017 12:49:14...	11/30/2017 12:50:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	socios.	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	Praesent	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	sed dictum	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	malesuada	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	vehicula.	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	Donec	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	porta	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	duis.	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item
11/30/2017 10:48:14 AM	duis.	Adding Queue Item	SideBot		High	Completed	11/30/2017 12:48:39...	11/30/2017 12:49:14...	SideBot	Remove Queue Item

One last thing that you should know is that the number of the In Progress Items is being influenced by the Concurrency Limits that you have set for the "Consume Queue Item" Process and Robots that are running the Process!

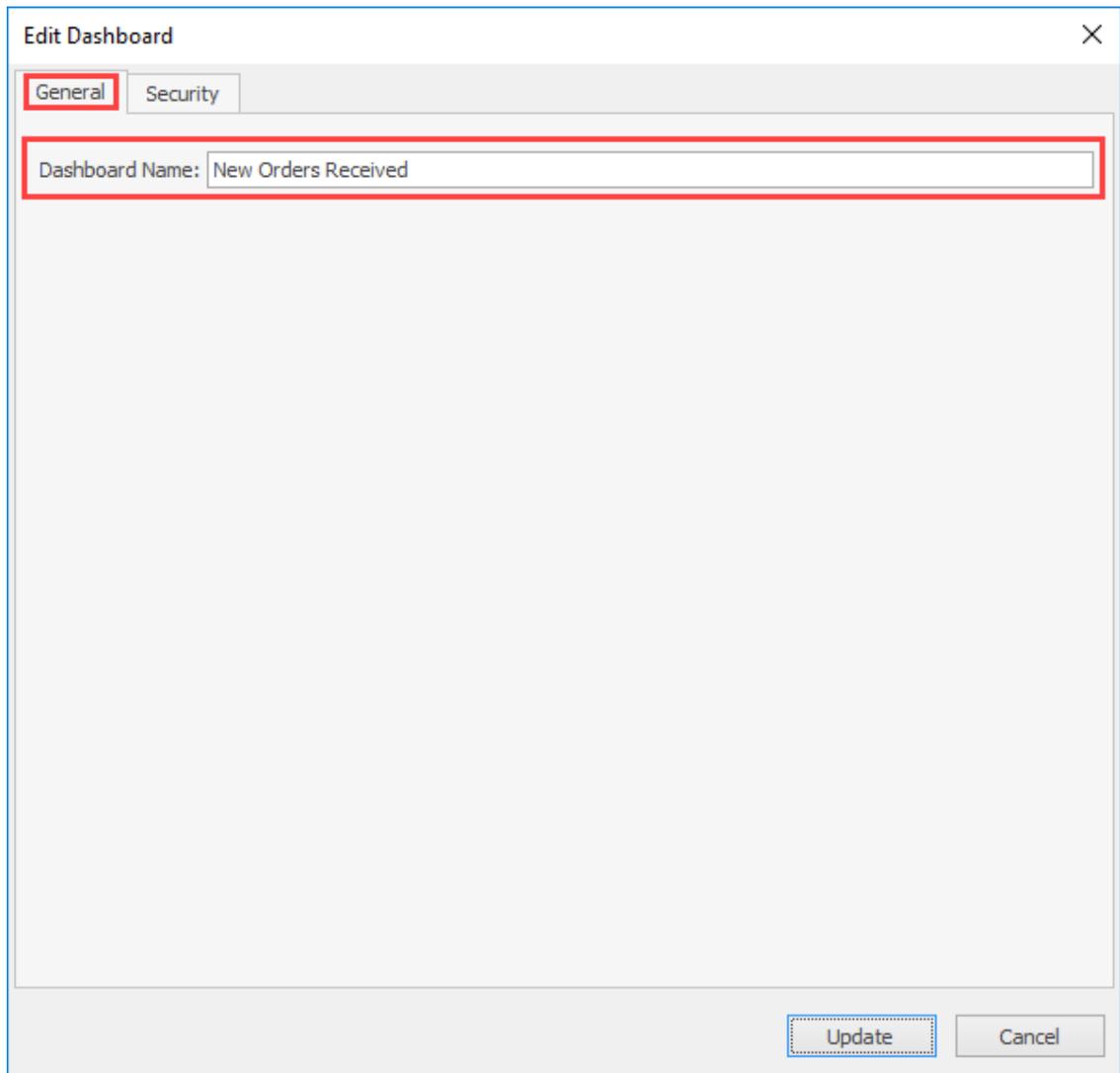
2.1.2.9.9 Dashboards

From Settings > Dashboards, users can edit or delete Dashboards that have been created in the Dashboard Designer.



When editing a Dashboard, there are two configurable properties:

The 'Dashboard Name' under the 'General' Tab:



The screenshot shows a dialog box titled "Edit Dashboard" with a close button (X) in the top right corner. Below the title bar are two tabs: "General" (which is highlighted with a red border) and "Security". Under the "General" tab, there is a text input field labeled "Dashboard Name:" containing the text "New Orders Received". This input field is also highlighted with a red border. At the bottom right of the dialog box, there are two buttons: "Update" and "Cancel".

The Permissions to view and edit Dashboards for each Role under the 'Security' Tab:

Edit Dashboard
✕

General

Security

Object: New Orders Received

Roles:

Manager

Add

Delete

Permissions:

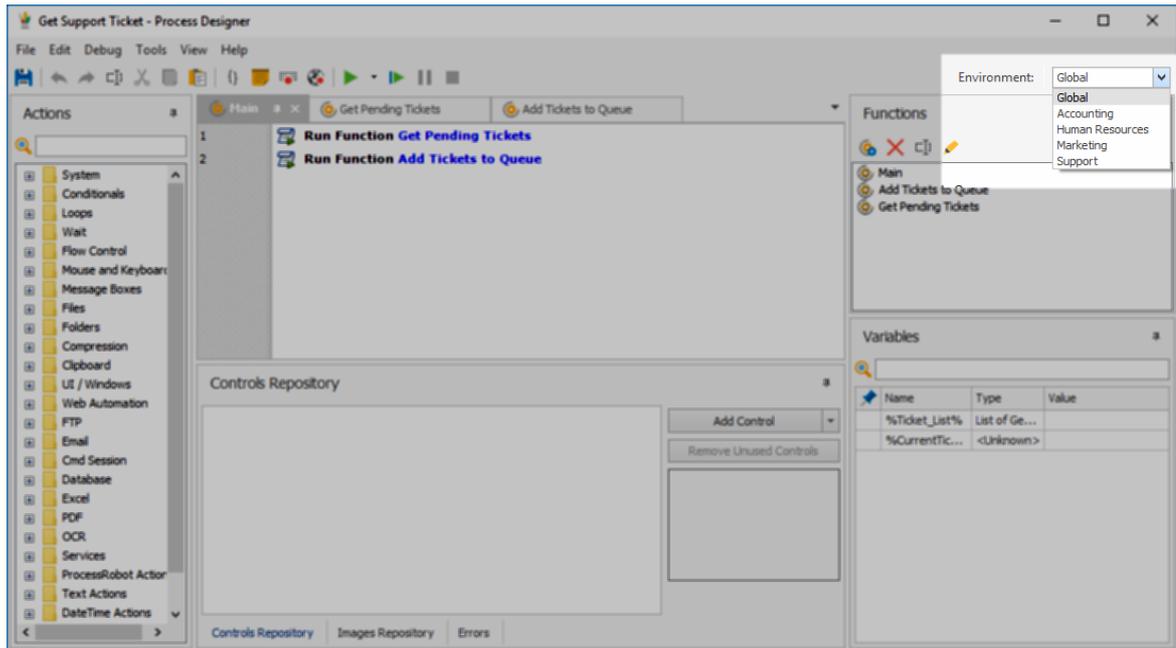
Permission	Allow	Deny	
Dashboard	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
Edit Dashboard	<input type="checkbox"/>	<input type="checkbox"/>	
View Dashboard	<input checked="" type="checkbox"/>	<input type="checkbox"/>	

Update

Cancel

2.1.2.9,10 Environments

When you are building a Process in the Process Designer, in the top right corner it is possible to specify an Environment in which the Process will run when executed from the Debugger. This simulates how the Process will behave when it is actually assigned to a Robot in that Environment when in Production. The choice of Environment in the Process Designer does not affect the Production Environment.

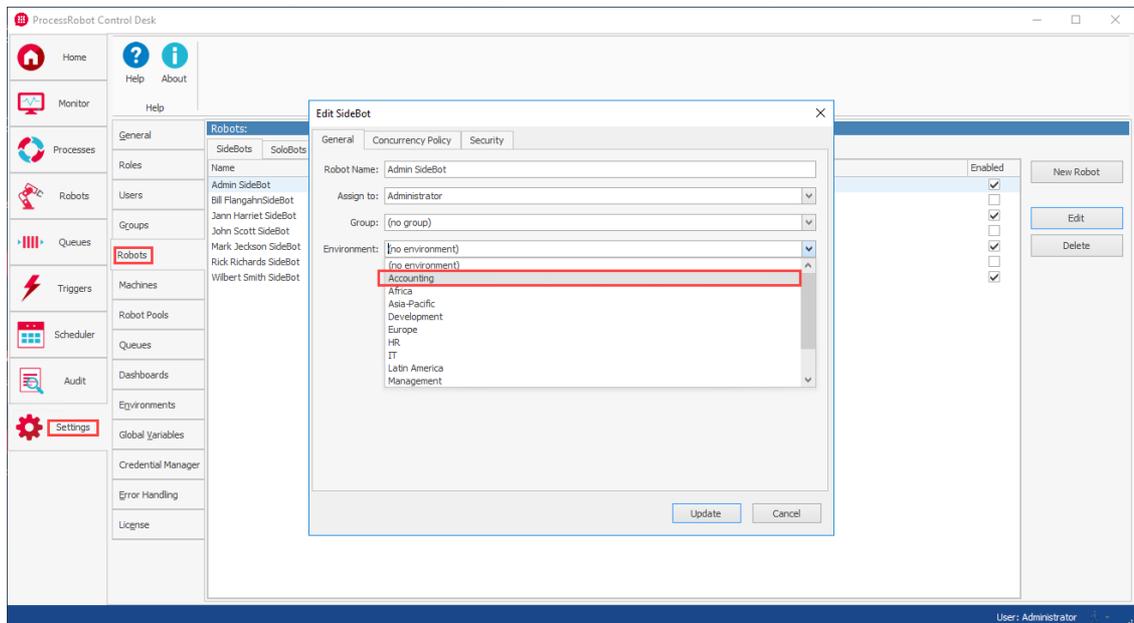


The Environments in the Robot Designer

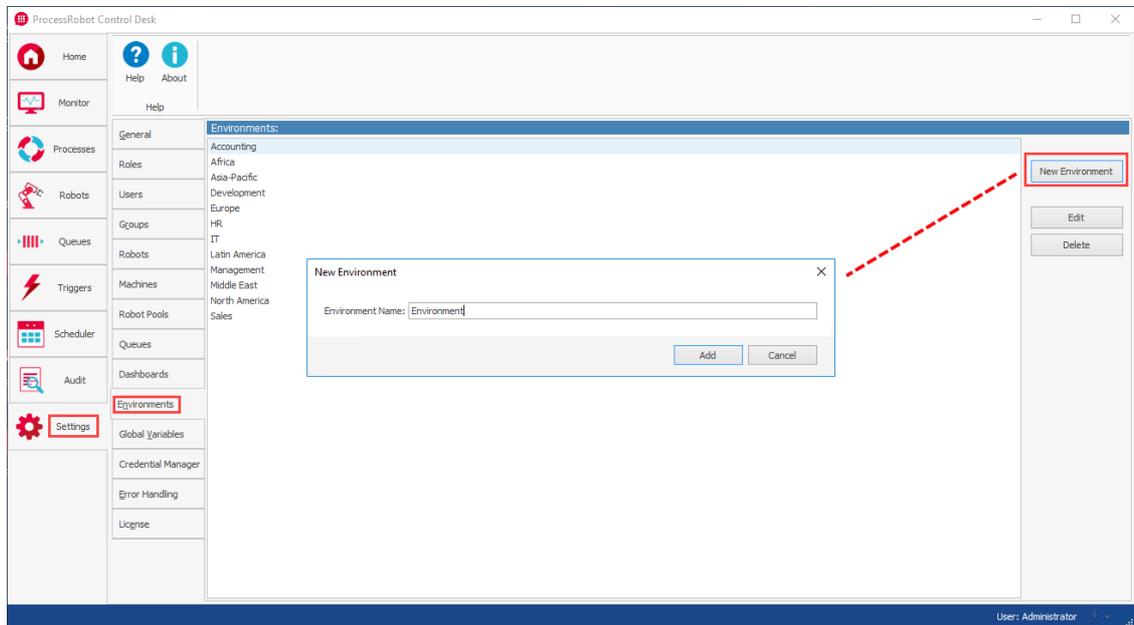
The Environments are associated with Global Variables that the user can set in the "[Settings > Global Variables](#)"²³⁷ Tab in the Control Desk.

In the "Settings > Environments" tab the user can create a "New Environment" and name it, "Edit" or "Delete" it.

Regulate in which Environment a Process that is in Production will run, by executing it in a Robot that belongs to the corresponding Environment:



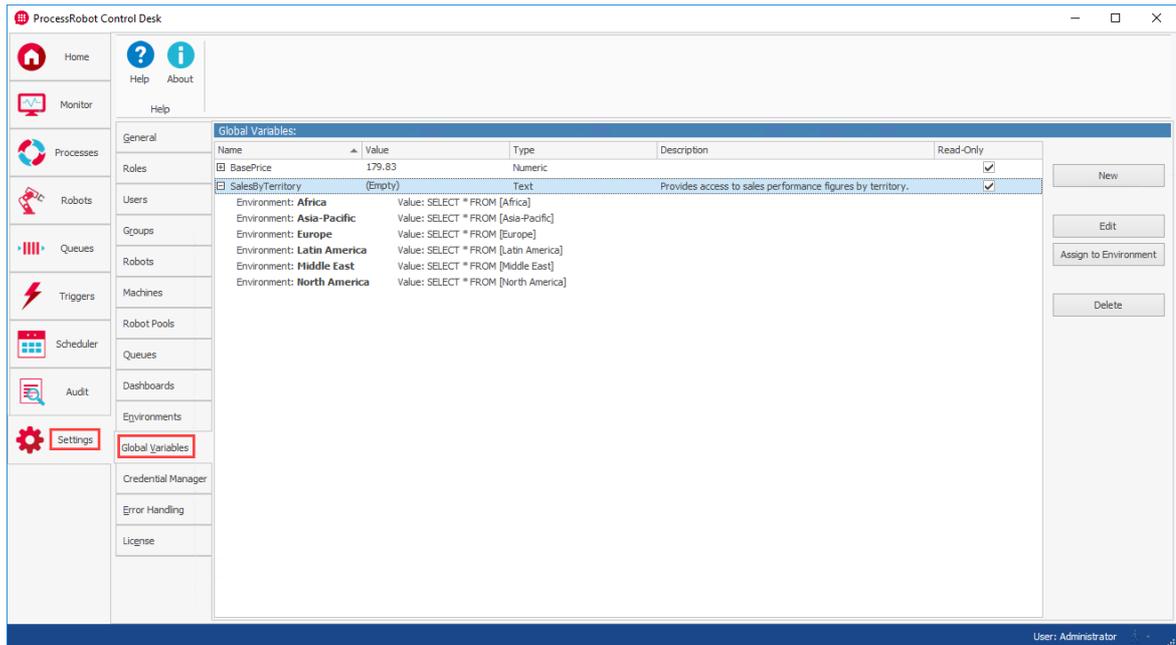
The use of Environments and their association with Global Variables is explained in the [Global Variables](#) ^[237] topic.



Control Desk Settings > Environments

2.1.2.9,11 Global Variables

Global Variables are variables that can be set and used from all the Robots and Processes. You can create your Global Variables in the "Settings > Global Variables" in the Control Desk by clicking on the "New" button.



Control Desk Settings > Global Variables

After pressing on the New button then you are prompted to enter:

Name: GlobalVariable

Description:

Type: Text

Read Only

Value: (Empty)

Name: The Name of the Global Variable.

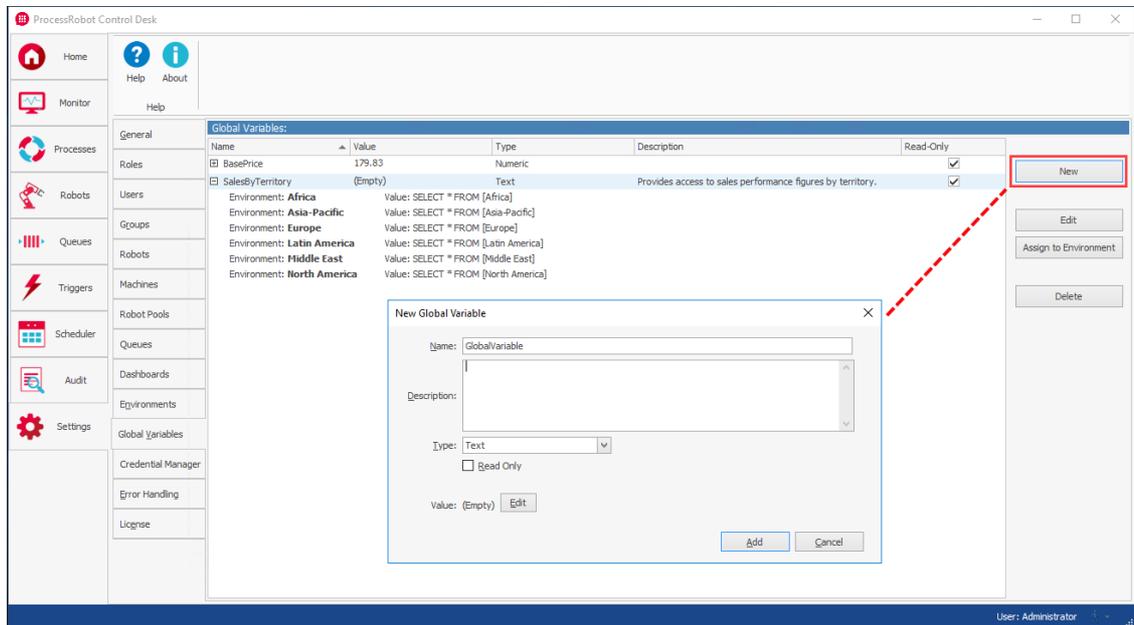
Description: A description about what the Global Variable is used for.

Type: The type of the Global Variable which can be set to be Text, Numeric, Boolean or List.

Read Only: Check this box if you wish the variable to be "Read Only". It makes sense that you will not be able to use the "Set Global Variable" action on this variable, if you do so.

Value-Edit: Press on the Edit button to write the value of the Variable.

Once you have specified all the fields above based on your desired settings, then you can click on "Add" and your Global variable will be created.

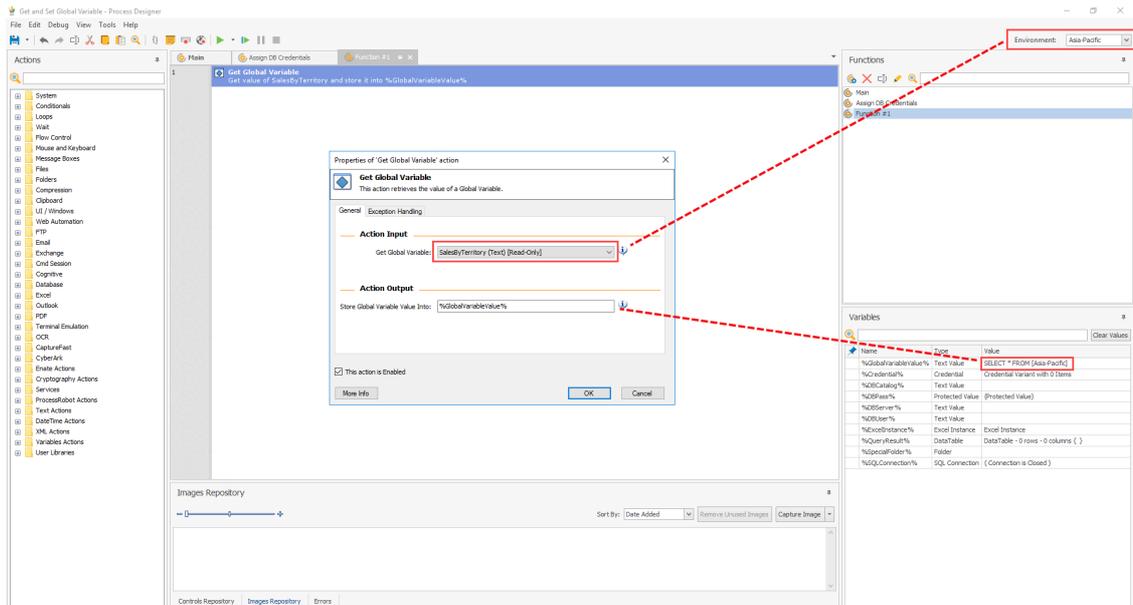


Add New Global Variable

The actions that can set and get a Global Variable's value are the "[Set Global Variable](#)"¹⁰⁰⁷ and "[Get Global Variable](#)"¹⁰⁰⁶" respectively.

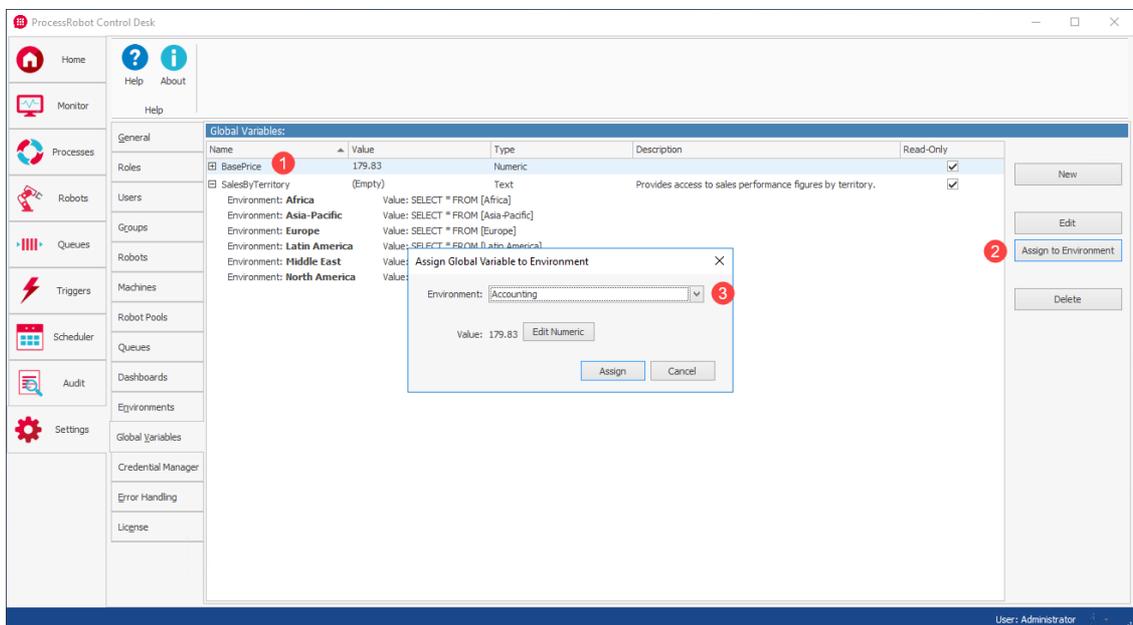
In every Global Variable you can assign an Environment from the Environments that you have previously set in the "[Settings > Environments](#)"²³⁵ in the Control Desk.

A Global Variable can be assigned to more than one Environments having a different value for each one. For example, database queries that correspond to a company's sales territories, like Asia-Pacific, can be saved in a Global Variable. The Global Variable will store a different value for each Environment. From the screenshot in the beginning of this topic, if a Process is run in the Asia-Pacific Environment then the "Get Global Variable" Action will retrieve the value stored for that Environment, "SELECT * FROM [Asia-Pacific]".



Assign a Global Variable to an Environment:

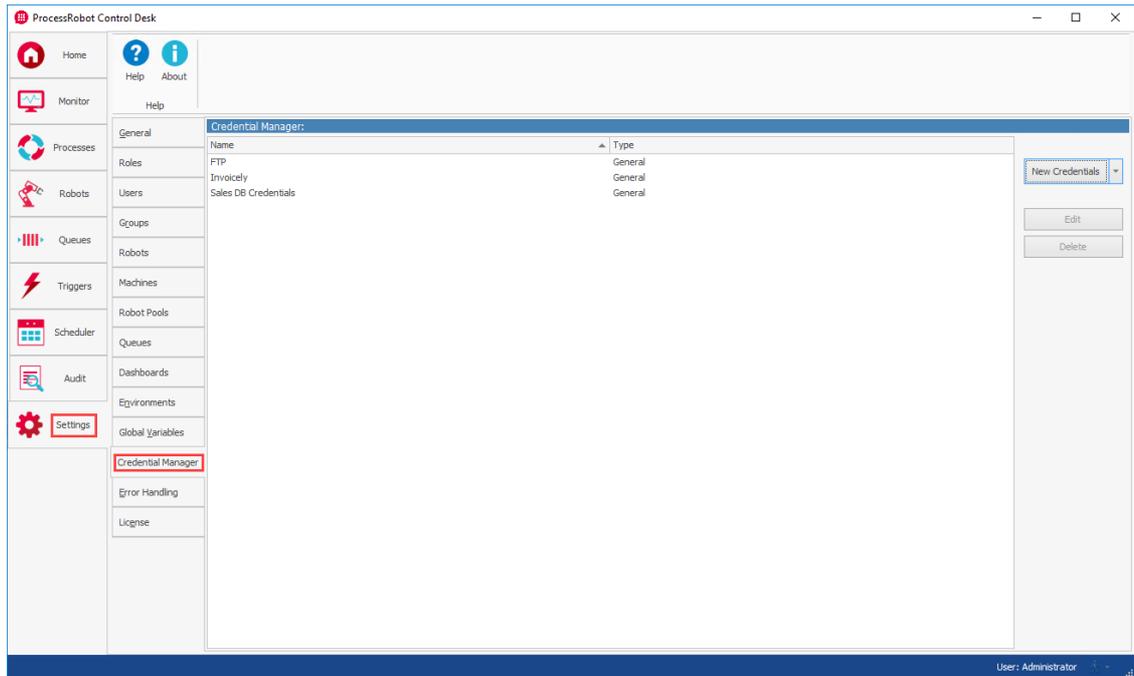
In order to assign a Global Variable to an Environment, first click on the Global Variable [1], then click on "Assign to Environment" [2]. Choose the desired Environment from the drop-down list of the resulting pop-up Window [3], click on the Edit button to set the value that the Variable will have for this Environment and then click the Assign button .



Assigning a Global Variable to an Environment

2.1.2.9, 12 Credential Manager

In the Credential Manager tab you may add credentials for desktop or web applications and encrypt them. Once you add the credentials that you want to use in your Robots, then the "[Get Credential](#)"¹⁰¹⁵ or "[Update Credential](#)"¹⁰²⁷" can be used to get or update a credential's value respectively.

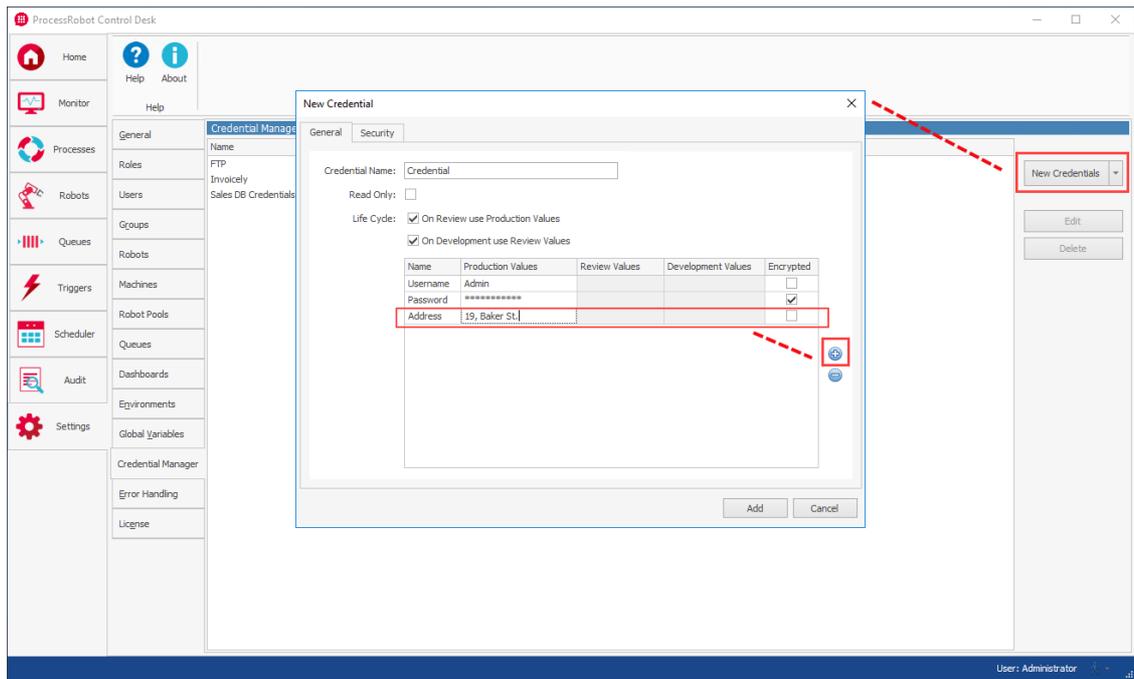


Control Desk Settings > Credential Manager

The Credential Manager Tab has three buttons:

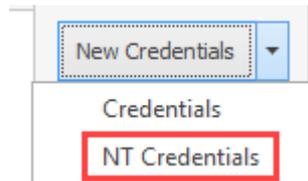
-New Credentials:

- Credentials:** Click this button to add a new general credential group. A general credential has two default properties, Username and Password, but the User can add additional properties. Similarly, the User can also remove properties. Both of these capabilities are achieved by clicking on the plus '+' or minus '-' buttons in the right hand area of the Credential window.



New Credential

- **NT Credentials:** Click 'NT Credentials' from the dropdown list under 'New Credentials' to add a new NT Credential group. An NT Credential group corresponds to the login credentials of an Active Directory user. NT Credentials can be used for the Solobot [Auto Login](#) feature, in case the 'Provide Password from Credentials' is selected.



An NT Credentials group has three properties: Username, Password and Domain. Unlike general credentials, none of these can be removed, nor can additional properties be added.

New NT Credential

General Security

NT Credential Name:

Read Only:

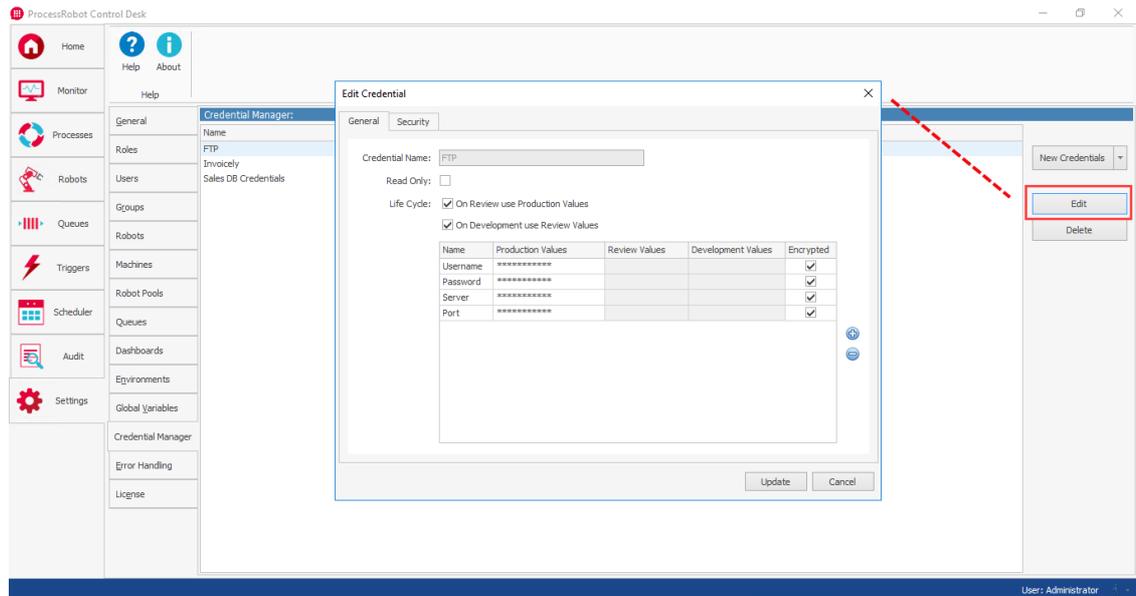
Life Cycle: On Review use Production Values
 On Development use Review Values

Name	Production Values	Review Values	Development Values	Encrypted
Username				<input type="checkbox"/>
Password				<input checked="" type="checkbox"/>
Domain				<input type="checkbox"/>

Add Cancel

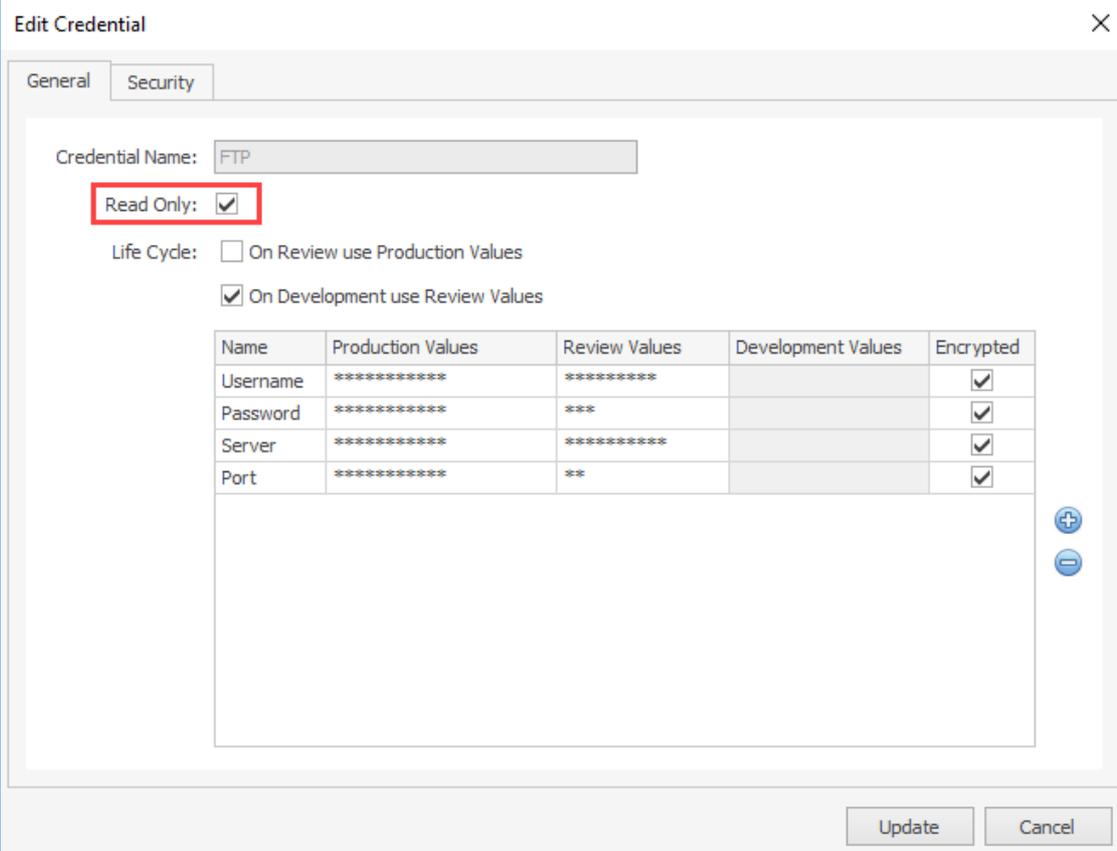
New NT Credential

-Edit: Edit an existing credential group and edit the existing credentials or add new ones.



Edit Credentials

Credentials can be designated as Read-only using the relevant checkbox. Doing so will prohibit the Credential from being updated through the [Update Credential](#) ¹⁰²¹ Action.



Edit Credential

General Security

Credential Name: FTP

Read Only:

Life Cycle: On Review use Production Values
 On Development use Review Values

Name	Production Values	Review Values	Development Values	Encrypted
Username	*****	*****		<input checked="" type="checkbox"/>
Password	*****	***		<input checked="" type="checkbox"/>
Server	*****	*****		<input checked="" type="checkbox"/>
Port	*****	**		<input checked="" type="checkbox"/>

Update Cancel

Credentials can also be assigned different values based on the Lifecycle stage of the Process which retrieves them. When both the 'On Review use Production Values' and 'On Development use Review Values' checkboxes are checked, only one set of values can be entered in the Credential.

General Security

Credential Name:

Read Only:

Life Cycle: On Review use Production Values
 On Development use Review Values

Name	Production Values	Review Values	Development Values	Encrypted
Username	*****			<input checked="" type="checkbox"/>
Password	*****			<input checked="" type="checkbox"/>
Server	*****			<input checked="" type="checkbox"/>
Port	*****			<input checked="" type="checkbox"/>

Update Cancel

Unchecking the 'On Review use Production Values' checkbox, for example, will allow the User to enter a different set of values for the same Credentials group to be used only when the Process is in Review. The same can be done for the Production stage as well.

Edit Credential [X]

General Security

Credential Name:

Read Only:

Life Cycle: On Review use Production Values
 On Development use Review Values

Name	Production Values	Review Values	Development Values	Encrypted
Username	*****	*****		<input checked="" type="checkbox"/>
Password	*****	***		<input checked="" type="checkbox"/>
Server	*****	*****		<input checked="" type="checkbox"/>
Port	*****	**		<input checked="" type="checkbox"/>

[+]
[-]

Update Cancel

Another noteworthy feature of the Credential Manager is that if you uncheck the Hidden box for a Credential Property that were up until recently hidden, it is erased. That way even the [Administrator](#)^[193] of the Control Desk cannot take advantage of their position and steal Credentials that might belong to someone else like the CEO or CFO of your organization.

New Credential [X]

General Security

Object: 'Credential'

Roles:

[Empty List Box]

[Add] [Delete]

Permissions:

Permission	Allow	Deny
[Empty]		

[Add] [Cancel]

Credentials Security Tab

In the Security Tab, permissions can be assigned to specific credentials, to allow or restrict access to certain ProcessRobot User Roles.

-Delete: Delete an existing group of credentials, just by clicking on it to highlight it and press on the Delete button.

2.1.2.9,13 Error Handling

Behavior Tab

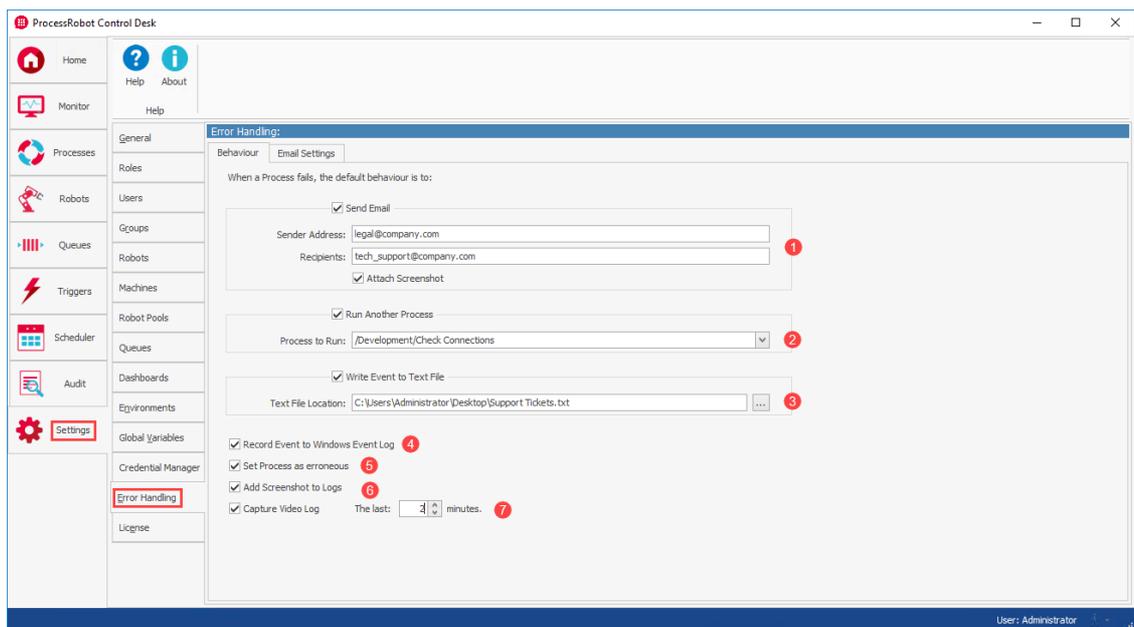
If a Process fails to execute successfully, you can specify what will happen here in the Options > Error Handling Tab. You can choose none, one, some, or ALL of the following:

1. **Send an email**, possibly with multiple recipients (separated by semi-colons), after the details set in the Error Handling > Email Settings Tab.

Please note that here you have the option to capture and attach a Screenshot of your entire screen the moment of failure if you tick the **Attach Screenshot** check box. This option will **not** Add a Screenshot to Logs. For this to happen you need to check [6] Add Screenshot to Logs.

2. **Run another Process** from your Processes database.

3. **Write Event to text file** whose path you can specify: The Event will be written at the end, so you can keep a running log of any Process failures.
4. **Record to the Windows Applications Event Log.** You can view the Windows Event Logs through Control Panel -> Administrative Tools -> Event Viewer on your workstation.
5. You can choose to **set** the failing **Process as Erroneous**. Setting a Process as **Erroneous** means that any Triggers or Schedules using this Process will not fire until a [new version](#)^[455] of it emerge from Review in Production replacing the one that caused failure.
6. **Add Screenshot to Logs** will do exactly that: produce a screenshot at failure that will be available for viewing and saving through the [Audit Tab](#)^[179] (last column at the right without title).
7. **Capture Video Log** of the error that occurs on a Process running on a Solobot machine so as to access the recorded video through the [Audit Tab](#)^[179] and reduce the troubleshooting time to minimum.

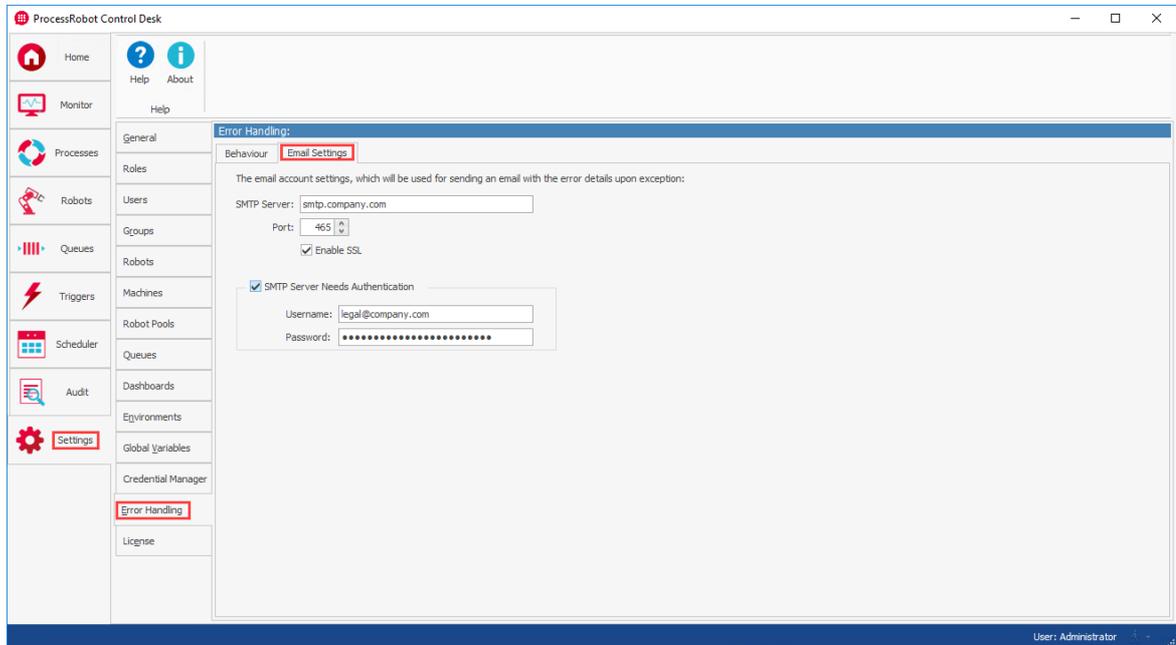


Control Desk Settings > Error Handling Tab

In any case, the execution of any Process along the result of the execution will always be recorded in the [Audit Tab](#).

Important! The settings and details you will enter here will be used ONLY by the [Global Error Handling](#)^[248] that defines the default behavior of your Control Desk when a Process (any Process) fails. This is the [4th level of Error-Exception handling](#)^[510] on Process Robot.

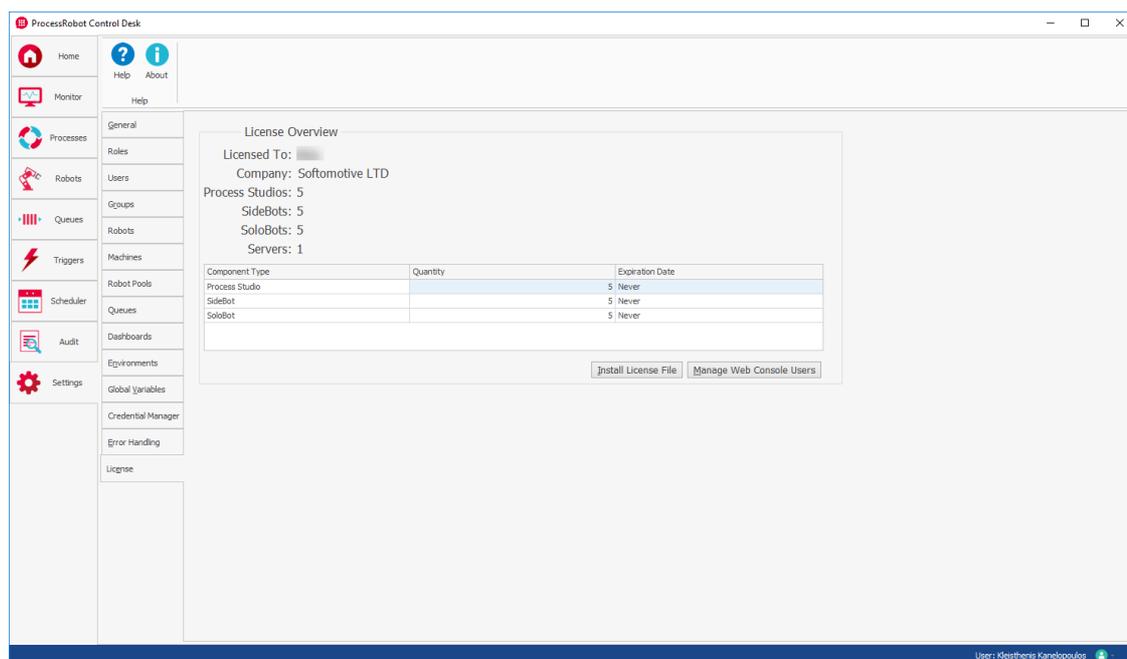
If you specify to send an email on the unsuccessful execution of any Process as default behavior on [Settings > Error Handling](#)^[248], you will need to fill in all settings and relative details here like SMTP Server, Port, Enable SSL, Username and Password in order for Process Robot to be able to send an e-mail:



Error Handling > Email Settings Tab

2.1.2.9,14 License

In the License Tab you can view your License info. It shows all the details of your registered licenses.



Control Desk Settings > License

Enter your License key to your ProcessRobot Console:

By clicking on the "Install License File" button, file selection window will pop-up prompting you to select the license file with extension ".lic". Select the license file and press "Open".

Note that the license file can also be directly passed at "C:\ProgramData\Softomotive\ProcessRobot\Licenses" on the Server machine where ProcessRobot is installed.

Managing Web Console Users

Press "Manage Web Console Users" to specify which ProcessRobot Users may have access to the Web Console. **Important:** only ProcessRobot Users that have been added with the Active Directory type may be added as Web Console Users.

2.2 Control Desk Web Console

The Control Desk Web Console is nothing more than a simplified, web version of the [Process Robot Control Desk](#)¹²⁰.

If you have any inquiries on how the Control Desk works, you should have a look on the Control Desk Topic of this Help File.

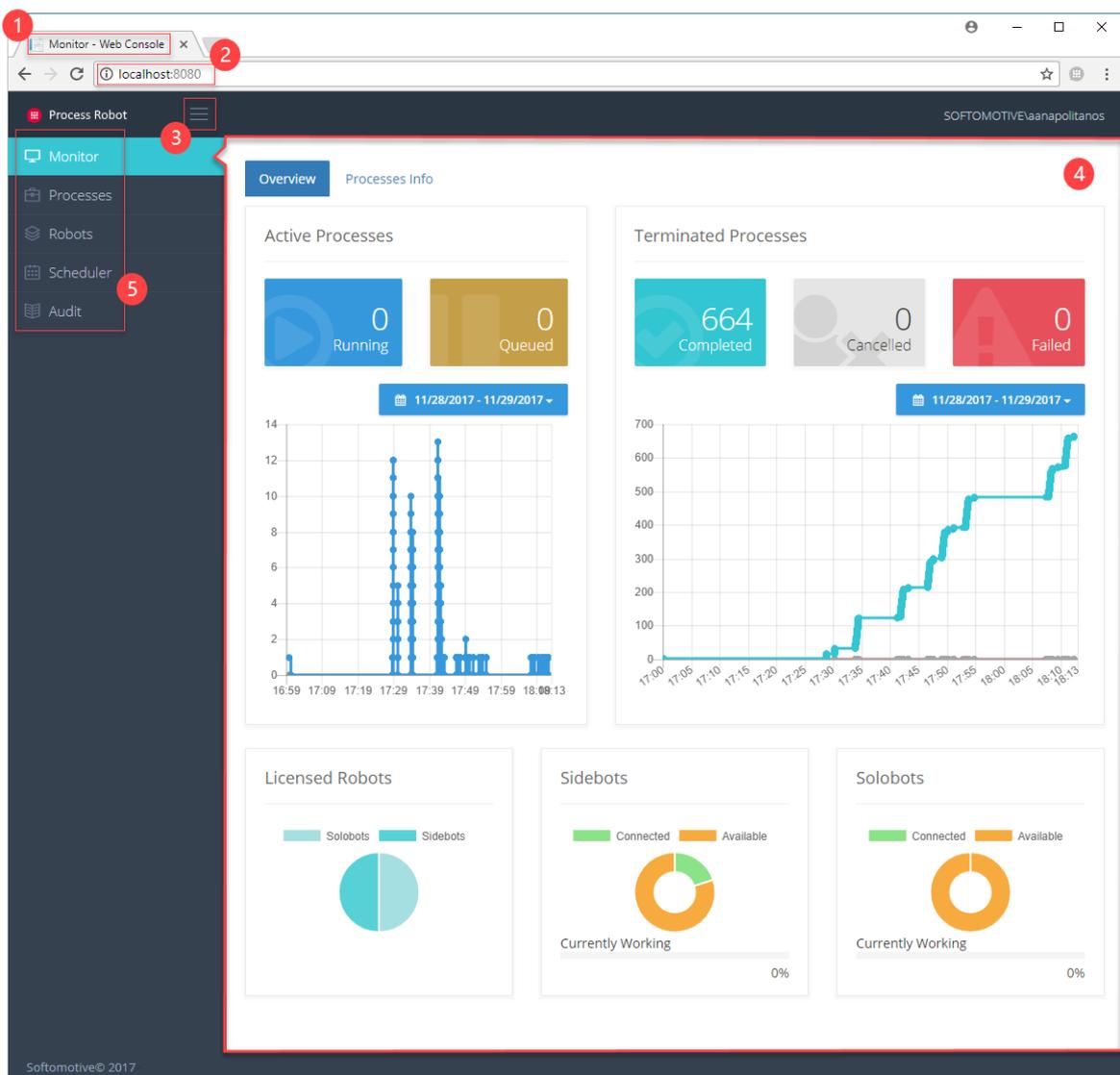
The purpose of the present article is to quickly and accurately inform *readers who are already familiar with the Control Desk* on what they can actually do through the Web Console.

Having that in mind, please feel free and very welcome to browse through the Topics of this article.

Important: only ProcessRobot Users that have been added with the Active Directory Type can be assigned as Web Console Users

2.2.1 Structure of Web Console

Let us observe the structure of the Control Desk Web Console as this appears on Internet Explorer 11.6:



1

Web Page Title:

Depending on the Web console tab that is currently highlighted, Monitor, Processes, etc, the title of the browser tab will be "Highlighted Tab - Web Console". In case of the screenshot above is "Monitor - Web Console".

2

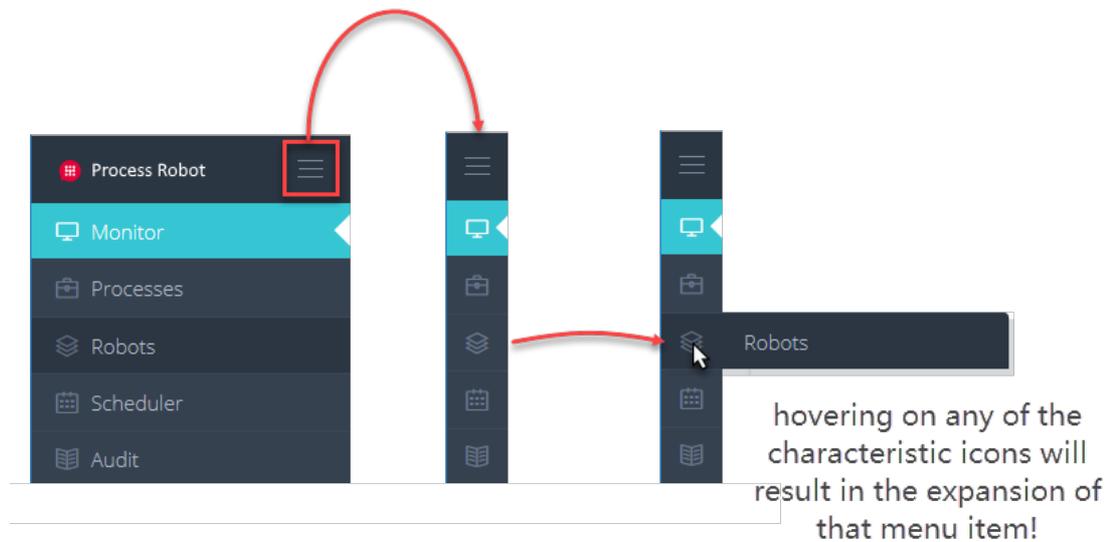
Accessing your Web Console:

You can access your Web Console via an http request to your computer. After the installation of the "ProcessRobotWebConsoleSetup" executable, just open a browser and type in the address bar "localhost" and a port that is not currently used. The Web version of the Control Desk will appear.

3

The Navicon:

This trigram has become synonymous of the ability to hide or show a menu and this is exactly what it does on the PR Web Console. Clicking on it will expand or collapse (partially) the Vertical Tab menu that allows you to navigate from one Tab to the other, giving you the ability to control the often valuable real estate of your monitor screen.



4

Screen Tab:

This is the viewport that allows you to interact with the Tab of your choice.

5

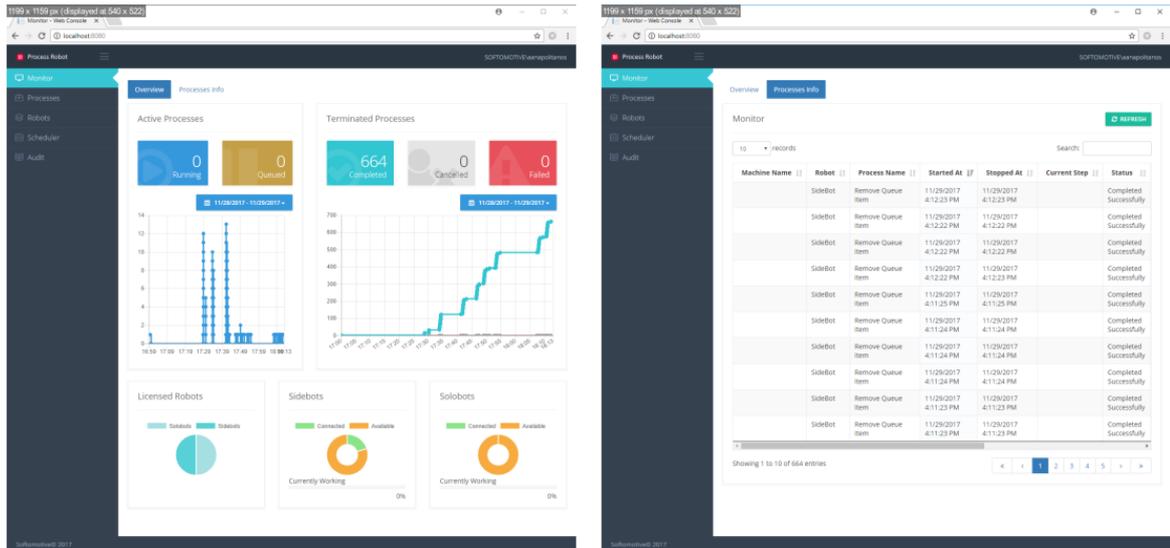
The Vertical Tab Menu:

This Tab Menu is the backbone of the Web Console, allowing users to navigate from one Tab to the other. It consists of the "[Monitor](#)^[254]", "[Processes](#)^[257]", "[Robots](#)^[260]", "[Scheduler](#)^[262]" and "[Audit](#)^[268]" tabs explained to you thoroughly in the following topics.

2.2.1.1 Monitor

The Monitor Tab on the Web Console consists of two screens accessible through a horizontal tab menu on the top left:

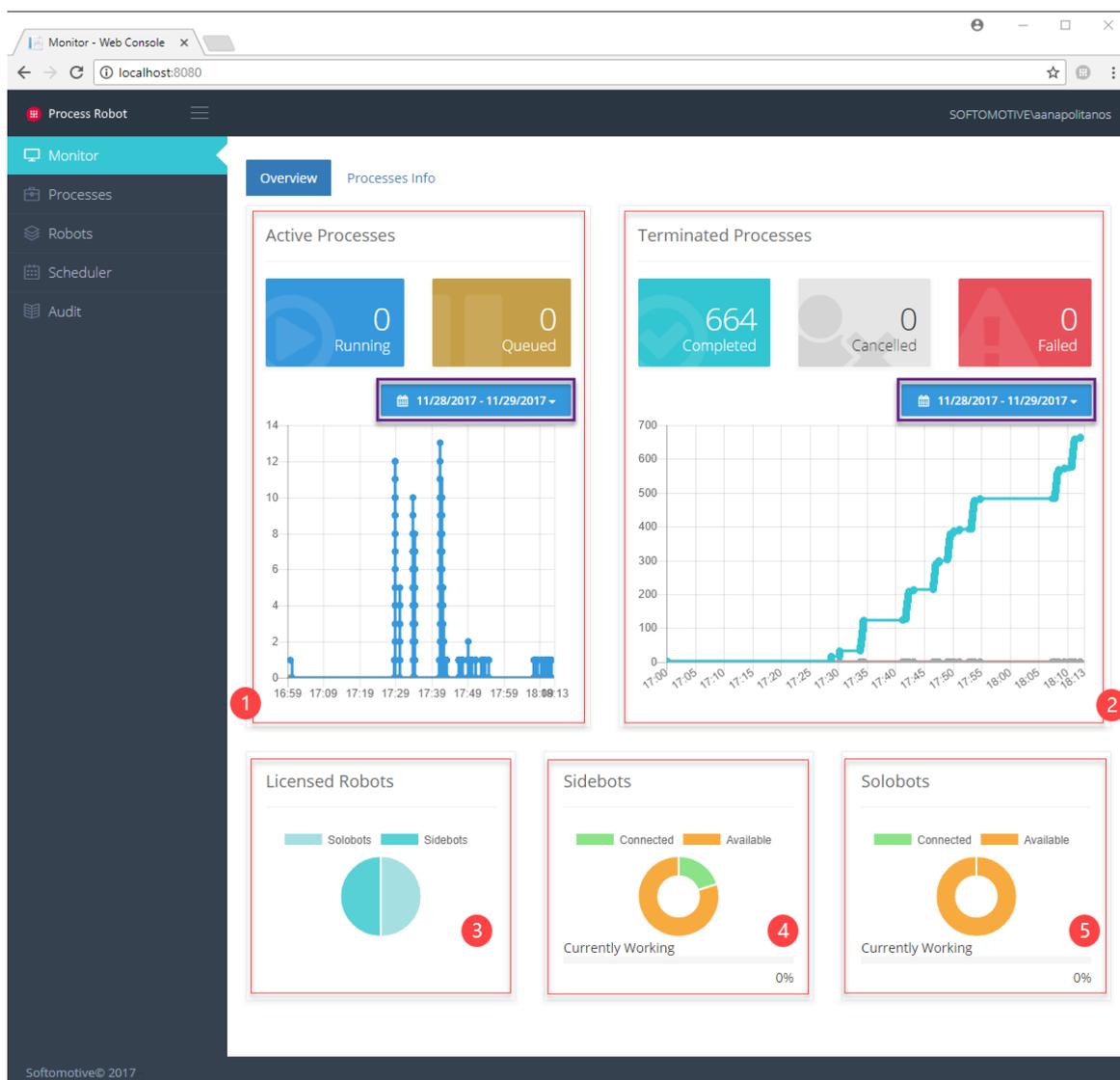
- a) The Overview
- and
- b) The Processes Info



The Overview and Processes Info Tabs

a) The Overview

The Overview Screen allows you to check the number of active or [queued](#)^[138] processes [1] as well as the number and status of all terminated processes [2] in the length of time that is specified at the Time Range Setter [3]:



The Cumulative Flow Diagrams ([1], [2]) are being produced instantly for the time range you are interested in, helping you to understand how can you divide the workload carried from your Process Robot resources over time.

Under these two panes, we found the License Utilization [3] and Currently Working panes [4], [5] that inform you quickly and accurately through their interactive graphs about the number of Robots you are able to deploy in order to satisfy your automation needs.

b) The Processes Info

Processes Info provides detailed information about Processes that are currently running or have run successfully during the last two hours:

The screenshot displays the 'Monitor' section of the Process Robot web console. The interface includes a sidebar with navigation options: Monitor, Processes, Robots, Scheduler, and Audit. The main content area is titled 'Monitor' and features a 'Processes Info' tab. A 'REFRESH' button is located in the top right corner of the table area. Below the title, there is a dropdown menu set to '10 records' and a search input field. The table below lists process records with the following columns: Machine Name, Robot, Process Name, Started At, Stopped At, Current Step, and Status. All records show a status of 'Completed Successfully'.

Machine Name	Robot	Process Name	Started At	Stopped At	Current Step	Status
	SideBot	Remove Queue Item	11/29/2017 4:12:23 PM	11/29/2017 4:12:23 PM		Completed Successfully
	SideBot	Remove Queue Item	11/29/2017 4:12:22 PM	11/29/2017 4:12:22 PM		Completed Successfully
	SideBot	Remove Queue Item	11/29/2017 4:12:22 PM	11/29/2017 4:12:22 PM		Completed Successfully
	SideBot	Remove Queue Item	11/29/2017 4:12:22 PM	11/29/2017 4:12:23 PM		Completed Successfully
	SideBot	Remove Queue Item	11/29/2017 4:11:25 PM	11/29/2017 4:11:25 PM		Completed Successfully
	SideBot	Remove Queue Item	11/29/2017 4:11:24 PM	11/29/2017 4:11:24 PM		Completed Successfully
	SideBot	Remove Queue Item	11/29/2017 4:11:24 PM	11/29/2017 4:11:24 PM		Completed Successfully
	SideBot	Remove Queue Item	11/29/2017 4:11:24 PM	11/29/2017 4:11:24 PM		Completed Successfully
	SideBot	Remove Queue Item	11/29/2017 4:11:23 PM	11/29/2017 4:11:23 PM		Completed Successfully
	SideBot	Remove Queue Item	11/29/2017 4:11:23 PM	11/29/2017 4:11:23 PM		Completed Successfully

Showing 1 to 10 of 664 entries

2.2.1.2 Processes

The Processes Tab on the Web Console consists of a single screen, comprised by a treeview that contains all Processes in Production [1], a standard pagination and search facility that is being used throughout the web console [2] and a table displaying the processes that belong on the respective highlighted folder of the treeview [3]:

The screenshot shows the Process Robot console interface. On the left is a sidebar with navigation options: Monitor, Processes (selected), Robots, Scheduler, and Audit. The main area is titled 'Processes' and contains a treeview on the left and a table on the right. The treeview shows a hierarchy: My Processes > Examples > 2 - Intermediate (highlighted). The table displays 10 records of processes, each with columns for Process, Enabled, Triggers, Erroneous, and Actions. A search bar and pagination controls are also visible.

Process	Enabled	Triggers	Erroneous	Actions
01 - Paste to Text File	True	None		▶ START
02 - Count Lines of Text File	True	None		▶ START
03 - Sort Lines of a Text File	True	None		▶ START
04 - Create New Folder with DateTime	True	None		▶ START
05 - Rename Multiple Files	True	None		▶ START
06 - Concatenate Text Files	True	None		▶ START
07 - Days of your Life	True	None		▶ START
08 - Display Notifications while Running	True	None		▶ START
09 - Find and Delete Empty Files	True	None		▶ START
10 - Delete All Text Files in a Folder Hierarchy	True	None		▶ START

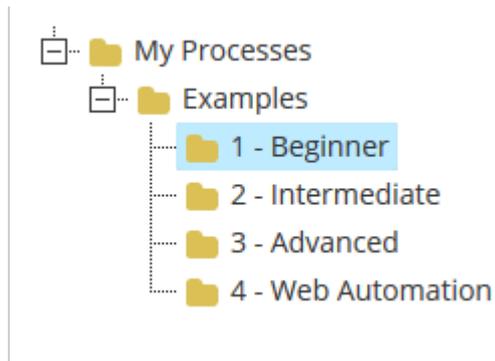
What is important for you to understand, is that through the Console you can only work with processes that are already in production. The Web Console gives you no Lifecycle options.

If you are unsure what Lifecycle or Versioning are (or how they work), we strongly encourage you to have a look on the topics under the [Version Control](#) article of this help guide.

1

Treeview:

The treeview is important in the sense that whatever folder is highlighted there, its contents will be "projected" on table [3]. Yet, its workings have nothing special to elaborate on:



It works as any other treeview with the exception that it is going as deep as *folder level* .

2

Pagination & Search:

For an extensive analysis on how to use the Pagination, Search and Refresh Control Elements, please read the following article:

[Standard Control Elements](#) ^[271] on the Web Console.

3

The Processes Table:

The Processes Table allows you to see whether a Process is enabled or not, what kind (if any) Triggers & Schedules are using a specific process and whether a process has been declared **Erroneous** or not.

Processes

Process	Enabled	Triggers	Erroneous	Actions
Feeder	True	1		▶ START
Input	True	None		▶ START
Output	True	None		▶ START
runtimeError	True	None	Erroneous	▶ START

Showing 1 to 4 of 4 entries

What is important to have in mind here are the following observations:

- a) The Sorting button on the right of every Column Title in the Main Table is sorting the data by placing that column's rows in standard numerical or lexicographical order. You can click the button twice: once to sort in ascending order (smaller to largest value) and twice to sort the data in descending order (larger to smallest value).
- b) It is important to know that from the Triggers Column you can see *both Triggers & Schedules*. You can easily recognize what is assigned where by the respective icon used to mark  Schedules and  Triggers. The number on the left of the icons is denoting the number of Schedules and/or Triggers in which the process is being used.
- c) You can Run a Process by pressing the Start button on the Actions Column.
- d) You can check whether or not a process has been marked as Erroneous in the Erroneous Column. Two things to note here are the following:

Since all of these Processes are already in Production, these Errors we are talking about are RunTime Errors and not Design Time errors. If you have difficulties telling the difference from one kind to the other, you might find it interesting to read this article >[The Errors pane](#)^[294].

A process will be marked as erroneous only if you tell the program to set it as Erroneous on Run time failure through ProcessRobot's Error Handling options (either for the specific process via [Process Properties](#)^[128] or for all processes via the [Error Handling](#)^[248] tab on Control Desk's Settings).

2.2.1.3 Robots

The Robots Tab on the Web Console consists of the standard pagination, search and refresh control elements [1, 2] and the main Table in which you can see all Robots that you have added [3].

Robots

10 records

SEARCH REFRESH

Connected	Type	Name	Register At	Assigned To	Machine Name	Group	Environment	Total Processes
	Sidebot	MySideBot	11/14/2017 11:41:25 AM		TECH-WRITER-1		Accounting	0 running, 0 queued
	Sidebot	SideBot-2			N/A		Accounting	0 running, 0 queued
	Sidebot	SideBot-3			N/A	IT	Accounting	0 running, 0 queued

Showing 1 to 3 of 3 entries

1

Controlling Pagination:

For an extensive analysis on how to use the Pagination, Search and Refresh Control Elements, please read the following article:

[Standard Control Elements](#) ²⁷¹ on the Web Console.

2

Search and Refresh:

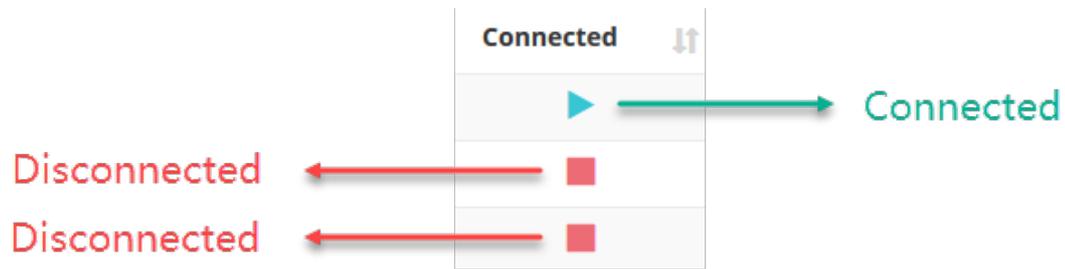
For an extensive analysis on how to use the Pagination, Search and Refresh Control Elements, please read the following article:

[Standard Control Elements](#) ²⁷¹ on the Web Console.

3

Robots Table:

If Robots are connected there will be a green arrow in the "Connected" column, otherwise there will be a red square, indicative of the Robot's disconnection in the Connected column.



You will also get information about its Type (Sidebot or Solobot), Name, when was Registered at, who was Assigned to, the Name of the Machine that the Robot is running on, the Group and Environment it belongs to and the Processes that are running or they are being Queued on this specific Robot.

What you need to understand here, is that all this information is there to make your life easier:

By having the ability to know for example in which Machine, Group and Environment a Robot belongs to, it is easy to locate it both in the active directory (i.e. the physical infrastructure your PR is running on) and the PR arrangement (Group, Environment) in which it belongs to.

2.2.1.4 Scheduler

The Web Console Scheduler allows you to Create Schedules [1] or Recurring Schedules [2] as well as Editing [3] or Deleting [4] existing ones. It also gives you the option to enable or disable any schedule by ticking the relevant checkbox (under the "Enabled" titled column) [5] or even sort your data based on ascending or descending order of a particular attribute [6]. The Web Console offers extensive pagination options [7], giving you the capability to manage your Schedules efficiently. Last but not least, we have the Refresh button and Search-Box facilities [8] we have seen previously before in this guide but we'll see here again since they provide indispensable services to your Enterprise Grade RPA.

The screenshot shows the 'Scheduler' interface in Softomotive. It features a table with columns: Type, Description, Execution Target, Process Name, Process In Production, Enabled, and Actions. The table contains three entries. Callouts 1-8 point to: 1. '+ NEW SCHEDULE' button; 2. '+ NEW RECURRING SCHEDULE' button; 3. 'EDIT' button in the Actions column; 4. 'DELETE' button in the Actions column; 5. 'Enabled' checkbox; 6. 'Description' column header; 7. 'records' dropdown; 8. 'REFRESH' button.

Type	Description	Execution Target	Process Name	Process In Production	Enabled	Actions
Schedule	Will sent a predefined number of e-mails to clients.	My_SideBot	Feeder	True	<input checked="" type="checkbox"/>	EDIT DELETE
Schedule	This will fire daily at noon-1min up until 14-11-2017!	SideBot	8 - Rotate Excel Data	True	<input checked="" type="checkbox"/>	EDIT DELETE
Recurring Schedule	Fires every 10 minute(s)	SideBot	1 - Get Metadata of Web Page	True	<input type="checkbox"/>	EDIT DELETE

Showing 1 to 3 of 3 entries

1

New Schedule:

Pressing the +New Schedule Button will launch a Window that will give you the opportunity to set all Properties related to your Schedule:

Properties
Process
Execution

Custom Description:

Schedule Type: Daily v

At the Following Time(s): 11:59 AM ✕

+ ADD TIME

If Process is late: Don't start Process v

Schedule Expires on

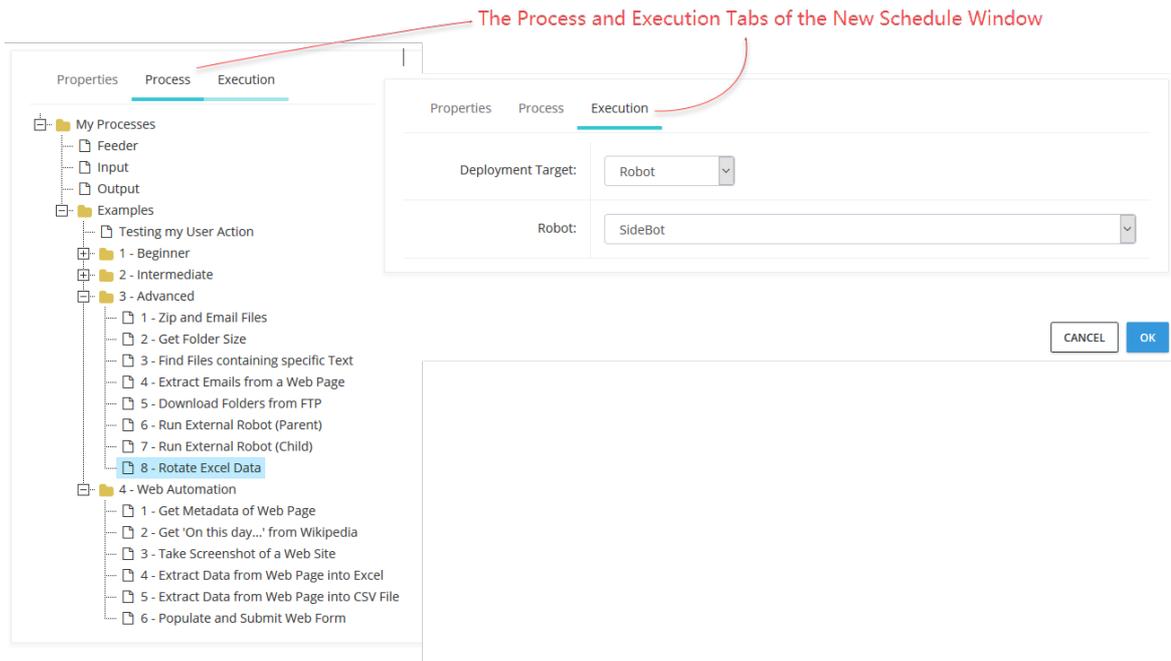
14-11-2017 📅

CANCEL
OK

These Properties will be later displayed on the Scheduler's Main Table, along with the Processes' Names we want to run and the Robot or Robots we want these selected Processes to run on (Execution Targets).

Type	Description	Execution Target	Process Name	Process In Production	Enabled	Actions
Schedule	Will sent a predefined number of e-mails to clients.	My_SideBot	Feeder	True	<input checked="" type="checkbox"/>	EDIT DELETE ✕
Schedule	This will fire daily at noon-1min up until 14-11-2017!	SideBot	8 - Rotate Excel Data	True	<input checked="" type="checkbox"/>	EDIT DELETE ✕
Recurring Schedule	Fires every 10 minute(s)	SideBot	1 - Get Metadata of Web Page	True	<input checked="" type="checkbox"/>	EDIT DELETE ✕

The Execution Target and Process Name are being given by the respective (namely Process & Execution) two other Tabs of the same Window:

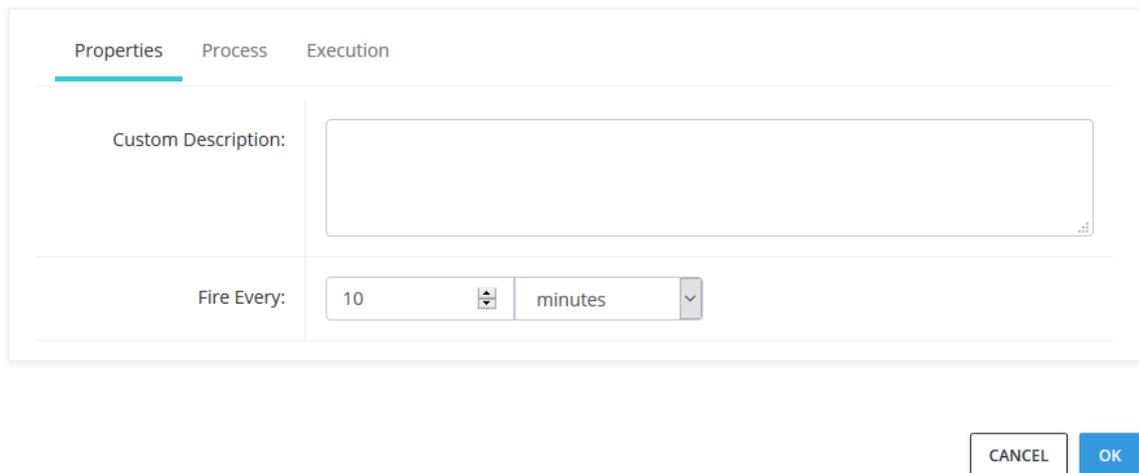


2

New Recurring Schedule:

Pressing the +New Recurring Schedule button will launch another Window in order to define the Recurring Schedule's Properties.

This Window differs only in the Properties tab from the New Schedule Window:



3

Edit Schedule:

This button will launch the exact same Windows the +New Schedule [1] or +New Recurring Schedule buttons [2] would launch for the editing of a Schedule or Recurring Schedule respectively.

4

Delete Schedule:

This button will launch a 'Warning from possible loss of data window' that will give you a second chance to reconsider the consequences of your action before you are able to delete the Schedule of your choice.

The screenshot shows the Process Robot Scheduler interface. A modal dialog box titled "Delete Schedule?" is displayed in the foreground, with "CANCEL" and "DELETE" buttons. The background shows a table of scheduled tasks with columns for Type, Description, Execution Target, Process Name, Process In Production, Enabled, and Actions. The table contains three entries:

Type	Description	Execution Target	Process Name	Process In Production	Enabled	Actions
Schedule	Will sent a predefined number of e-mails to clients.	My_SideBot	2 - Open a Folder	True	<input checked="" type="checkbox"/>	EDIT DELETE
Schedule	This will fire daily at noon-1min up until 14-11-2017!	SideBot	8 - Rotate Excel Data	True	<input checked="" type="checkbox"/>	EDIT DELETE
Recurring Schedule	Fires every 10 minute(s)	SideBot	1 - Get Metadata of Web Page	True	<input type="checkbox"/>	EDIT DELETE

Showing 1 to 3 of 3 entries

5

Enable/Disable Checkbox:

By Disabling a Schedule it will no longer fire the Process until you Enable it again.

This Process won't run anymore as Scheduled					Enabled
Recurring Schedule	Fires every 10 minute(s)	SideBot	1 - Get Metadata of Web Page	True	<input checked="" type="checkbox"/>
					<input checked="" type="checkbox"/>
					<input type="checkbox"/>

6

Sorting your Schedules:

The Sorting button on the right of every Column Title in the Main Table is sorting the data by placing that column's rows in standard numerical or lexicographical order. You can click the button twice: once to sort in ascending order (smaller to largest value) and twice to sort the data in descending order (larger to smallest value).

Let's have a quick look on how this would work on the Process Name column:

Random Ascending Descending

Random	Ascending	Descending
<div style="border: 1px solid #ccc; padding: 5px;"> <p>Process Name </p> <p>2 - Open a Folder</p> <p>8 - Rotate Excel Data</p> <p>1 - Get Metadata of Web Page</p> </div>	<div style="border: 1px solid #ccc; padding: 5px;"> <p>Process Name </p> <p>1 - Get Metadata of Web Page</p> <p>2 - Open a Folder</p> <p>8 - Rotate Excel Data</p> </div>	<div style="border: 1px solid #ccc; padding: 5px;"> <p>Process Name </p> <p>8 - Rotate Excel Data</p> <p>2 - Open a Folder</p> <p>1 - Get Metadata of Web Page</p> </div>

7

For an extensive analysis on how to use the Pagination, Search and Refresh Control Elements, please read the following article:

[Standard Control Elements](#)^[27] on the Web Console.

8

For an extensive analysis on how to use the Pagination, Search and Refresh Control Elements, please read the following article:

[Standard Control Elements](#)^[27] on the Web Console.

2.2.1.5 Audit

The Audit tab allows you to inspect your Logs. Although it is offering you limited filtering options compared to the Control Desk Audit Tab, you can still isolate and sort the entries you are most interested in through the combined action of the select Date Range facility [1], the Search box [2] and standard Web Console pagination facility [3]:

Audit

NOVEMBER 7, 2017 - NOVEMBER 13, 2017 REFRESH

10 records Search:

Type	Date & Time	Message	Path	Process Name	Robot Name	User Name	Event Id
i	11/13/2017 1:48:15 PM	Trigger Created					3101
i	11/13/2017 1:47:40 PM	Trigger Created					3101
x	11/13/2017 1:09:02 PM	Process 'runtimeError' completed with result: Error in Function 'Main' at Action#2 'Extract Data from Web Page'	/My Processes	runtimeError	SideBot		1003
i	11/13/2017 1:08:55 PM	Process 'runtimeError' started	/My Processes	runtimeError	SideBot		1001
i	11/13/2017 1:08:10 PM	Process 'runtimeError' Properties Changed	/My Processes	runtimeError			1104
x	11/13/2017 1:06:36 PM	Process 'runtimeError' completed with result: Error in Function 'Main' at Action#2 'Extract Data from Web Page'	/My Processes	runtimeError	SideBot		1003
i	11/13/2017 1:06:29 PM	Process 'runtimeError' started	/My Processes	runtimeError	SideBot		1001
x	11/13/2017 12:59:28 PM	Process 'runtimeError' completed with result: Error in Function 'Main' at Action#2 'Extract Data from Web Page'	/My Processes	runtimeError	SideBot		1003
i	11/13/2017 12:59:21 PM	Process 'runtimeError' started	/My Processes	runtimeError	SideBot		1001
x	11/13/2017 12:57:15 PM	Process 'runtimeError' completed with result: Error in Function 'Main' at Action#2 'Extract Data from Web Page'	/My Processes	runtimeError	SideBot		1003

Showing 1 to 10 of 181 entries

1

Select Date Range:

By clicking on the Select Date Range facility, a drop down menu appears, inviting you to enter a range of dates after a number of predefined (and one custom) range options:

Message	Path	Process Name	SideBot	Username	Count
Trigger Created					
Trigger Created					
Process 'runtimeError' completed with result: Error in Function 'Main' at Action#2 'Extract Data from Web Page'	/My Processes	runtimeError			
Process 'runtimeError' started	/My Processes	runtimeError			
Process 'runtimeError' Properties Changed	/My Processes	runtimeError			
Process 'runtimeError' completed with result: Error in Function 'Main' at Action#2 'Extract Data from Web Page'	/My Processes	runtimeError			
Process 'runtimeError' started	/My Processes	runtimeError			
Process 'runtimeError' completed with result: Error in Function 'Main' at Action#2 'Extract Data from Web Page'	/My Processes	runtimeError			
Process 'runtimeError' started	/My Processes	runtimeError	SideBot	Alexandros Anapolitanos	1001
Process 'runtimeError' completed with result: Error in Function 'Main' at Action#2 'Extract Data from Web Page'	/My Processes	runtimeError	SideBot		1003

NOVEMBER 7, 2017 - NOVEMBER 13, 2017 REFRESH

Today 11/07/2017

Yesterday

Last 7 Days

This Month

Last Month

Custom Range

APPLY
CANCEL

11/13/2017

Nov 2017

Su	Mo	Tu	We	Th	Fr	Sa
29	30	31	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	1	2
3	4	5	6	7	8	9

<< < 1 2 3 4 5 > >>

2

The Search box:

For an extensive analysis on how to use the Pagination, Search and Refresh Control Elements, please read the following article:

[Standard Control Elements](#) on the Web Console.



Controlling Pagination:

For an extensive analysis on how to use the Pagination, Search and Refresh Control Elements, please read the following article:

[Standard Control Elements^{\[271\]}](#) on the Web Console.

2.2.1.6 Standard Control Elements

The Web Console is using three standard control elements in order to assist users on their daily tasks:

- 1) The Pagination interface,
- 2) The Search box and
- 3) The Refresh Button.

- 1) Controlling Pagination:

You can control the way that your records/data are being displayed in the main Table of each Tab through the the Console's standard Pagination facility.

Let us examine how this interface works through an example using the Scheduler Tab.

In order to arrange your data in pages:

- a) Select one of the predefined options on how many Schedules you want to be displayed per page,
- b) Have a look at how many entries (Schedules) you currently have and
- c) Navigate your way through your data by either moving to the last or first page [c1] or simply the previous or next page [c2]

Scheduler

+ NEW SCHEDULE + NEW RECURRING SCHEDULE REFRESH

10 records

Search:

	Description	Execution Target	Process Name	Process In Production	Enabled	Actions
Schedule	Will sent a predefined number of e-mails to clients.	My_SideBot	1 - Run Calculator	True	<input checked="" type="checkbox"/>	EDIT DELETE
Schedule	This will fire daily at noon-1 min up until 14-11-2017!	SideBot	8 - Rotate Excel Data	True	<input checked="" type="checkbox"/>	EDIT DELETE
Recurring Schedule	Fires every 10 minute(s)	SideBot	1 - Get Metadata of Web Page	True	<input type="checkbox"/>	EDIT DELETE

Showing 1 to 3 of 3 entries

<< < 1 > >>

c1

c2

2) The Search box:

The Search box on the Web Console is filtering your data dynamically (on every key-stroke), applying the input value of your search to all possible data rows and data columns. For example observe how the Search box will filter the table below when we type "none" or "err" on it.

The example below is using data from the Processes Tab:

Search:

	Process	Enabled	Triggers	Erroneous	Actions
	Feeder	True	1		
	Input	True	1		
	log message	True	None		
	Output	True	None		
	runtimeError	True	None	Erroneous	

Showing 1 to 5 of 5 entries

« < 1 > »

Search:

	Process	Enabled	Triggers	Erroneous	Actions
	log message	True	None		
	Output	True	None		
	runtimeError	True	None	Erroneous	

Showing 1 to 3 of 3 entries (filtered from 5 total entries)

« < 1 > »

Search:

	Process	Enabled	Triggers	Erroneous	Actions
	runtimeError	True	None	Erroneous	

Showing 1 to 1 of 1 entries (filtered from 5 total entries)

« < 1 > »

3) The Refresh button:

There is one important advantage in using the Refresh button over the browser's default reload current page facility.

Hitting the Refresh button will allow you to refresh the information being displayed while keeping in memory your filtering (date range and sorting) options:

Audit

10 records

Search:

Type	Date & Time	Message	Path	Process Name	Robot Name	User Name	Event Id
⚠	11/14/2017 1:37:39 PM	User [redacted] @TECH-WRITER-11 With SID S-[redacted] 185 does not have SideBot assigned					2004
⚠	11/14/2017 1:51:28 PM	Server Warning: Cannot move Process. The process is probably open by [redacted]					2004
⚠	11/14/2017 2:22:11 PM	Server Warning: Cannot move Process. The process is probably open by [redacted]					2004
⚠	11/14/2017 2:57:47 PM	Server Warning: Cannot Checkout the Process. The process is probably open by [redacted]					2004
ℹ	11/14/2017 1:33:19 PM	ProcessRobot Server Started					2001

NOVEMBER 13, 2017 - NOVEMBER 14, 2017 REFRESH

These non-default sorting/filtering options will be reset if users decide to refresh the page via their browser instead of clicking REFRESH

The reader should be aware that the Refresh button in the Web Console is designed to replace the (F5) Refresh key that is available throughout the Desktop components of ProcessRobot.

3. Process Studio

3 Process Studio

3.1 Process Studio - Introduction

Process Studio is where the developer can build, debug and test the Processes which will later on be assigned to run on Robots (to Solobots or to Sidebots).

Here is where the "construction" takes place, using the Process Studio building blocks (aka actions) and where all the tasks performed by humans, are now becoming actual processes that will be automatically executed by Robots.

With Process Studio you can have a code-free automation development, leveraging the highly visual Process Designer. Process Studio offers advanced testing and debugging, leveraging what-if scenarios before and during deployment.

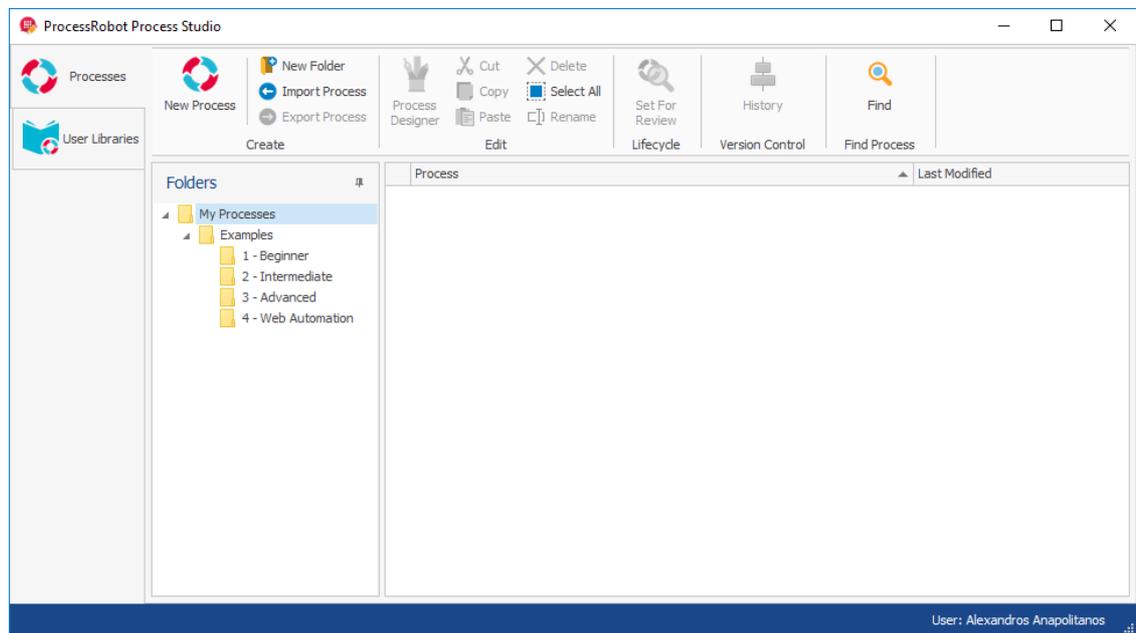
Process Studio provides you with robust multiprocessing automation, fully scalable and extensible while ensuring seamless integration across the enterprise.

The developer can build reusable [Functions](#)^[301] or [User Libraries](#)^[394] for ease of use and maintenance of complex Processes and can have instant access to help tips into the User Interface and to even the deepest level of technical detail.

ProcessRobot includes numerous powerful features that make it the ideal tool for automating your repetitive tasks. The most important features are:

- Visual Process Editor that lets you build automation Processes using Drag & Drop.
- 300+ predefined actions to combine for easy Process-building.
- Macro Recorder that lets you record user interaction to automate tasks easily.
- Web Recorder that lets you record your activities while you navigate the web and convert them into a Process.
- UI Automation technology that allows direct handling of the different controls within a window.
- Task Scheduler that lets you schedule and execute Processes while you are away.
- Triggers that let you monitor your system and respond to events (e.g., when a file is created or modified, when a HotKey is pressed, when system is idle and more).
- Integrated Debugger to inspect and debug your tasks while running.
- Support for Variables and Data Types.
- Autologin to unlock/login to a workstation upon running a Process.
- Mainframe automation, with Terminal Emulation.
- Advanced Flow Control capabilities, Conditionals, Loops, nested Loops and Functions.

- Advanced Error Handling to create Processes that compensate for and handle errors gracefully.
- OCR functionalities and Image Recognition technology that literally "sees the screen" to help you automate non-standard interfaces or applications over remote sessions.
- Control and Image Repository for storing all the accessible controls and Images of a Process and advanced editing.
- Real Time Process Status monitoring.
- Regular Expressions support for text parsing or for advanced data extraction
- Dynamic debugging with the ability to change variable values on the run.
- Find Process facility in order to find easily your Processes no matter how complex is your Folder structure.
- Email Automation.

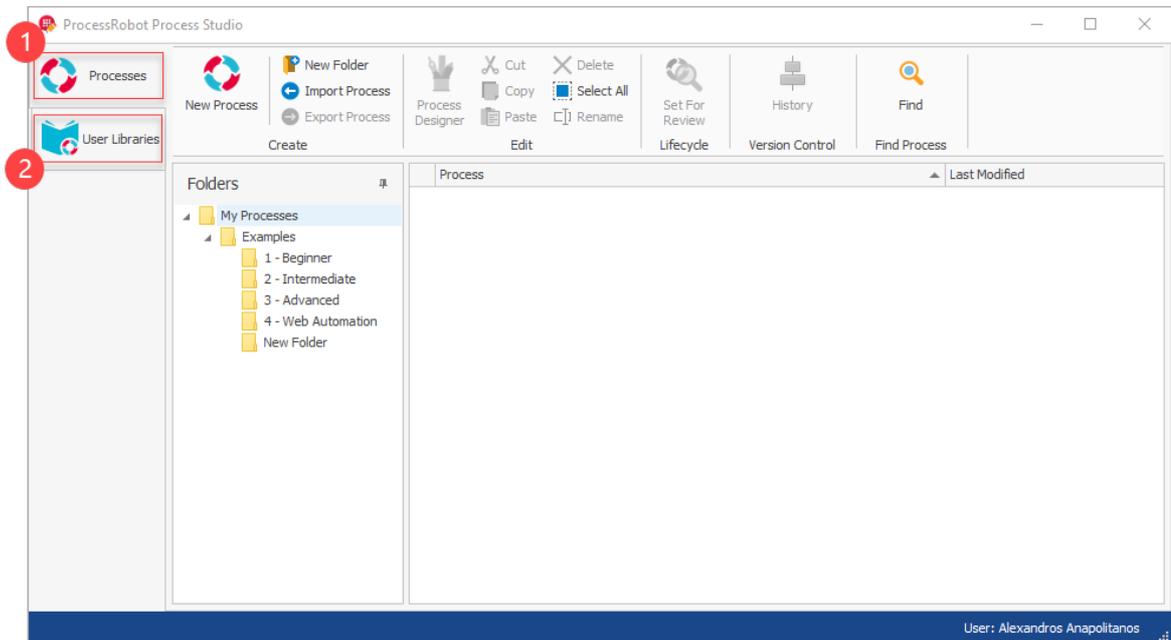


Process Studio

3.2 Process Studio

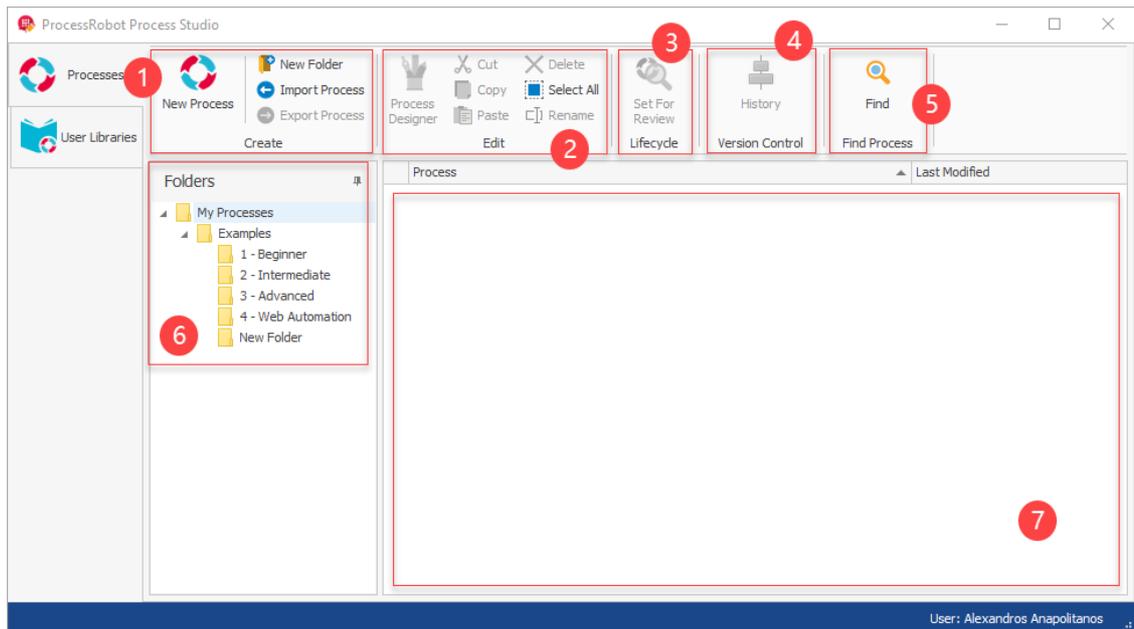
3.2.1 ProceStudio Window

Process Studio allows you to create and manage Processes [1] and [User Libraries](#) [394] [2] via the options in its main vertical tab menu:



If you want to create or manage your **User Libraries** please click on the appropriate tab and follow this [link](#)³⁹⁴ that describes in detail your options.

If on the other hand you want to create or manage your **Processes** please continue reading. The Console's Processes Tab allows you to:



1

Create:

In the Create group you can [create a New Process](#)^[281], [create a new folder](#)^[281] to save your Processes in, [Import/Export a Process](#)^[283] in/out of the console.

2

Edit:

In the Edit group you can Edit a Process in the Process Designer by clicking on the "Process Designer" option, Delete a Process, Select all Processes in the folder that you are currently in, Rename/Cut/Copy/Paste a Process.

3

Lifecycle:

The Process Lifecycle is explained in the [Processes](#)^[124] topic. A process is passed in the in the Process Studio for Development. Once development is done then the Process will be passed to the Review state and will be available in the Control Desk>Processes.

4

Version Control:

In this pane you are able to view the History of a Process. You can view information about the Process's versions and the stage of [Lifecycle](#) that each version is or has been in the past.

5

Find:

This facility allows you to find easily your Processes no matter how complex is your Folder structure.

6

Refresh:

Hitting the Refresh button will allow you to refresh the information being displayed.

7

The Processes' Folders List pane:

In this pane you are able to view your saved Processes' folders structure.

8

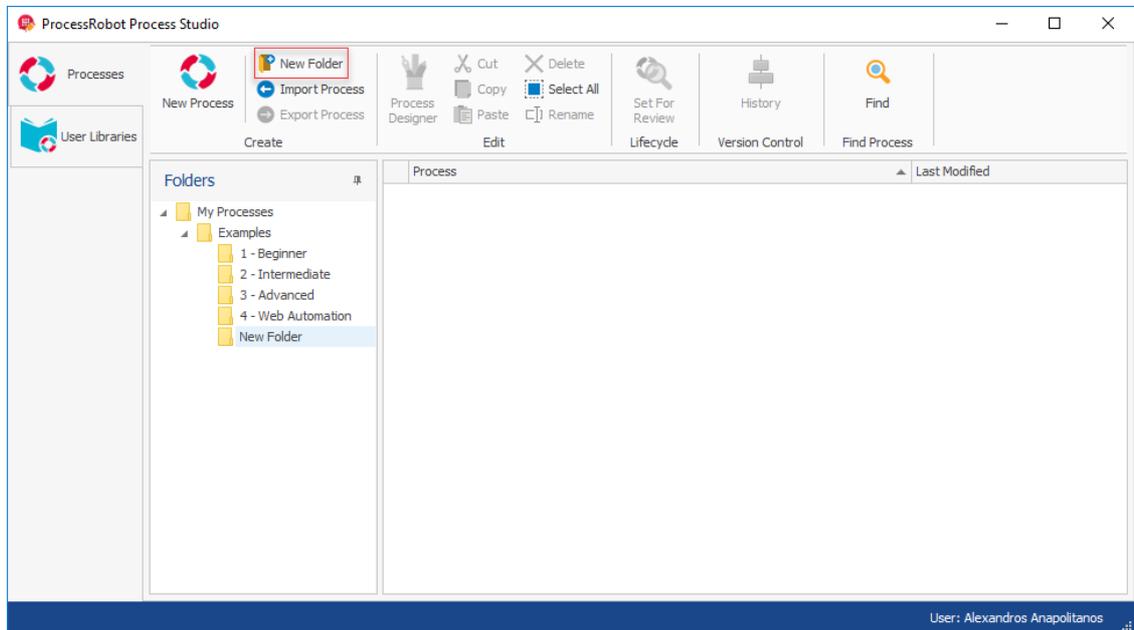
The Processes List pane:

In this pane you can see all the Processes that are saved in a specific folder. You can click on a folder in the Processes' Folder pane and all its Processes will be listed in the Processes list pane.

The Columns in the Processes List pane can be rearranged should you click and drag them according to your desirable display.

3.2.2 Create a new Folder

You can create a new folder by the "Create" menu group in the Processes tab after clicking "New Folder", or by right clicking on the folders pane and selecting "New Folder". This creates a new folder that is waiting for you to rename it.



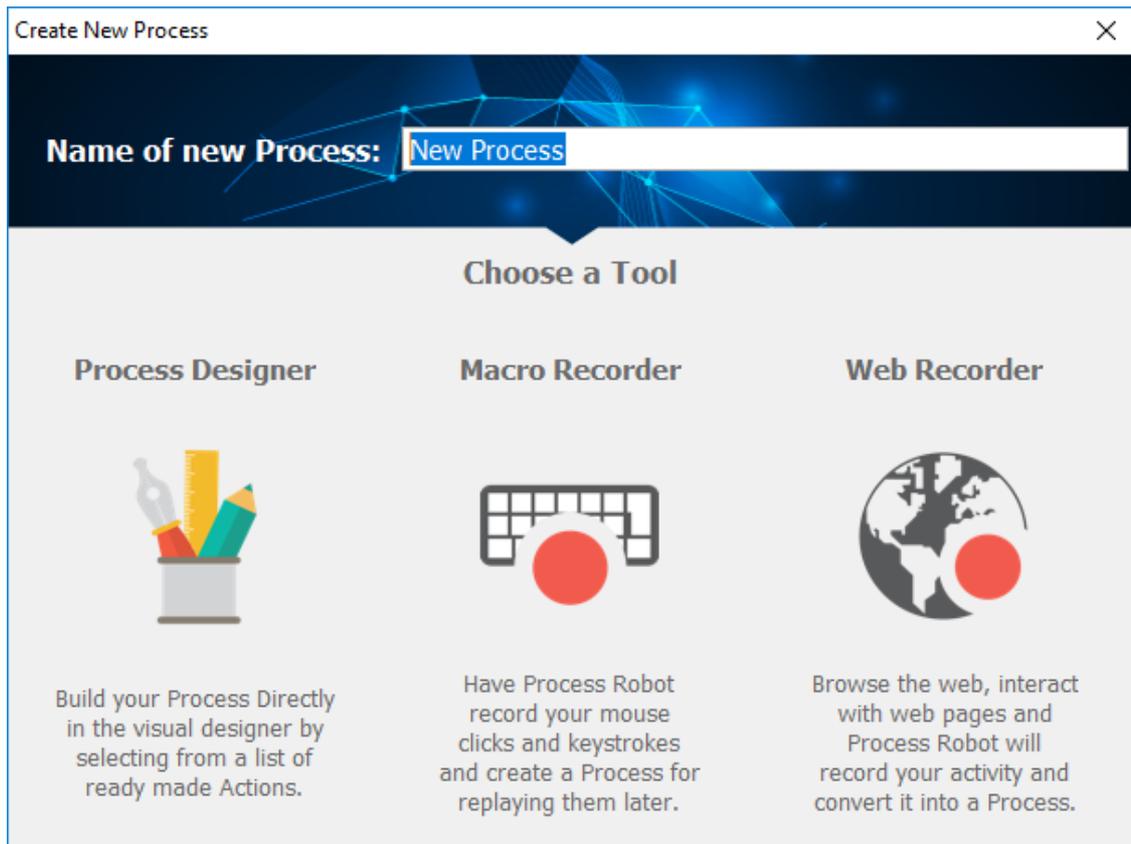
The new folder will be created as a subfolder, under the folder which is selected in the Folders List. It will always be a subfolder of whatever folder you have selected on the left screen. Therefore you cannot create another folder at the same level as 'My Process', as you cannot select a higher folder to place the new one in.

You can also drag and drop a folders into another folder in the Folders' list pane.

Of course, you cannot have two folders with the same name under the same folder.

3.2.3 Create a new Process

You can create a new Process by pressing Ctrl+N or by clicking on the New Process option on the create sub menu. This will open the "Create New Process" Dialog, prompting for a name for the new Process.



After setting the Process's name you have three options.

1. You can start with the Process Designer and start building your Process by combining actions
2. You can jump to the Macro Recorder to record your mouse clicks and keystrokes or use the smart recording, and then go back to the Process Designer
3. Finally you can open the Web Recorder and start navigating in a browser window where all your web activities will be recorded and converted into action when you finish the recording and return to the Process Designer.

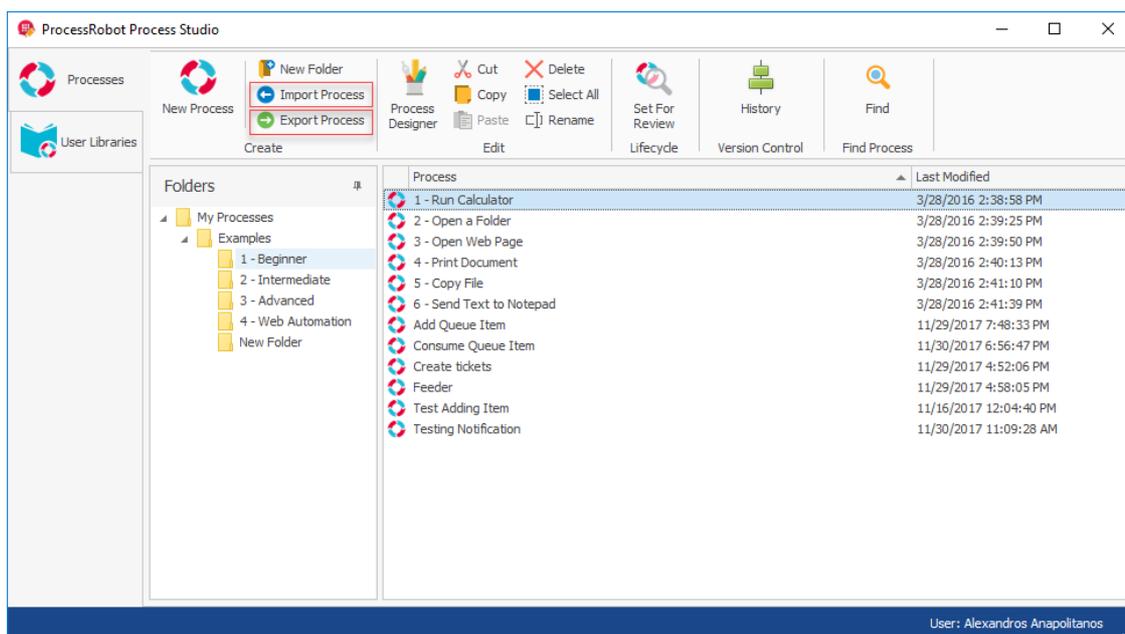
Note that you can always enter the Process Designer and access the Recording tools (Macro Recorder and Web Recorder) from within the Process Designer's window. The tools are available in the Process Designer's toolbar and under the "Tools" menu.

To re-edit a Process later, double-click on it, or select it and click on the "Process Designer" button  on the "Edit" sub menu group or right click on the Process and select Edit Process (Ctrl+E).

3.2.4 Import/Export a Process

ProcessRobot stores all Processes in your SQL Server Database. Therefore, there are no obvious Process files to copy or move. To copy a Process onto another computer that has ProcessRobot installed, export the Process - this will create a file with the extension .prp (process robot process).

This file may be moved to another machine with ProcessRobot and then be imported into its database. Now, the new computer will also have a copy of the Process. This is done through Import Process and Export Process options on the Create sub-menu group.



Import a Process:

To import a Process in your ProcessStudio, select the "Import Process" option.

Having done so, a dialog will appear and prompt you to select the Process's file you wish to import (*.prp)

Once the file has been selected and you have pressed the "Open" button the dialog window will close and within the ProcessStudio, in the Processes panel you shall see the Process you selected to import.

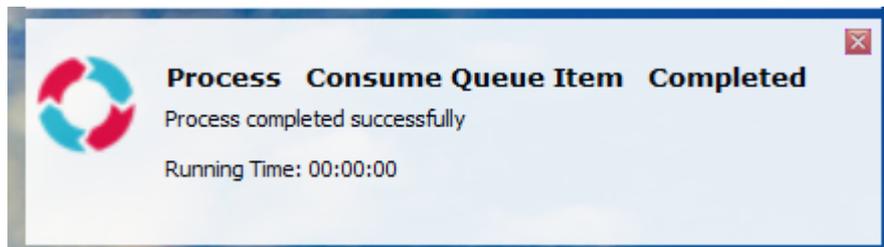
Export Process:

To export a Process, locate the Process you wish to export from the Processes List pane and select the "Export Process" option in the Create sub-menu group. Having done so, a dialog will appear and prompt you to select the folder where the Process should be exported to.

Once the folder has been selected and you press the "OK" button in the dialog, the dialog window will close and within the folder you selected a new file will be created named after the Process's name and with a *.prp extension.

3.2.5 The Notification Popup Window

The Notification Popup Window is the small window that opens in the bottom right corner of the screen while a Process is running (and for a short time thereafter) from the ProcessRobot Console, a Schedule or with a Trigger. It will not be displayed when you run a Process from within the Process Designer, as this is considered debugging, not running.



The Notification Window displays the status of the Process (Running or Completed), which Action is currently being executed, the total running time, and a small link which says "Stop Process" (and stops that instance of this Process only).

You can also display custom messages through the Notification Window, using the [Display Notification Action](#) ⁶³⁸ under Message Boxes. This is a great way to write messages to the user without pausing the Process for a standard Message Box. Enter the text (and/or Variable value) you want to display. As you go through the Process, you can change the message with further use of the Display Notification Action, and using an empty message will remove the message you had previously displayed.

Additional Notification Windows will pop-up above the previous one, so if you run several Processes simultaneously, the Windows will form a column.

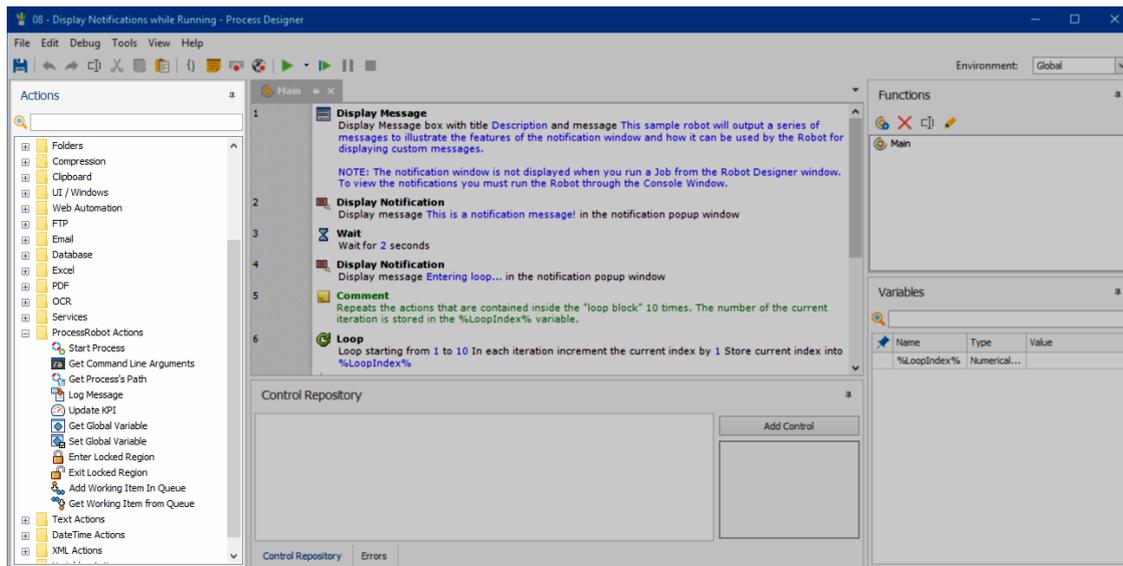
You can close any open Window at any time by clicking on the red (X) button on the top right corner of the Window. This will hide only the Notification Window, while the Process instance will continue to run. To stop the Process itself, click on the "Stop Process" link in the Notification Window.

3.3 Process Designer

Process Designer is the interactive visual interface that allows you to build powerful processes.

3.3.1 The Actions Pane

In the Actions pane you will find all the Actions that can be chosen to build your Process, organized in Actions groups in a folder structure.



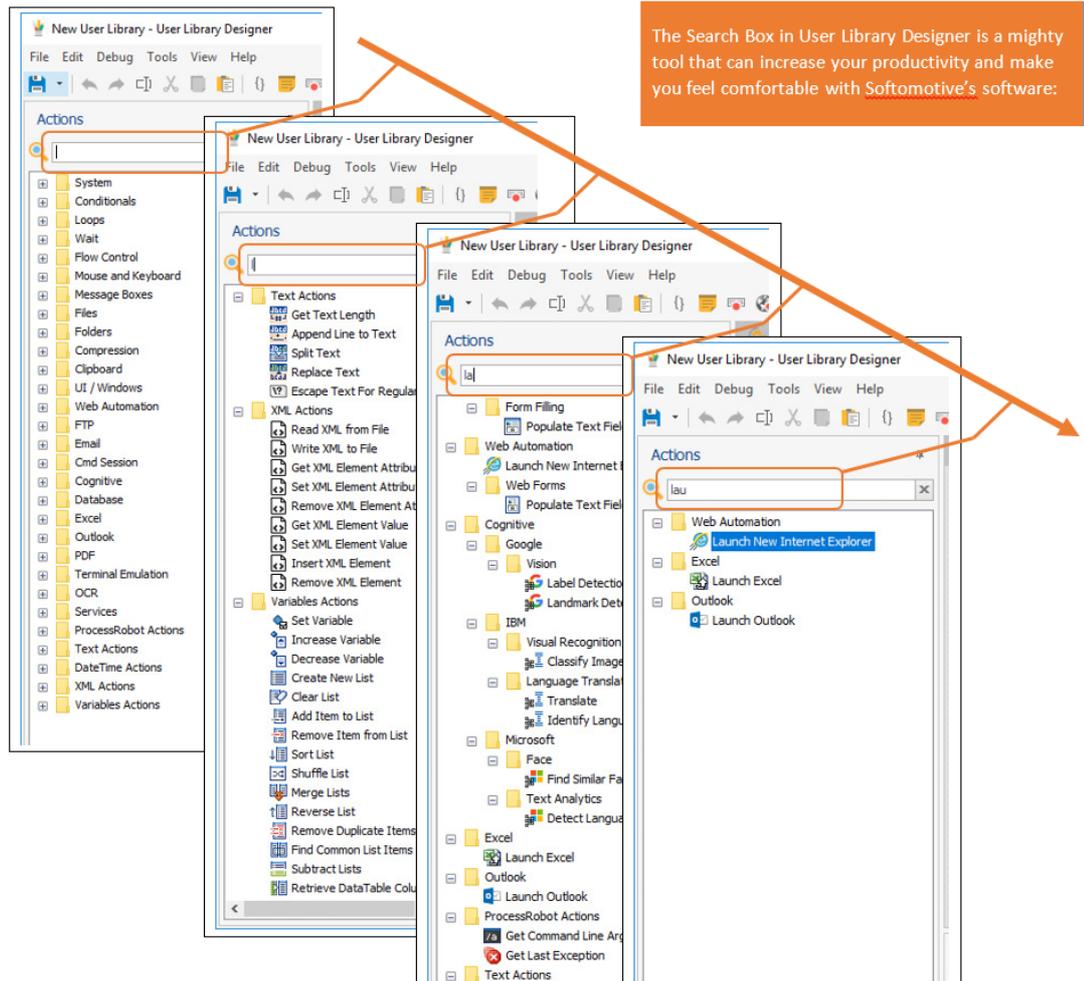
The Actions Pane of the Process Designer

You can either expand the group of actions in order to select the one you want, or you can search an action with its name, or even part of it, in the Search field at the top of the Actions pane.

You can drag an action to the Workspace or double-click on the Action to move it automatically there. Either way, this will open the Properties dialog box if the Action has properties.

The search box on the top of the pane can help you to quickly and conveniently locate an Action from the treeview Folder structure in case you remember (even part of) its name but you have difficulties recollecting its position in the Directory.

The text in this field is dynamically filtering the contents of the treeview as you type or delete characters in it, so in order to see the whole Folder structure again, you have not but to delete any characters you might have typed here.

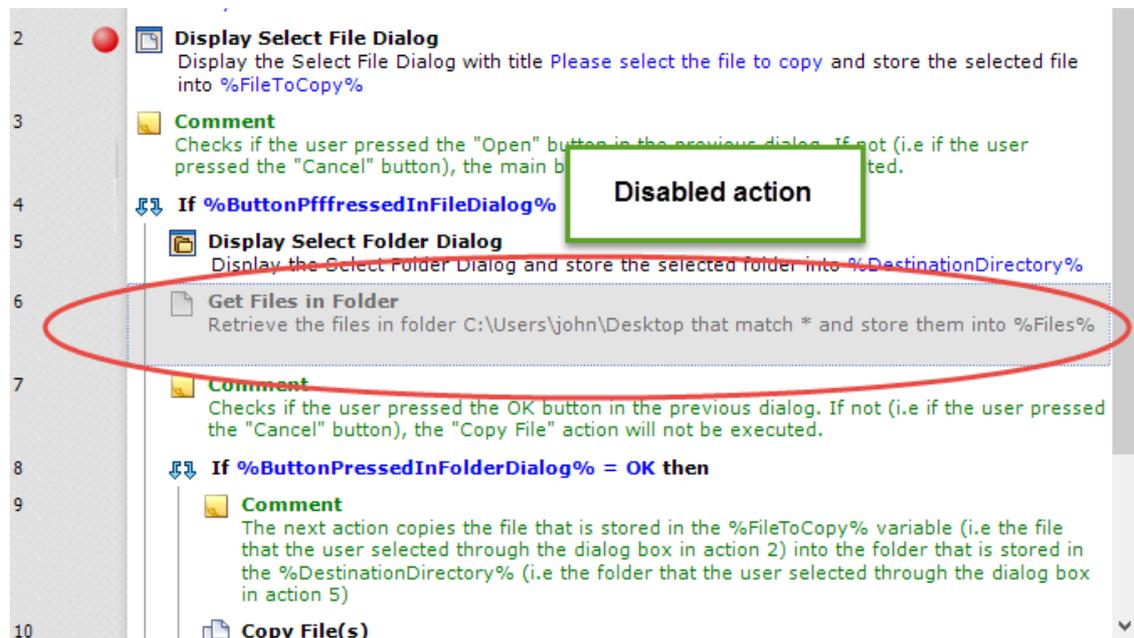


3.3.2 The Workspace

This pane shows the flow of the Process, displaying Actions in the order that they will be executed. Numbers are displayed in the gray column before each Action, as well as Icons to show errors, and breakpoints. If you are running (debugging) the Process, the current Action you are on will also be highlighted here.

Breakpoints : You can add a breakpoint by clicking to the left gray space of an Action. A red dot will appear and the Process will pause on that step, until you click to continue. This can also be used while running a Process from the Process Designer to pause while you look at some part of the Process. Note, that when you run a Process outside the Process Designer on your Robots, then any breakpoint will not have any effect to the Process whatsoever. It is solely for running the Process through the Robot Designer.

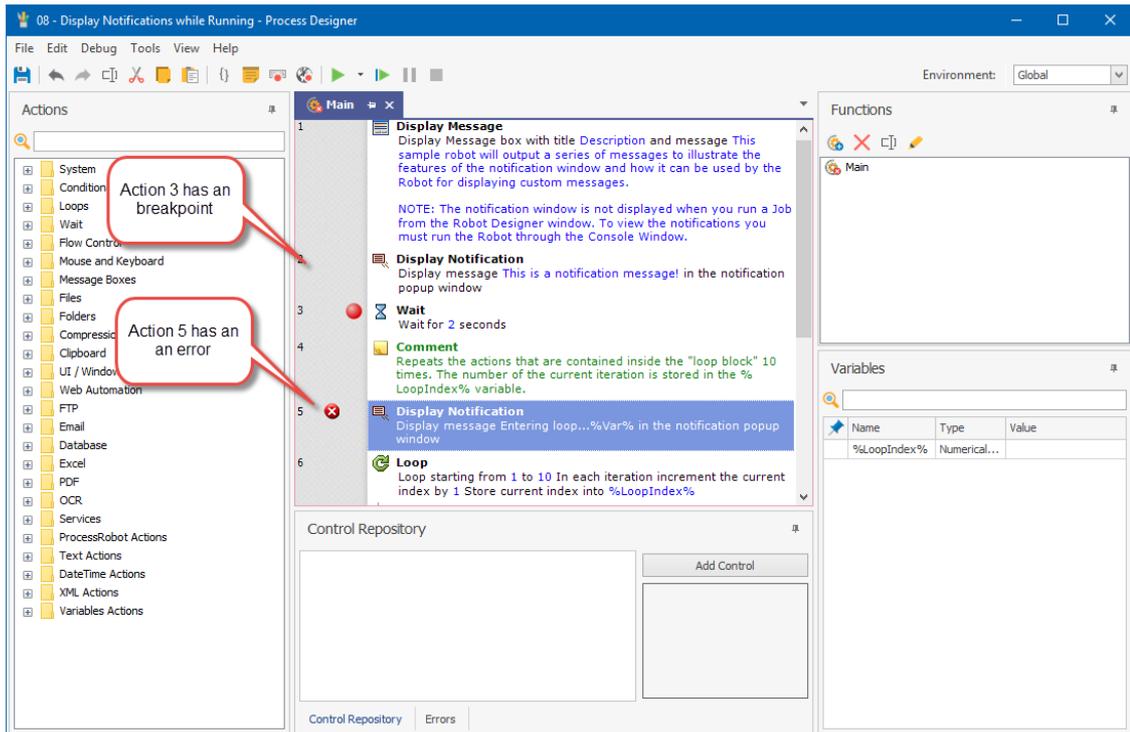
Disabled Actions: Actions can be disabled, so that they are ignored by the Process while executing, but they still exist. This is useful for trying different versions of a Process without constantly deleting and re-writing Actions. Disabled Actions become gray to differentiate them from enabled Actions.



A grayed out Disabled action

In the Workspace, Actions, their Icons and their properties are displayed. A Process is described in text in the Workspace and you can understand a Process just by reading through the Actions' properties in the Workspace. To change the display of the icons and text for this Process, click on [Tools > Process Designer Options](#)³¹².

To manipulate the Actions within a Process, treat them like any other icon setup in Windows. Click on an action to select it, click on Actions with the Ctrl or Shift keys to select multiples, right-click for the Action menu, and use any of the regular HotKeys to Copy, Cut, and Paste. To change the order of Actions, Drag and Drop or right-click and choose to Move Action Up or Move Action Down.

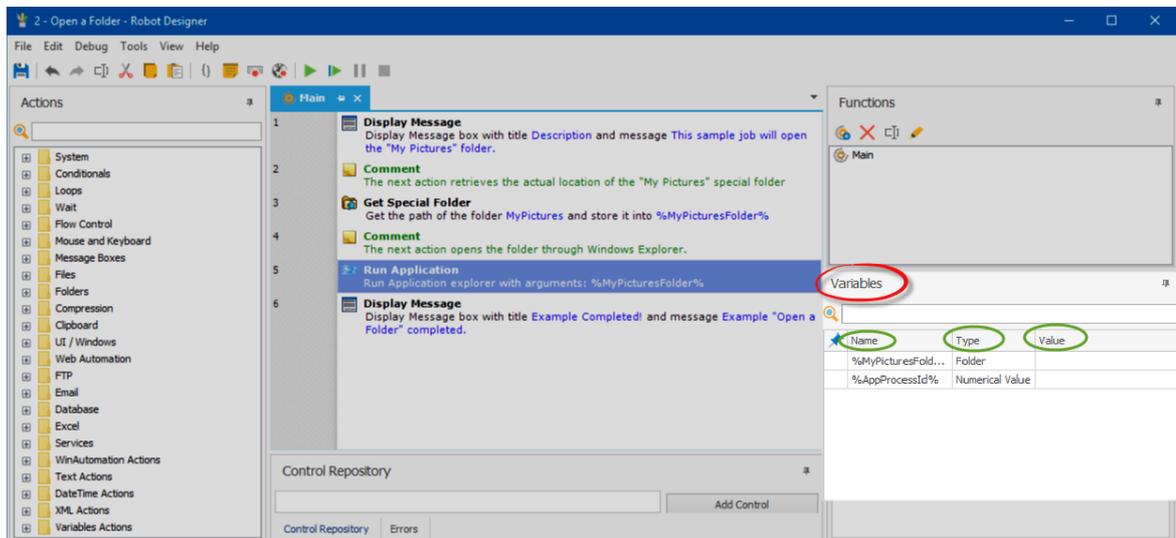


The Workspace Pane of the Process Designer Window

3.3.3 The Variables Pane

This pane is located by default next to the Workspace under the Functions pane. Variables are defined through Actions or through the Variables Manager.

There are four columns: Pin, Name, Type, Value.



The Variables Pane in the Process Designer Window

- A Pin option is there to select one or more variables and pin them on top of the Variables pane. This way they will be displayed on the top of the variables list that you have declared within your Process.
- Name, is the name that was assigned to the Variable in an Action or in the Variables Manager (Trigger Variables are never displayed here)
- In column three, the Type of Variable is displayed. Variable types can be: Text, Numerical, DateTime, File, Folder, Data Row, Data Table, Excel Instance, FTP File, or FTP Folder, as well as a list of the above, among others.
If you use the same Variable name as an output for two or more Actions that create different Types of Variables, the Variable type will change during the execution of a Process, so ProcessRobot cannot determine a constant value, and will show '<Unknown>'. If both Actions determine an output Variable with the same type of value, it will show that type, as there is no ambiguity. For this and other reasons, we recommend that you use different Variable names for each Action output - unless, of course, you want to overwrite the previous information.
- Value, is the value set when a Variable is created through the Variables Manager Window and only appear if you created the Variable there, with an initial value. Otherwise, the Value will be filled for each variable during the execution of the Process.

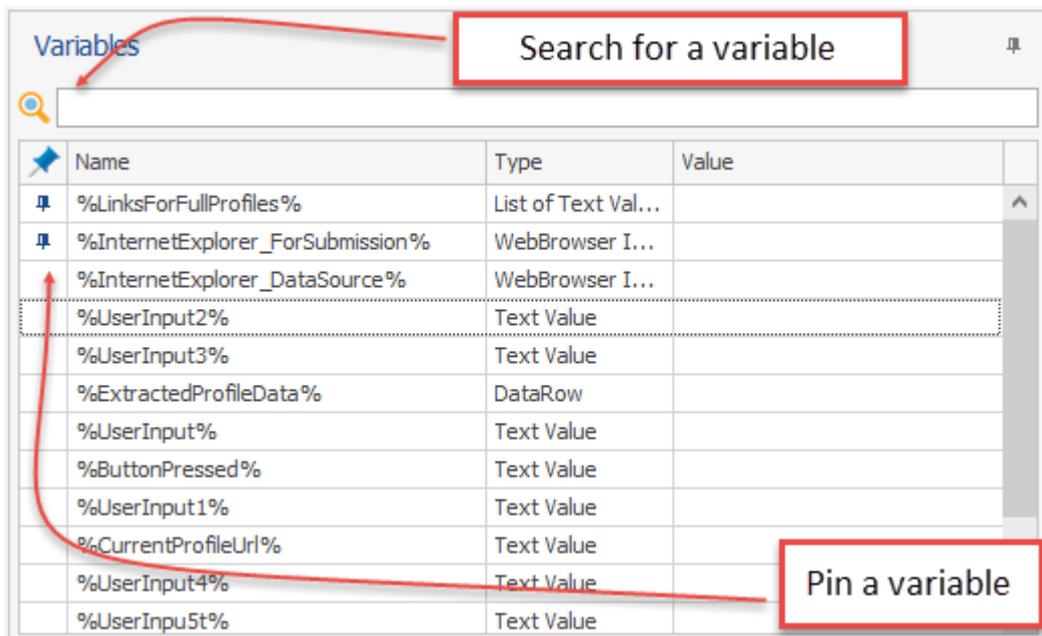
In the variables pane you are able to:

Search for variables by name. In the search field, type the name of the variable you are searching for and all the variables that contain that typed sting will be listed.

Pin Variables. You can pin a variable at the top of the display by clicking in the pin column at the left side of the variables name. Once you pin the variable it will automatically move to the top of the variables list.

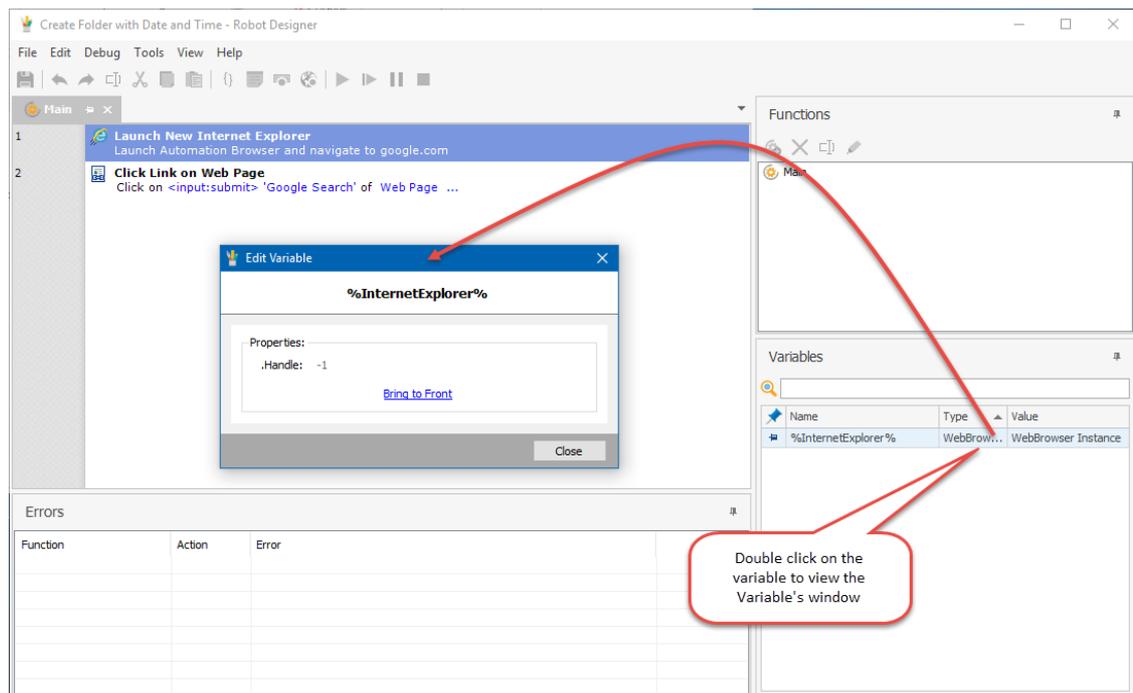
Re-order Variables. Click on a variable to move it up or down and reorder the variables list. (Of course, pinned variables cannot be moved)

The actions above can be performed while a Process is being executed or not.



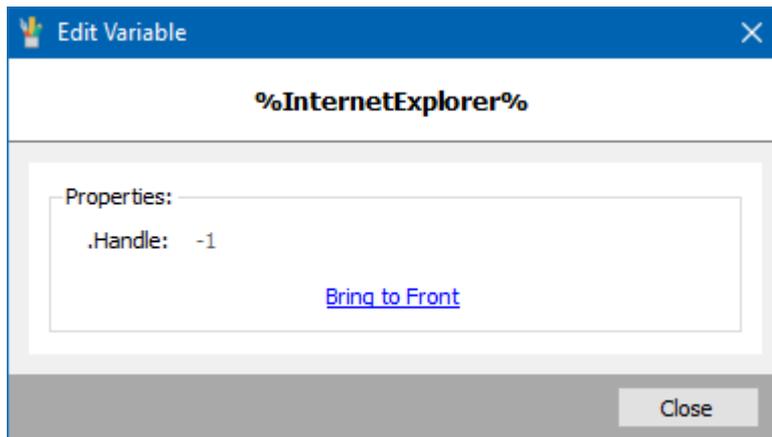
3.3.4 The Variables Visualizer Window

The Variables visualizer window can be displayed for each of the variables in a Process upon running it step by step, or pausing it, or if it has been paused by a breakpoint, or stopped BUT no action has been opened yet.

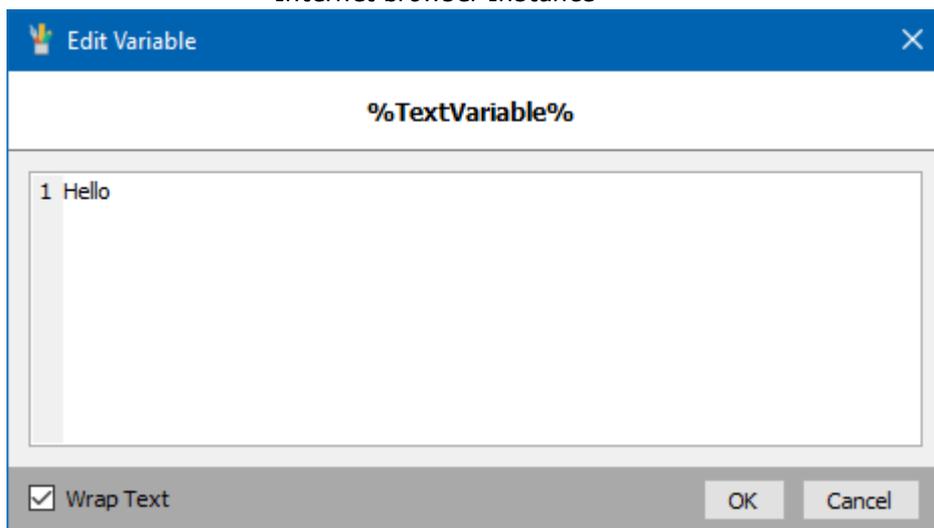


Double click on a variable in the variables pane and the Variable Window will appear, with information about the variable's value at this point. You can edit the variable and change its value on the flow before resuming the execution.

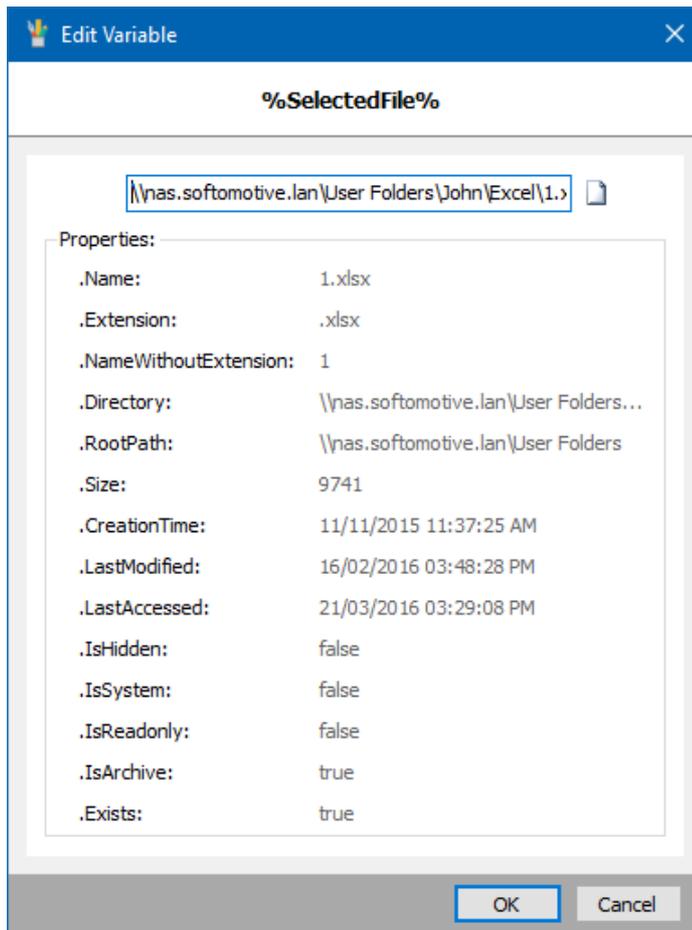
The variable window fields and information will depend on the type of the variable that you are accessing. Below you will find the different windows that will be displayed for different variable data types.



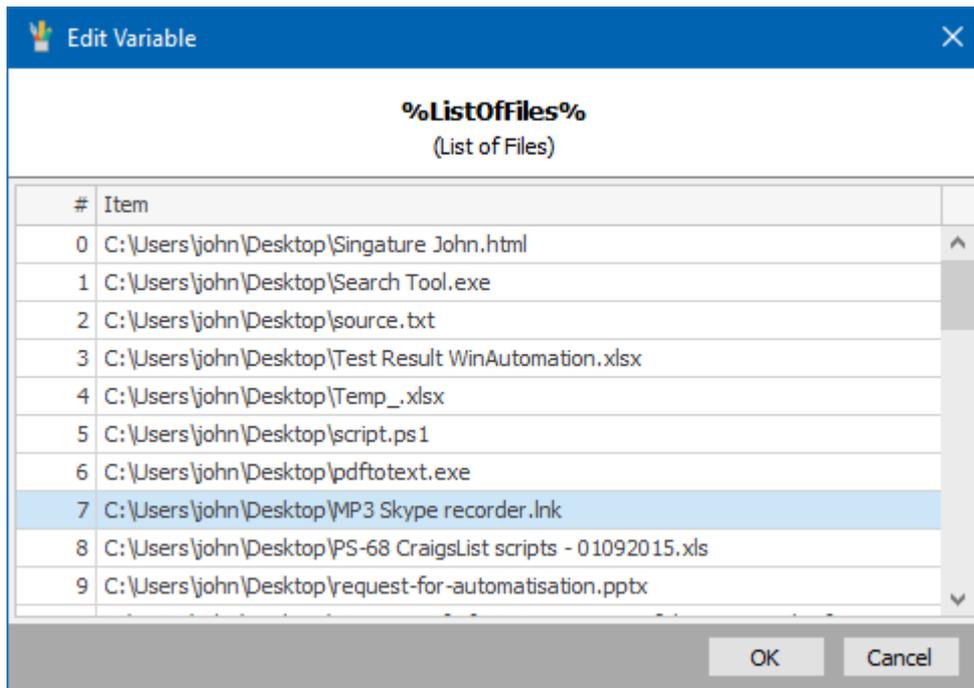
Internet browser Instance



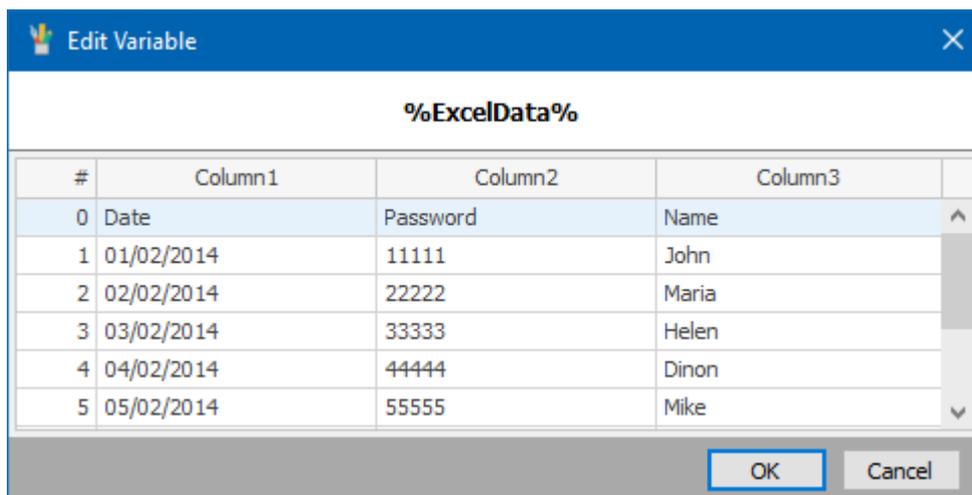
Text Variable



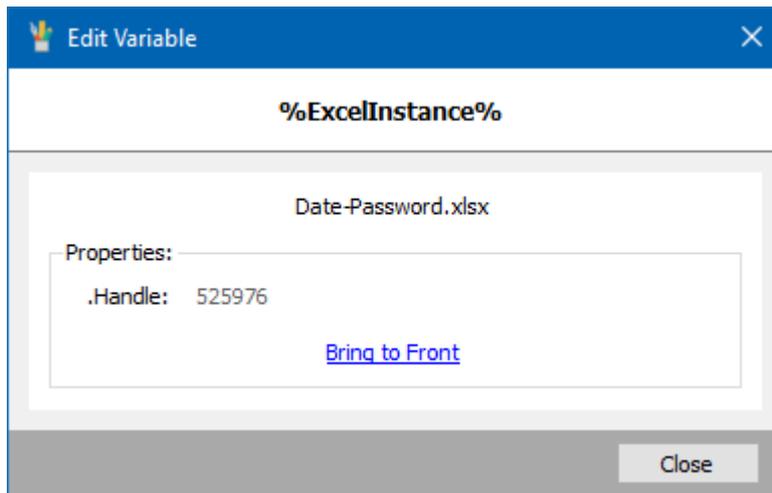
File Variable



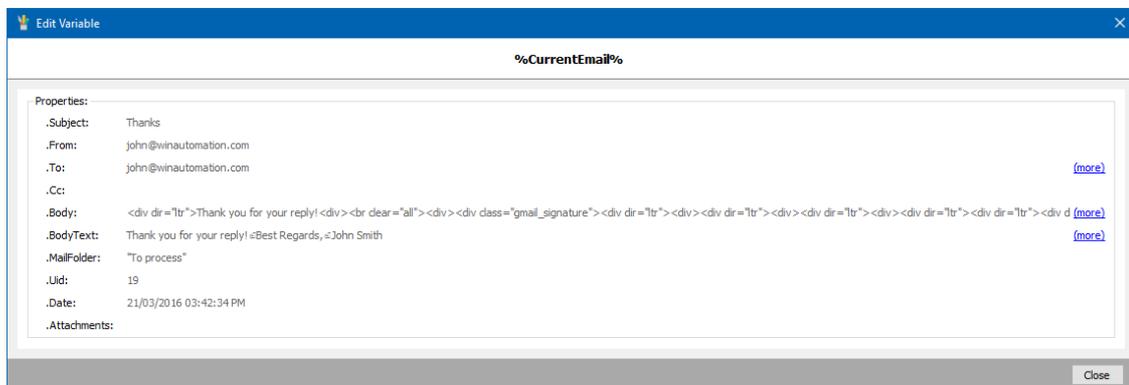
List Variable



Data table Variable



Window Instance Variable



Email Variable

3.3.5 The Errors Pane

The Errors pane is combined with the Control Repository, in tabs in the default Layout. In the Process Designer mode, errors are shown here. Of course you are free to change the default display!

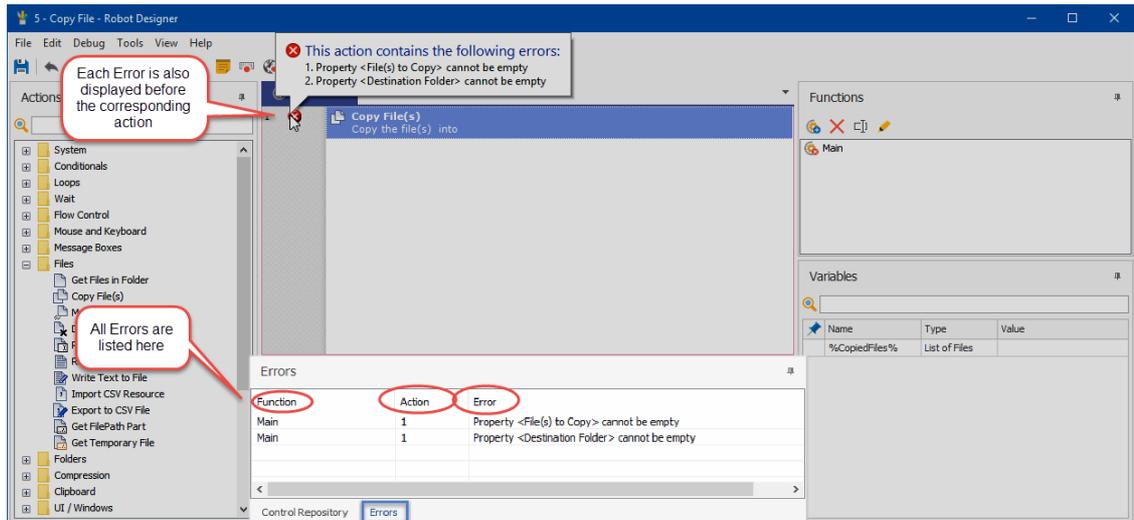
The Errors information is split to three columns: **Function, Action, Error**

Function: The Function Name that contains the action which threw the error.

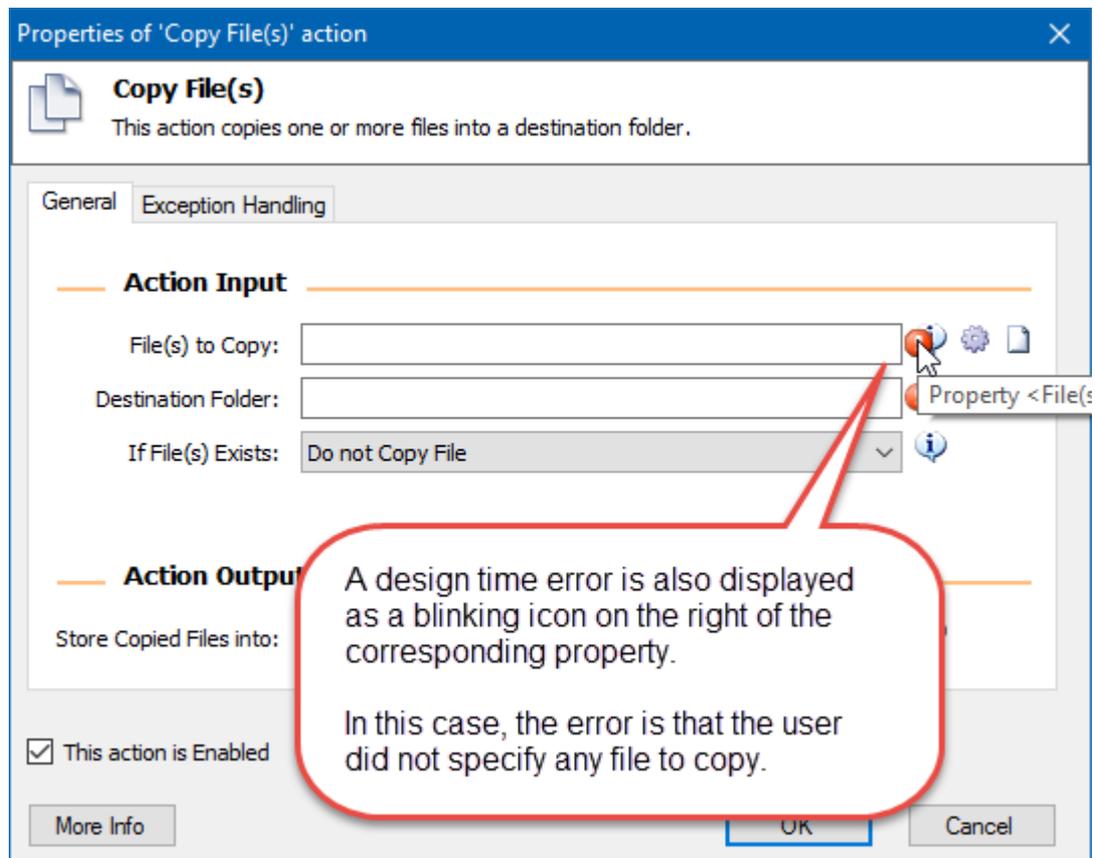
Action: The Action number in the Function which threw the error.

Error: The error message.

Each Error is also displayed before the corresponding action.



The Errors Pane of the Process Designer Window



The Properties Dialog of an action that has a Design-Time error

There are two kinds of errors: **Design Time** and **RunTime** errors:

Design Time Errors which are also called configuration errors, are visible in the configuration of the Action or Process and keep the Process from running.

RunTime Errors, or Exceptions are not obvious errors in the design but come up when you run the Process. Run Time Errors will cause the Process to fail unless you have planned for it in the Exception Handling Tab of most Actions.

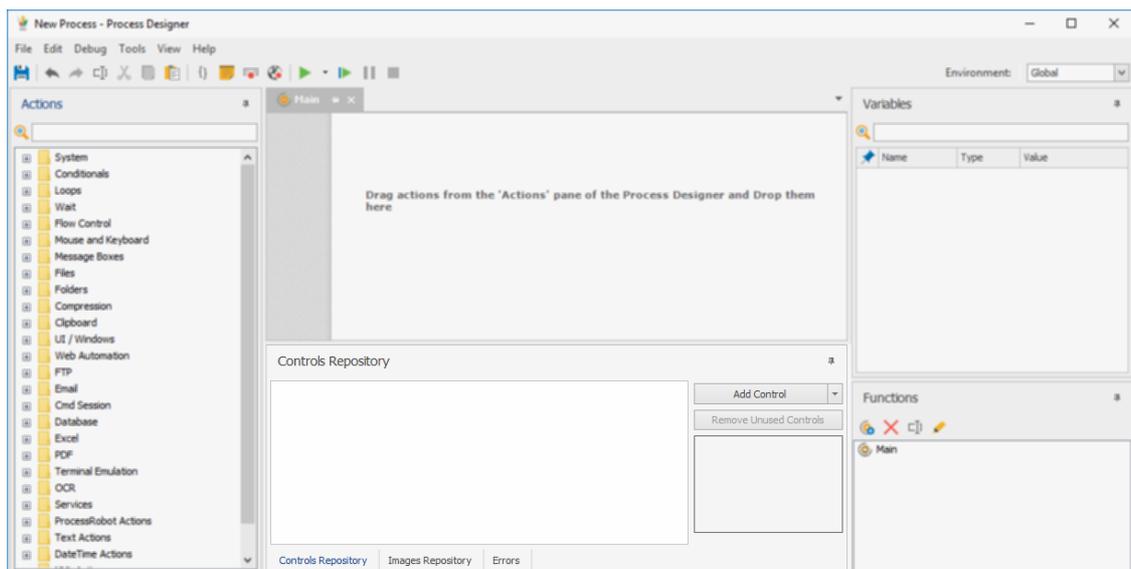
For example, let's say that you create a Copy File Action. If you leave the File to Copy field blank, that is a problem with the configuration and immediately shows up as a Design Time Error. If you put in a path and the file does not exist when you run the Process, that is a Run Time Error.

3.3.6 The Control Repository Pane

The Control Repository is placed at the bottom center part of the Process Designer, in the default layout.

It will hold all the controls from UI/Windows and Web Automation actions. All the elements that your Process will access (buttons, elements, items, text fields, checkboxes, radio buttons, windows etc...) will be stored in the Control Repository.

Upon creating a new Process the Repository will, of course, be empty.



The Control Repository

To add controls, you should use UI/Windows or Web Automation actions.

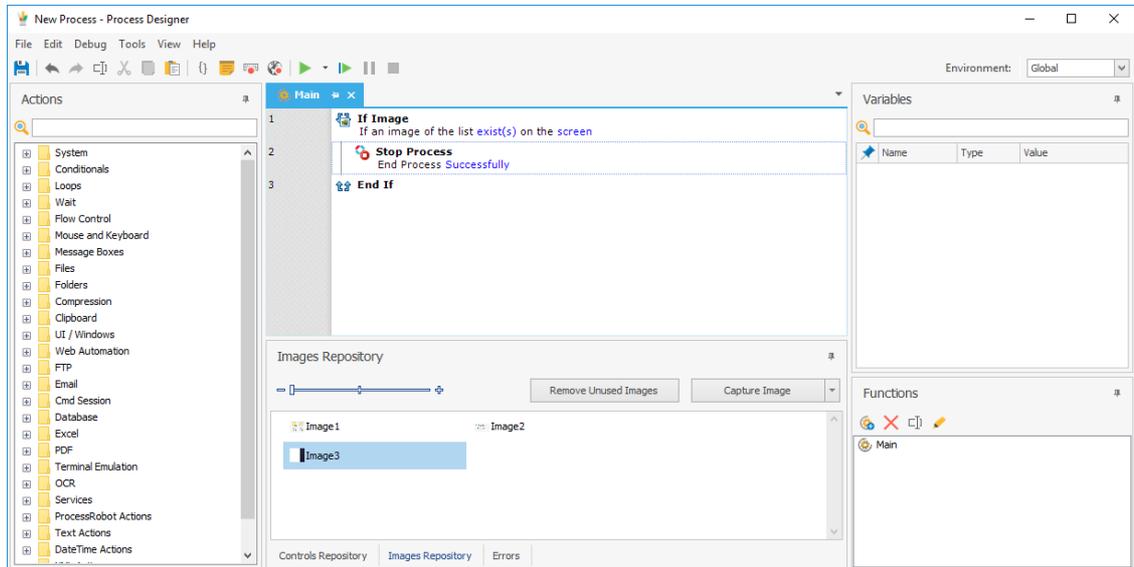
In the topics [Building a WebAutomation Process](#)^[353] and [UI Automation> UI Elements - Select Controls](#)^[338] you will find a detailed explanation about how to add/manage controls and edit their UI or CSS Selectors.

3.3.7 The Image Repository Pane

The Image Repository is placed at the bottom center part of the Process Designer, in the default Layout.

It will hold all the images that you have captured for Image related actions. All the Images that you have saved using Image recognition actions will be stored in the Image Repository.

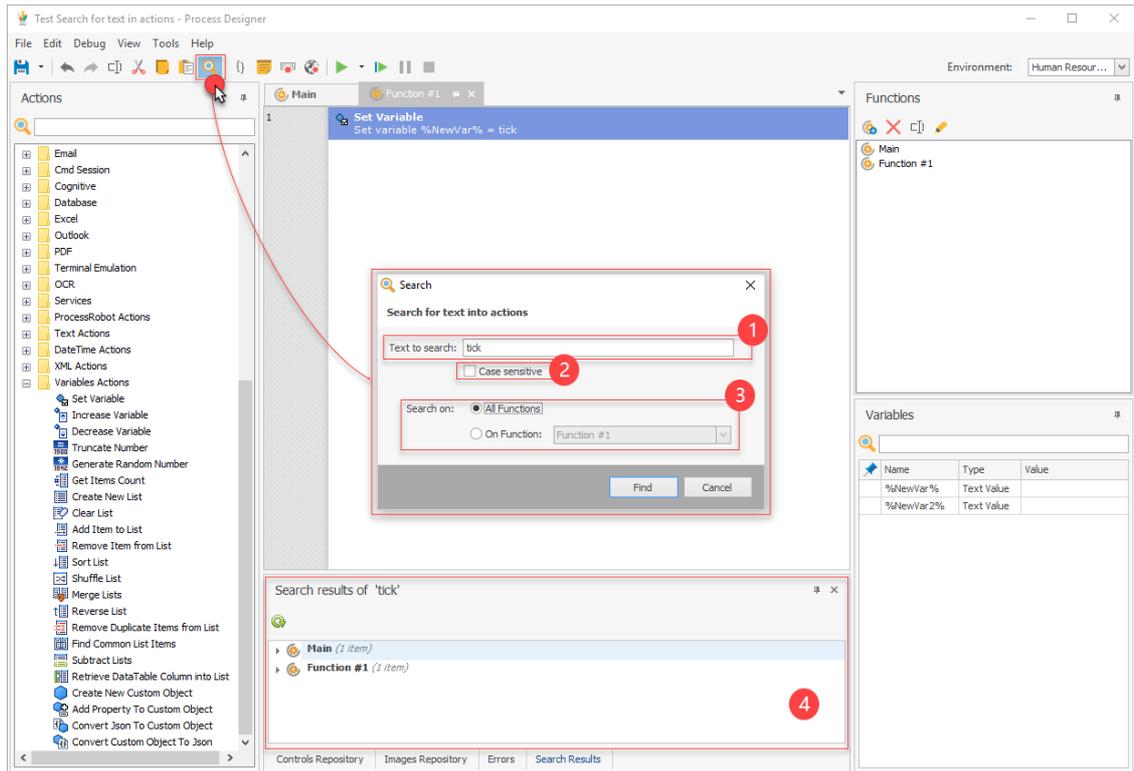
Upon creating a new Process the Repository will, of course, be empty.



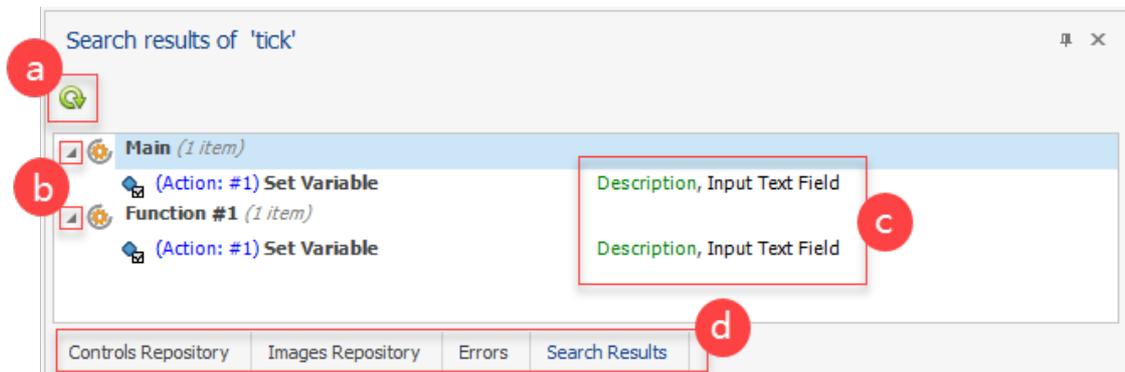
The Image Repository

3.3.8 The Search Results Pane

'Search for text into actions' is a very handy interface you can easily invoke through the Process Designer's Toolbar [1] in order to search for specific actions [5] that contain in them some specific text [1]:

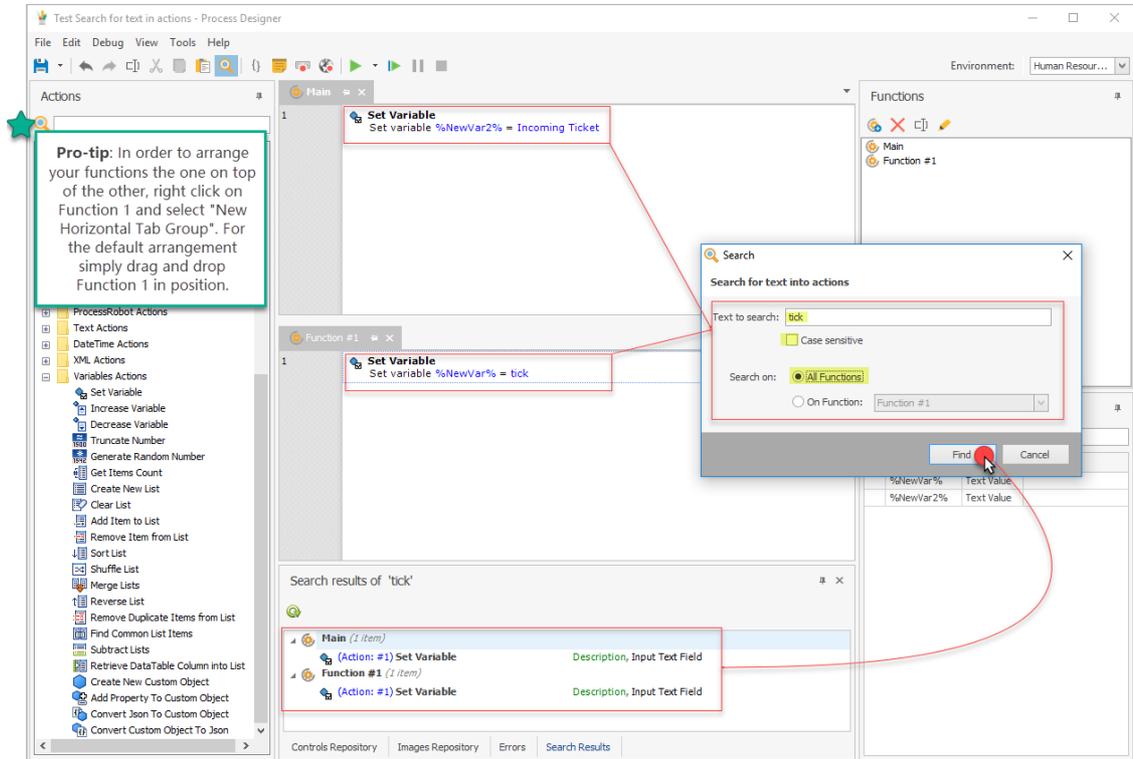


The Search Window allows you to find actions that contain the specified text potentially across your functions, a feature you can enable by clicking the appropriate radio button of the 'Search on' [3] property. The results of your search will be displayed in the Search Results pane [4] that informs you on the number of hits in each function. The pane also allows you to examine interactively your results, by clicking the little arrow [b] in front of the function that interests you:



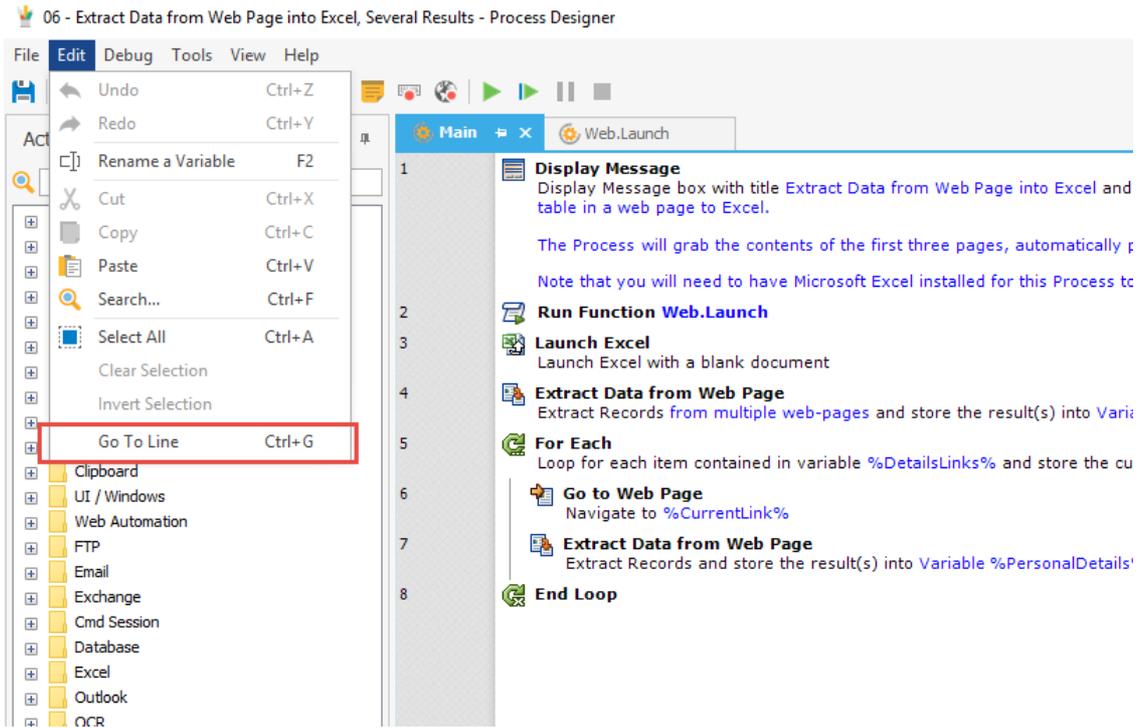
You should have in mind that you can easily transition from Search Results to Errors pane, Images or Control Repository at any time by clicking the appropriate option in the Tab Menu [d] at the bottom. The pane even informs you on where the hits were recorded [c] exactly in order to save you time from examining results that might be of little interest to you.

Another thing that should not go unnoticed is that even if you search for some text that is part of some longer sentence, the Search Window still will return a hit:



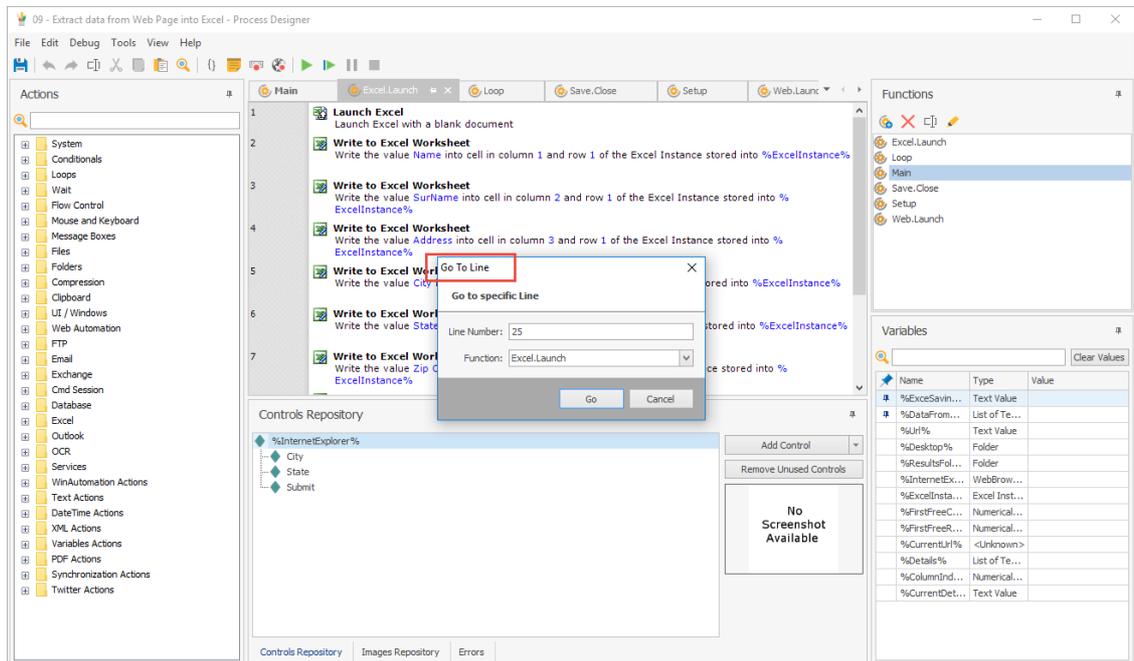
3.3.9 Go to Line

The "Go To Line" functionality is very helpful on large Processes. It will help you navigate to a specific line within your process in a Function. It can be accessed either by the "Edit > Go To Line" option in the menu bar, or by simply hitting Ctrl + G on your keyboard.



Go To Line

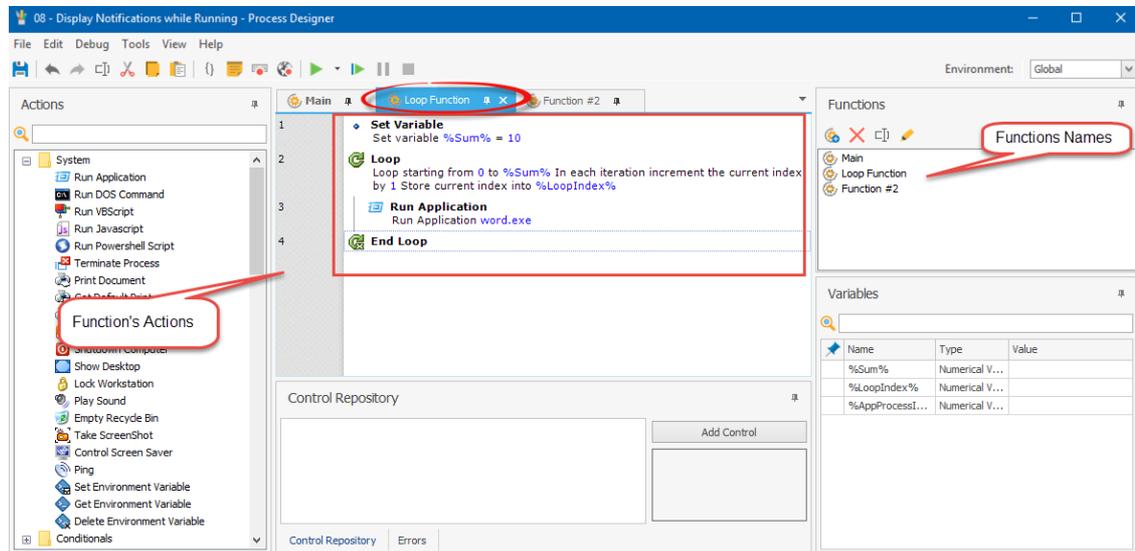
The Go To Line window will prompt you to enter the Line number that you wish to go to the Function on which you wish to be transferred. Just hit go and you will be landed in the action whose number you have entered in the Function of your selection, with the action being highlighted.



Go To Line Window

3.3.10 Functions

A function in ProcessRobot consists of actions grouped together so they can be invoked by a function name.



To create a new function, click on the New Function button (). Alternatively, you may also press *Ctrl+N*. Functions can be created, removed, renamed and have their tabs re-ordered freely, with the sole exception of **Main**, from the Functions pane.

To invoke a function, place a [Run Function](#)^[612] action anywhere in the Process. When this action is reached, the execution flow will jump to the first action of the invoked function and continue from there. Sooner or later, the actions of the invoked function will all be executed or an "Exit Function" will be reached. At that point, the execution flow will return to its former position, namely the next action after the invoking [Run Function](#)^[612] action.

Functions usually perform a specific task and it is always considered a good practice to name them in a manner indicative of their purpose. A function name is unique, i.e. two functions cannot share the same name while residing in the same Process.

New Processes, by default, contain only one function, the **Main**. The **Main** function is always called when the Process first executes. Other functions may be called by means of the [Run Function](#)^[612] action, by the [Exception Handling](#)^[502] mechanism of any action, or as a result of a button press inside a [Custom Dialog](#)^[636].

Scope: Variables in a Process are "global" in the notion that every action can access any variable of the Process from any function.

[Go To](#)^[611] statements on the other hand, can jump to labels inside their own function ONLY.

Recursion: A function cannot call itself; It can [Exit](#)^[613], [Run](#)^[612] another function or exhaust its actions and return automatically.

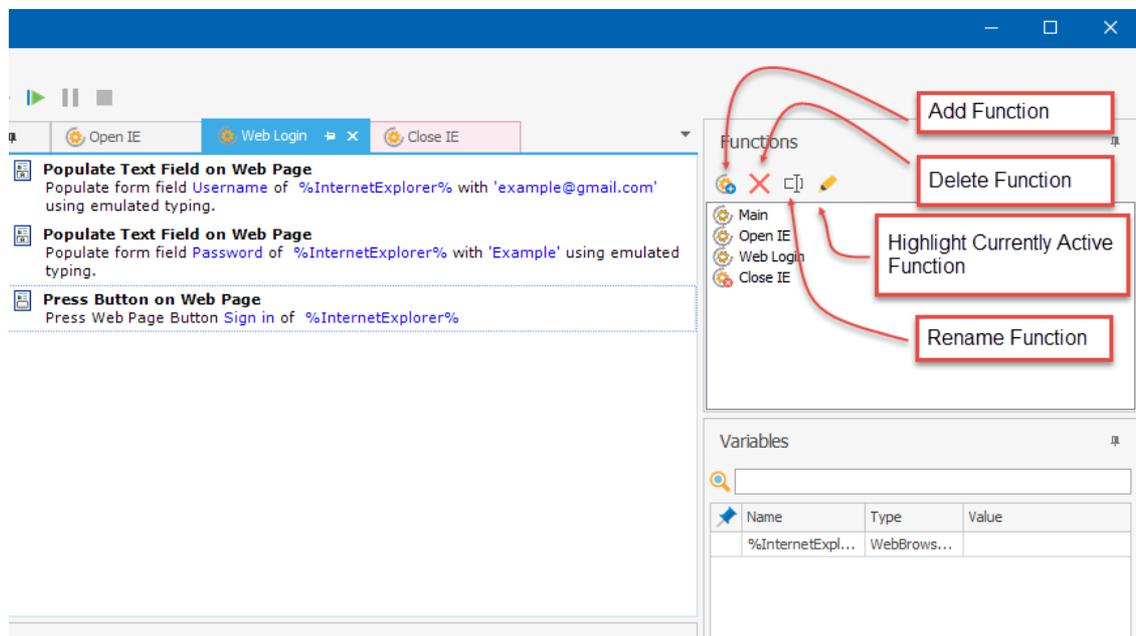
Some of the benefits of using functions are:

- Enhanced readability
- Code re-use without duplication
- Modular testing capabilities

3.3.11 Managing Functions

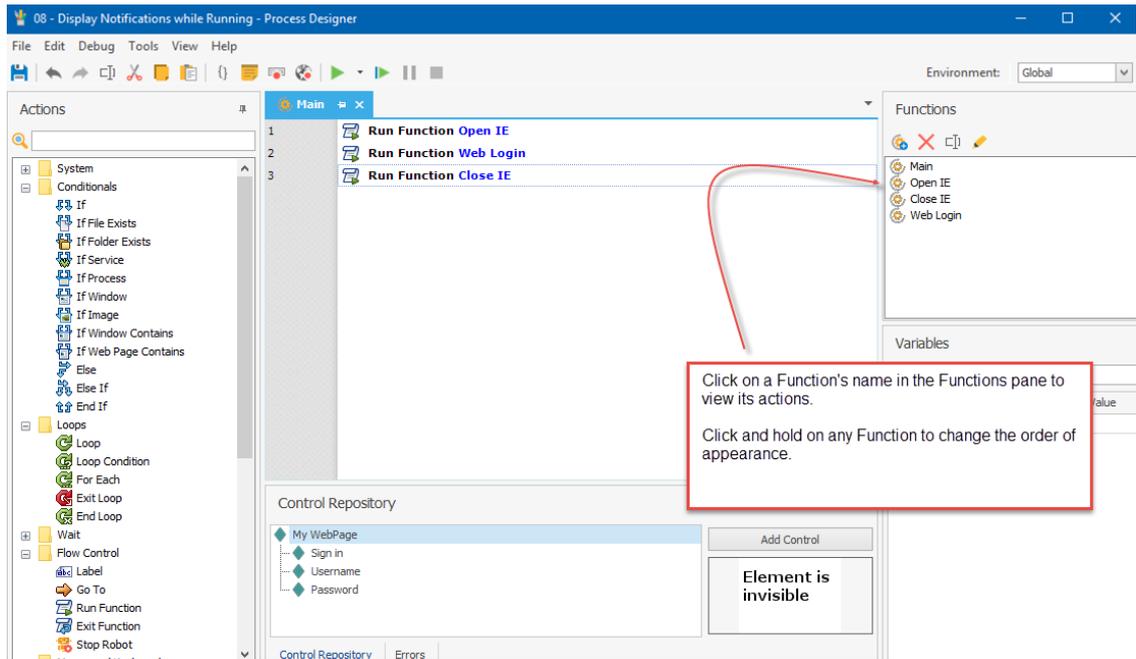
Functions work exactly like most Windows applications. You can select an Action by clicking on it, or multiple Actions using Ctrl or Shift in conjunction with the mouse. Once you have selected the Action(s), you may delete them using the Delete

option; or cut or copy them using the Edit menu, the Cut and Copy Icons. You may also Drag them to another part of the Process.



Managing Functions (1)

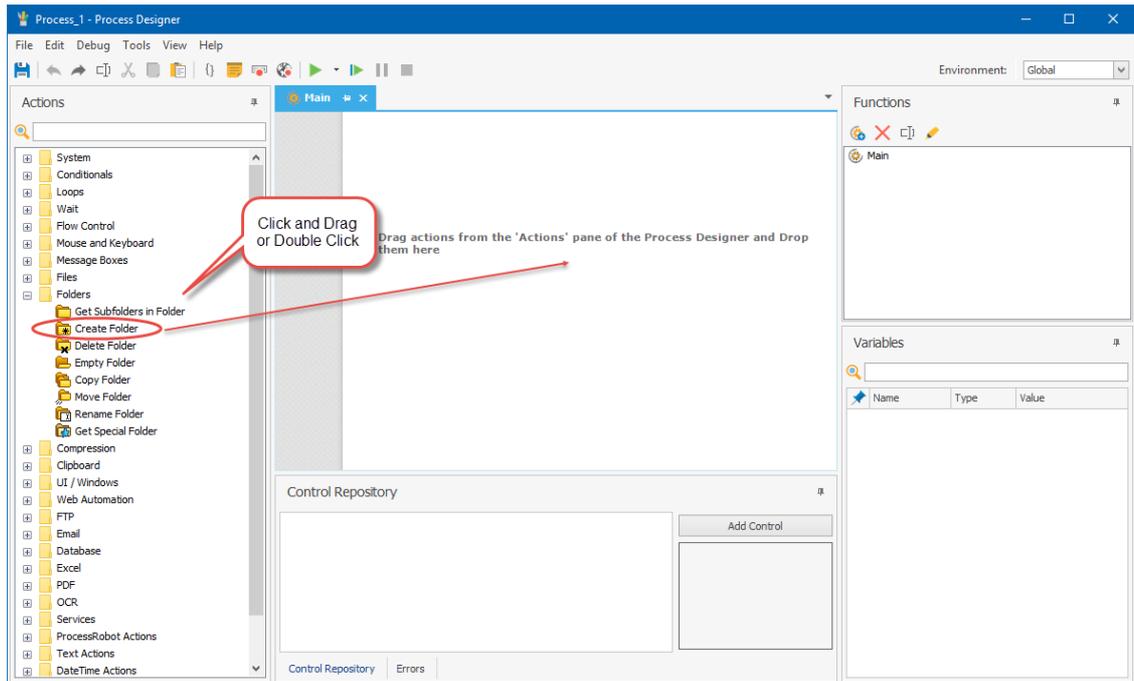
You can click on a function in the Functions pane and move it up or down for reordering. You can also close them, if you do not want them to be visible in your workspace. Keep in mind that if you run the Process from the designer then each Function will be visible while being executed.



Managing Functions (2)

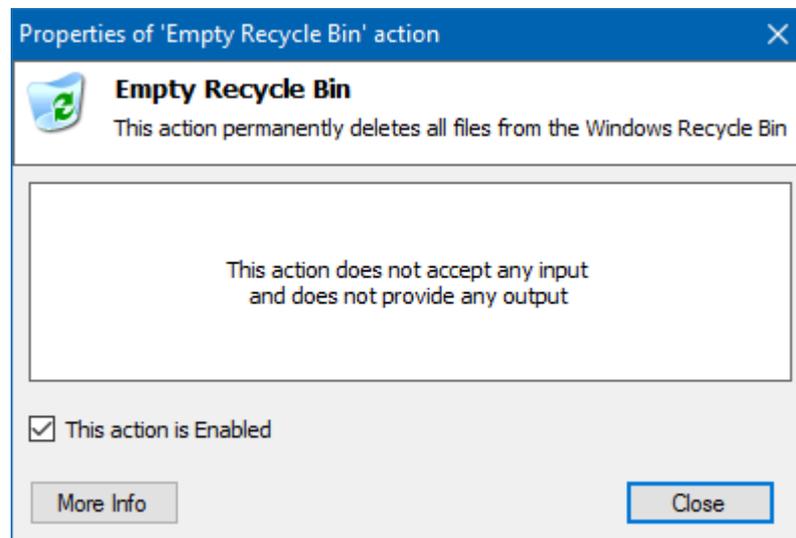
3.3.12 Adding an Action

You add an Action from the Actions pane to the Workspace pane by double-clicking on it or dragging it and dropping it where you want. Double-clicking will always place the Action at the end of the current Process, but can be dragged or moved by highlighting the Action and clicking on the Up or Down Arrows in the Toolbar.



Inserting a new Action into a Process by Drag & Drop

Once you have added the Action, the Action Properties Dialog Box will open automatically if the Action has any configurable properties. As a counter-example, the Empty Recycle Bin action has no properties to set, so will just appear in the Workspace without opening a Properties Dialog Box.



3.3.13 Configuring an Action

Adding an Action to the Workspace will automatically open the Action Properties Dialog box if there are any properties to be determined. If you want to open a previously created Action,

double-click on it or right-click on it and click on Action Properties.

Once you have an open Properties Box, you will see **Action Inputs** and **Action Outputs**:

Action Inputs require information from you, to determine what happens - text to be displayed, a Variable to be used by the Action, a file path or URL, and so on.

Action Outputs will only be Variable names so that information generated or collected by the present Action can be passed on to a later Action.

Please, note that the "Action Output" refers to data and is not considered to be a program run or a Message Box shown - those are *effects* of the Action, not Output.

To the right of a property field, you will see an "i" Icon. This is a ToolTip Help that you can access by pointing the cursor over it. You may also see a gear icon "⚙️", a file icon "📄", or a folder icon "📁". These show where you can browse for a file, a folder, or a Variable (gear).

If there is an Error thrown by one of the properties (a missing file name, improperly written Variable, etc), there will also be a blinking red Error icon "🚫" the next time this Action Property Dialog Box is opened.

If the Action could cause an Error, you might want to look over the Exception Handling options (click on the tab at the top) to make sure you like that result.

As an example, we will look at the [Display Input Dialog](#) properties. There are four different Input properties:

1. Input Dialog Title - what you want to show at the top of the Box, for example "Enter Name"
2. Input Dialog Message - the text or question you want to show in the box, for example "Hello user, what is your name?"
3. Default Value - do you want to have a default setting, or an example like "<Enter Name Here>"
4. Input Type - how do you want their input formatted? This is a drop-down menu, so choose whichever seems most appropriate to the information.

Notice that all of these fields require some information that you currently have - what you want to write, how you will format, etc. This is Input - from you, now. Notice that the Action will require input from the user when the Process is running, but that is not Action Input here - only your choices now are Action Input.

Properties of 'Display Input Dialog' action

Display Input Dialog
This action displays a dialog box that prompts the user to enter text

General Exception Handling

Action Input

Input Dialog Title: Enter your Name

Input Dialog Message: Hello User, what is your name?

Default Value: <Enter your name here >

Input Type: Single Line

Keep Input Dialog Always on Top

Action Output

Store User Input into: %UserName%

Store Button Pressed into: %ButtonPressed%

This action is Enabled

More Info OK Cancel

Also notice that the Action Output is only information stored as Variables. The Output is determined by the Action as the Process is running and is used as Input in later Actions - this is the definition in ProcessRobot of a Variable: a 'container' used to carry values from one Action to another.

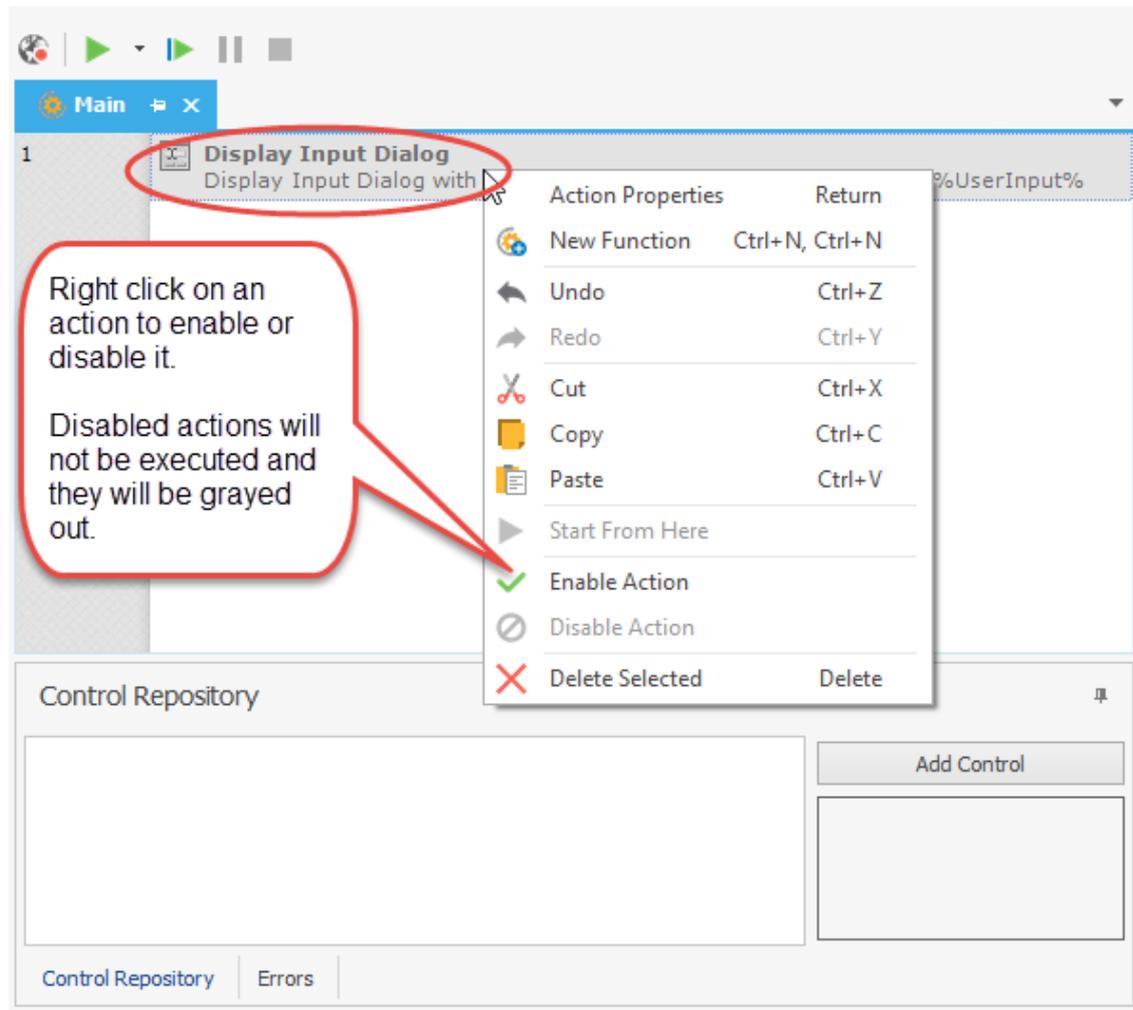
Choose Variable names that mean something to you, so you can easily remember them later. Also, read about [Variable Types](#) so you are clear with your use later. Once you have defined Variables through Output, you can browse them using the gear Icon to use them as input in a later Action. In this case, we stored the text entered by the user (which is supposed to be its name) in "%UserName%" and the Button pressed into "%ButtonPressed%".

3.3.14 Enable/Disable Actions

You can Disable or Enable Actions by right-clicking on the Action and select "Disable Action" or "Enable Action".

This allows you to remove an Action from a Process without erasing it and having to re-type everything if you want to use it again later.

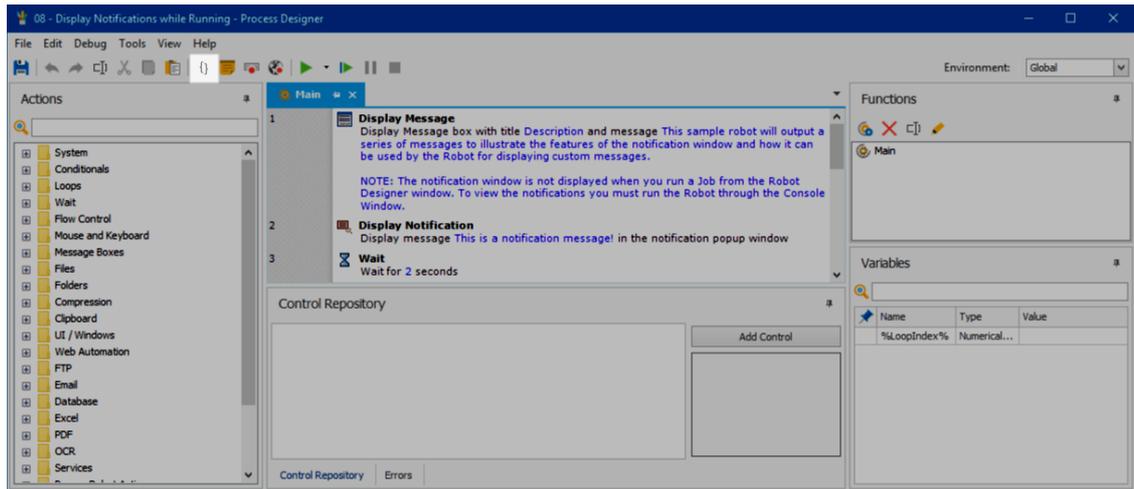
Often, this will be used to test different versions of a Process to see what works best and has the desired results. You could also suspend usage of a part of a Process while something has changed.



Enable/Disable actions

3.3.15 Organize your Actions with Regions

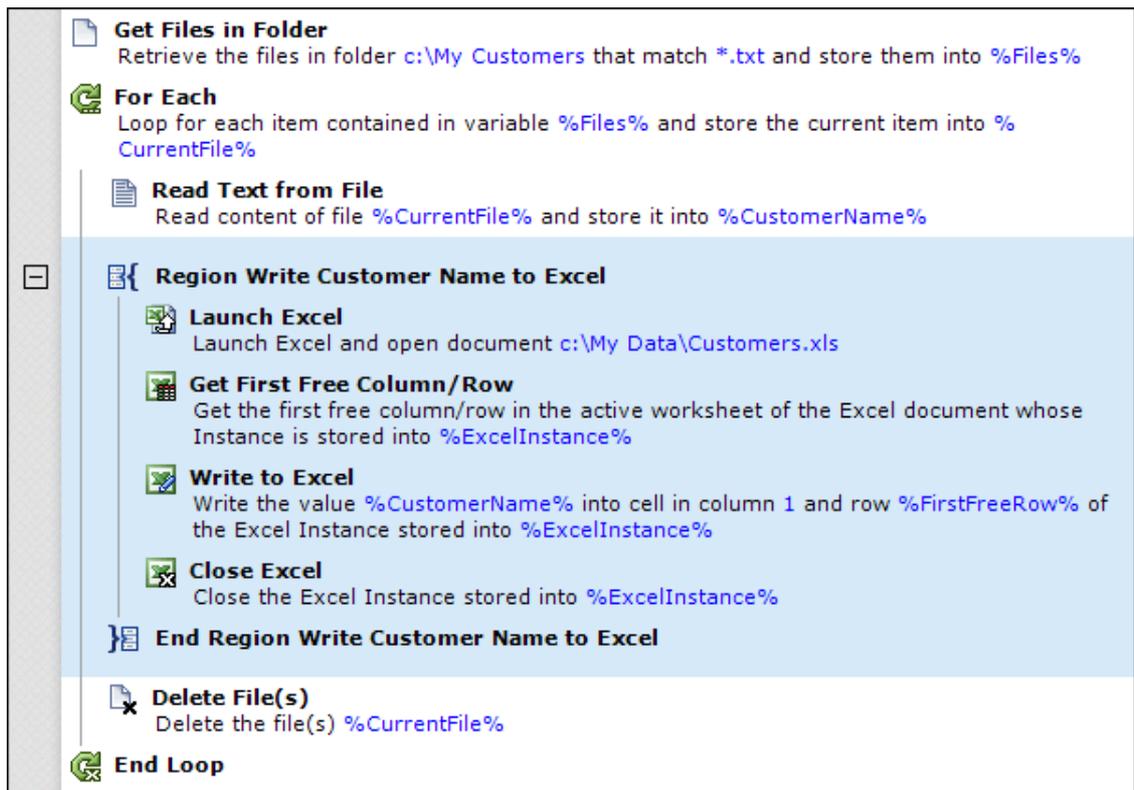
In order to organize the actions in your Process and make long Processes or Functions more manageable you can use the Regions feature of the Process Designer.



Regions

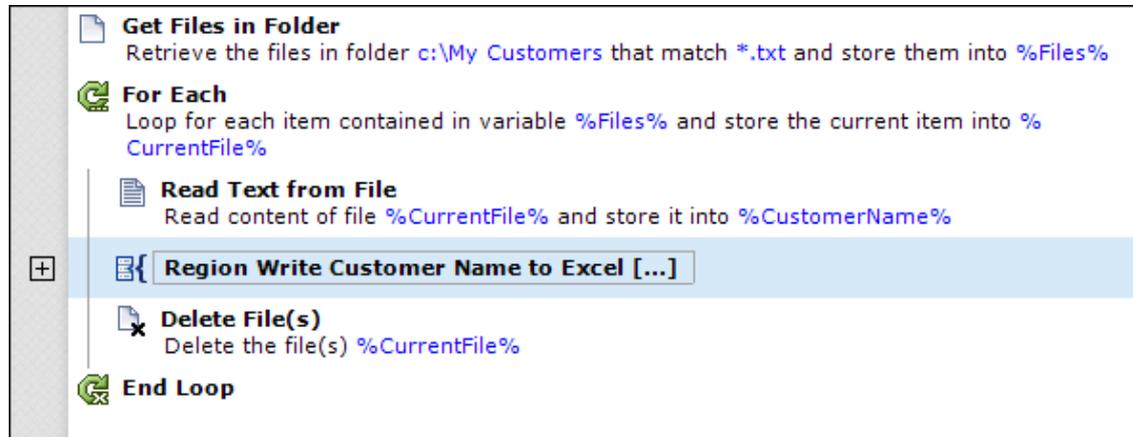
The Region block consists of two actions the "Start Region" and "End Region" action.

These actions have absolutely *NO IMPACT* in the behavior of the Process and their only task is to mark the beginning and the end of a logically separate block of actions.



An expanded Region within a Process

Each Region (as defined by the two aforementioned actions) may have a name to describe the block of actions contained within. You can also collapse a region by clicking on the [-] icon on the left of the "Start Region" action. This way the actions of the region will not be visible, allowing you to review your Process at a higher level of abstraction.



The same Region in collapsed state

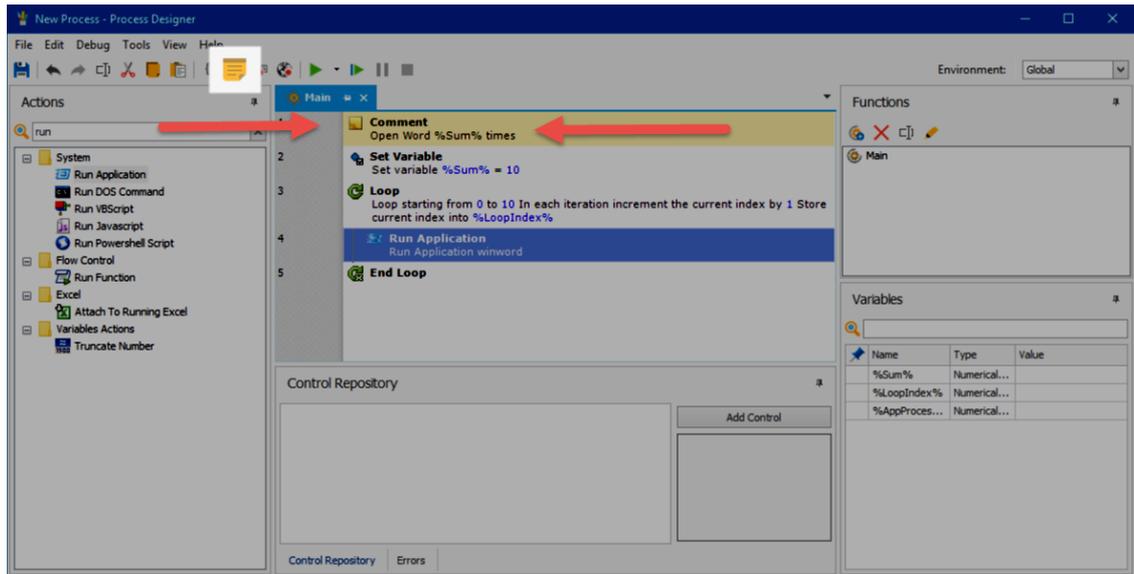
To create a new Region you can click on the "Region" button  on the toolbar. If no action is selected this will insert the "Start Region" and "End Region" actions at the end of the Process. If one or more actions are selected the "Start Region" action will be inserted right before the first selected action and the "End Region" will be inserted right after the last selected action enclosing the selected actions into the newly created region.

Regions have a different background color than the rest of the Process. The background color of the regions is customizable and can be changed through the [Process Designer Options](#)  dialog.

Regions can also be nested within other regions to represent submodules of your Process within other modules. In this case the background color of each nested region becomes automatically a bit darker than the background color of the parent region.

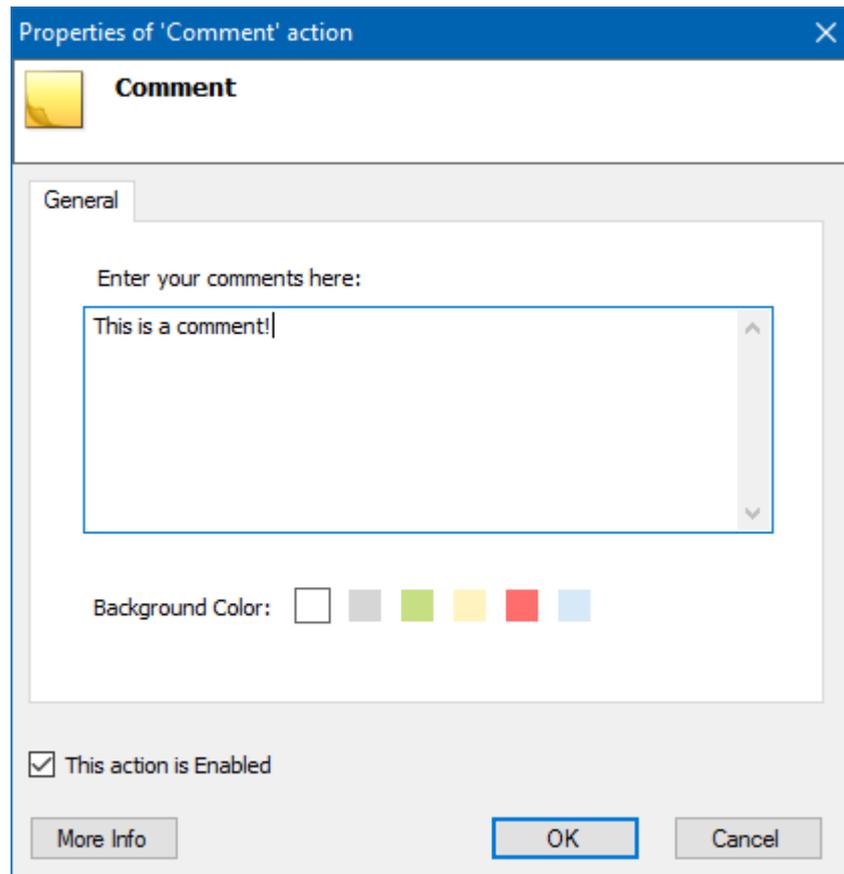
3.3.16 Adding Comments

The yellow sticky-pad button  in the Toolbar is used for adding comments. A comment is a special kind of Action in the Process that is used for documentation and for adding explanatory notes between the actions.



Comments in Process Designer

The background color of each comment can be set to a different color as a visual clue for differentiating sections of the Process and giving more emphasis to some comments compared to others.



Neither adding nor deleting Comments has any affect on the behavior of the Process.

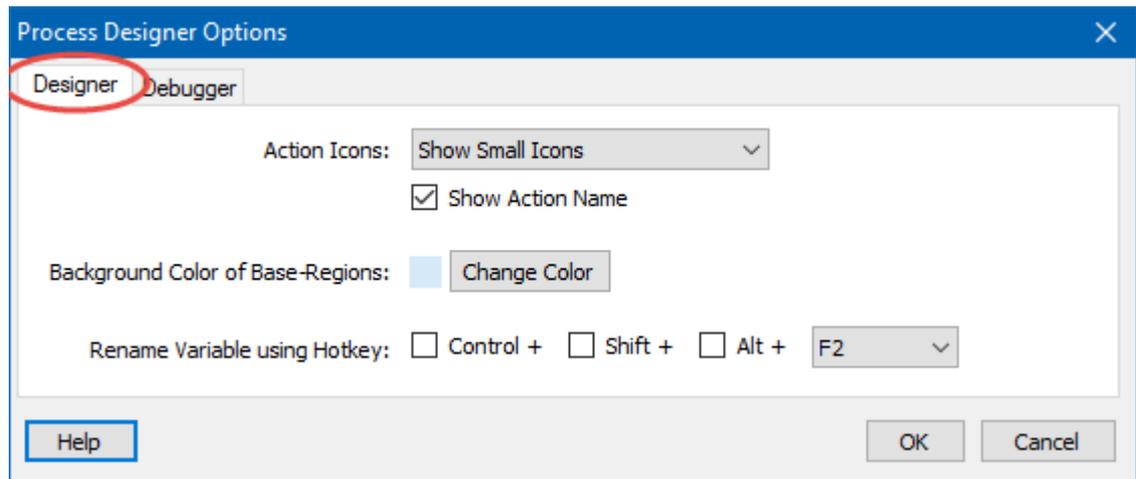
3.3.17 The Variables Manager Window

You may access the Variables Manager through [Tools -> Variables Manager](#) while in the Process Designer. This is where you can find a list of all Variables used in the Process. If you want to define a new Variable and assign an initial value to it, you can do it here, by pressing the Add New Variable button.

Keep in mind that an alternative way to declare a new variable is to simply enter a variable name into an Output property of any Action of the Process. If this variable name does not refer to an existing variable, a new variable with that name will be created automatically. A variable created this way can not be edited or deleted through the Variables Manager Window

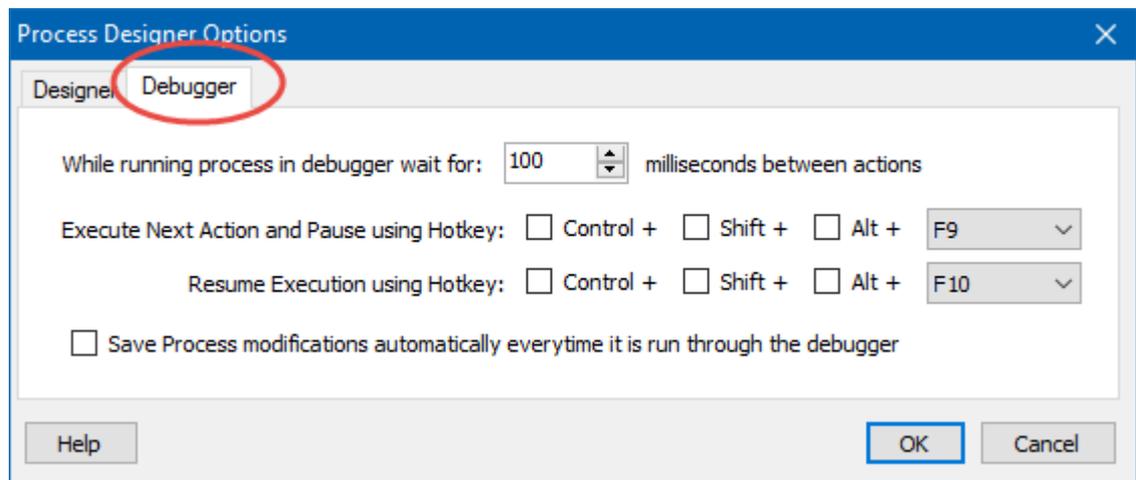
Once you open the Variables Manager Window, you will see two tabs - User Defined Variables (ones you set) and Trigger Variables (predetermined to manage Triggers, these will be empty unless you use that Trigger to start the Process). The Trigger Variable tab contains only the names of the Variable and a description. The User Defined Variables have five columns:

You have the option to rename a Variable using a Hotkey of your choice.



Process Designer Options > Designer tab

The Debugger Option allows you to set a delay time between the execution of each Action. This is an **artificial delay**, that is useful if you want to follow the actions visually as they execute. **This delay is taken into account only when you run the Process through the Designer**, when you run a Process **through the Console** then **the Process runs at full speed**.



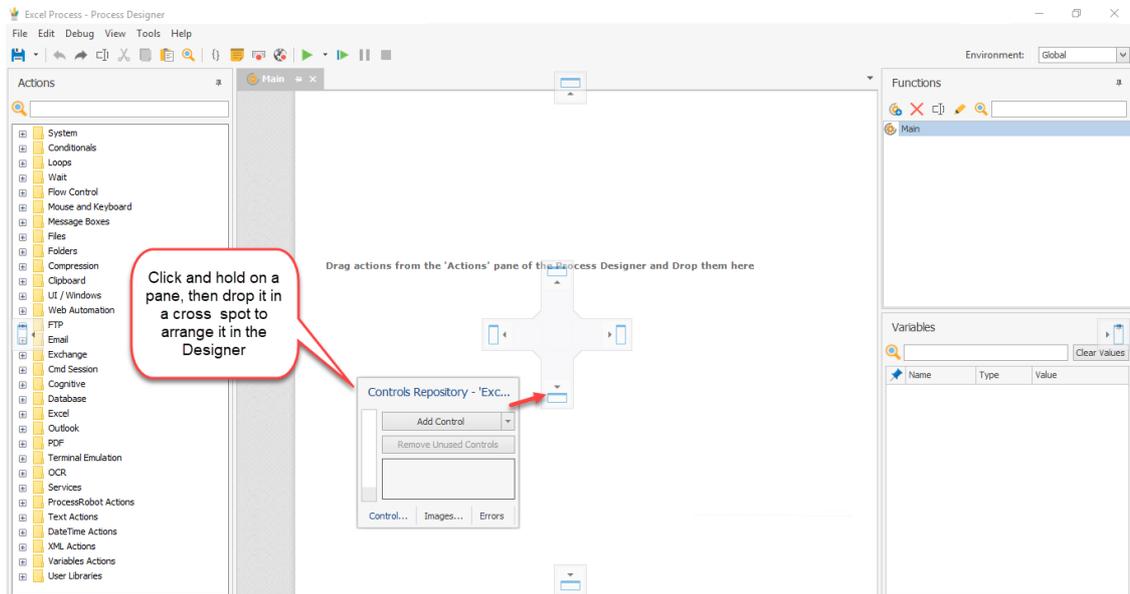
Process Designer Options > Debugger tab

The second setting is the designated hotkey for performing [Step by step execution](#)^[318] and to resume the execution while debugging your Process. The hotkeys specified here will execute the next action when the Process is paused in the Debugger or Resuming the execution. This is useful especially for Processes that control the mouse where it is inconvenient to use the mouse in order to click the "Execute next action" button on the toolbar.

The last option lets you specify whether you want any modifications of the Process to be saved automatically every time you run it through the debugger.

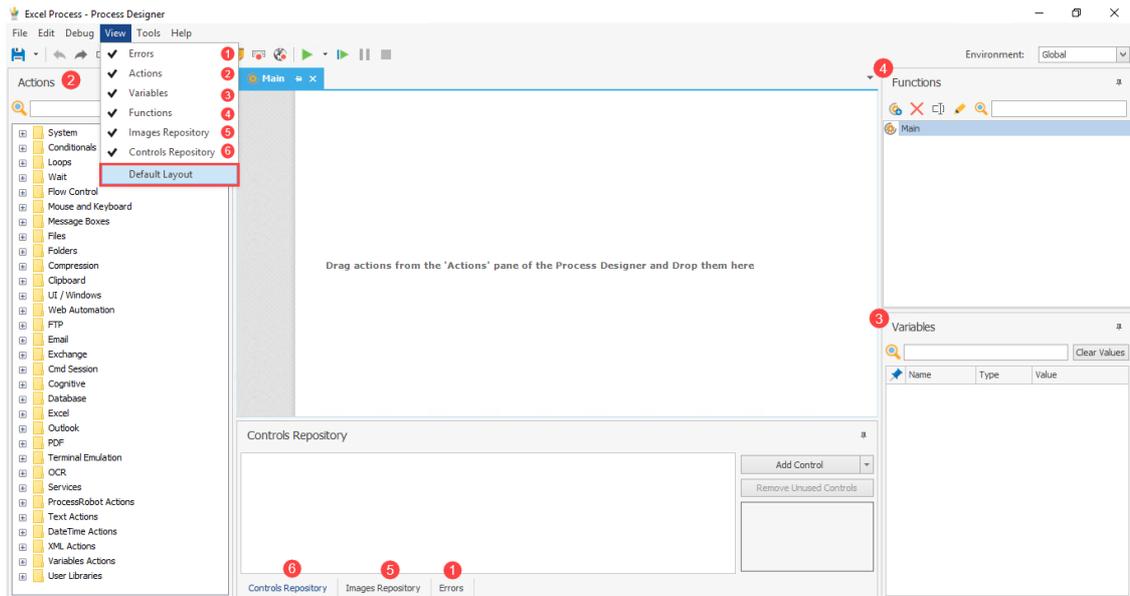
3.3.19 Process Designer Custom Layout

You have the ability to customize the panes' location in the Process Designer. You can click and hold on a pane and you can drop it in a location on the cross in order to pin it in a specific position.



You can also select to View all or less of the panes in the Process Designer, by clicking on the 'View' menu option and tick/untick the panes that you want to have visible on your designer.

You can always click on the "Default layout" option in order to rearrange all the panes in their default location.

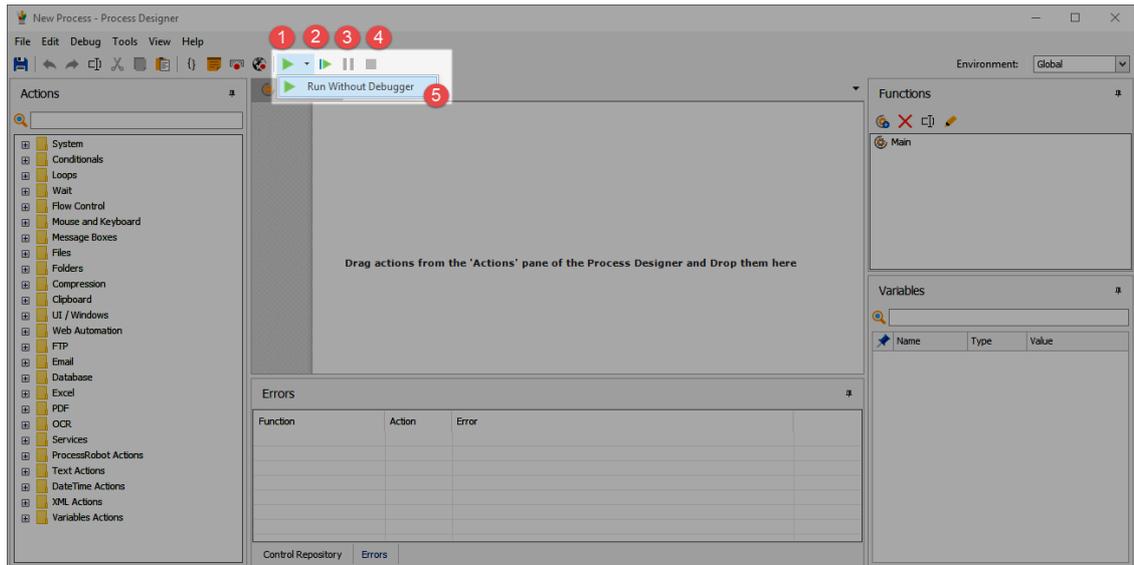


3.4 The Debugger

3.4.1 Run/Stop/Pause a Process from Process Designer

Running a Process in the Process Designer is considered **Debugging**.

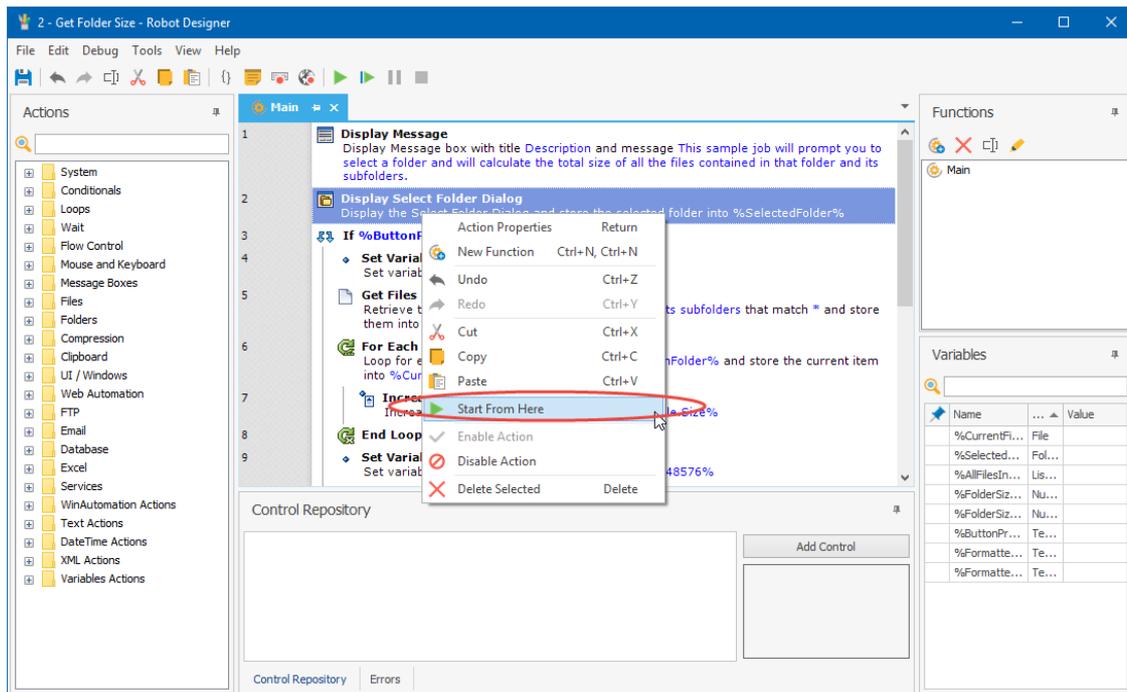
1. You can run a Process by clicking on the Run button .
2. You can execute a Process action by action by clicking on the button next to "Start" .
3. You can pause a Process while it is running by clicking on the Pause button .
4. You can stop a Process while it is running by clicking on the Stop button .
5. You can run a Process choosing the "Run Without Debugger" option. This will run the Process as if it was running on the target robot and not from the Process Designer. The Notification Window will also pop-up, exactly as it happens when a process is running on the target robot.



Run/Stop/Pause a Process from Process Designer

Since Debugging is expected to involve running the program, making a change, running it again, etc, no Logs are recorded when you run the Process from the Process Designer. In the Process Designer Window, you can also see error icons to the left of any Action that has errors "  " and therefore cannot be run, and to the right of any property field in the Action Property Dialog Box "  " to help you pinpoint errors to Debug the Process.

If you want to start running the Process from a specific action: You can right-click on this action and select "Start from Here" on the context menu that will appear. This will start the debugging from the selected action as if it was the first action of the Process, ignoring all actions above it. Please note that this may lead to unexpected results, as the ignored actions will not populate the variables that may be needed by the executed actions. Keep in mind that you will not be able to start running a Process from an action within a Loop, as it makes sense that this would not be applicable to the logic of the script in terms of what we are looping through.

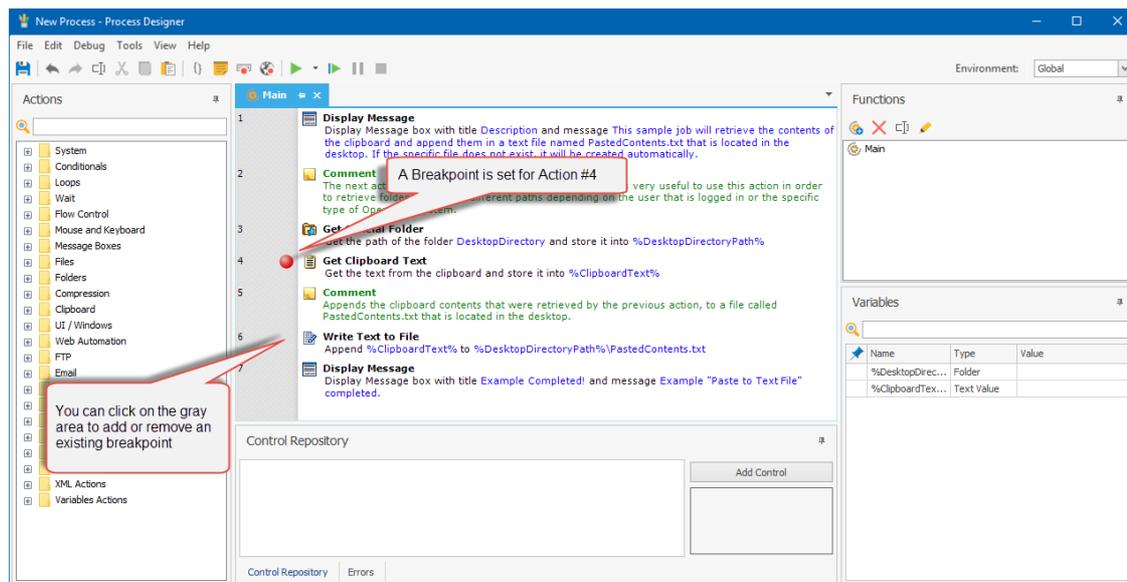


Start execution from a specific action

3.4.2 Adding Breakpoints

You can add **breakpoints** to stop the Process at any Action. Simply click in the gray column to the left of the Action you wish to pause at.

A red dot "●" will appear to mark the breakpoint, and the Process will pause at the beginning of that Action. Pressing the Run button will continue the Process, though if the breakpoint is in a Loop, it will pause again when the process gets to that Action. Since there is an Icon to [Execute Next Action](#), you can add a breakpoint at the first Action of a Process and then use the Execute Next Action button to move through the Process manually, one Action at a time.



Breakpoints

Breakpoints are considered, only when you are running the Process through the Process Designer (debugging). When you run a Process through the Console all breakpoints set in the Process are completely ignored.

3.4.3 Execute step by step

You can move through a Process one step at a time by clicking Debug -> Execute Next Action, or by clicking on the Execute Next Action button  on the toolbar. This will start the Process, but pause before the first Action. You can then click Execute Next Action to move through the Process. By using the Variables Pane ²⁸⁸, you can watch the variable values as they change.

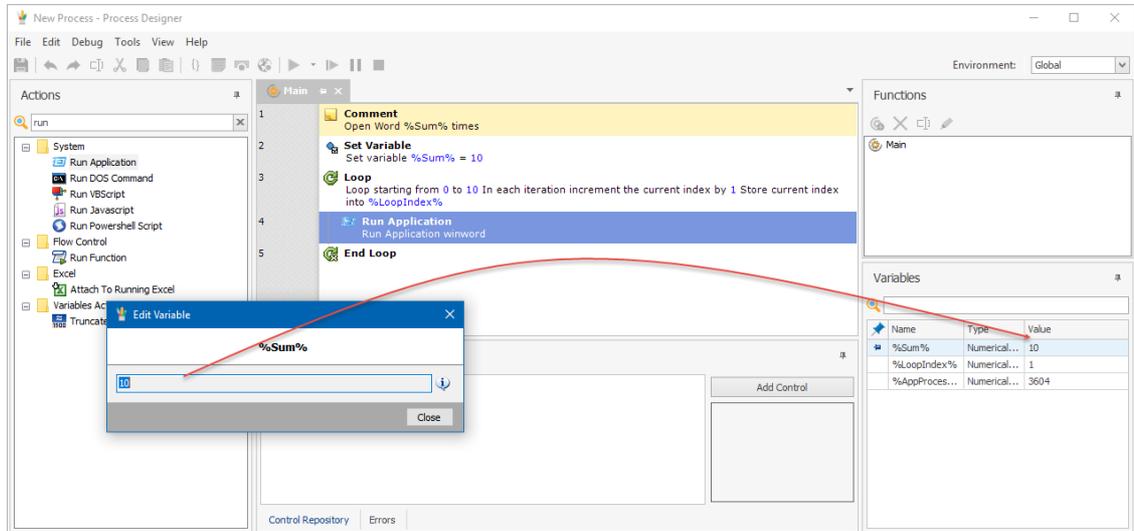
Additionally, you can use this button after a Process being stopped by a breakpoint to execute only the subsequent action instead of pressing the Start button that will resume the execution of the Process.

Variable inspection during execution:

You can examine a Process at a particular Action step, by adding a breakpoint at that step as mentioned above, or by moving through the Process manually with the Execute Next Action Icon. Once you have stopped at the Action you want, you can view the Variables through the Variables Pane ²⁸⁸ of the Process Designer Window.

You are able to see what is the last value stored in each variable once the execution is completed. *Also, once the execution pauses on a breakpoint you have the option to change a variable's*

value dynamically through the [Variables Visualizer Window](#)²⁹⁰ set a different one and resume the execution to see how the process will react to that value.

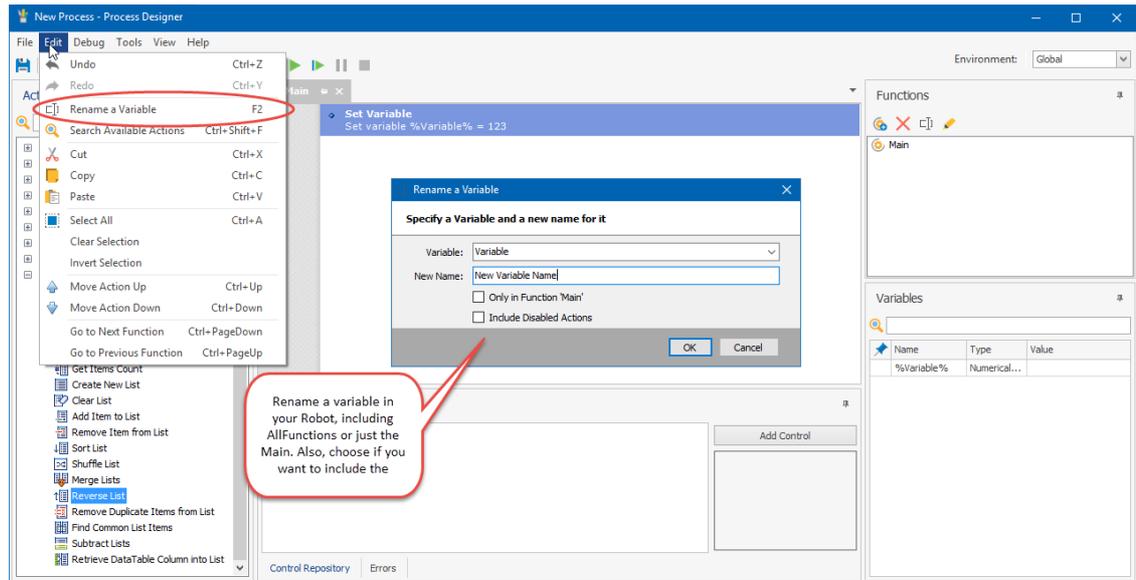


As a last note, instead of clicking on the button you may as well press the designated Hotkey. This is especially useful when debugging Processes that control the mouse and you don't want to use the mouse at the same time to click on the button. The default key for executing the next action is F9, but you can change this through the [Process Designer Options](#)³¹² dialog.

3.4.4 Rename a Variable

You have the option to massively Rename a variable in your Process while in the Process Designer by clicking on Edit > Rename a Variable or by clicking on the Rename Variables icon "  " in the toolbar.

In the window that pops up you can select to rename the variable on in the 'Main' function and also include disabled actions.



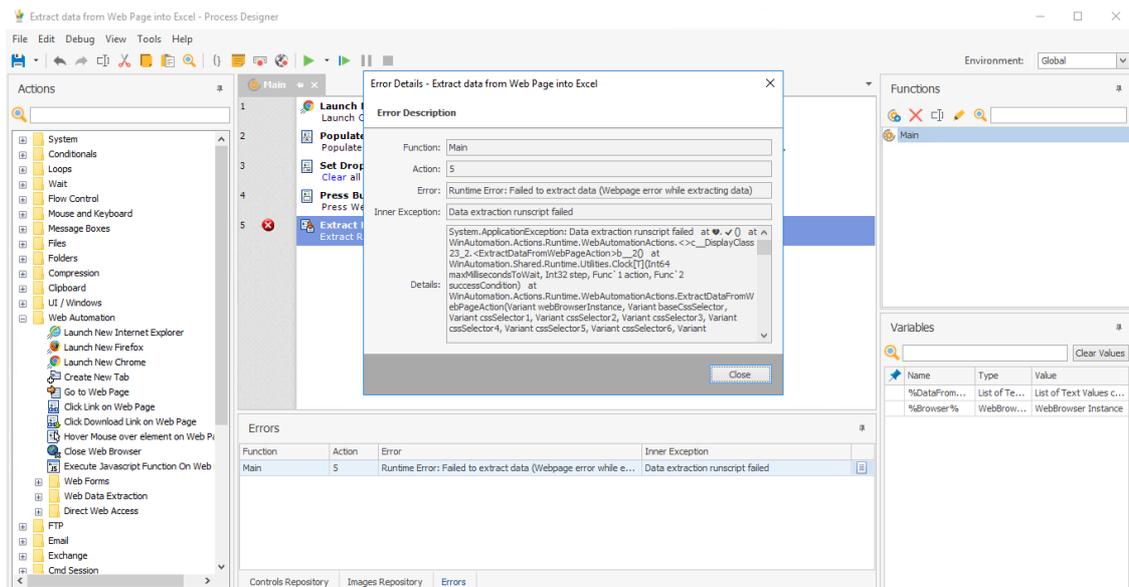
Rename a Variable

3.4.5 Full Error View

When running a Process from the Process Designer, then the only thing that the user is able to see is the error message. The Additional details of the error is only available in the Audit/Logs, only when running the Process in the Review or the Production lifecycle stage.

Now, when a Process throws an exception, then you can go to the Errors pane at the bottom of the Designer, and double click on the error. Once you do so, the Error message window will popup. This window will have information about:

1. The Function that the error took place
2. The action number in the Function that the Error took place
3. The Error message short description
4. The Inner Exception of the Error
5. The additional Details of the Error. These details are very helpful when debugging as they are giving a clear message to the user about why the error happened and what is its real cause.



Full Error View through the Designer

3.5 The Macro Recorder

3.5.1 Simulating the activity of a user

A user (like YOU) inputs his/her commands and data through the Mouse and Keyboard. Through Mouse, Keyboard and UI Automation Actions, ProcessRobot can simulate you (a user). You can create this simulation by choosing Mouse, Keyboard and UI Automation Actions or by using the Macro Recorder - which will automatically generate an appropriate set of Actions that mimic your own, recorded activity.

Remember - since you are creating an [Interactive Process](#)⁵⁰⁷, ProcessRobot cannot simulate a user (and run the Process) when no users are logged on. Here is where the AutoLogin feature on Solobots comes handy.

3.5.2 Recording vs Building a Robot

In addition to building a Process by combining and configuring Actions, you can also use ProcessRobot to simulate a user's activities by controlling the mouse and the keyboard. When you open the Macro Recorder, it records your activities and creates a set of Actions out of it, as a Process.

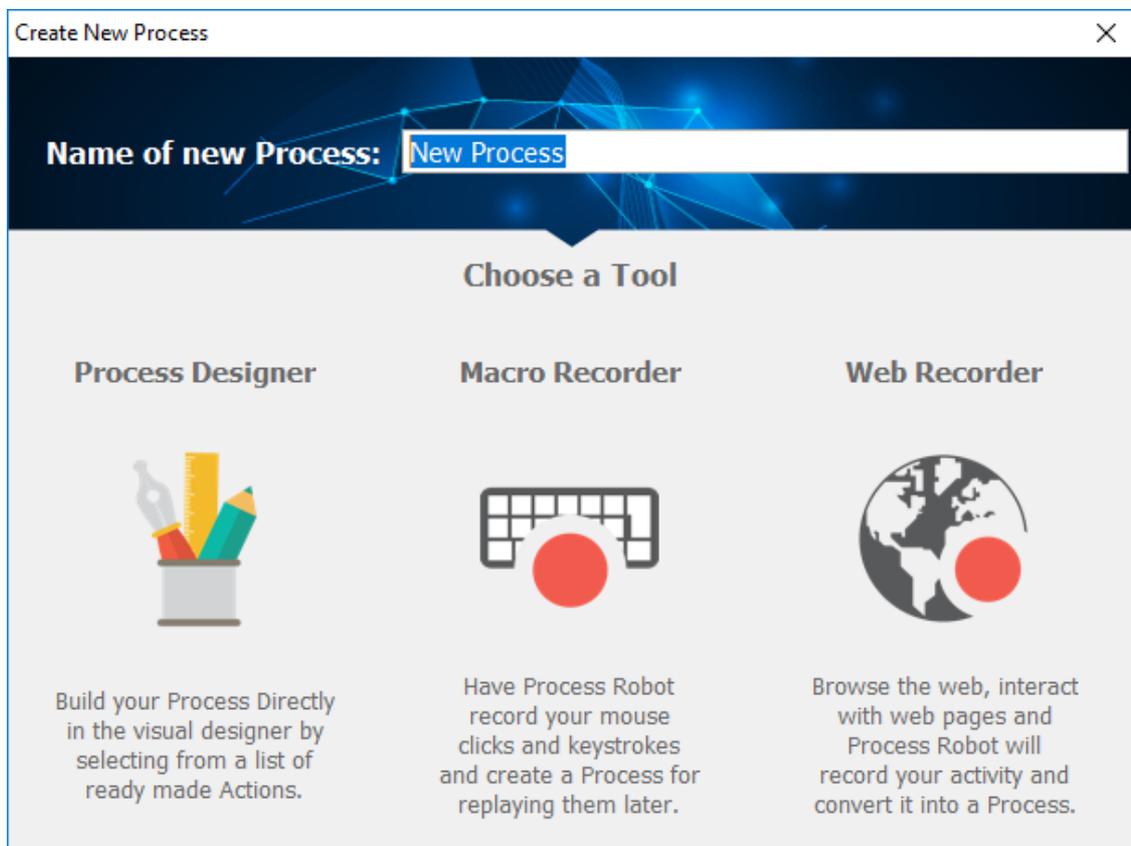
Launch the Macro Recorder and it will record everything you do. When you finish recording, then ProcessRobot converts this to a set of Actions for your Process. These can be edited and added to later. If you use the Recorder with a Process that already has Actions, it will add the steps above the highlighted Action or at the bottom if you don't have any highlighted. The Macro Recorder is really just a way to input Actions in an easier interface than writing them all out would be.

Keep in mind that you can use the Macro Recorder to record the backbone of your script. Of

course there are tasks that will have no issue running, only by recording them once and not modify any of the actions generated. Still there are cases when the generated process has to be modified to work as intended. For example you might need to add a Loop in the script, or there are redundant actions generated that you should remove. In cases as such we add/remove the appropriate actions from the ones generated from the Recorder, and if any error occurs while executing we identify the problem or the error and we try to use alternative actions to cope with the situation and have it run flawlessly.

3.5.3 Macro Recorder

To start the Macro Recorder when you create a Process, after you name the Process, choose option 2: Open Macro Recorder and click OK.



Create New Process

If you have already created the Process, you can open the Macro Recorder in the Process Designer Window (that is, inside the Process you wish to edit) by clicking Tools -> Record Macro or by clicking on the Record Macro button  on the toolbar.

While you can build a Process in ProcessRobot by combining and configuring actions from the list of ready made actions in Process Designer, you may find it more efficient to use Macro Recorder to simulate your activities by just controlling the mouse and the keyboard.

Launch the Macro Recorder and perform the activities you want your Process to reproduce. Macro Recorder will record everything you do and when you finish recording it, it will convert this to a set of Actions for your Process. Of course, these can be edited or added to other actions within the Process Designer.

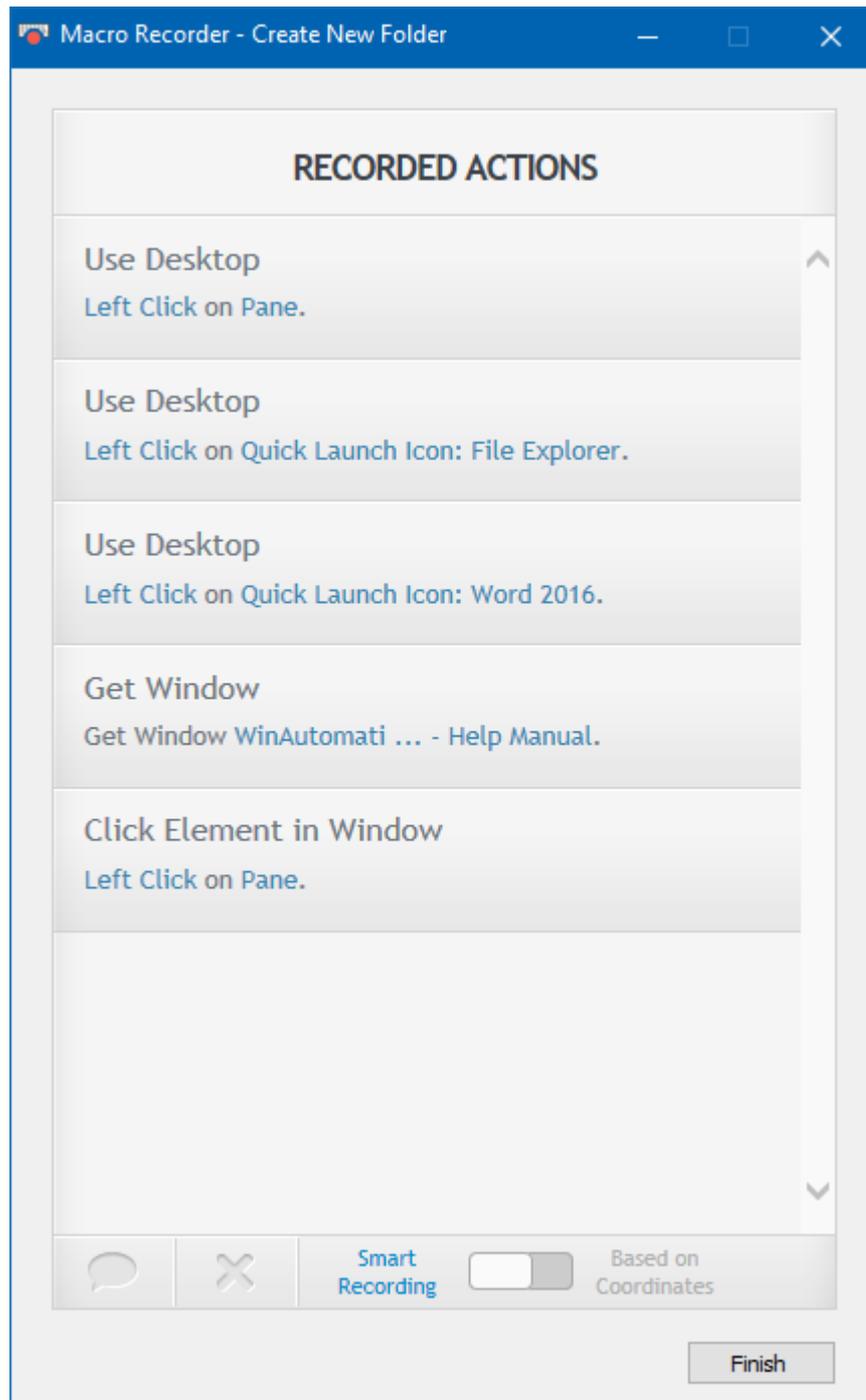
Using the Macro Recorder is a quicker and easier way to define Actions that compose a Process than manually selecting them and inputting them in the Process Designer.

There are two ways that the Macro Recorder can be record your activity; one is to record based on Coordinates and the other is Smart Recording. The first method creates mouse and keys actions but the Smart recording is created UI/Windows action. **The Macro Recorder by default recognizes the different elements on the screen (such as buttons, text boxes and other standard Windows controls) and injects the appropriate UI/Windows actions to build your Process with.**

Starting the Macro Recorder

In order to create your Process using the Macro Recorder, choose option "Macro Recorder" from the tools of the "Create a Process" window, or (if you are already within the Process Designer) choose the Macro Recorder button on the toolbar of Process Designer window (inside the Process you wish to edit) or choose Tools > Macro Recorder.

Launching the Macro Recorder opens the sidebar where you can preview the recording while performing your activities.



The Macro Recorder can record in two modes: Smart and Coordinate-based.

Smart Recording (Default): The actions produced by this type of recording will click on elements such as windows, buttons and menu items based on UI selectors rather than absolute locations. UI selectors are very similar to the [CSS Selectors](#)^[485] utilized by Web Automation actions. This type of recording is the recommended one, since it produces flexible and portable Processes.

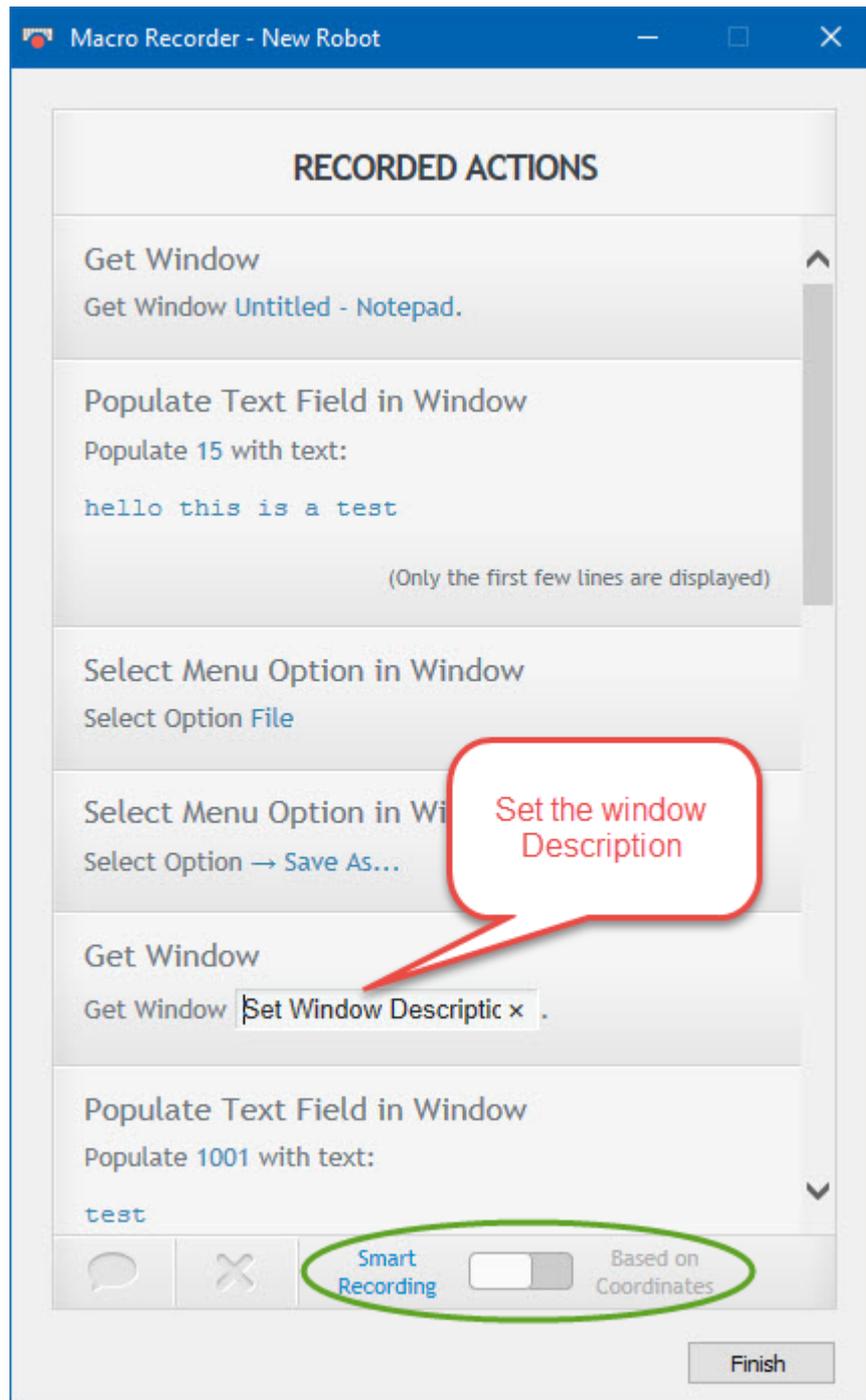
Coordinate Based: Coordinate based actions will move the mouse to the specific, absolute recorded locations inside the active window. Consequently, any change in the absolute positions of the target elements could potentially render the actions inoperable. Absolute positions depend on screen resolution, operating system and application settings as well as various other factors. This type of recording is only recommended for users that face compatibility issues with the default recording setting.

Each activity you perform will be recorded in the sidebar and once you have completed the desired sequence of activities and click the "Finish" button, you will be transferred back to the Process Designer where it will have automatically been translated into a set of UI Automation actions.

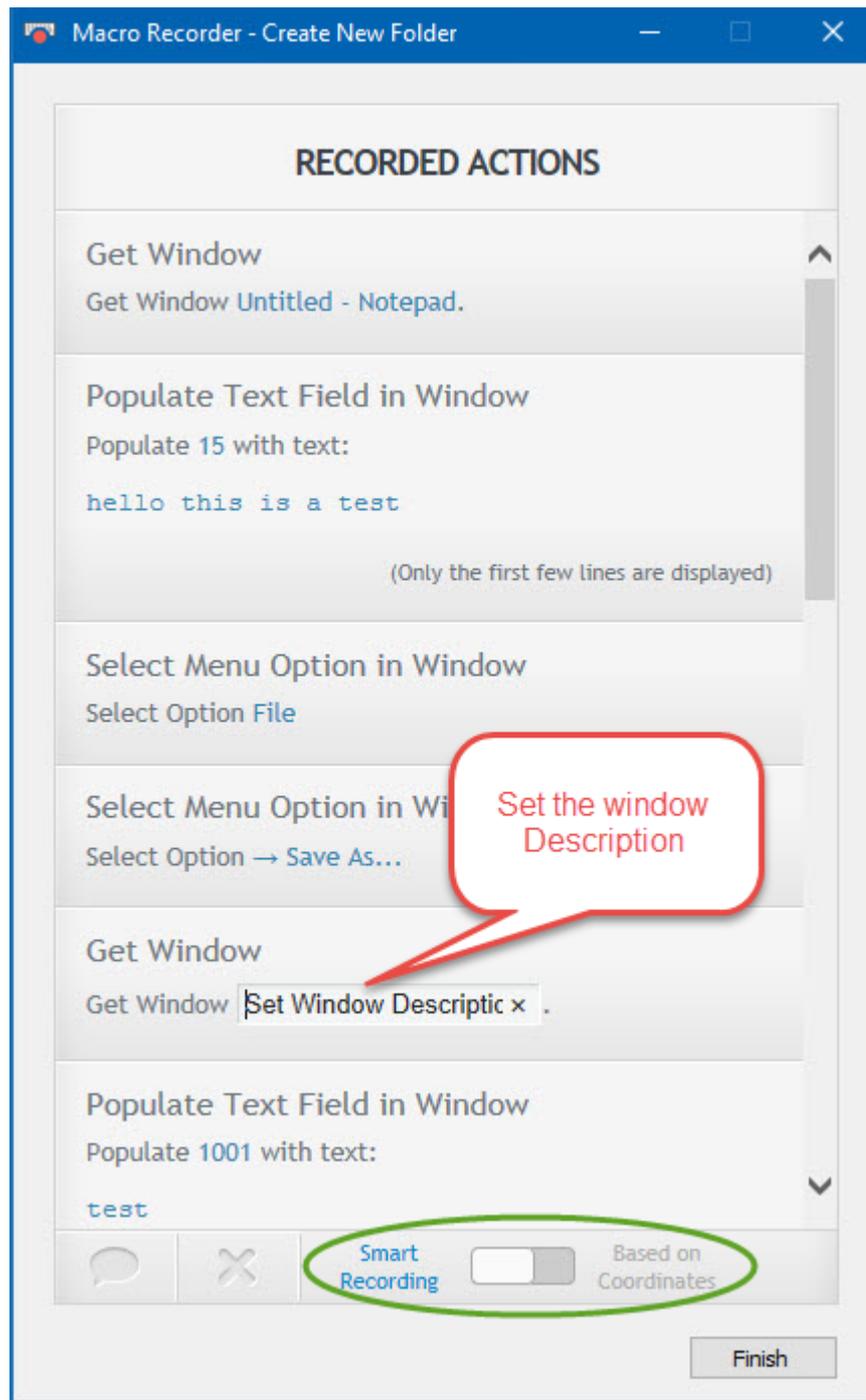
While recording, if you perform an activity unintentionally and you do not want it to be included in the final sequence of actions, you can move the mouse over the action that you want to delete in the sidebar and a red "✖" button will appear in the action's top right corner, as shown below. By clicking on the ✖ button the action will be removed from the list of recorded actions.

If you want to change parameters of the actions recorded (the actions' parts you can edit are displayed in blue) you can do so by clicking them and editing their text.

Note: Changing the description of an Element will not change the actual selector



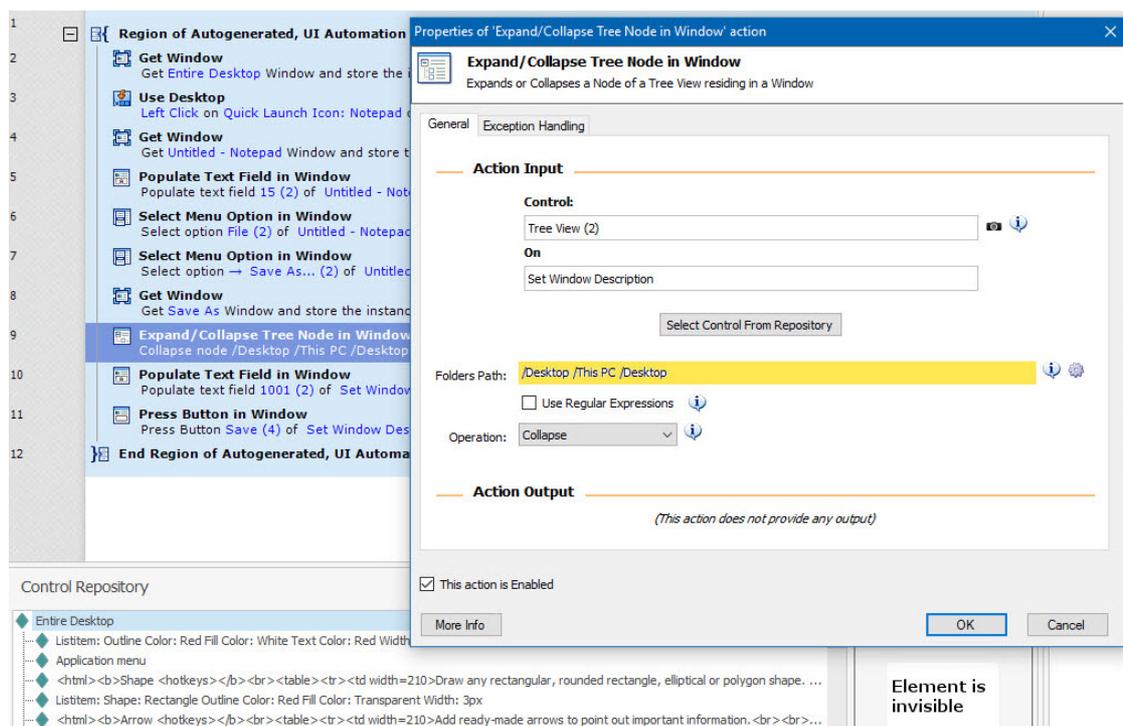
You can also take notes that will make the design of your Process more meaningful for you and that may be useful if you chose to edit the actions later in the Process Designer. To do so, you click on the "Insert Comment" button  and a comment action will be appended in the sidebar. Your comments will also be displayed (along with the corresponding action) in the Process Designer.



To discard all the actions you have recorded and restart your recording from scratch, press the "Reset Recorded Actions" button (which is located next to the "Insert Comment" button). A message box will appear for you to confirm the delete.

To complete the recording session and return to the Process Designer, click the Finish button on the sidebar. The recorded Actions will be inserted into the Process Designer Window within a region named "Region of Autogenerated, UI Automation Actions".

Of course you can further edit each of the actions (e.g. use variables as input where applicable) within the Process Designer.



You may also insert additional Actions to your Process to be performed before of after the Macro Recorded section Actions, either by manually selecting and configuring them from the "Actions" pane, or by using once again the Macro Recorder.

3.6 The Custom Dialog Designer

3.6.1 Designing a Custom Dialog

Dialog vs. Custom Dialog

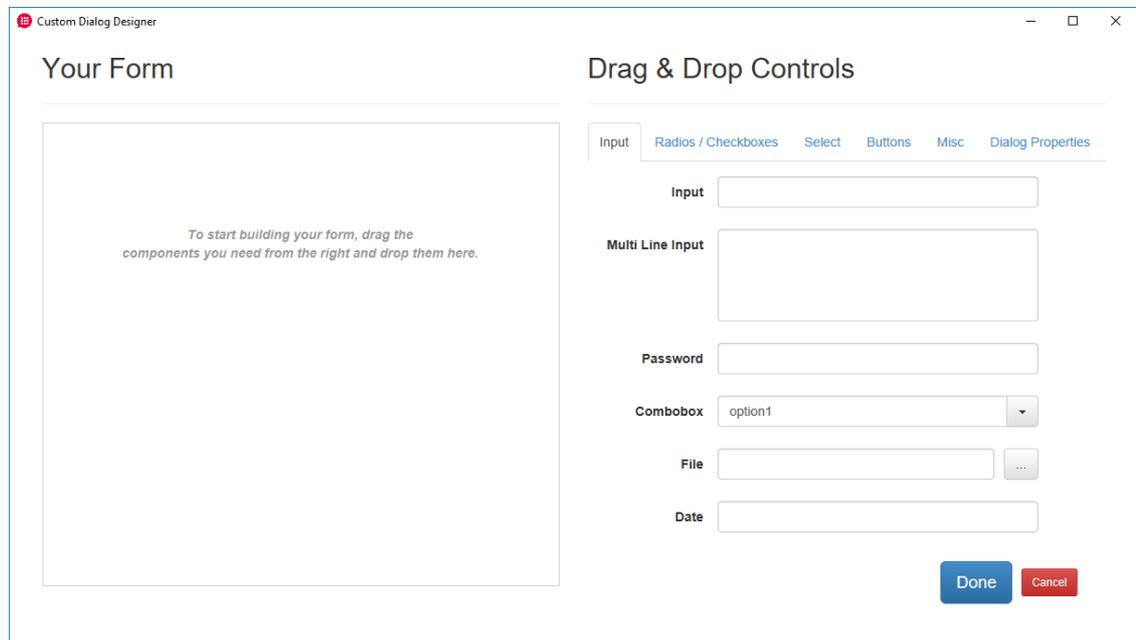
ProcessRobot offers a wide variety of dialog-producing actions, categorized under "Message Boxes" in the Actions' Pane. Selection dialogs allow you to assign a value to a variable (be it [Text](#)^[638], [Date](#)^[640], [List Selection](#)^[642], [File](#)^[645] or [Folder](#)^[647]) while the Process is running. Once the value is entered, the user can either press the "OK" or the "Cancel" button. This button choice can also be assigned to a variable (%ButtonPressed%). Simple, default dialogs come in handy when there is only *one configurable parameter* that needs to be set. For example, it is very common to include a "[Display Select File Dialog](#)^[645]" action at the beginning of a Process and use the value assigned to the %SelectedFile% throughout the remaining actions.

With a Custom Dialog on the other hand, you may *configure multiple parameters* in a single action. Custom Dialogs buttons are not limited to "OK"/"Cancel" and can also invoke [Functions](#)^[301], with the Dialog itself optionally persisting throughout the Function execution. Nearly every aspect of the Custom Dialog and its controls is configurable including, but not limited to, their general appearance, positioning and displayed text.

Given such flexibility, the "[Display Custom Dialog](#)^[636]" action will allow you create a fully featured Graphical User Interface (GUI) for your Process, with the ease of drag and drop.

The Custom Dialog Designer

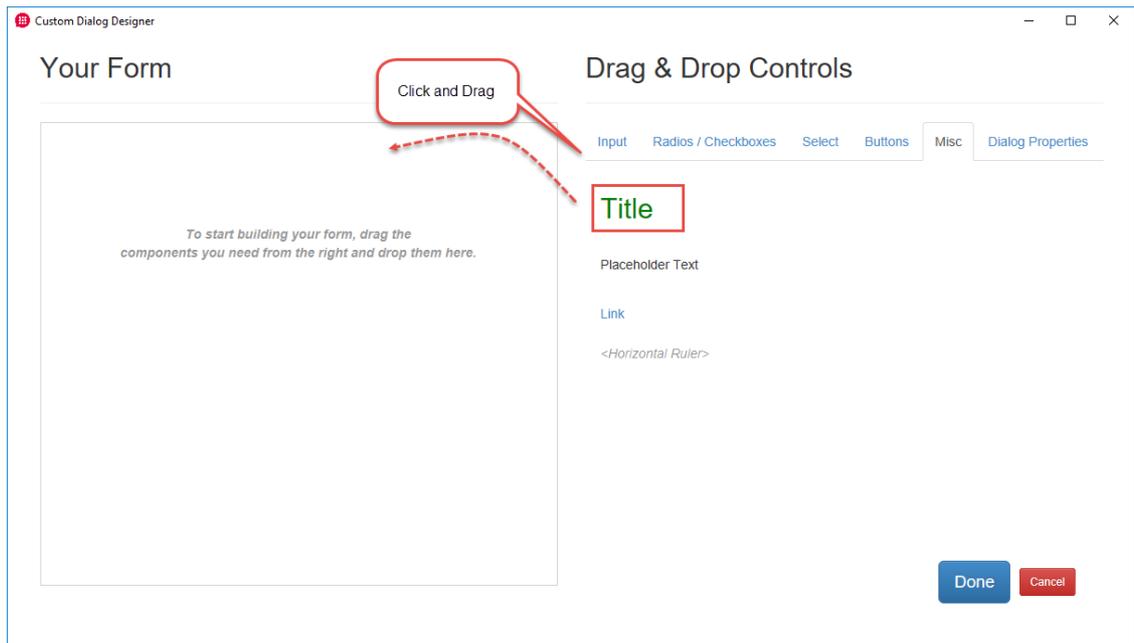
When you initially place a "[Display Custom Dialog](#)^[636]" action into your Process, the Custom Dialog Designer automatically opens up:



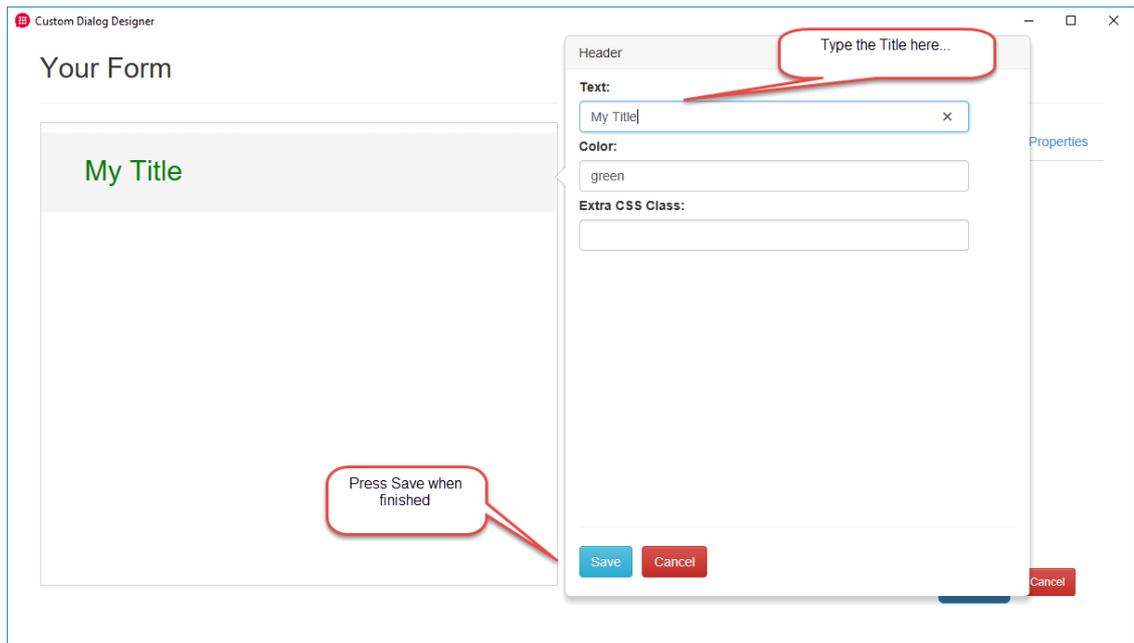
To bring it up again for any consequent modifications, double click on the action and press on the "Open Designer" button.

The Designer Window consists of two panes: **Your Form** on the left and the **Drag & Drop controls** on the right. You may select any of the tabs of the **Drag & Drop controls** pane, to reveal the controls of the corresponding category or edit the Dialog's properties. Controls can be added or rearranged by drag and drop.

For example, to add a "Title" control, select "Misc" and drag the Title from the pane on the right, to your form on the left:



When a control is added to the form, the control's property editor comes up:



With the help of the editor, you may modify the control's properties. As always, you can insert preexisting variables in any of the fields by clicking on the gear (⚙️) icon. When you are done editing, press the "Save" button. Saving also updates the live preview of the dialog presented on **Your Form**.

Your Form



If you need to modify any of the control's properties at a later point, left click on the placed control so that the property editor reappears.

Control Types

Input

Simple Input Holds a single-line text value, the most common form of input.

Multiline Input Holds a multi line text value, suitable for short messages or lists. The value returned is still a single Text variable containing line breaks, which can be [split](#) if necessary.

Password Similar to the Simple Input, but masks typed characters so that passwords do not appear on the screen as plaintext.

- Combobox* The Combobox allows the user to either make a selection from a pre-defined list or type their own, custom, value. The value returned is a list variable containing the text of the selected item(s).
- File* Lets you select one or multiple Files or a Folder. The value returned is a File/Folder variable for single items, or a File List Variable when you allow multiple item selection.
- Date* Holds and returns a Date Time variable. If no value is entered by the user, the current date and time is returned.

Radios / Checkboxes

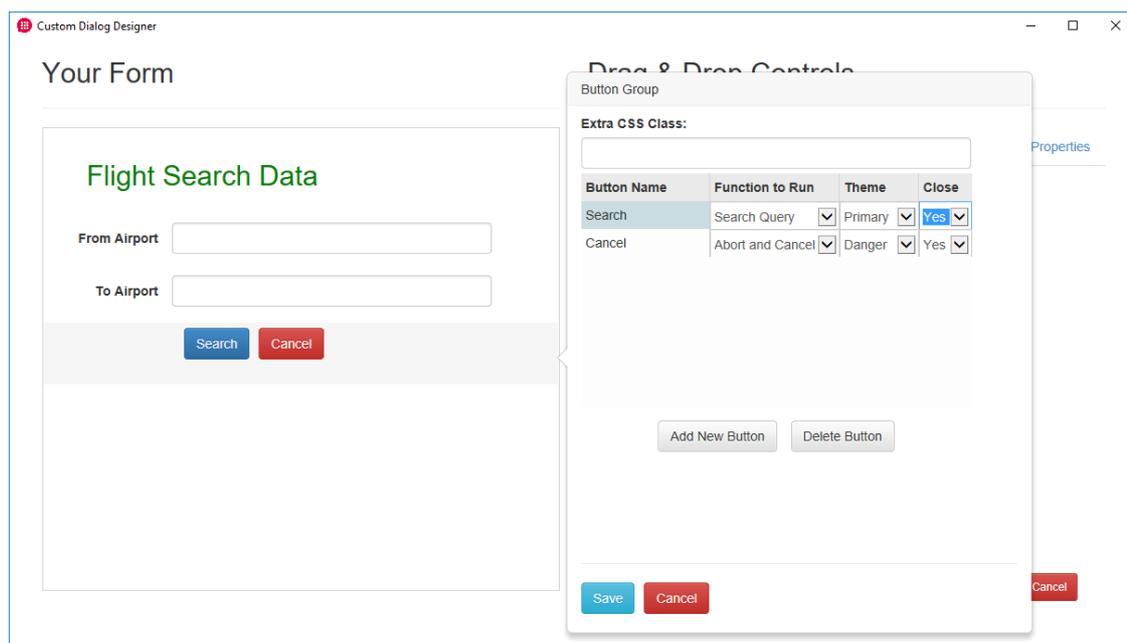
- Inline Radios* Select among a group of radio-buttons that are distributed horizontally. Only single item can be selected inside a group. The text/label of the selected item is the value returned.
- Multiline Radios* Select among a group of radio-buttons that are distributed vertically. Only single item can be selected inside a group. The text/label of the selected item is the value returned.
- Checkbox* A single checkbox that the user can leave checked, or uncheck. Returns a Boolean variable with the value %True% when the box is checked, %False% otherwise.

Select

- Select* Similar to the Combobox, but does not allow the insertion of custom values: the user can select only among the existing options.
- Multiple Select* Allows the selection of multiple items from the provided list. The value returned is a single Text variable containing the text/labels of the selected items, separated by linebreaks.

Button Groups

Button groups can contain one or more buttons, each one with its own name, style and assigned operation. You may modify the buttons belonging to a group by bringing up the property editor:



You will find more information regarding the configurable button properties, in the *Button Properties* subsection, below.

Miscellaneous

Title A Title control is a large text label, suitable for titling the entire form or sections of it. Does not return any value.

Placeholder Text Simple text control. You can use it for displaying any desired text at any part of the form. Does not return any value.

Link Adds a text control that when clicked, opens up the system's default browser and navigates to the specified linked URL. Does not return any value.

Horizontal Ruler Adds a horizontal dividing ruler, suitable for dividing controls into logical groups. Does not return any value.

Control Properties

General Properties

ID / Name Used for retrieving the value entered/selection made for this specific control. Must be unique and cannot contain spaces.

Label / Text The text displayed as label to the control or on the control itself.

Link Specific to Link controls - The URL of the target page for this Link control.

Enter Password Specific to Password Inputs - Allows the entry of unescaped '%' characters inside the default password property.

Default Value The value that the control will be prepopulated with, when the dialog is first shown.

<i>Default Password</i>	Specific to Password Inputs - Similarly to the default value, enter here the value that the Password control will hold when the Custom Dialog opens.
<i>File Selector Title</i>	Specific to File Inputs - The text to be used as title of the "Select File Dialog" that emerges when the "..." button of the control is pressed.
<i>Pick</i>	Specific to File Inputs - Specifies whether the file selection dialog will allow File(s) or Folder selection.
<i>Multiple Selection</i>	Specific to File Inputs - When selecting files, specify whether the user is allowed to select only one or multiple.
<i>Short Description</i>	A small, optional description that gets displayed the legend of the control.
<i>Popup Help Text</i>	The pop-up text to be displayed when the user hovers over the info icon on the right of the control.
<i>Size</i>	The size that the control should have. (Overrides CSS style)
<i>Extra CSS Class</i>	You can assign an extra CSS class on the control, which is useful for custom styling.

Button Properties

<i>Button Name</i>	Defines name of the button and also the text displayed on it.
<i>Function to Run</i>	Select a Function from the drop down list to assign it to the button. Once the User presses the button, the selected function will be invoked. You may have only one function executing per custom dialog at a time.
<i>Theme</i>	Select among various appearance styles for the button.
<i>Close</i>	Choose whether the custom dialog should close when the button gets pressed.

Whenever a button that has not be set as the "Cancel Button" gets pressed, all input gets validated against the following rules, where they may apply:

Validation

<i>Required</i>	Specify whether populating this input field is mandatory.
<i>Allowed Value</i>	Specify the accepted format of the user-supplied input value. You may choose one of the pre-defined formats, or define your own, custom one.
<i>File Filter</i>	Specific to File Inputs - The file filter applied to the "Select File Dialog" that emerges when the "..." button of the control is pressed.
<i>Check if File Exists</i>	Specific to File Inputs - Specify whether the selected file(s) are required to exist.
<i>Min/Max Date</i>	Specific to Date Inputs - Restricts the range of the date supplied.

Validation fails will prevent function execution and Custom Dialog termination. Moreover, a validation fail always produces an appropriate error message on the form.

Dialog Properties

<i>Title</i>	Specify the title to be used for the Custom Dialog.
<i>Initial State</i>	Choose among Minimized, Maximized and Normal states for the Custom Dialog's Window.
<i>Initial Position</i>	Specify an initial position for the Custom Dialog's Window. When "custom" is selected, you have to also specify the coordinates (X,Y)
<i>Height</i>	Specify the height of the Custom Dialog's Window.
<i>Width</i>	Specify the width of the Custom Dialog's Window.
<i>Accept Button</i>	Specify a button to be automatically pressed when the user presses Enter with the dialog open.
<i>Cancel Button</i>	Specify a button to be automatically pressed when the user presses Escape with the dialog open. Validation of input is skipped.
<i>Dialog Icon</i>	Specify a custom icon for the dialog. The icon is visible both on the task bar and the upper left corner of the Custom Dialog's Window.
<i>Keep on Top</i>	Specify whether the Custom Dialog's Window should remain on top of all other windows.
<i>Allow Resizing</i>	Allow/Disallow resizing of the Custom Dialog's Window.

Retrieving Entered Values from a Custom Dialog

The input controls of a Custom Dialog hold useful values, entered by the user in runtime. All of the populated values get stored inside a Data Row variable as defined in the "[Display Custom Dialog](#)^[636]" action's properties (default variable name: %CustomDialogResults%). Values are available to functions invoked by a button press on the Dialog and actions following the "[Display Custom Dialog](#)^[636]" action.

Input controls store their value on a "column" with the ID/Name of the corresponding control. For instance, the value populated into the "Combobox1" control can be accessed as such:

%CustomDialogResults["Combobox1"]% (Assuming that the default names were kept)

Some controls may hold a variable with [additional properties](#)^[439] or even a list. You may access the properties and the list items as well:

%CustomDialogResults["File"].Size% is the size of the selected file and **%CustomDialogResults["MultipleSelect"][0]%** is the first selected item.

Advanced Customization of the Dialog's appearance

You may add your own CSS rules applying to the Custom Dialog and its controls, in the Advanced Tab of the "Display Custom Dialog" action. Styling rules entered inside the "CSS Theming Overrides" box are effective immediately; you can preview them on the Custom Dialog Designer, or view them directly at the resulting Dialog that is produced during runtime.

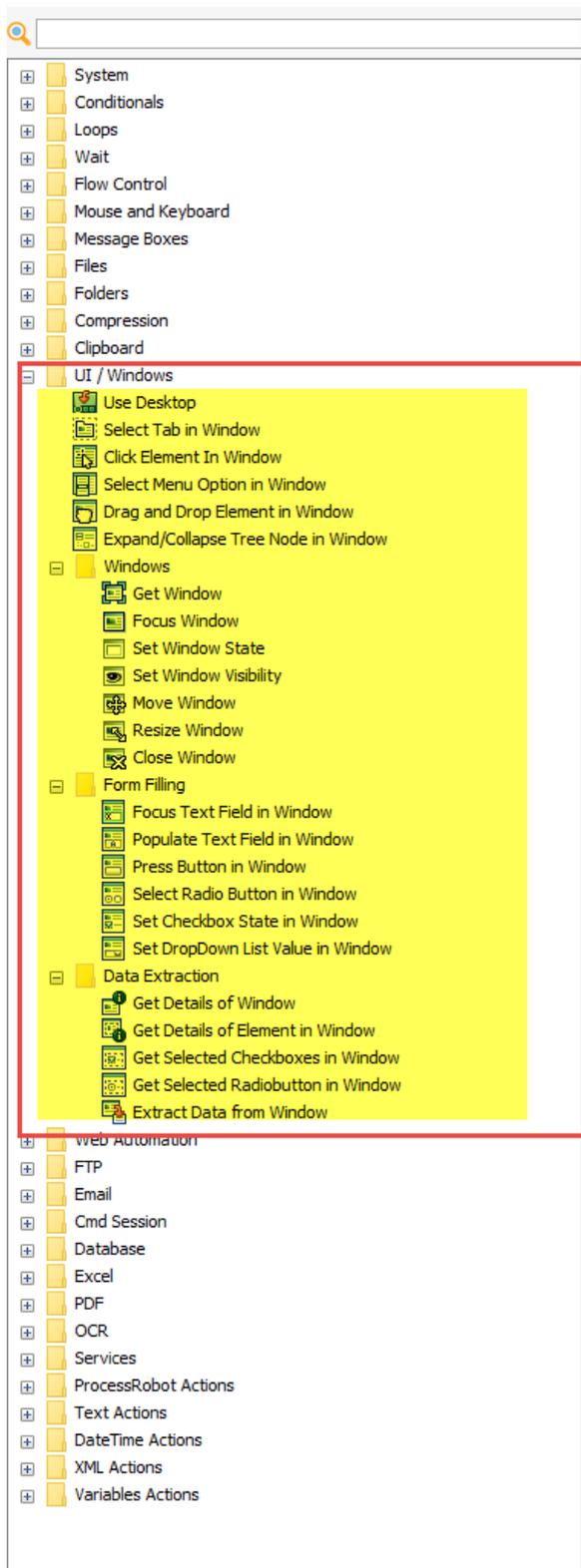
Each control can be assigned an "Extra CSS Class" through the corresponding property. A CSS class allows you to set a particular style for one or many HTML elements with that particular class in your custom CSS rules.

3.7 UI Automation

3.7.1 UI Automation Overview

ProcessRobot actions under the "UI/Windows" category are capable of directly manipulating Windows applications, or any window for that matter (apart from web application that can be automated with "Web Automation"), their controls, as well as extracting data from them.

Processes can click on buttons, menu items, populate edit boxes directly, etc. and all that without the use of absolute coordinates or image recognition!



Drag actions from the 'Actions' pane o

Controls Repository

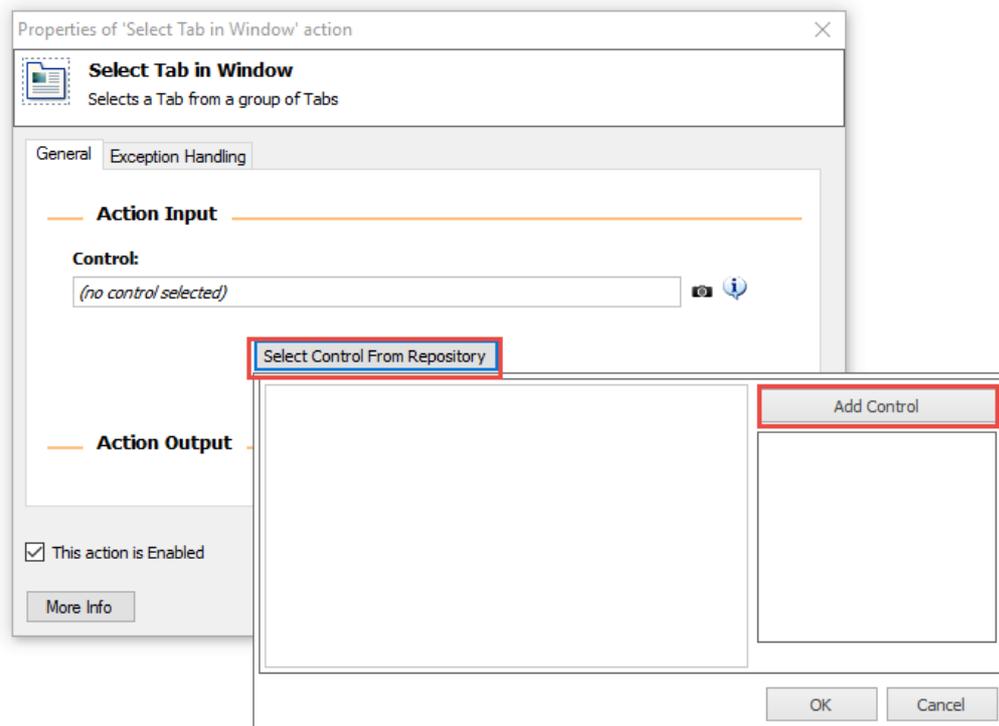
The UI Group of actions

UI Automation based Processes can be conveniently authored with the help of the [Macro Recorder](#)^[322] (with the Smart Recording option enabled). Alternatively, you can build your Process manually by combining and configuring the appropriate actions in the Process Designer.

Window handling actions, such as "[Focus Window](#)^[692]", "[Move Window](#)^[701]" etc. can both act on a specific window from your Control Repository.

Control handling actions, on the other hand (e.g. "[Click Element in Window](#)^[682]"), act on a target control of your repository.

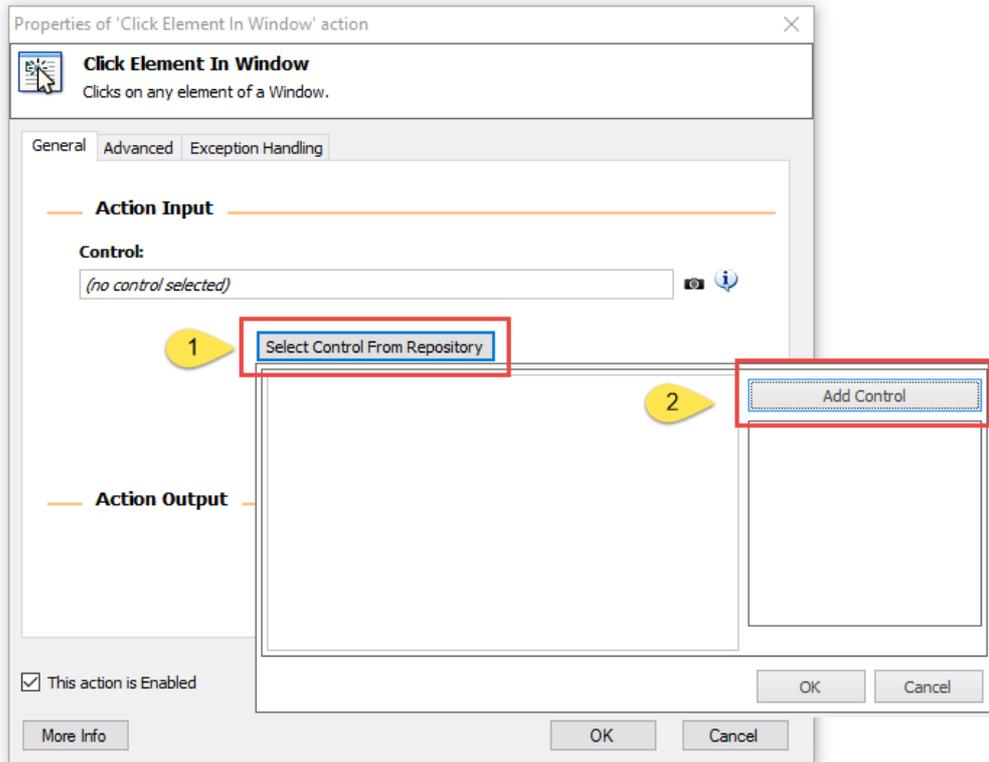
Using the "Select Control from Repository" in the UI/Windows actions' properties and adding Controls using the "Add Control" option the most convenient method for specifying the target control(s) for your actions.



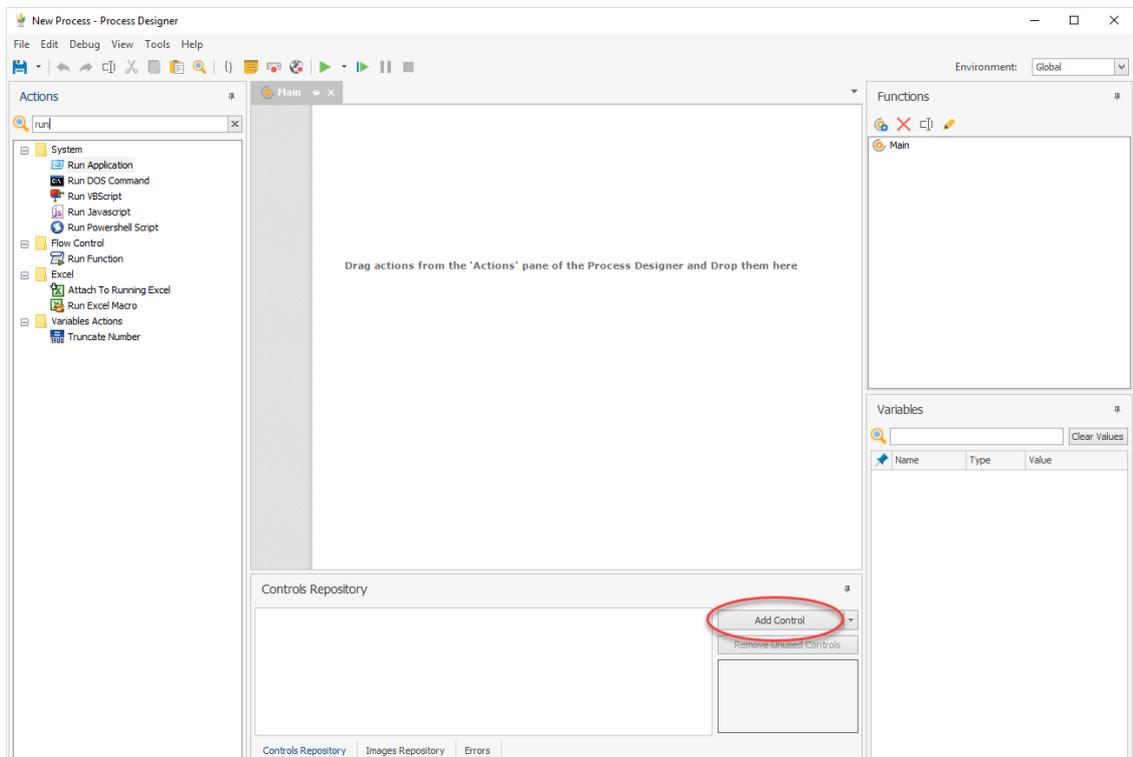
3.7.2 UI Elements - Select Controls

The "Select Control from Repository" or "Add Control" in the Control Repository pane, allows you to easily select a desired target control, on any currently open application window. To select a control simply click on the "Select Control from Repository" option in the action's properties. Having done so you have two options:

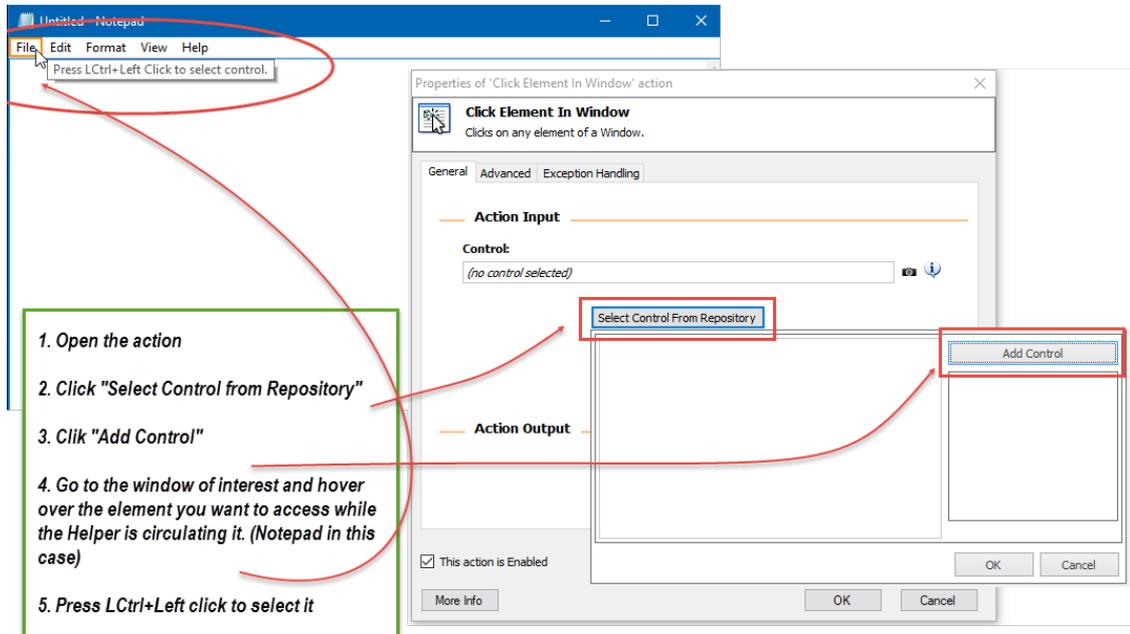
1. "Add Control":



OR...

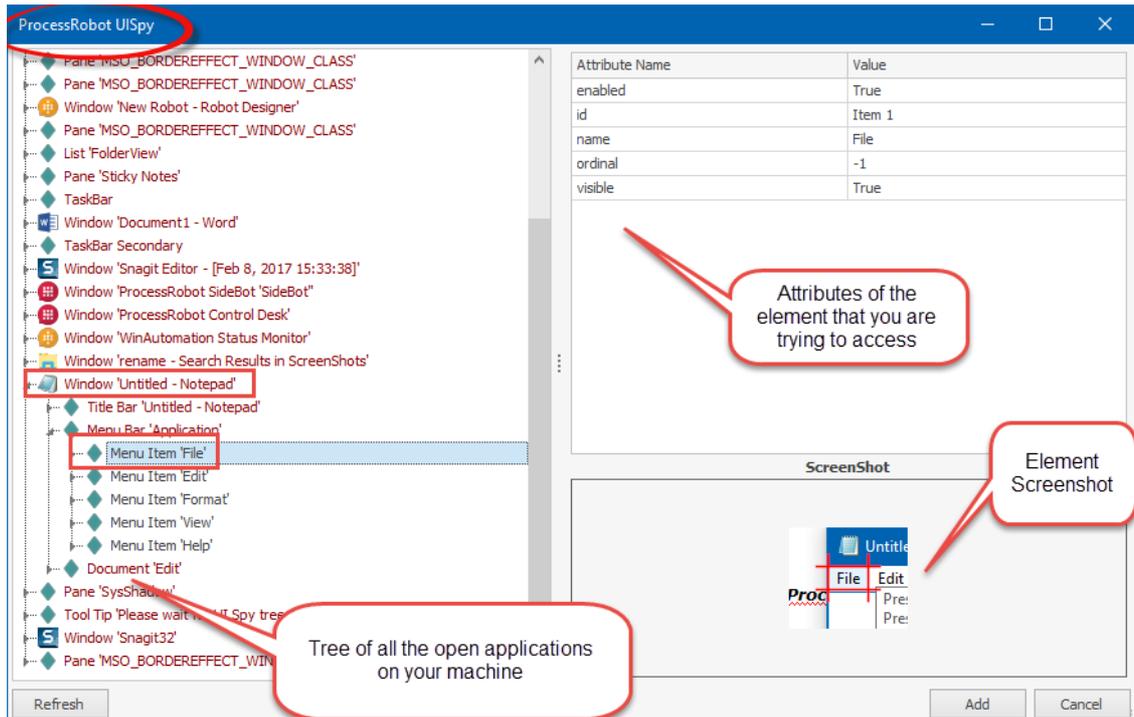


After selecting to Add a control with either ways mentioned above, you are able to select the element you wish. The Live Helpers (orange boxes circulating the accessible elements) will appear and you can navigate to the window of interest in order to access the element. Once this element is highlighted by the helpers, then simply press "LCtrl+LShift+Left Click" to select the control.



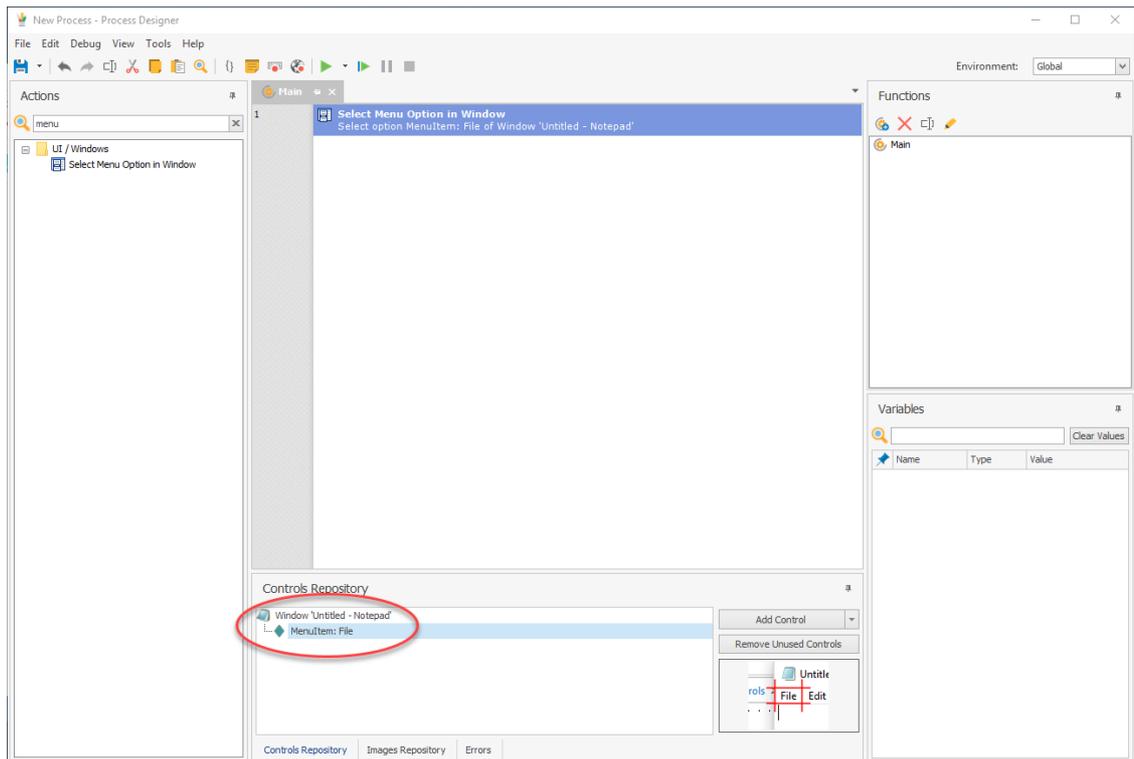
Having done so the **ProcessRobot UISpy** window will appear with three panes.

1. the tree of the all open applications and their controls, on your machine,
2. the attributes for the element that you are trying to access and
3. a screenshot of the element that you selected to add.



Click on the "Add" button to add the Control to your Repository. The Control Repository pane in your Process Designer will now have the Window and the element.

Should you wish the ProcessRobot UISpy not to be displayed and have the control directly passed into your repository, then instead of "LCTRL+LShift+Left Click" simply use "LCTRL+Left Click" (without the LShift).

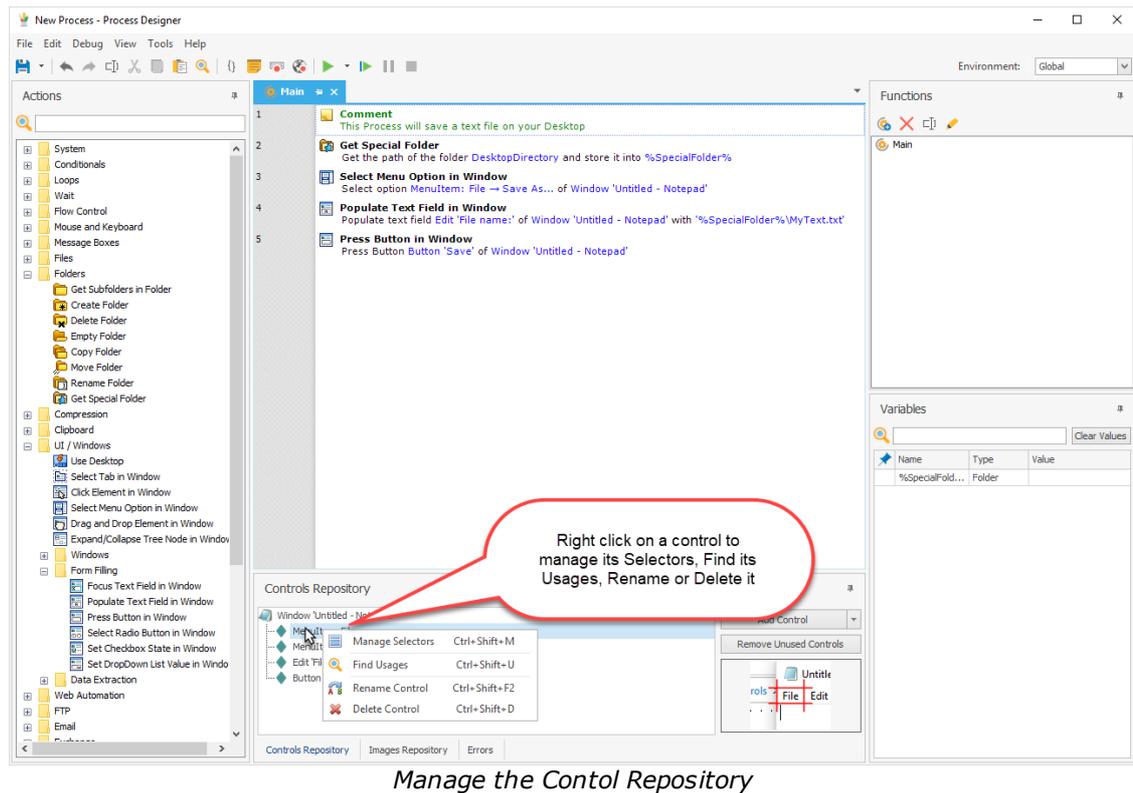


You can go on adding more Controls in your Repository, building your Processes with different actions until you have completed your Process.

3.7.3 Manage Control Repository

You can right click on an item in the Control Repository to:

1. **Manage** the Selectors that were generated from ProcessRobot to change them or make your own if you believe that they can be more efficient. In case you want you can also add variables in the selectors.
2. **Find** its **Usages** along the Process.
3. **Rename** the control. You may want to give a different name to the control.
4. **Delete** the control, in case you no longer wish to have it in your Repository.



To manage the selectors of a control, you can either double click on it, or right click and select "Manage Selectors". On the window that pops-up the selectors for the element will be listed. They are usually more than one and they work in the notion that if the first one fails to access the element, the Process will fall back to the second one. If the second one still fails to access the element the Process will fall back to the third one...and so forth. The action will error out only if all the selector fail.

3.7.4 Manage UI Selectors

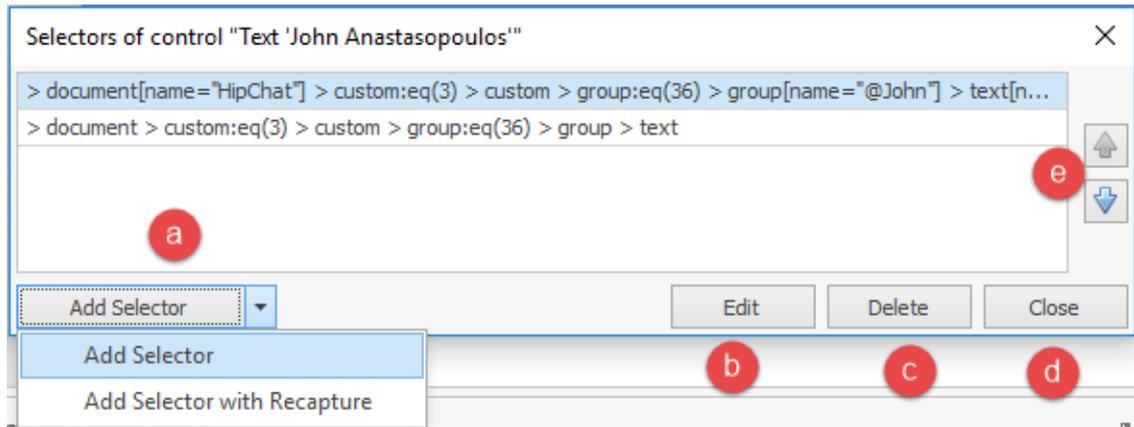
To manage the selectors of a control, you can either double click on it, or right click and select "Manage Selectors".

The same goes for CSS selectors for Web Automation actions, also.

In the window that pops up you are able to:

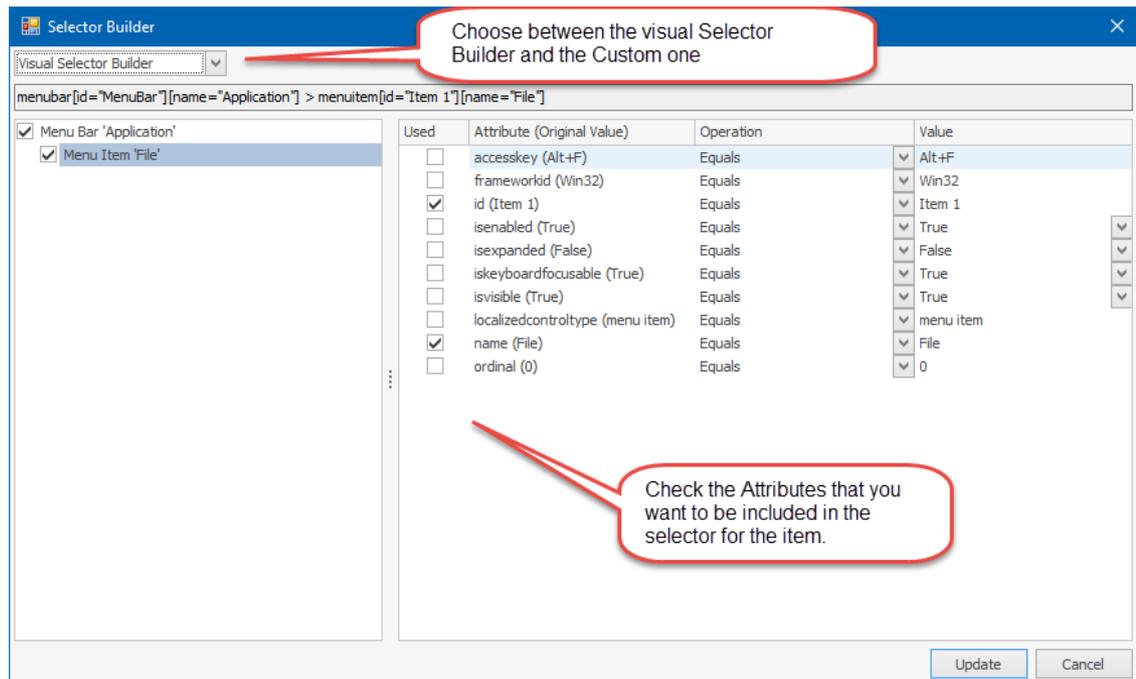
a) **Add Selector.** This button will give you two options.

- i) Add a selector based on the existing one, in the Selector Builder Window where you can check or uncheck the attributes to use for this element or
- ii) Add a selector by recapturing it, in which case the helpers will again appear so that you can LCtrl+Left click on the element.
- b) **Edit**. Click on the selector you want to edit, to pop up the Selector Builder window.
- c) **Delete**. Click on the selector that you want to delete.
- d) **Close**. Close the Window.
- e) **Reorder**. Reorder the sequence of the controls and move them up or down, based on the order that you wish them to attempt to access the control.



When you choose to Edit a Selector the the "Selector Builder" window pops-up. In here you can see the UI or CSS selector and you can choose the attributes that you want to be included in the selector for the item.

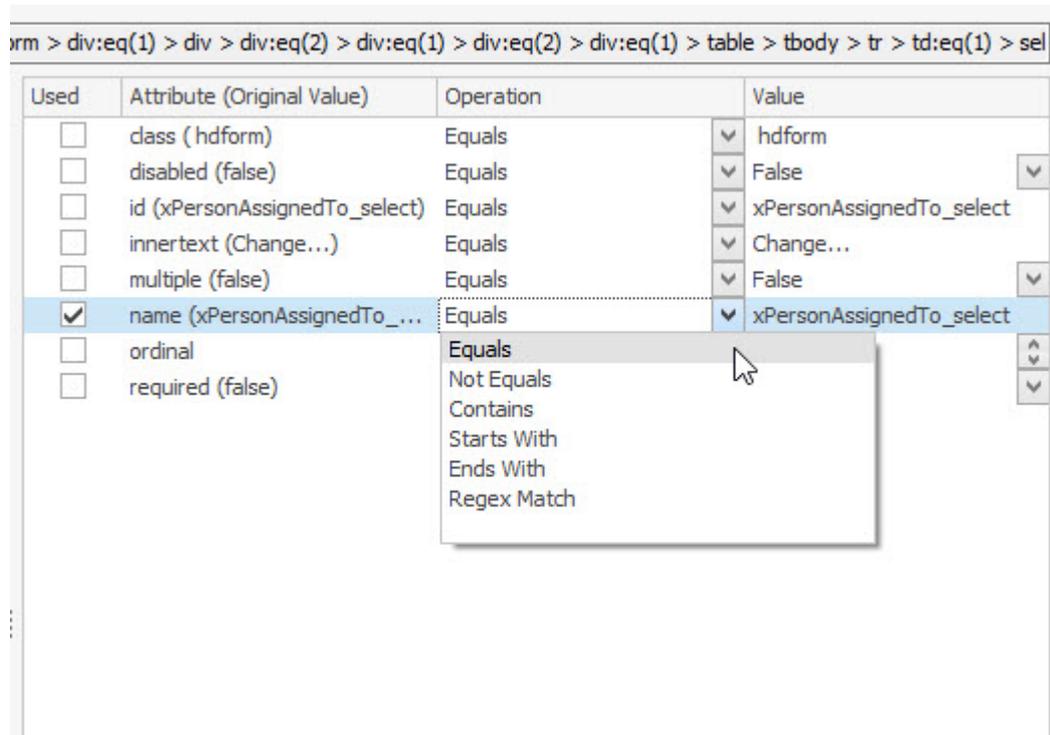
You can also choose whether you want the Visual Selector Builder or the Custom builder.



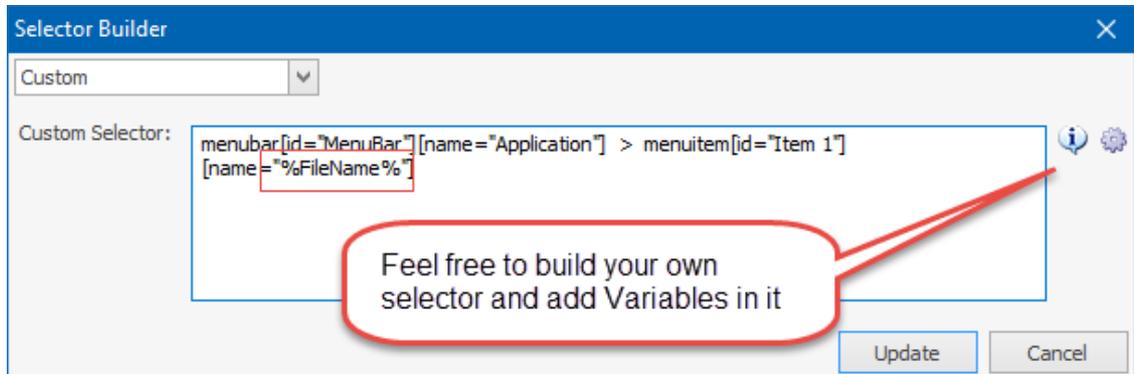
Selector Builder

Notice that in the "Operation" column, you have the option to select what how the value will relate to the checked attribute. While the "Value" column of the Selector Builder is editable, you can click on the Operation arrow and choose the relation between the attribute and the value to be:

- Equals
- Not Equals
- Contains
- Starts with
- Ends with
- Regex match



If you choose the Custom builder, from the drop down list menu option on the top left part of the window the the following window appears where you are free to build your own selector and even use Variables from the gear icon.



3.7.5 Building UI Selectors

The UI selectors are very useful when using UI/Windows actions, as they help you access any windows/elements you want.

UI selectors are generated for each control you add in your Control Repository and can be shown should you double click on a Control.

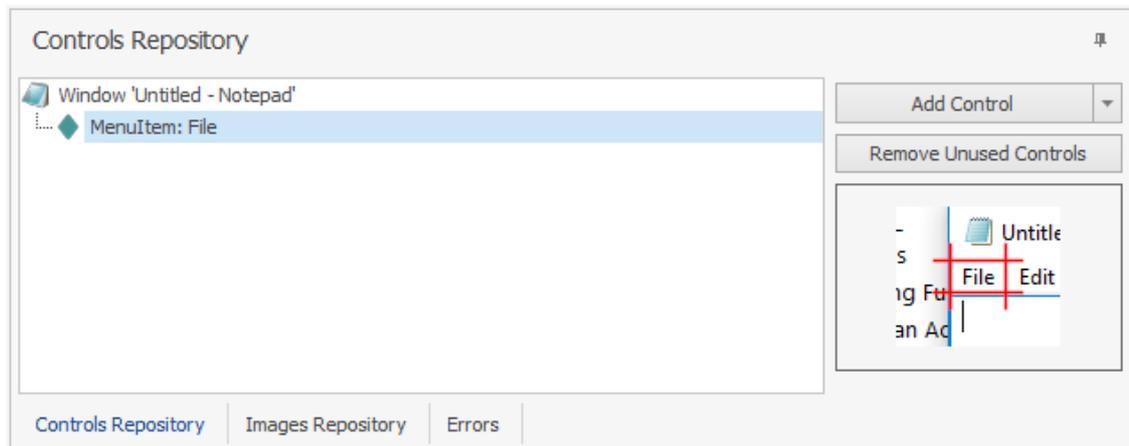
ProcessRobot automatically generates one or more selectors. This means that, if the first selector fails to access the element, then we fall back to the second selector; and in case of a new failure we fall back to the third and so on. The action will error out, if none of the selectors are able to get the element and there is no Exception handling in the relative tab of the action.

You do have the option to modify the existing generated selectors (delete one or two of them, enhance them, or build them yourself from scratch) so that you can get the desired window or element.

Let's say that you have an "Untitled – Notepad" window open on your desktop which you wish to get.

Selectors are read from left to right using ">" to show that the element that follows is contained within the previous one in a "parent > child" notion. Let's say that you want to access a Notepad Window and then click on the File Menu option.

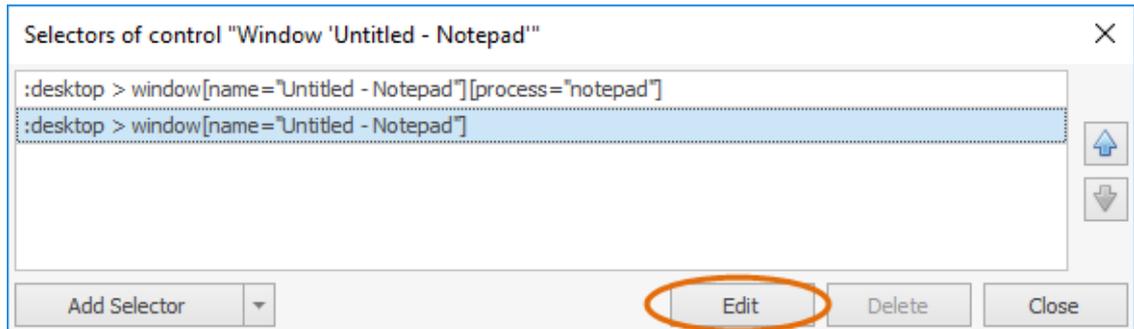
You would have to use a "Click Element in Window" action and add the Control to the Repository. You will end up with the control as follows:



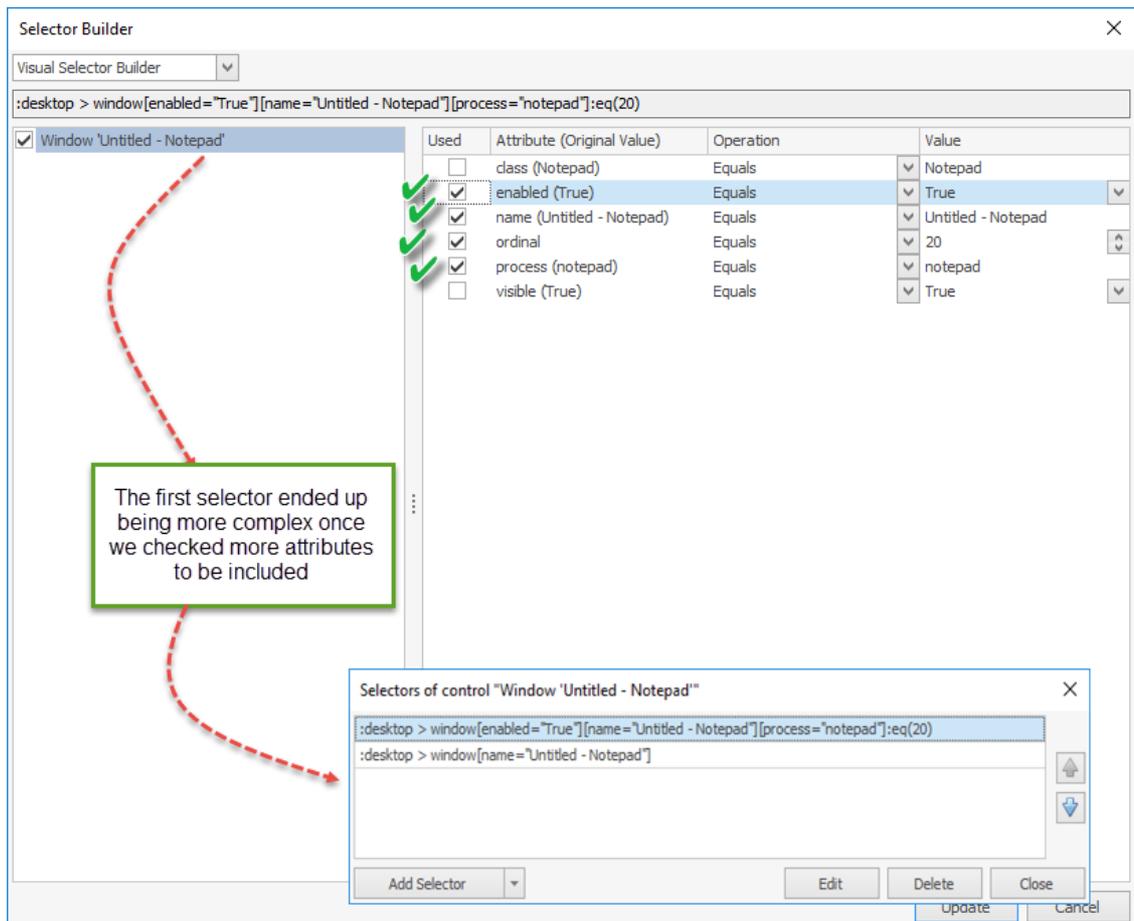
The Window '**Untitled-Notepad**' will have its selectors and once accessed the Process will move on to the Menu Item:File for this window.

What if you want to get a text document with another specific name??

Double click on the Window 'Untitled-Notepad' and in the window that pops up hit the Edit button for each selector.



In the selector builder Window that will appear you can check other attributes that you want to include in the existing Selector for the window. Thus, the selector will become more complex.



OR, you can choose to build a Custom Selector from the drop down list menu option on the top left part of the Selector Builder window.

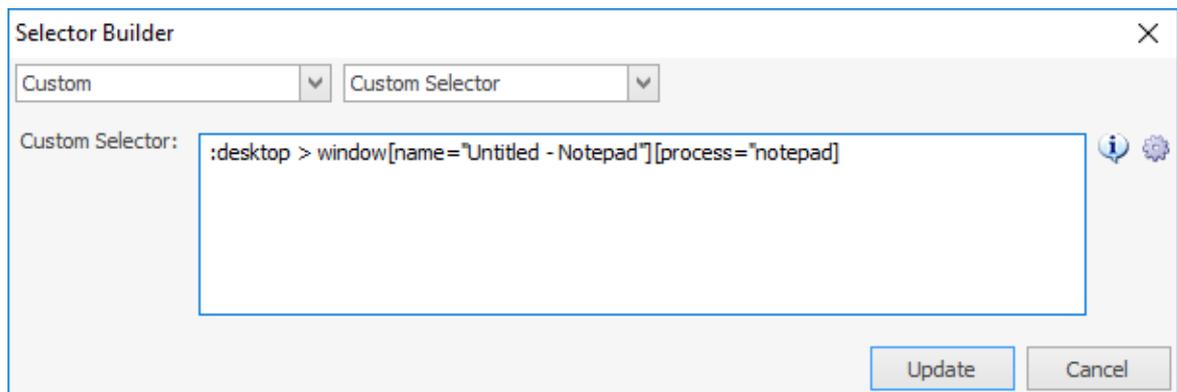
In general the selector for a specific window with "n" attributes is:

```
:desktop > window[Attribute1="Attribute1Name"][Attribute2="Attribute2Name"]...  
[Attributen="AttributenName"]
```

So, for the notepad with title "Untitled" the selector can be:

```
:desktop > window[process="Notepad"][name="Untitled - Notepad"][class="Notepad"]
```

...you can delete one or more attributes and you can just keep the ones that you think will make your selector more efficient, in the Selector Builder Window.



- **Using Variables in Selectors:**

If you have a variable that holds the name of the text file then in the Custom Selector you can insert that variable. Let's say that you are certain that the file you want to access is named "Example.txt".

The selector to access that window can simply be:

```
:desktop > window[process="Notepad"][name="Example"][class="Notepad"]
```

or

```
:desktop > window[name="Example"][process="Notepad"]
```

Further on, if you have the name of the file in a variable like %FileName%=Example then the selector can be:

```
:desktop > window[process="Notepad"][name="%FileName%"][class="Notepad"]
```

- **Useful Tips for more efficient Selectors:**

```
:desktop > window[name^="Untitled"][process="notepad"]
```

would get any Notepad window that its title starts with "Untitled".

```
:desktop > window[name$="Notepad"][process="notepad"]
```

would get any Notepad window that its title ends with Notepad.

```
:desktop > window[name~="Untitled"][process="notepad"]
```

would get any Notepad window that contains the word Untitled in its title.

NOTE:

:eq(n) is a very frequently used selector attribute. It selects the element at index "n" within the matched set (keep in mind that it always zero based). If the index is negative then the $(|n| - 1)$ element is picked counting from the last element to the first.

For example if you have several buttons in a window and you write something like:

... > button:eq(0) > ... this means that you are looking for the first button, while **button:eq(1)** would be the second button and **button:eq(2)** would be the third one and so on.

3.7.6 UI Data Extraction

ProcessRobot's UI Automation technology allows the extraction of textual data from any open application window. You will find all the relevant actions under the "Data Extraction" subcategory of the "UI and Windows" group, inside the Action's Pane.

"[Get Details of Window](#)^[717]" and "[Get Details of Element in Window](#)^[719]", are suitable for extracting certain useful attributes from the target Window or Element, such as the Window's Title or the Element's location. Both of the actions are also capable of getting the text displayed by the element, including any text belonging to visible child elements.

The following two actions, "[Get Selected Checkboxes in Window](#)^[721]" and "[Get Selected Radiobutton in Window](#)^[722]" are specialized actions, dedicated in extracting the current selection state of Checkboxes and Radiobuttons.

Last but not least, "[Extract Data from Window](#)^[723]" extracts *only the text displayed on elements* (and their visible children). For simple control elements, "[Extract Data from Window](#)^[723]" will get the exact same results as the equivalent "[Get Details of Element in Window](#)^[719]" action. Certain controls, on the other hand, will yield data in the form of List or Table variables. More specifically, Tree and

Tree Item controls (e.g. the tree view pane of Windows Explorer) as well as List controls produce the text data in a List variable. Data Grid controls always get their data extracted in the form of a table. In other words, "[Extract Data from Window](#)^[723]" allows the retention of the original organization of the data extracted, making any further processing significantly easier.

3.8 Web Automation

3.8.1 Web Automation Overview

With ProcessRobot you are able to automate your web related tasks in an intuitive manner with unprecedented speed and reliability.

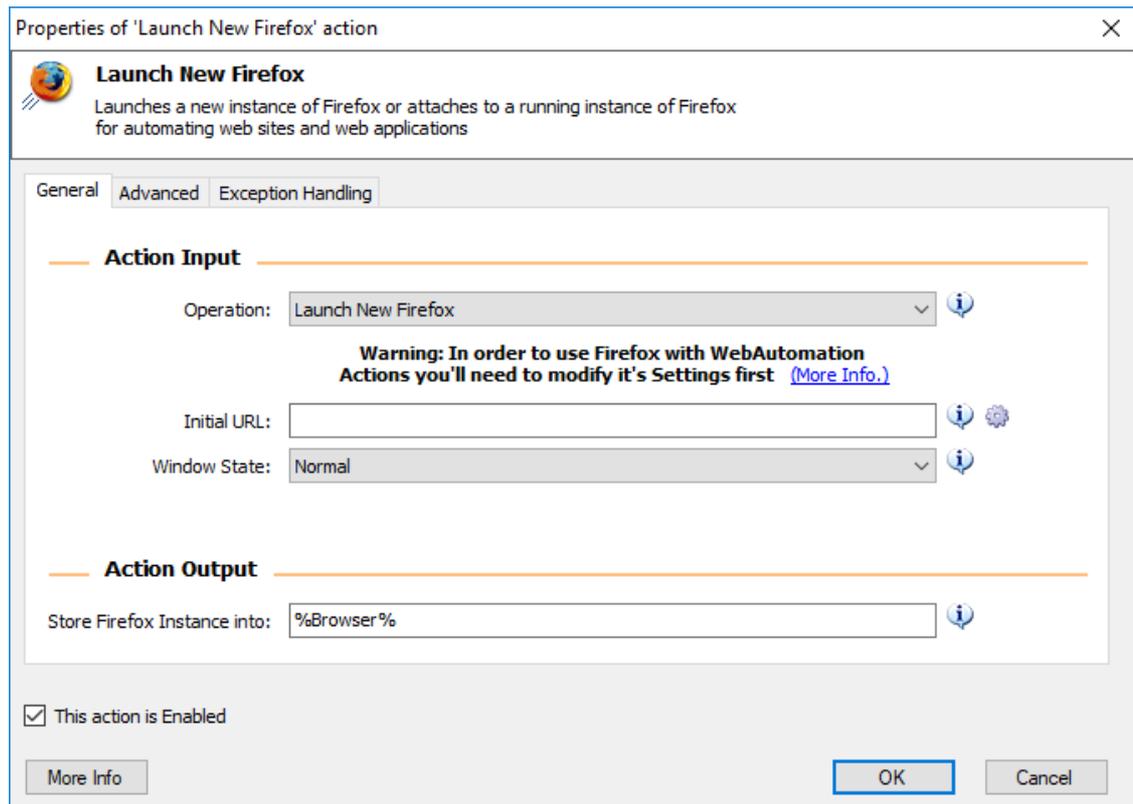
The Process Designer includes a number of actions under the "Web Automation" category, each of them corresponding to an action that a user would manually perform within a web browser.

The usual sequence for creating a Web Automation Process is the following:

1. You launch a browser window (or attach to an existing one) using one of the following three actions: "[Launch New Internet Explorer](#)^[726]", "[Launch New Chrome](#)^[734]" or "[Launch New Firefox](#)^[729]". These actions generate a variable containing the instance of the web browser that needs to be passed to subsequent actions (this is useful when you need to automate more than one browsers concurrently).

By default a Web Browser instance is stored in %Browser%, but you can change the name of the variable to the most convenient name for you. Do keep in mind that any web actions on this specific browser that you opened with these actions, should respond to this instance.

For example if you wish to use the "Click Link on Web Page" action, in the Web Browser Instance you should indicate the variable %Browser% (or the variable's name that you gave to the instance in any of the Launch new Browser actions).



Launch New Firefox, one of the three Web Browsers you may use in your automations with Process Robot!

2. You add the appropriate actions based on the steps that you want to automate. There are actions available for clicking on links, filling and submitting web forms, extracting data from web pages, etc. In each action you will need to pass the variable containing the web browser instance (for example %Browser%) to specify the browser on which the action will be performed, as previously mentioned in [1].
3. Optionally, you may close the browser used for the automation using the "[Close Web Browser](#)"^[746] action.

In order to build your Process you have plenty of tools at your disposal. The easiest way to start is to use the [Web Recorder](#)^[360] that gives you the ability to record your web activities as you perform them and convert them into a Process.

Alternatively, you can build your Process manually by combining (and configuring) the appropriate actions in Process Designer. The rest of the overview will focus on the Process Designer approach, for Web Recorder usage please refer to the [corresponding topic](#)^[360].

3.8.2 Building a WebAutomation Process

Building a Web Automation Process in the Process Designer:

As mentioned before, you will start with any of the Launch Web Browser actions ("[Launch New Internet Explorer](#)^[726]", "[Launch New Chrome](#)^[734]" or "[Launch New Firefox](#)^[729]") and specify the initial web address to visit:

In addition to the initial URL, you may also specify:

- The type of Internet browser you wish to work with, by choosing among launching the Automation Browser (if in IE), a new Web Browser, or attaching to an already running Browser instance.

The Automation Browser may be preferred as it offers a minimal and fully-functional browser interface, while allowing you to avoid inconsistencies and delays that Web Browser add-ons, toolbars and widgets may cause. (More information you can find in the [Actual Internet Explorer VS Automated Browser](#)^[393] topic)

- Whether the web browser window will be launched in normal, maximized or minimized (in case you want the Process to run in the background) state.

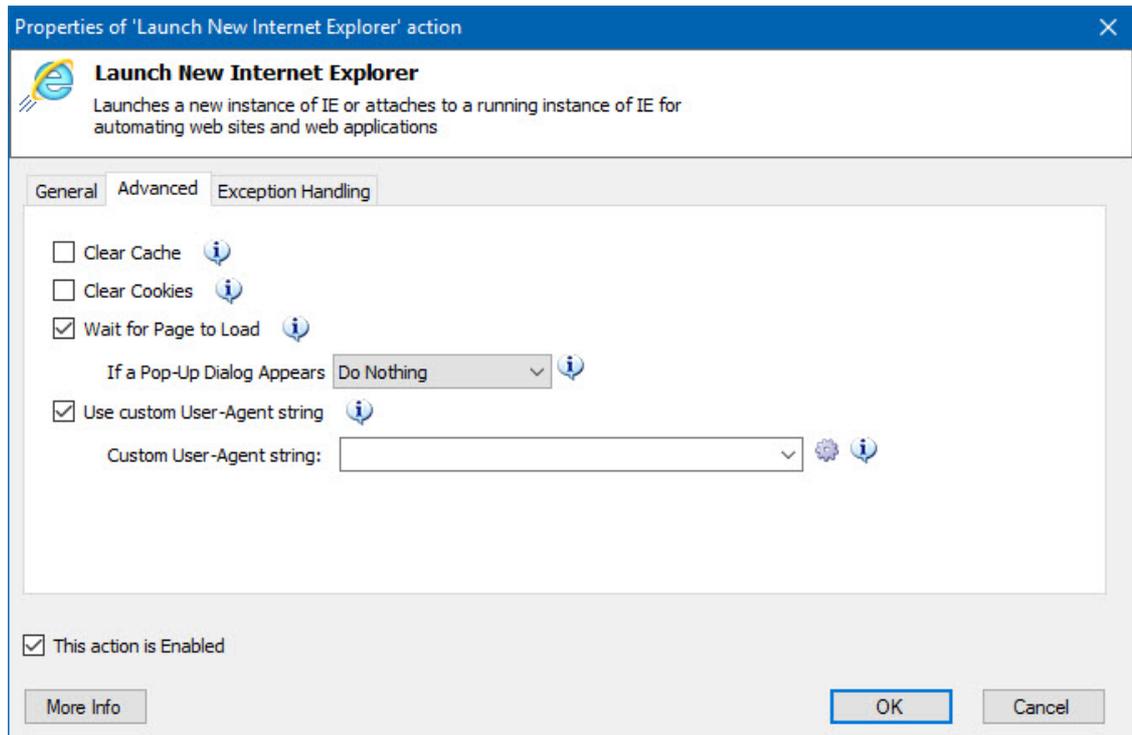
The main outcome of Launch New Web Browser actions is to generate a variable containing the instance of the web browser that will be passed as input to the rest web-related actions of your script. Since it is possible that within the same Process you may need to automate more than one browsers concurrently (if for example you may want to extract data from one web site and populate some fields of another), you can include multiple "Launch New Web Browser" actions and store the generated Browser Instances appropriately, so that you can pass them to the respective subsequent web-automation actions.

The most commonly selected option is to Launch a new Web Browser. However, by default, IE comes with security settings disallowing any external applications (ProcessRobot in our case) from controlling it and Firefox and Chrome come with other settings that needs to be changed for Process Robot to work as expected. Therefore, it is essential to configure IE security [settings](#)^[383] first if you plan to work with IE or change the settings of [Firefox and/or Chrome](#)^[367] if you wish to work with them.

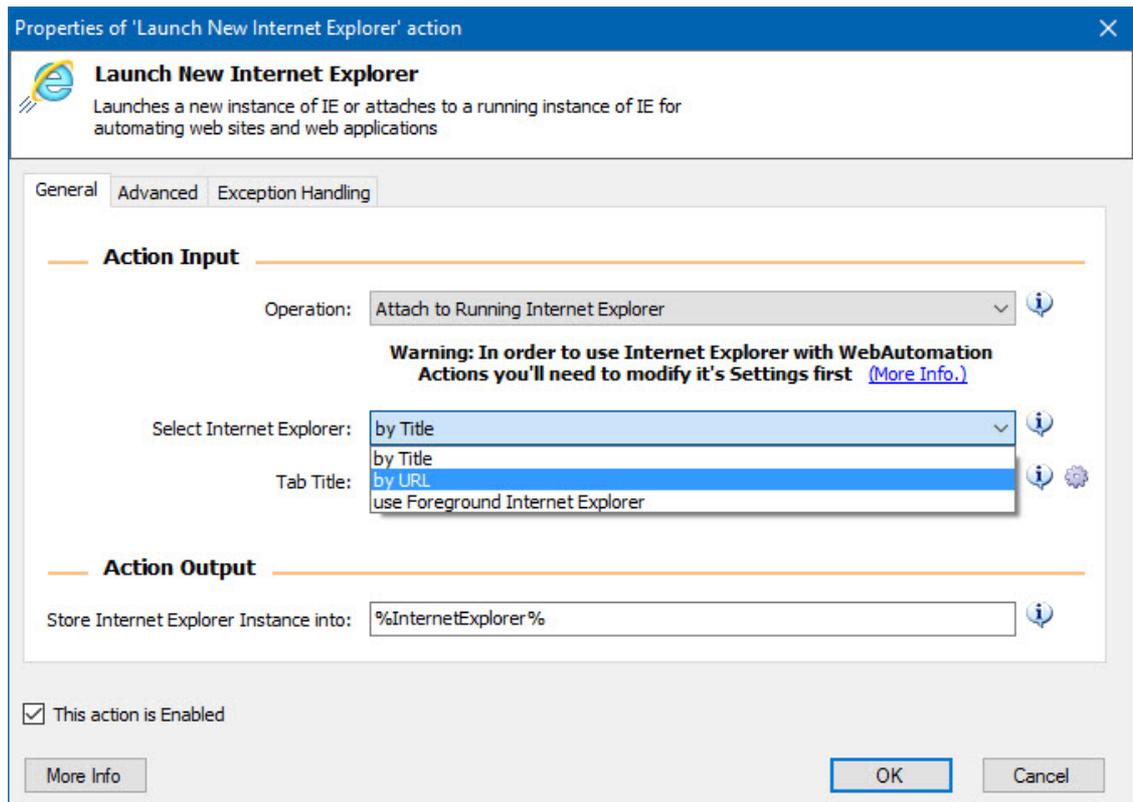
Alternatively, you can use the ProcessRobot's browser that does not require any configuration changes.

When selecting to launch a new browser, you can specify some further options, found in the **Advanced** tab of Properties of Launch New Web Browser action window:

- Whether to clear the browser's cache and/or any stored cookies right after launching it (depending on the size of cache or stored cookies this procedure may take a while).
- Whether you need the Process to wait for the new web page to load completely before proceeding to the next the action.
- How should the action respond if a pop-up dialog appears while loading the initial web page (i.e. close, ignore or interact with it).
- Specify the user-agent string that you want the runtime web helper to advertise to the website's that it is instructed to navigate to (only IE).

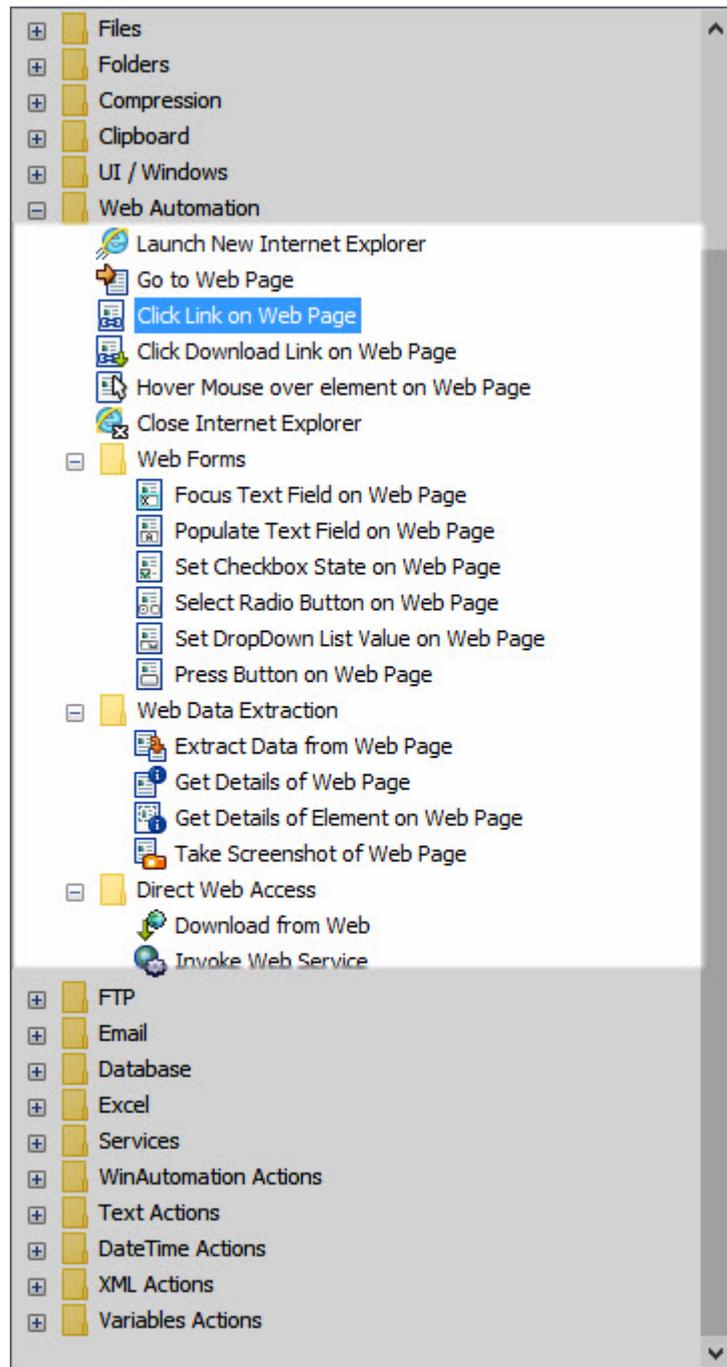


If you set the action to attach to a running Internet Explorer (instead of launching a new one) you are prompted to choose whether it will attach to an Internet Explorer Tab based on its Title, URL, or just pick the Active Tab of the Internet Explorer running as the Foreground Window.



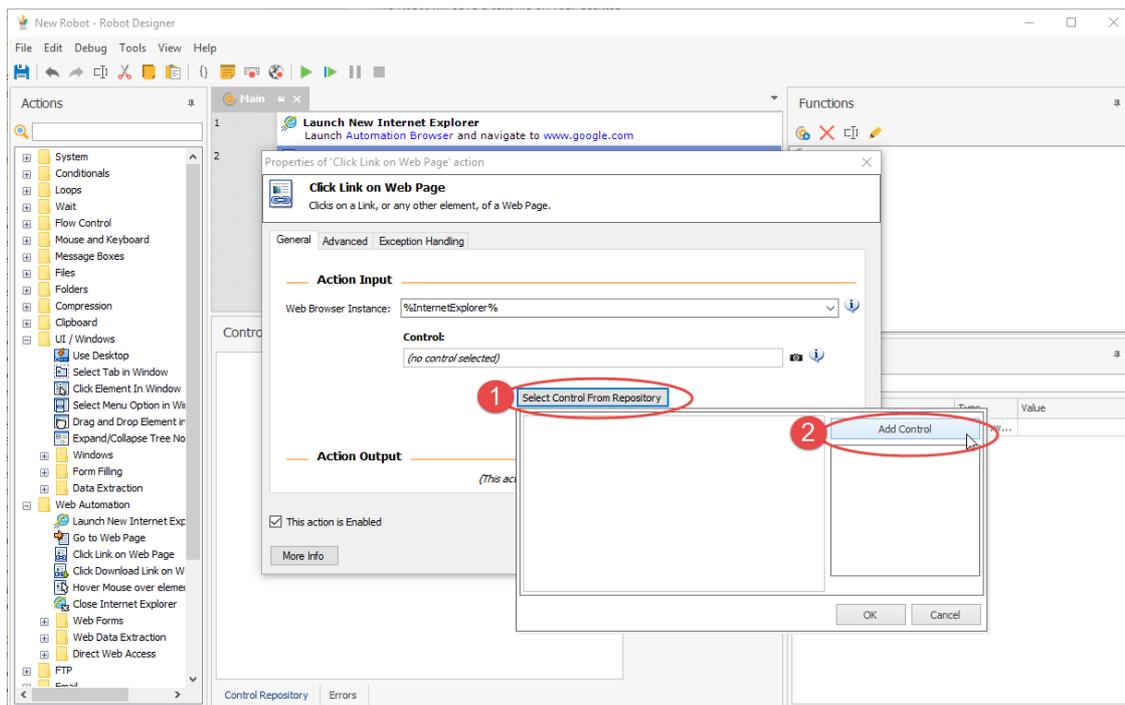
Next, you will append other actions to interact with elements in the web page.

There are actions available for clicking on links, filling and submitting web forms, extracting data from web pages, etc.



Although each action has its specific features there is an overall mode in setting the web-related actions' properties.

Say, for example that you need to click on a link. For that you will need to use the "[Click Link on Web Page](#)^[743]" action:



As you can see in the screenshot above, you will first need to select the Browser Instance on which the action will be performed. The next step is to specify the link you want to click on and add it in your Control Repository. You can do this in two ways:

1. Select the Control from your Repository, if it is already there from a previous selection, or
2. Click "Select Control Repository" and then click "Add Control". This will open the Live Helpers. Hover your mouse over the element that you want to access in the Internet Explorer and hit Left+LeftClick to select it.

Once you do the **ProcessRobot UISpy** window will pop up, that it will show you the HTML tree of the element as well as its attributes in the right pane of the window.

The screenshot shows a web browser displaying the Google homepage. Below the browser, the WinAutomation UI Spy tool is open. The tool's interface is divided into three main sections:

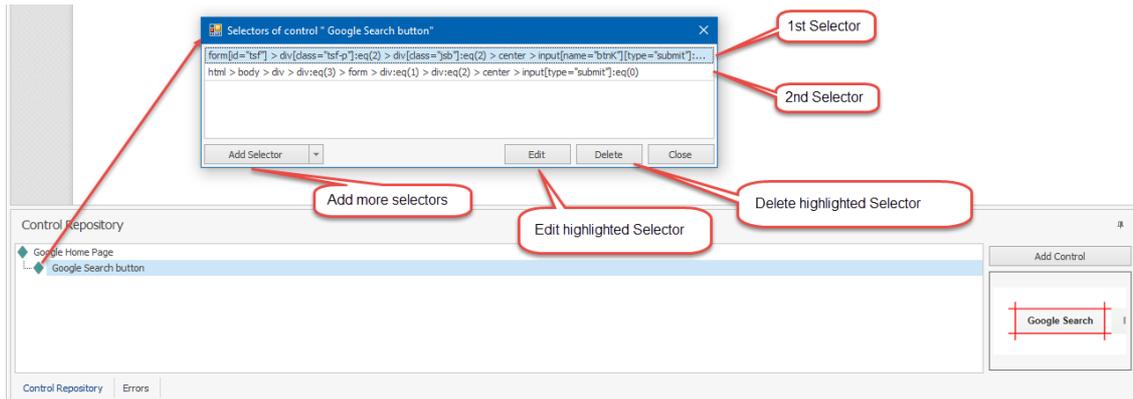
- DOM Tree (Left):** A hierarchical tree view of the page's HTML elements. The element `<input:submit> 'Google Search'` is selected and highlighted with a red box.
- Attributes (Right):** A table listing the attributes of the selected element. The table has two columns: "Attribute Name" and "Value".
- ScreenShot (Bottom Right):** A small screenshot of the selected button, with a red box around the text "Google Search".

A red arrow originates from the selected element in the DOM tree and points to the "Add" button at the bottom of the UI Spy tool. The "Add" button is used to save the selected control to the WinAutomation Control Repository.

Click on "Add" to add this control to your Repository.

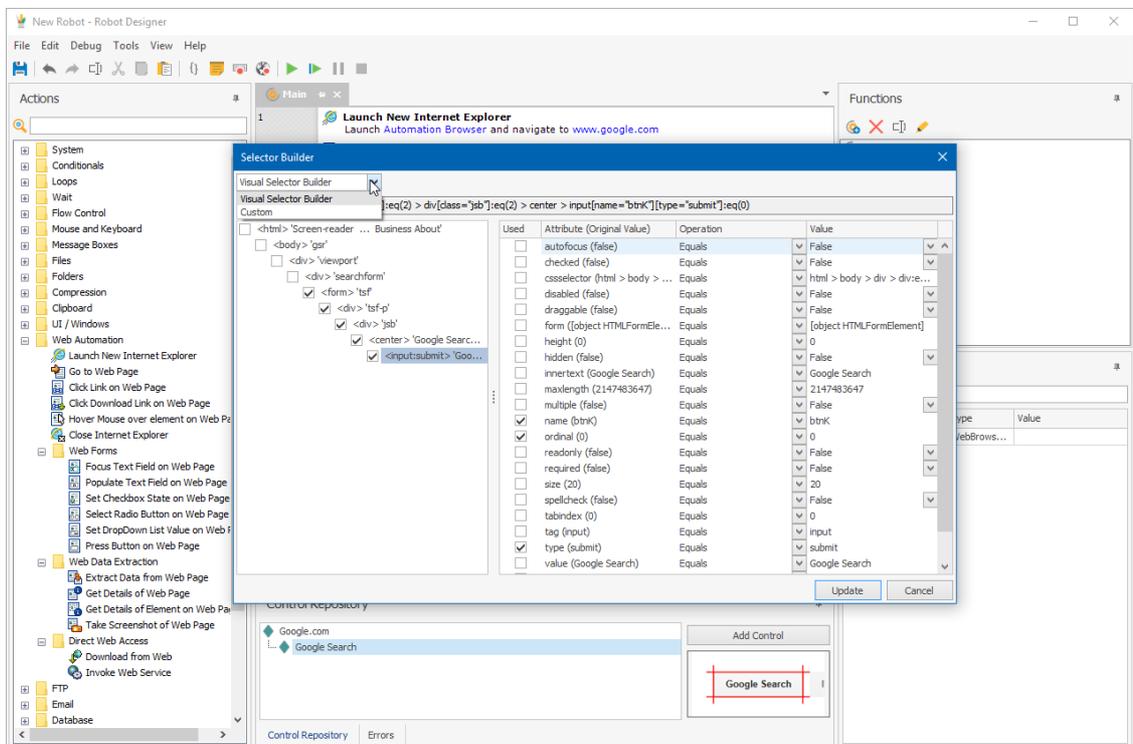
Under the hood:

What the Web Helper does is extract the CSS Path of the selected element and a meaningful description adding it to your Control Repository. Should you double click on the control, the Selectors for it will pop up as per the screenshot below:

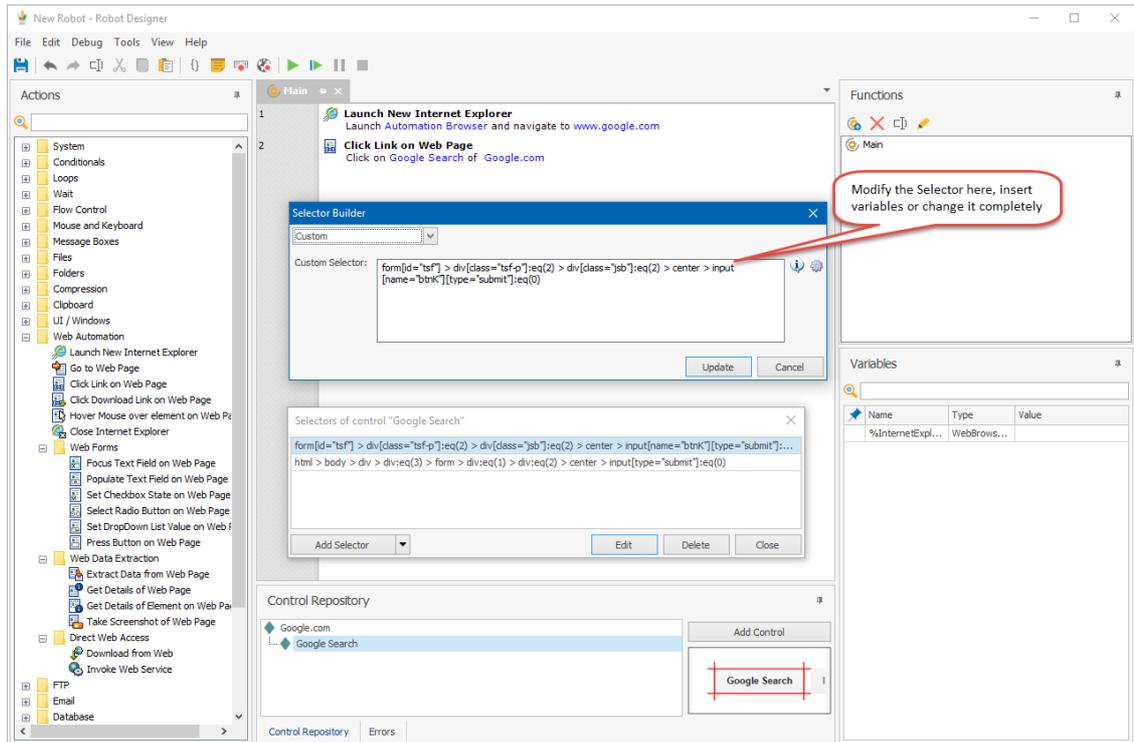


Should you click on "Add Selector" or "Edit" the Selector Builder will appear.

In the Selector Builder you are able to add or remove attributes to the CSS selector by checking or un-checking the relevant check-boxes. You can also choose to build a custom Selector from the Custom Option.



In the Custom selector Builder window you can modify the Selector, insert variables, or change it completely.



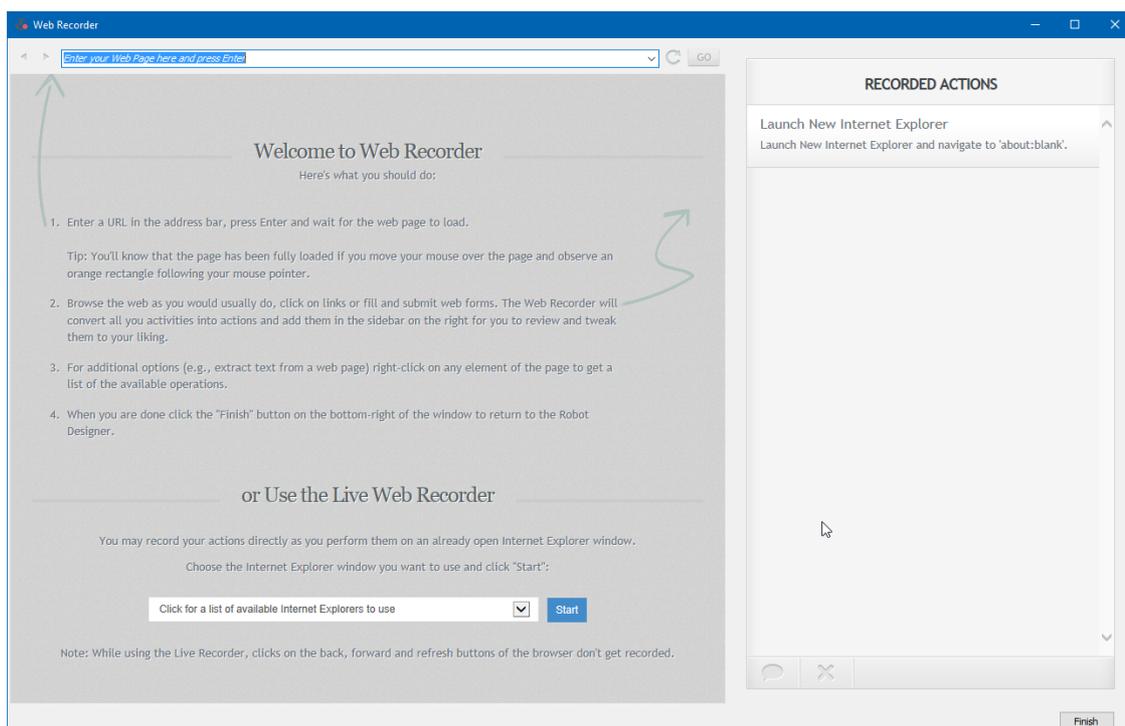
To learn more about the values you can enter in the "CSS Selector" field and how to select a single element within a web page you may refer to the [CSS Selectors](#)^[485] topic.

This is the logic behind any web-related action where you need to specify an element and perform an operation on it.

3.8.3 The Web Recorder

While you can create a Web Automation Process by combining and configuring actions in the Process Designer, as you would do for any other other Process, it is much easier to build your web related Process by using the Web Recorder

You can open the web recorder either while [creating a new Process](#)^[281] or from within the Process Designer by clicking on the Web Recorder button  on the toolbar.



This is the initial screen of the Web Recorder. It consists of two parts, the Web Automation browser on the left and the sidebar with the recorded actions on the right. At this point, you are presented with a browser choice. You can either record on the Web Automation browser (initially displaying the "Welcome to Web Recorder" page) or you may use any already open Internet Explorer window with the help of the Live Web Recorder.

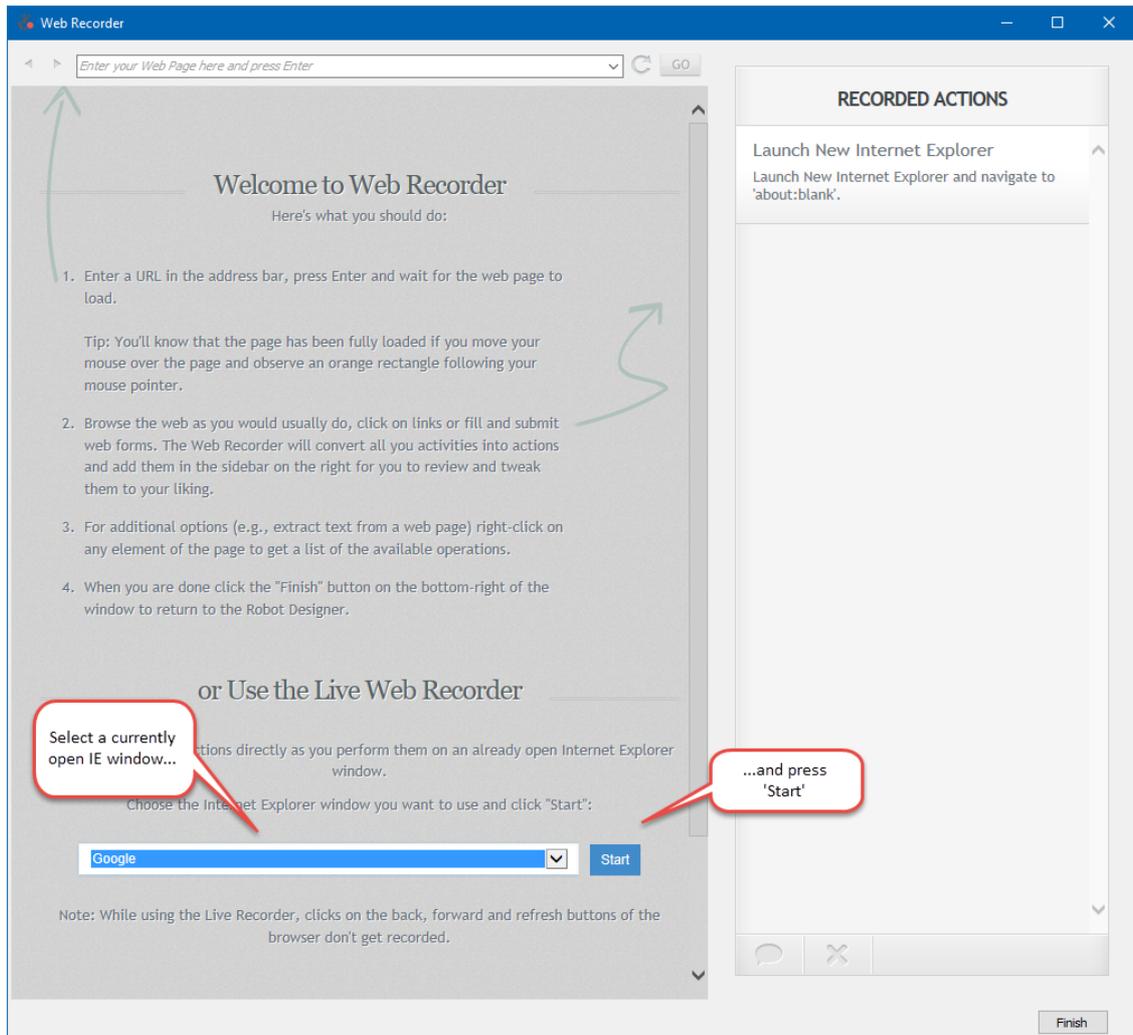
Use the Web Automation browser for the recording

As with the [Web Helper](#)³⁹³, the first step here is to enter in the address bar the URL of the site you want to visit and press "Go" (or Enter). Immediately you will see that the first action in the sidebar gets updated with the website address. This indicates that the recording has started and in

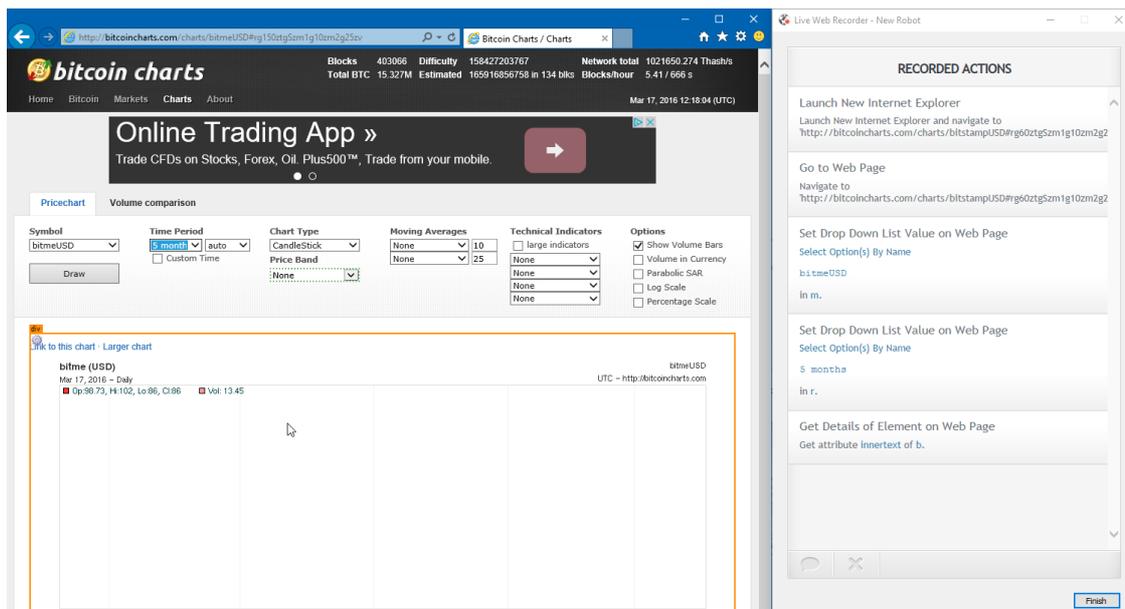
the sidebar you will see the corresponding actions that get added as you interact with the web page.

Record the actions on an already open Internet Explorer window (Live Web Recorder)

The Live Web Recorder does not require you type a URL. Instead, the starting web page is determined by your selection of the Internet Explorer window. In other words, the current address of the selected window is where the recording starts from. Therefore, when there are more than one Internet Explorer windows open, you should select the one displaying the site you want to visit before pressing the "Start" button:



With the correct browser window selected, press the "Start" button to start the recording. The sidebar will now appear on the top right part of the selected Internet Explorer and the Web Automation browser (which is now unnecessary) will hide.



Note that while using the Live Web Recorder, clicks on the back, forward and refresh buttons of the browser do not get recorded.

Making the Recording

As soon as you start the recording, the process is similar irregardless of your selection between the Live Web Recorder and the Web Automation browser.

Whatever you do gets recorded. So, for example, if you click on a link a new "[Click Link on Web Page](#)"⁷⁴³ action will be appended on the sidebar. Enter some text into a text field, submit a form, navigate back or click on another link; every activity you perform will be translated into the appropriate action and added to the sidebar.

If you perform an action unintentionally and you do not want it to be included in the final sequence of actions, you can move the mouse over the action that you want to delete and a red  button will appear in the action's top right corner, as shown below. By clicking on the x button the action will be removed from the list of recorded actions.

Live Web Recorder

RECORDED ACTIONS

Go to Web Page
Navigate to
<http://bitcoincharts.com/charts/bitmeUSD#rg150ztg5zm1g10zm2g25;>

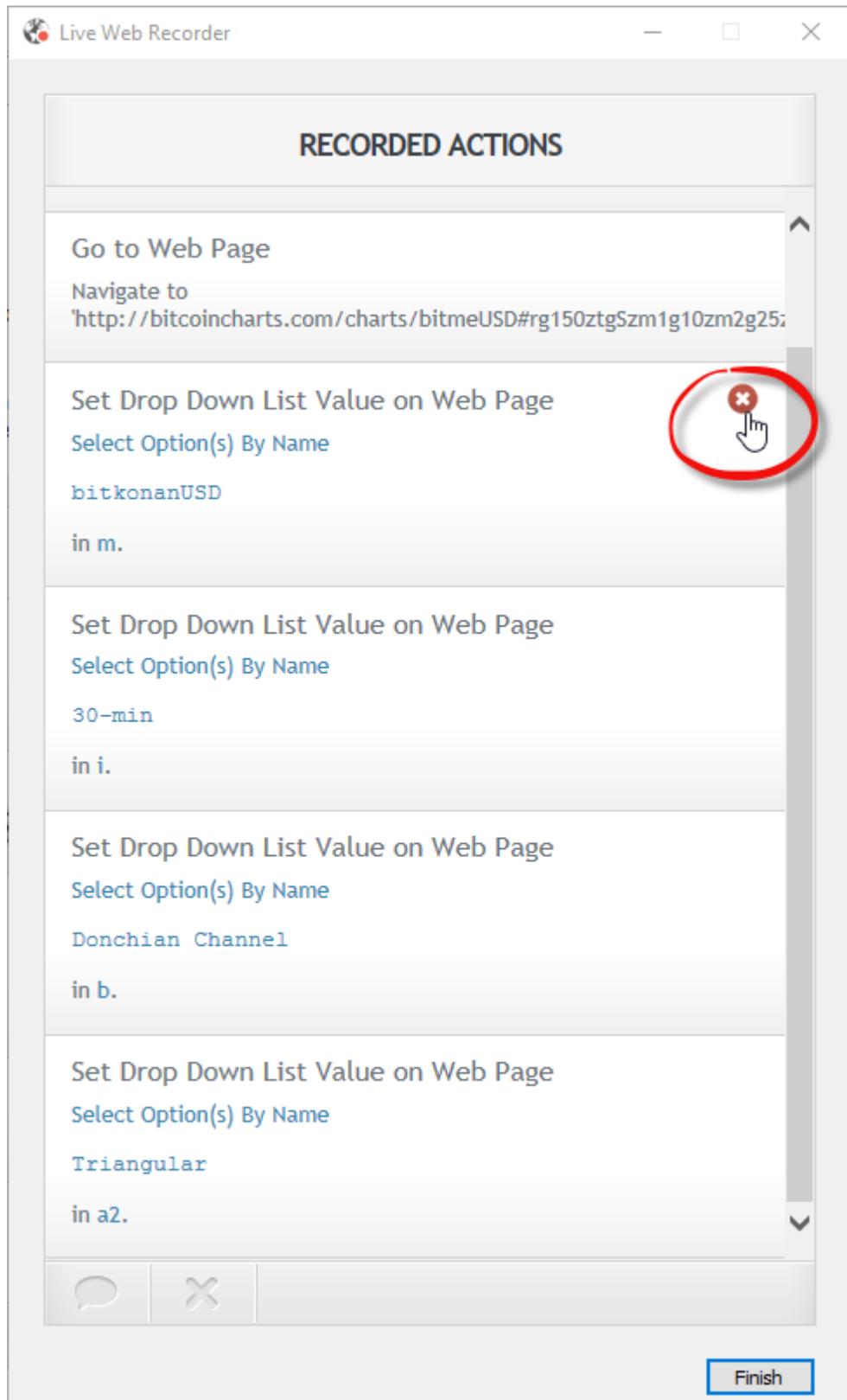
Set Drop Down List Value on Web Page
Select Option(s) By Name
`bitkonanUSD`
in m.

Set Drop Down List Value on Web Page
Select Option(s) By Name
`30-min`
in i.

Set Drop Down List Value on Web Page
Select Option(s) By Name
`Donchian Channel`
in b.

Set Drop Down List Value on Web Page
Select Option(s) By Name
`Triangular`
in a2.

Finish



Most actions have part of their text in blue. These pieces of text correspond to the parameters of the action and can be edited by clicking on them. As an example we'll click on the email address that we entered into the email field and change it:

Live Web Recorder

RECORDED ACTIONS

Go to Web Page
Navigate to
<http://bitcoincharts.com/charts/bitmeUSD#rg150ztg5zm1g10zm2g25;>

Set Drop Down List Value on Web Page ✖
Select Option(s) By Name
New Value

OK
in m.

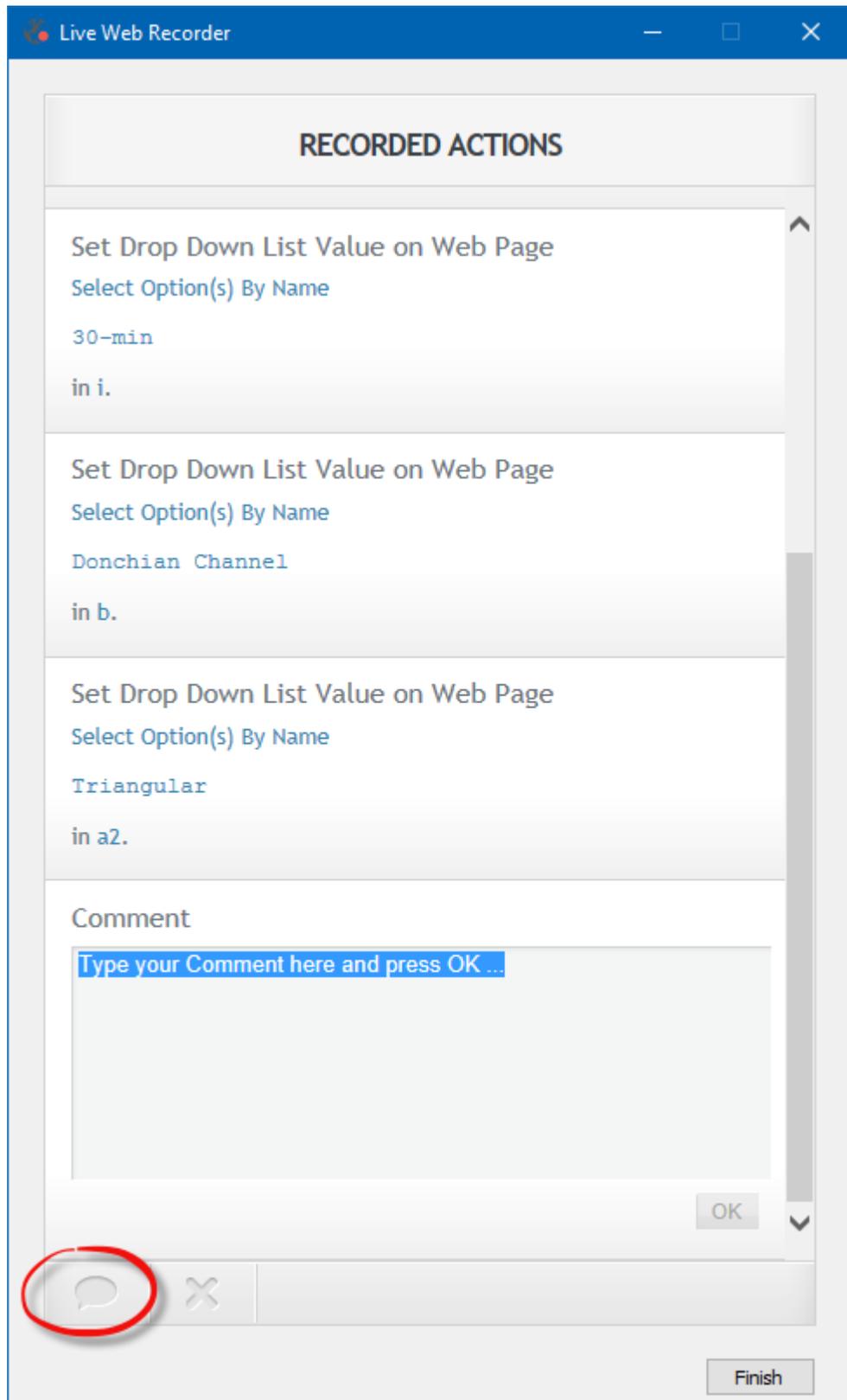
Set Drop Down List Value on Web Page
Select Option(s) By Name
30-min
in i.

Set Drop Down List Value on Web Page
Select Option(s) By Name
Donchian Channel
in b.

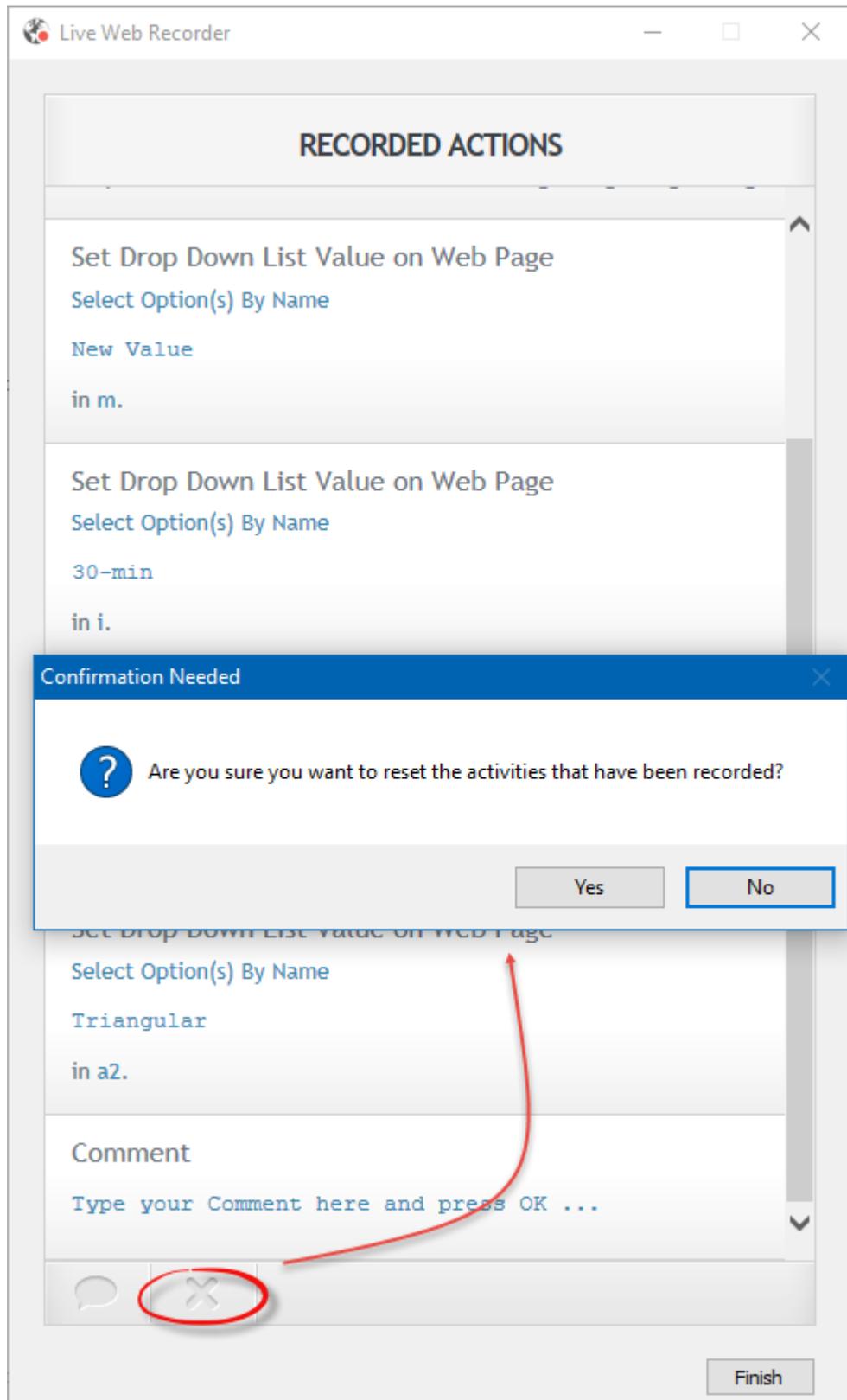
Finish

This feature is also useful for changing the name the Web Recorder assigns to each element. ProcessRobot makes its best effort to extract a name for every element you interact with, however this may not always be possible or you may want to replace the auto generated name with another one that makes more sense to you.

While recording you may want to take notes that will be useful if you chose to edit the actions later in the Process Designer. To do so you need to click on the "Insert Comment" button  and a comment action will be appended in the sidebar. This action, holding the comment that you entered, will be part of the recorded actions that will be sent back to the Process Designer.

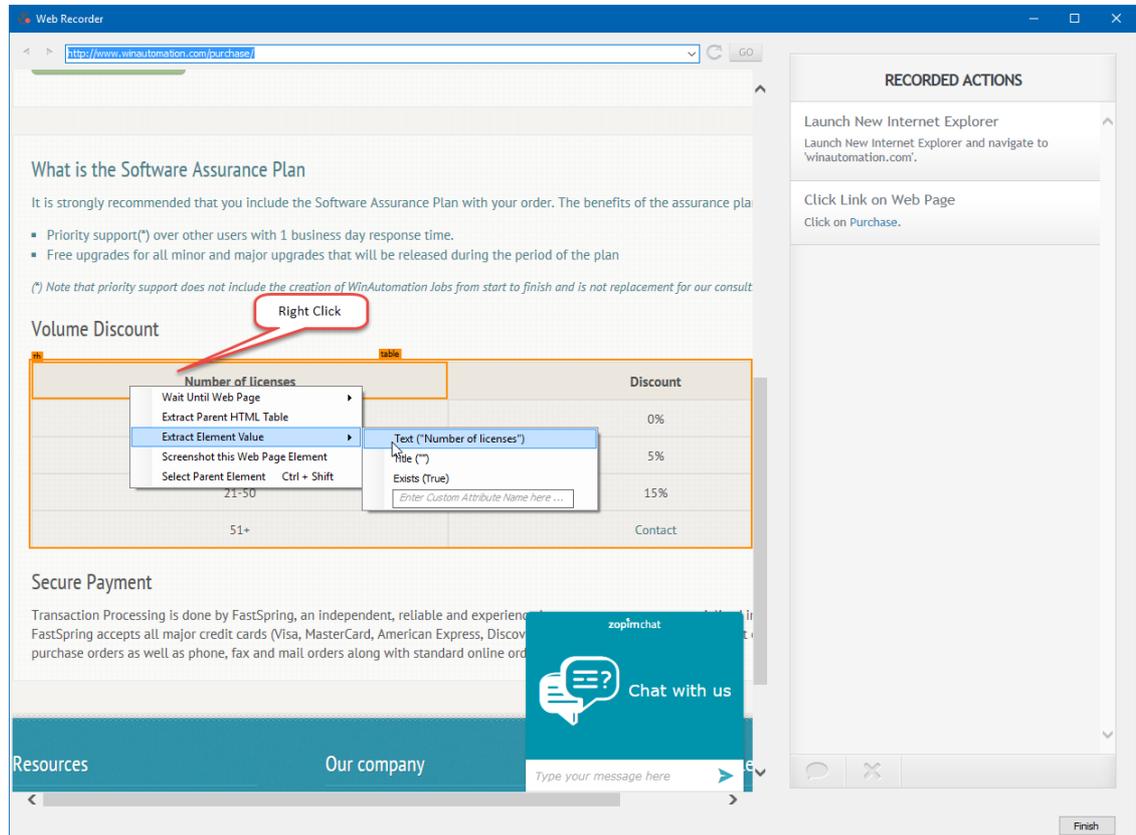


If, at any point you want to discard all the actions you have recorded so far and start from scratch, you can press the "Reset Recorded Actions" button. A message box will appear for you to confirm the delete.



Extracting Data while Recording

The Web Recorder also includes the functionality of the [Data Extraction Web Helper](#)^[373] so that while you are recording you can specify any data on the web page you are visiting that you want to extract. This is done simply by right-clicking on the element you want to extract and from the context menu that will appear select the "Extract Element Value" and then the attribute that you want to extract.



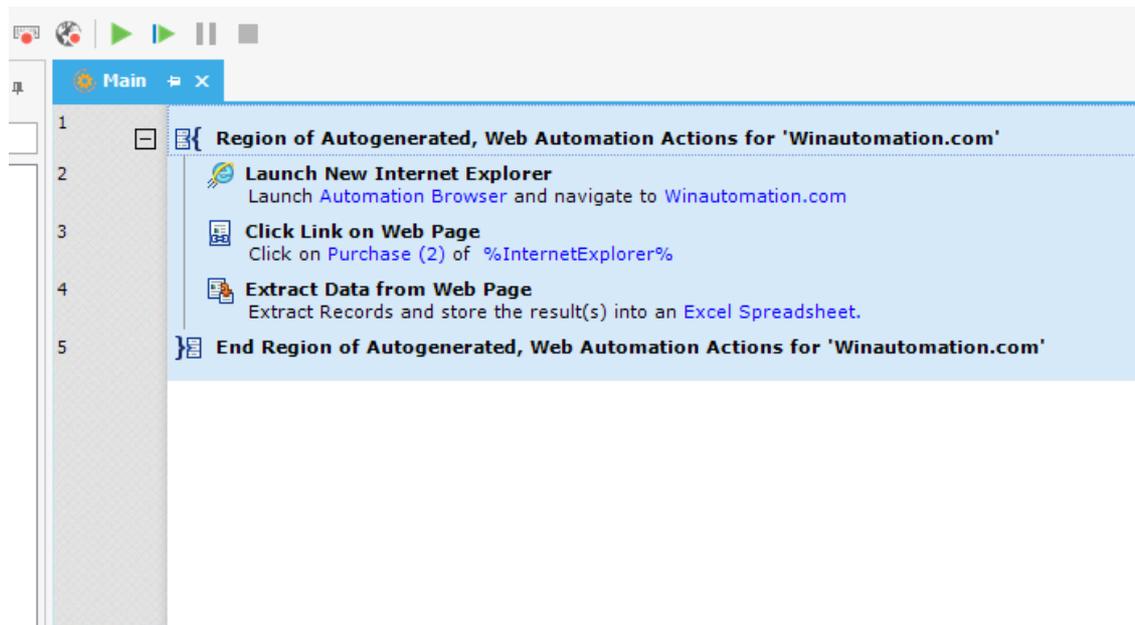
As with the [Data Extraction Web Helper](#)^[373] you may select multiple elements and the Extraction action will be updated accordingly. The action will also include a link to a preview of the selected data as it is going to be extracted.

The screenshot shows the Web Recorder interface with a browser window displaying a webpage. The 'RECORDED ACTIONS' panel on the right lists several actions, with the third action, 'Extract Data From Web Page', having a 'Show Preview' button circled in red. A red arrow points from this button to an 'Extracted Data Preview' window. This window displays a table with the following data:

Value #1	Value #2
1-10	0%
11-20	5%
21-50	15%
51+	Contact

By right-clicking on an element you can also insert an action that takes a screenshot of that element, or an action that waits until the text contained in that element appears/disappears from the page. The latter is useful especially for dynamic/ajax-powered websites that you need to make sure that the page has been updated before moving on to the next action.

Finally, once you are happy with the process you have recorded you can press the "Finish" button and all the actions will be transferred back to the Process Designer:



As you can see from the screenshot above, all actions (along with any comments you may have added while recording) have been inserted into your Process. From this point you can continue editing and modifying your actions as you would for any other Process whether it has been generated by the Web Recorder or not.

3.8.4 Web Data Extraction Part I

Extracting data from web pages is a big part of Web Automation. In ProcessRobot there are four actions dedicated to this task, with the "[Extract Data from Web Page](#)^[762]" being the more important and versatile.

The other actions allow you to [take a screenshot of a web page element](#)^[767], [retrieve details of a web page](#)^[765] such as its title or its HTML source and finally [retrieve any HTML attribute of any web page element](#)^[766] or even its text.

So far so good, however, very often, you want to retrieve information displayed into the page in the form of tables or lists, rather than technical values. And this is where the "[Extract Data from Web Page](#)^[762]" action comes into play:

Properties of 'Extract Data from Web Page' action

Extract Data from Web Page
Extracts Data from specific parts of a Web Page in the form of single values, lists, or tables.

General Exception Handling

Action Input

Web Browser Instance: %Browser%

Synopsis of Data to be Extracted: **No data specified for extraction.**

Specify Web Data to Extract

(bringing an actual Internet Explorer window to the foreground while this dialog is open will automatically activate the live version of this helper.)

Store Extracted Records into: an Excel Spreadsheet
an Excel Spreadsheet
a Variable

Action Output

Store New Excel Instance Into: %ExcelInstance%

This action is Enabled

More Info OK Cancel

As with any other web-related action you will first need to specify the web browser instance containing the page you want to extract data from. The next step is to specify the data itself and finally to select where the extracted data will be stored. The default value is to be written into a newly generated Excel spreadsheet, but to do so you need to have Microsoft Excel installed in your computer.

Alternatively you can have the data stored into a variable for further processing by later actions. Note that the extracted data can be in any of the following forms:

1. Single Value:

Say that from a web page containing info about a product you extract the product name only. In this case, if the extracted data is stored into a variable this variable will contain a text value.

2. Handpicked (multiple) values:

Say that, in our previous example you select to extract not only the product name, but also the description and its price. In this case three separate values will be extracted and the resulting variable will hold a value of type DataRow.

You will be able to access each one of the retrieved values using the following form: %DataFromWebPage[...]% where within the brackets you will enter either a number or the name of the value.

3. Lists:

You are no longer in the page containing the product info, but in a page containing the list of all products. If you choose to retrieve all the product names displayed in the page then you'll end up with a list. Subsequently, the variable holding the extracted data will be of type List.

4. Tables

In the previous example of the web page containing a list of products you select to retrieve both the name and the price for each product. In this case the resulting variable will hold a DataTable with a product in each row and two columns (with the product name stored in the first column and the product price in the second one).

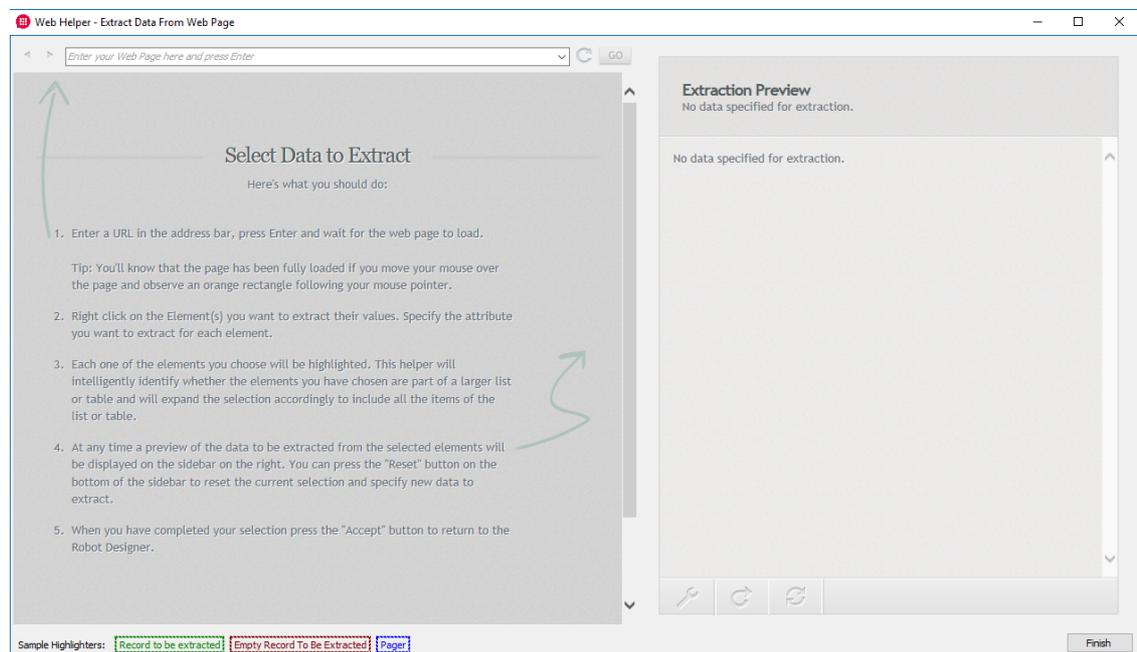
Selecting the Data to extract from a Web Page

To specify which data you want to extract from the web page you will need to use a Data Extraction [Web Helper](#)^[393]. The target data can be specified either by using the live version of the Web Helper, or the standard one.

Live Web Helpers conveniently work on an existing Internet Explorer window. Just have the action "Extract data from Web Page" open in your designer and click on the Internet Explorer of your interest.

The standard Web Helper on the other hand, is a browser window itself and opens by pressing the "Specify Web Data to Extract".

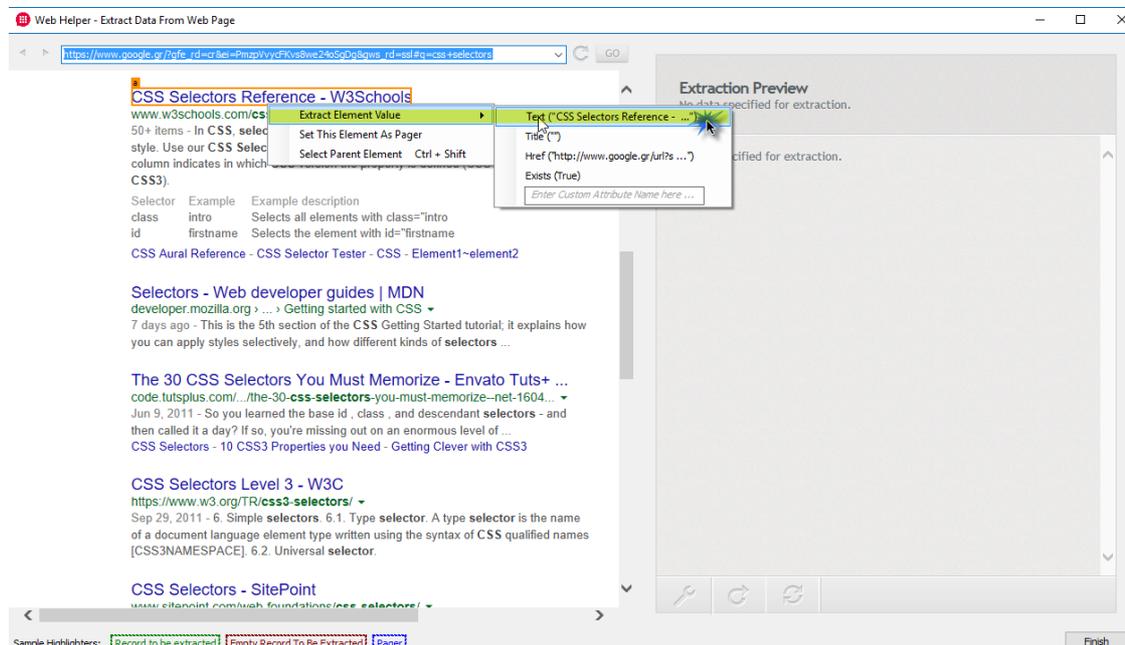
Should you click on the "Specify Data to Extract" button the Web Helper Window will appear.



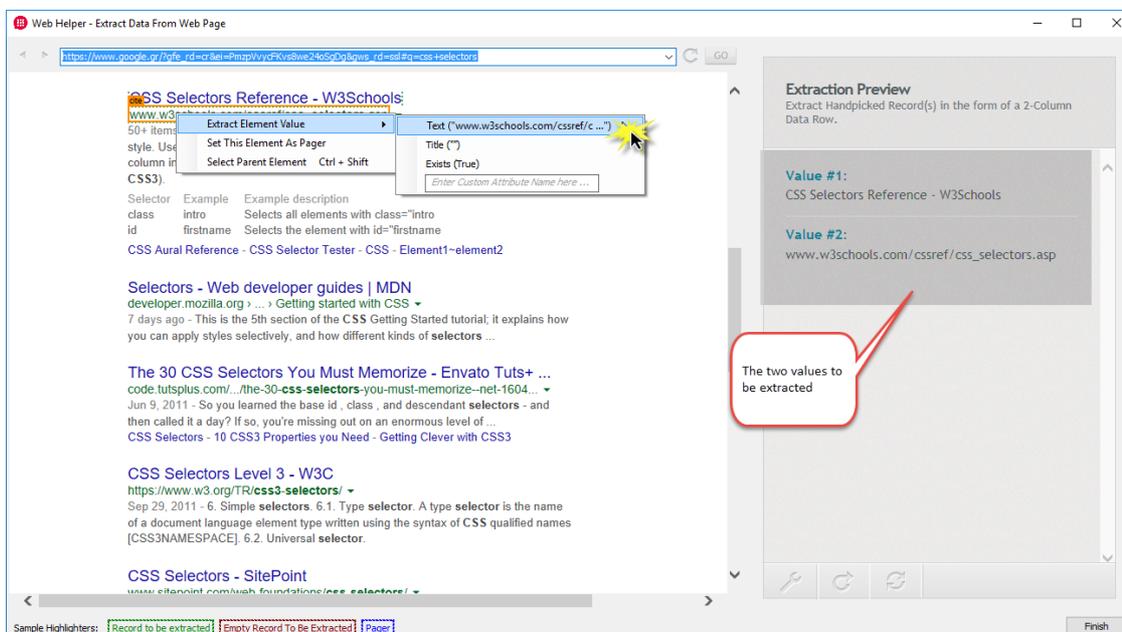
this window consists of two parts, the left pane which is the web browser and the right sidebar which displays a preview of the data selected for extraction.

As with the Web Helpers, the first step is to enter the URL in the address bar and navigate to the page containing the data to be extracted.

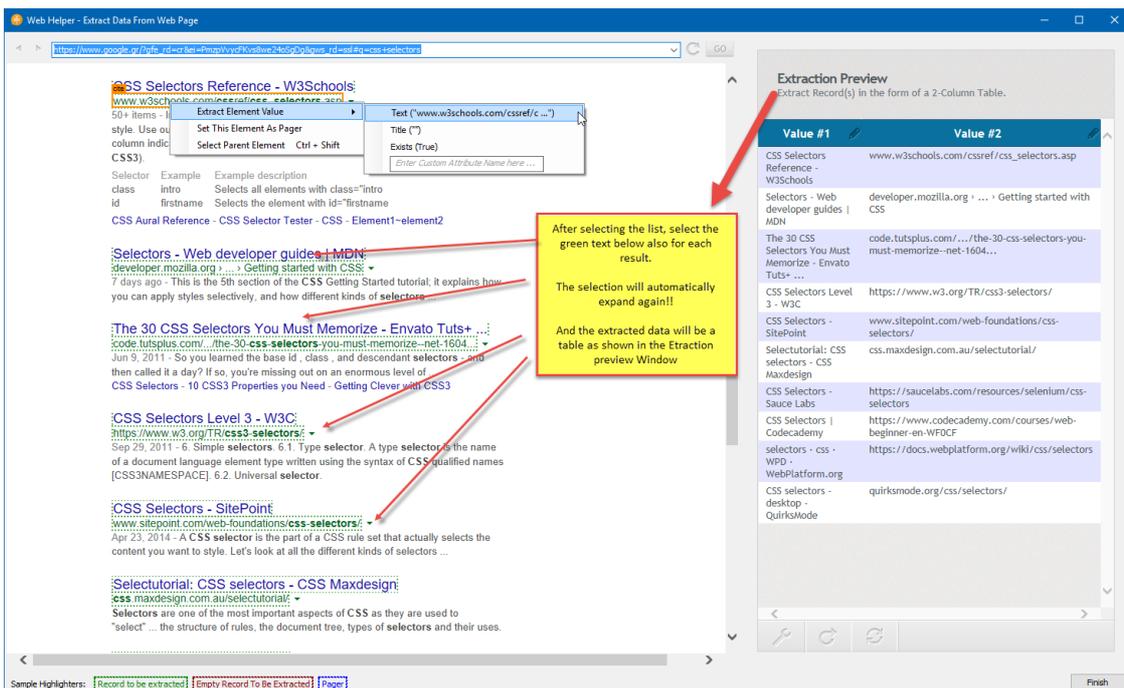
Next, all you have to do is right click on any element of the page that you want to retrieve and select the property you want to extract. Most often you will want to extract the text of the element, but you have also the choice to specify any HTML attribute you want to retrieve.



At any point you can press the "Accept" button and finish the process of selecting the data you want to extract, or you may continue by selecting more elements. Depending on the elements you select, the web helper may or may not expand the selection. For example, if the next element you select is the URL of the element chosen in the previous screenshot (shown in green in the screenshots) you will have just two elements selected:

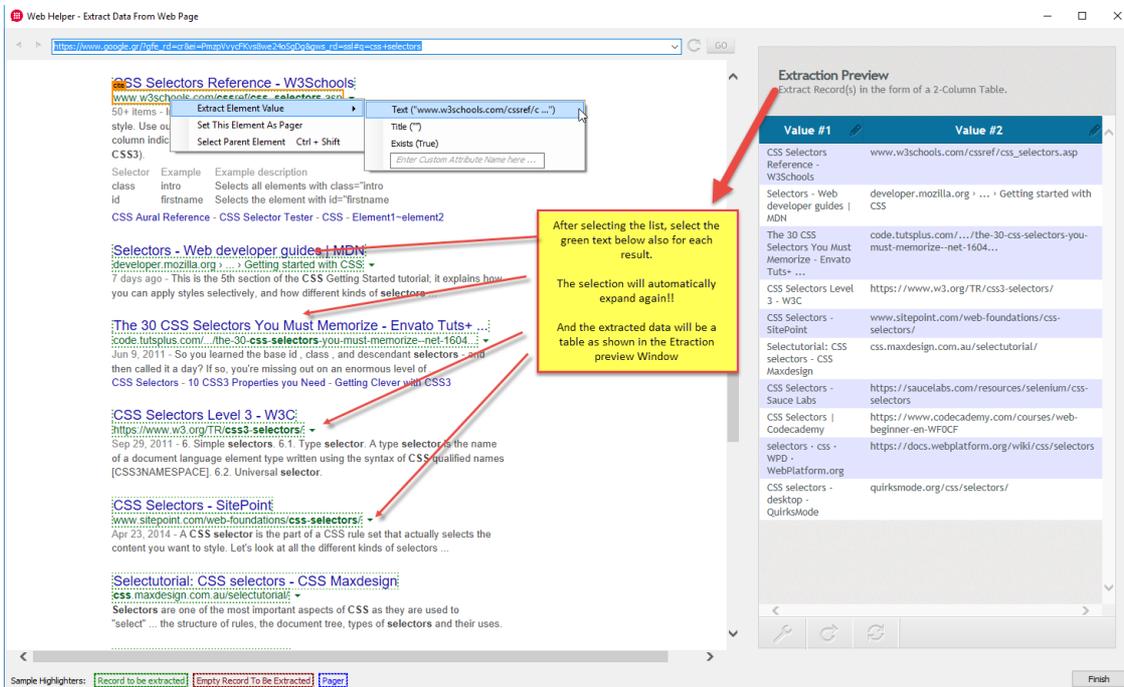


If, however, the second element you selected was another search result title, ProcessRobot would detect that you are extracting a list and would expand the selection to all items of the list



On the right sidebar you see the preview of the data to be extracted in the form of a list.

So now you have specified that you want to extract a list. If you select an additional element, ProcessRobot will extract the corresponding data for each element already in the list, returning the result in the form of a table:



By selecting an additional element, the table would simply get an additional column. You can edit the column names by clicking on them in the preview sidebar.

If the data spreads over multiple pages there will be a "Next" link somewhere that points to the next page. You can right click on that link and select "Set This Element As Pager". This way, ProcessRobot at runtime will not retrieve the data just from the first page but will continue and retrieve the same data from the next pages too.

At any point you can press the **"Reset"** button  to discard the selection made so far and start over. You can also examine and modify the CSS Selectors generated by the Web Helper that specify which info needs to be extracted by pressing the **"Advanced Settings"** button .

Finally you can press the **"Recalculate Now"** button  to highlight which data will be extracted from a web page based on the current selection. This can be useful if, for example, you select some elements to extract from a web page containing info about a product. You can then visit a page containing info on another product and click the "Recalculate Now" button to make sure that appropriate info will be retrieved from the second page too.

As mentioned before, after you have finished with selecting the data you want to extract you can press the "Accept" button to return to the action's properties dialog.

3.8.5 Web Data Extraction Part II

The web data extraction can also take place on an actual IE if you have the "Extract data from Web Page" action open while you move your mouse pointer to the page of interest.

Should you click in the webpage, then the "Live Web Helper- Extract Data From Web Page" window will pop up. In this Window you will be able to preview the extracted data.

Extracting a List:

Lets say that you wish to extract the title for all available results in a webpage.

Having the "Extract data from Web Page" action open, hover your mouse on the page (or click on a blank area). Then right click on the first result and extract its Text as in the screenshot below:

Do the same for the second result and the list a list of all the items' text will be automatically extracted. Click on the "Advanced Settings" icon to review the CSS selector which you can modify and make it even more efficient.

1. As you can see while extracting a list, we have the **Base Selector** and the **CSS selector**. The Base selector is the root element in the HTML code, under which the items of the list are listed. This means that the extraction starts from the "...div:eq(1) > ul > li"
2. For each list item from the list "...div:eq(1) > ul > li" and then it gets the "h3 > a" element.
3. The attribute that you are extracting is "Own Text" and it can be changed to "Title", "Href", "SourceLink", "Exists" or any other Attribute is available in the HTML code of the page for this element.
4. You also have the option to apply Regular Expressions on the extracted text, in order to get just a part of it.

Changing the selector by hand, then you can click on the "Recalculate now"  button to see the extraction's Result.

Extract Data From Web Page - Advanced Settings

Extract: List

Base CSS Selector: `html > body > div:eq(4) > div:eq(1) > div:eq(2) > div > div:eq(0) > div > div:eq(2) > div > w-root > div > div:eq(1) > ul > li` (1)

CSS Selector: `h3 > a` (2) Attribute: Own Text (3) RegEx: (4)

Use Paging

Extraction Preview
Extract Record(s) in the form of a List.

1. New Sony PS3 Super Slim 12GB Playstation 3 UK PAL Offic
2. SONY PS3 PLAYSTATION 3 SUPER SLIM 500GB CHARCOAL E
3. New listing Sony PlayStation 3 PS3 Slim 160 GB Console T
4. ORIGINAL SONY PLAYSTATION 3 CONSOLE SLIM 120 GB CE
5. Sony Playstation 3 slimline 500gb 2x wireless controllers
6. New listing Brand New PS3 12GB Including Game
7. Sony Playstation 3 super slim +2 fully working controllers
8. New listing Sony PlayStation 3 Super Slim MEGA BUNDLE /
9. New listing Sony PS3 Slim Console 120gb with leads & cor
10. Sony Playstation 3 PS3 500GB superstim console, good coi
11. Sony Playstation 3 PS3 12GB super slim console, good cor
12. New listing Ps3 250GB Super Slim inc Controller, 1 Game
13. SONY PLAYSTATION 3 PS3 SLIMLINE SLIM 120GB CHARCOA
14. New listing Ps3 Super Slim 320GB family MEGA BUNDLE M
15. PS3 CONSOLE 320GB CHARCOAL BLACK + 11 TOP GAMES II
16. New listing SONY PS3 SLIM (PAL) CONSOLE 250GB + 4 GAM

Extracting a Table:

In order to extract more than one piece of info for each result you would have to extract a table.

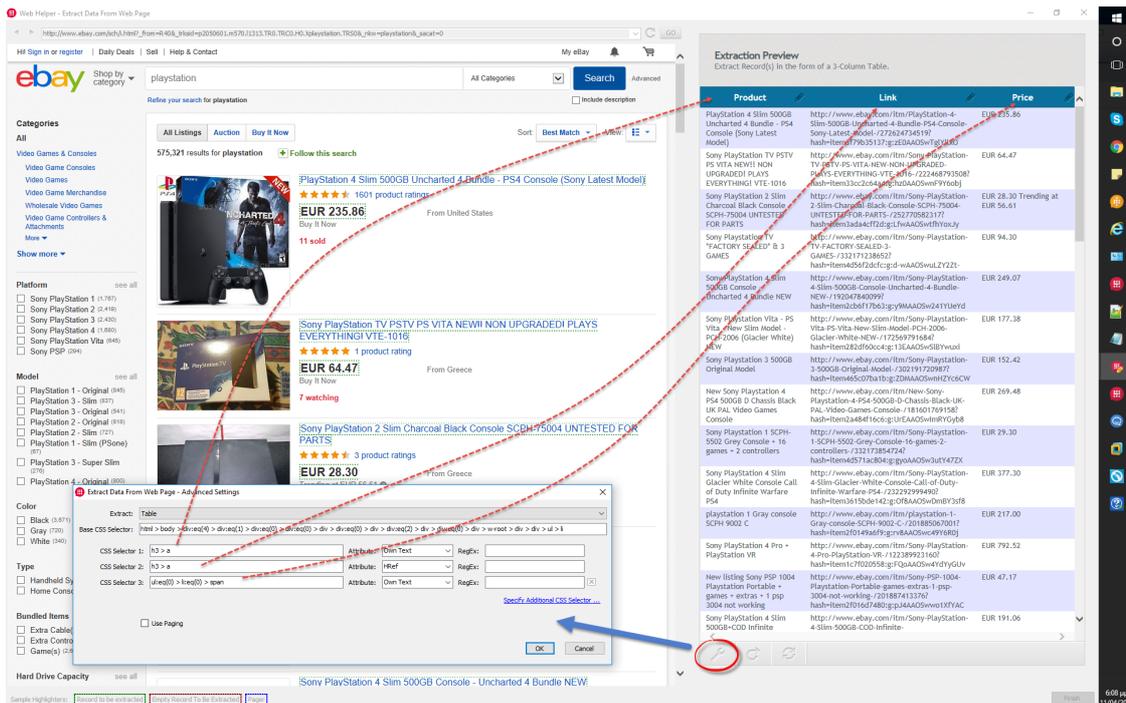
Let's say that we want to extract the Title of the product, the link behind it and the price.

For the first result we right click on the title, extract its "Text", then right click again to extract the "Href" and finally we right click on the price element to extract its "Text".

We move on to the second result/product to do the same and the table is automatically created in the extraction preview window.

For the table, in the same notion as extracting the list, we have the Base CSS Selector, which is the root element in the HTML code, under which the data of each result/product exist. This means that the extraction starts from the `.....div:eq(1) > ul > li` and then for each or the item we extract the

- `h3 > a` Attribute "Own Text"
- `h3 > a` Attribute "Href"
- `ul:eq(0) > li:eq(0) > span` Attribute "Own Text"



Attributes to extract:

In the Attribute field of the "Advanced Settings" of the "Extraction Preview" window, other than the attributes that are listed in the drop down list, you can specify any other attribute that the element has. For example if an element in the HTML code of the page is:

```
<li class="sresult lresult clearfix li shic" id="item463b90d307"
_sp="p2045573.m1686.l2210" r="3" listingid="301647057671">.....</li>
```

Then in the attribute dropdown list you can write "class" if you want to extract its class, "id" if you want to extract its id...and so on.

NOTE

- You can extract the plain html code of the element -and all its children elements- should you write "outerhtml"
- You can extract the plain html code of the all the children elements of the element should you write "innerHTML"

This is very helpful if you want to extract a piece of info that resides in the html for this element by applying some Regular Expressions on the extracted code.

3.8.6 Web Data Extraction Part III

Very often we want to extract data from a web page after a search, which has returned more than one page of results. In this case we want to perform the extraction to all, or to a number of pages.

This can be achieved with the "Extract Data from Web Page" action, if you select to set an element as "Pager". As shown below, you can right click on the pager element and select "Set This Element As Pager".

Having done so, upon extraction of the data, the Process will keep clicking on the "next page" button and extracting what you have set it to extract, up until when the pager element is enabled and present on the web page.

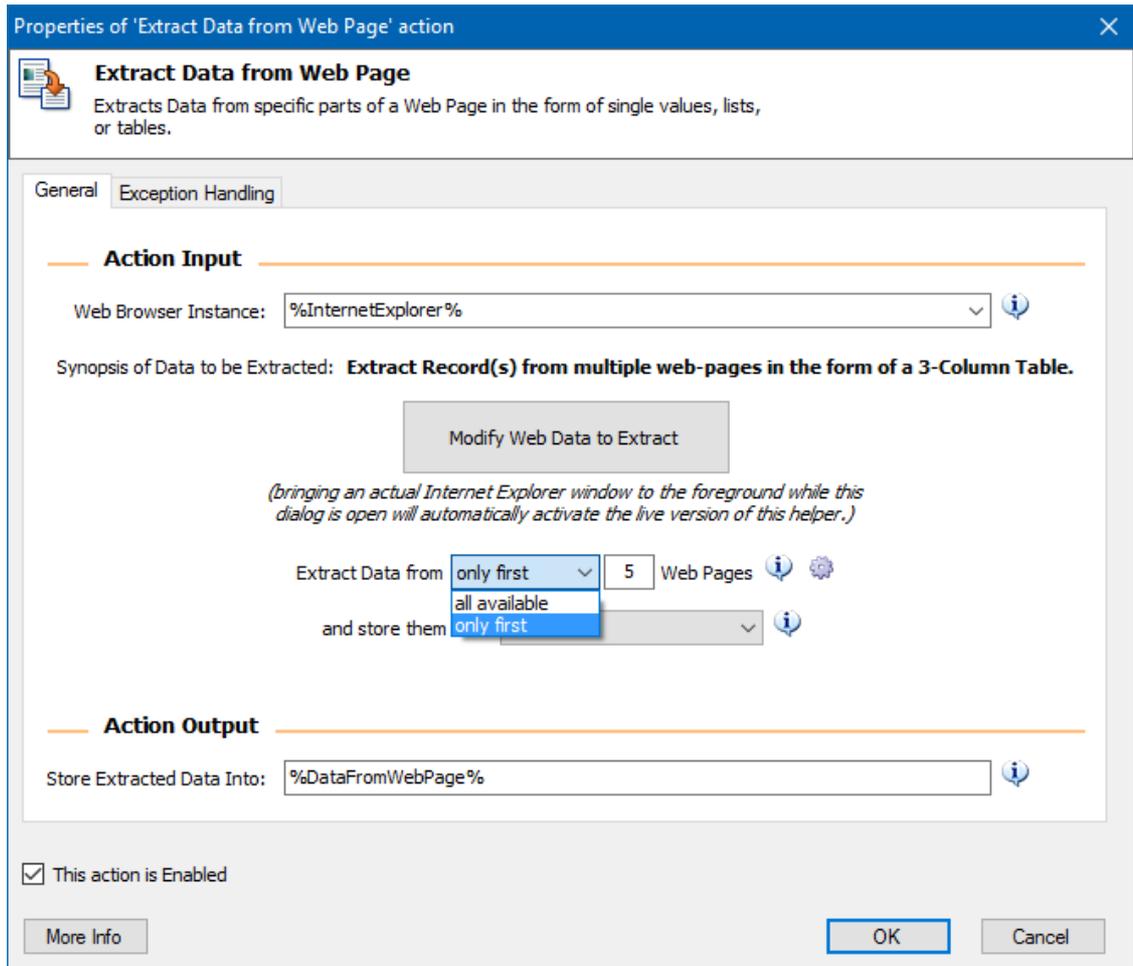
The screenshot shows a search results page for PS3 consoles. The 'Live Web Helper' window displays an 'Extraction Preview' of the results in a 3-column table. The table contains the following data:

Value #1	URL
New Sony PS3 Super Slim 12GB Playstation 3 UK PAL Official Latest Games Console	http://www.ebay.co.uk/itm/New-Sony-PS3-Supe
New Listing Sony Playstation 3 Console Slim 320Gb With 1 Controller And 5 Games Bundle	http://www.ebay.co.uk/itm/Sony-PlayStation-3
SONY PS3 PLAYSTATION 3 SUPER SLIM 500GB CHARCOAL BLACK CONSOLE BUNDLE	http://www.ebay.co.uk/itm/SONY-PS3-PLAYSTA
Sony Playstation 3 slimline 500gb 2x wireless controllers and 23 games bundle	http://www.ebay.co.uk/itm/Sony-Playstation-3
New listing Sony Playstation 3 PS3 Slim 160 GB Console Terraria FIFA Call Of Duty Bundle New	http://www.ebay.co.uk/itm/Sony-PlayStation-3
ORIGINAL SONY PLAYSTATION 3 CONSOLE SLIM 120 GB CECH-200AA In Black WITHOUT	http://www.ebay.co.uk/itm/ORIGINAL-SONY-PL
New listing Brand New PS3 12GB Including Game	http://www.ebay.co.uk/itm/Brand-New-PS3-12G

The CSS selector for the pager is shown in the "Advanced Settings" for the Extraction Preview window. You can modify it and use your own if you are familiar with CSS selector and maybe use its id and class attributes to make it 100% consistent.

Once you have set the data to extract, press "Commit". Back in the "Extract Data from Web Page" action you can choose to extract the results from:

- All Available pages, which means that the extraction will finish once there are no more result pages, or
- only first 'X' Web Pages, which means that the extraction will finish after extracting the data from the X web page.



Feel free to have a look at the Create CSS Selectors topic.

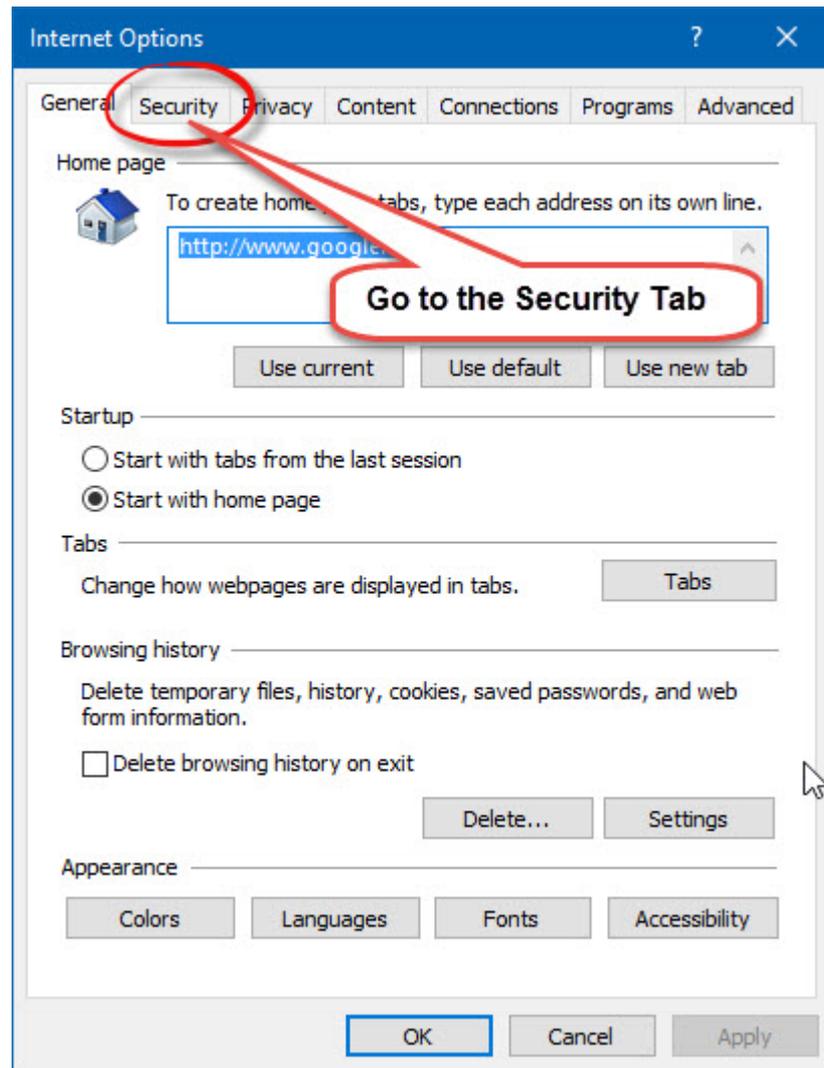
3.8.7 Configure IE for Web Automation

For automating your web related tasks you can either use the ProcessRobot's automation browser or an Internet Explorer window. The latter choice is especially useful when you want the task to be performed through an IE browser window that is already open.

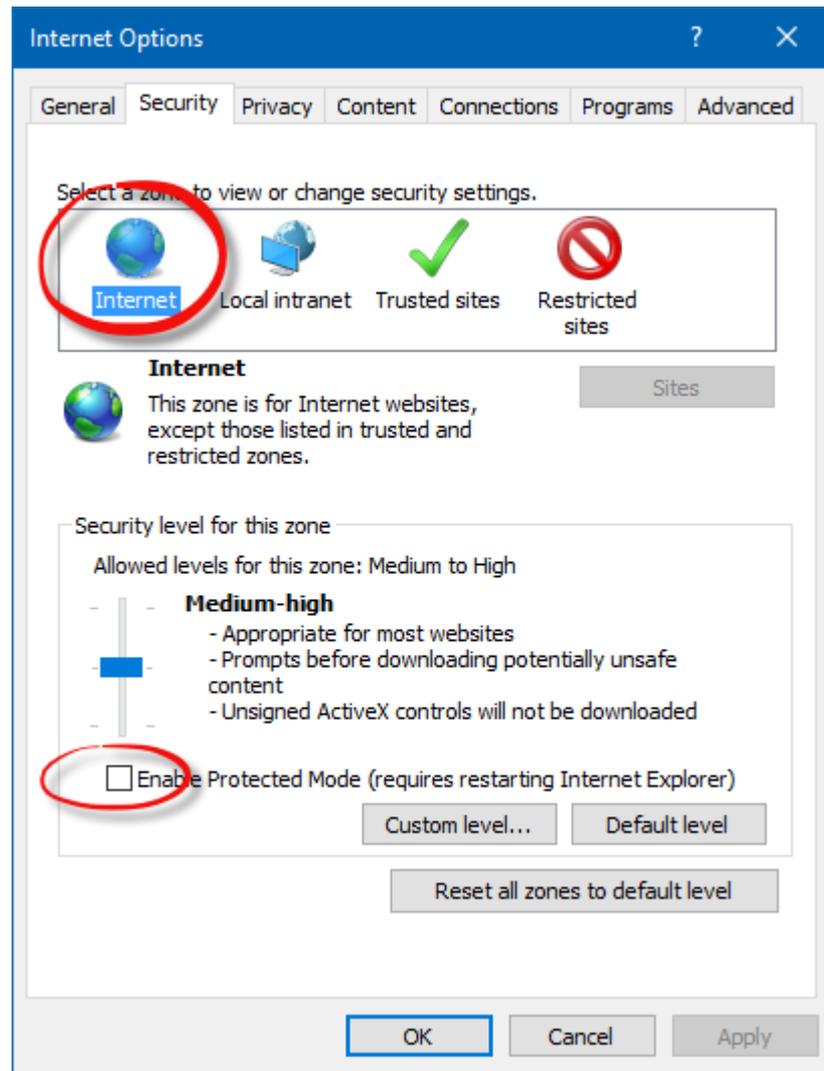
While you can use the ProcessRobot's browser without the need to configure anything, for being able to control IE you will need to modify some of its settings. By default, Internet Explorer comes with security settings enabled that do not allow an external application (ProcessRobot in our case) to control it.

To change this, you perform the following steps

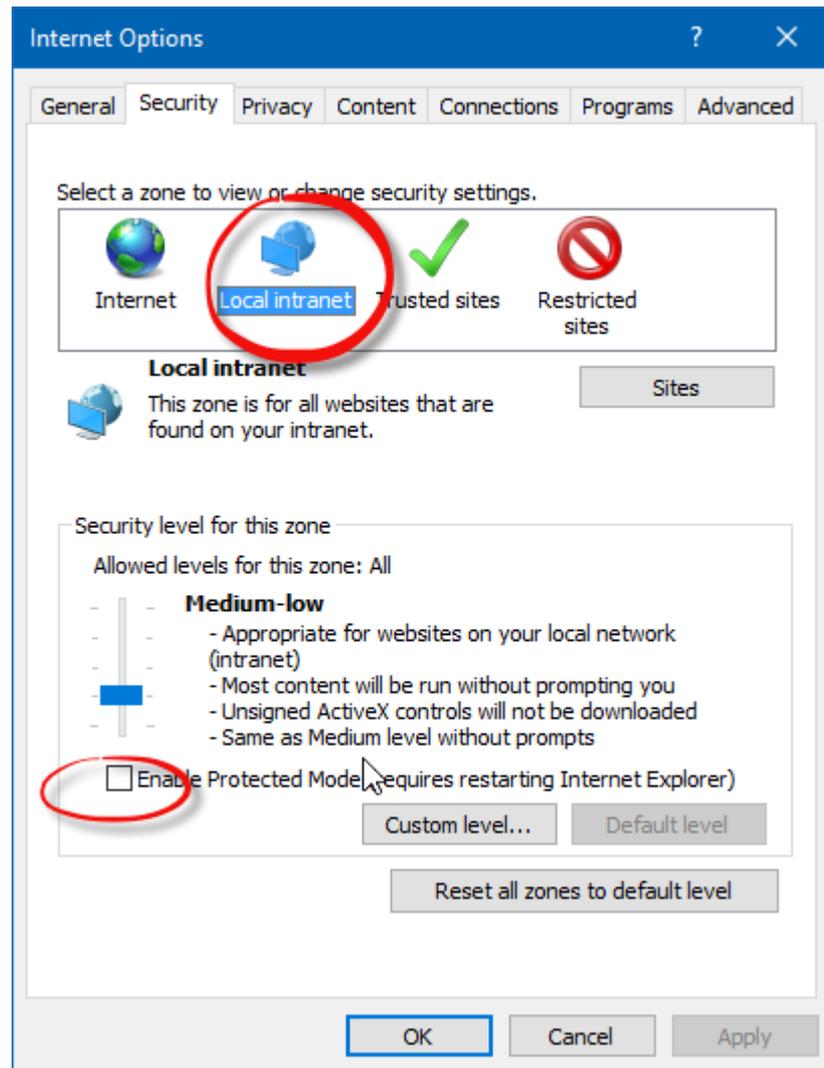
Open the Internet Options dialog in Internet Explorer and click on the "Security" tab:



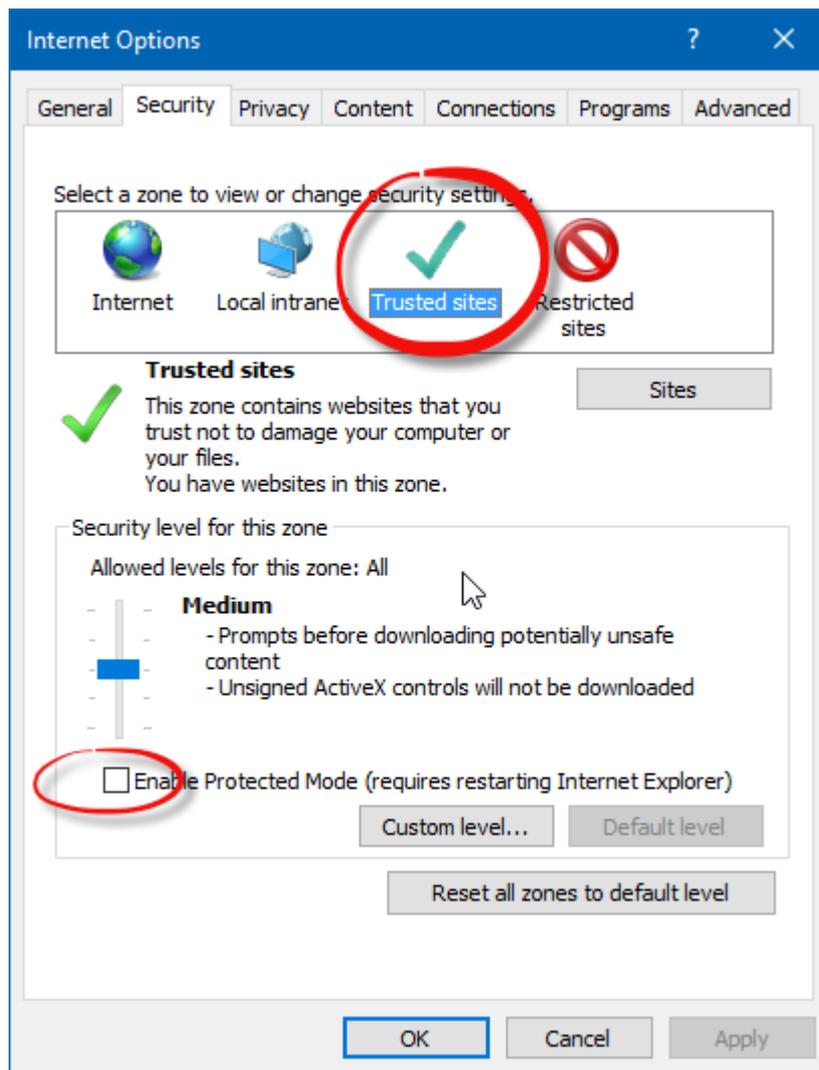
Select the "Internet" zone and UNCHECK the "Enable Protected Mode" checkbox:



Select the "Local intranet" zone and UNCHECK the "Enable Protected Mode" checkbox:



Select the "Trusted Sites" zone and UNCHECK the "Enable Protected Mode" checkbox:



Finally click the OK button and restart Internet Explorer. You will now be able to control Internet Explorer through ProcessRobot for automating all kinds of web sites and web applications.

3.8.8 Configure Chrome and Firefox for Web Automation

Unlike with IE, WinAutomation uses extensions to communicate with Chrome and Firefox.

These extensions are included in the WinAutomation installation, during which the user will be prompted to install them.

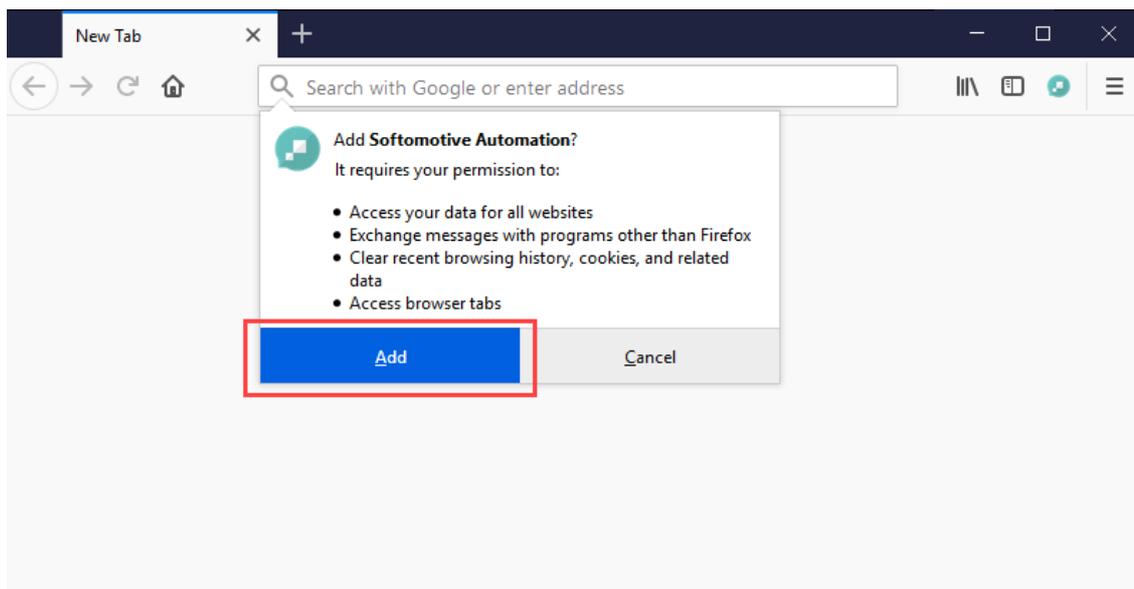
However, should the user not want to install the extensions during installation, they can do so at a later point through the Process Designer.

This article demonstrates how to install the extensions and how to enable them or disable them in the browser:

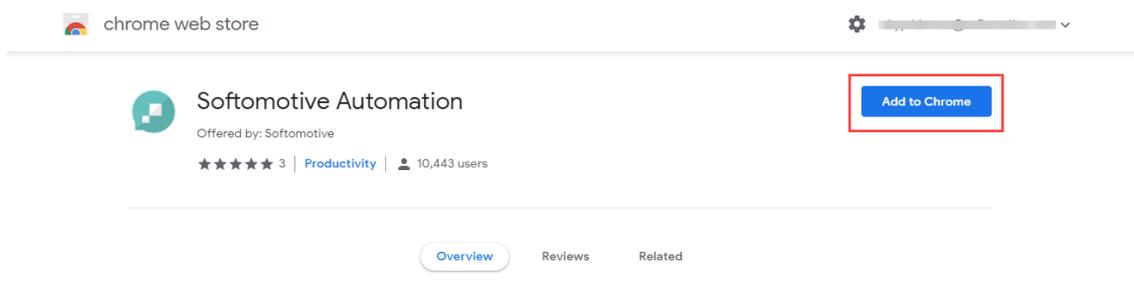
(a) Installing extensions for Firefox or Chrome

1. Open "Process Designer > Tools > Browser Extensions" and select the Browser for which to install the extension (Firefox or Chrome).

This will launch, or bring to the foreground, Firefox or Chrome with the following screen:

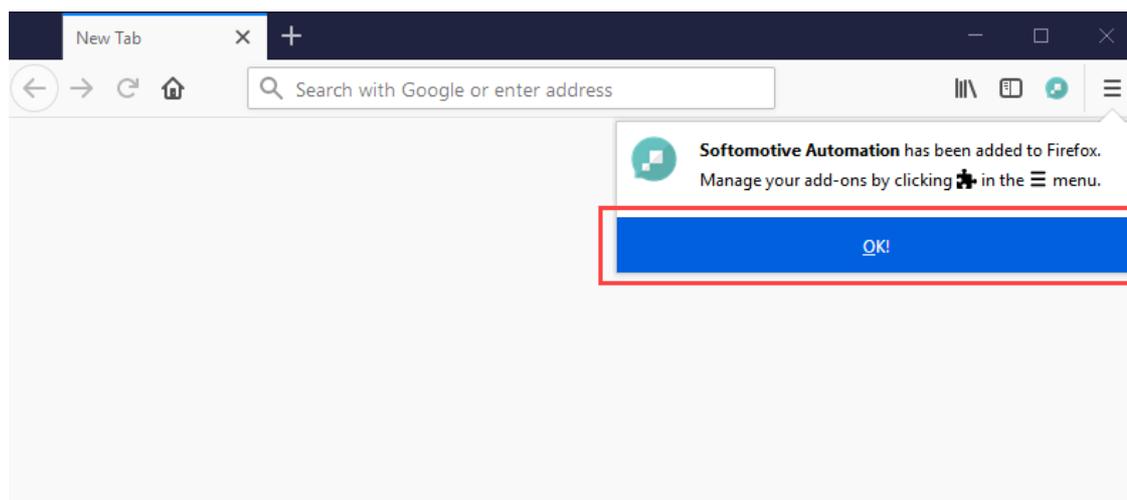


Firefox

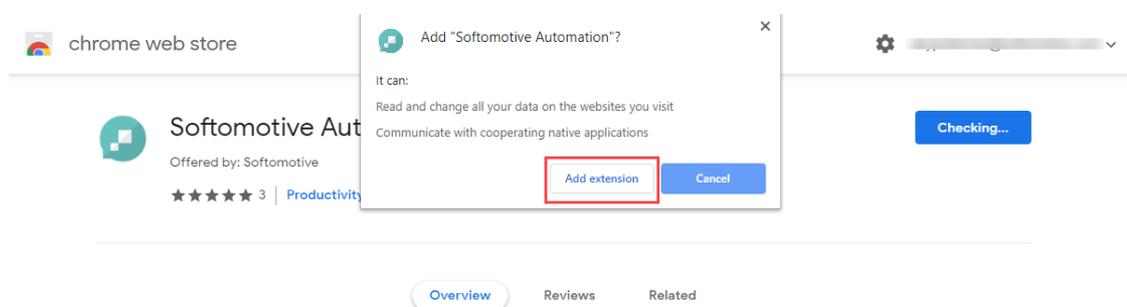


Chrome

2. Click Add and then OK or "Add Extension".

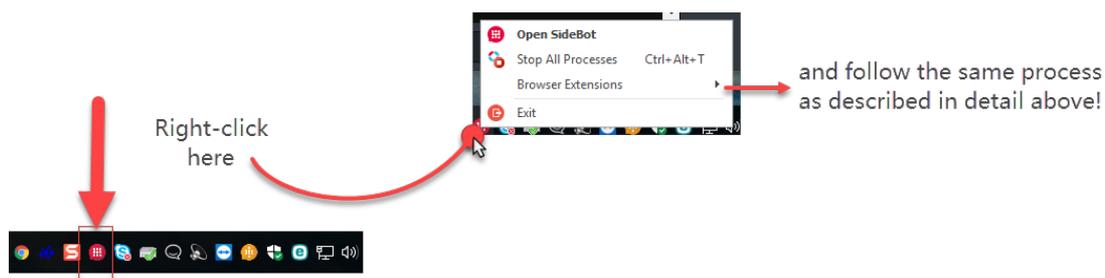


Firefox



Chrome

If the user does not have access to Process Designer, but does have access to a Sidebot, they can follow the same steps described above, with the difference that they will start the process by right-clicking on the ProcessRobot icon in the System Tray:



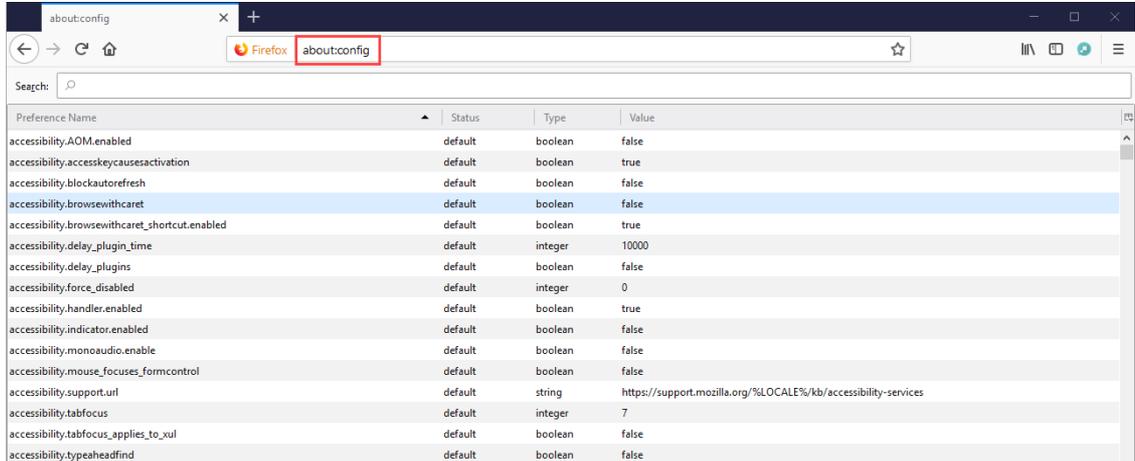
(b) Additional Settings that Might Influence Automation for Firefox or Chrome

For Firefox:

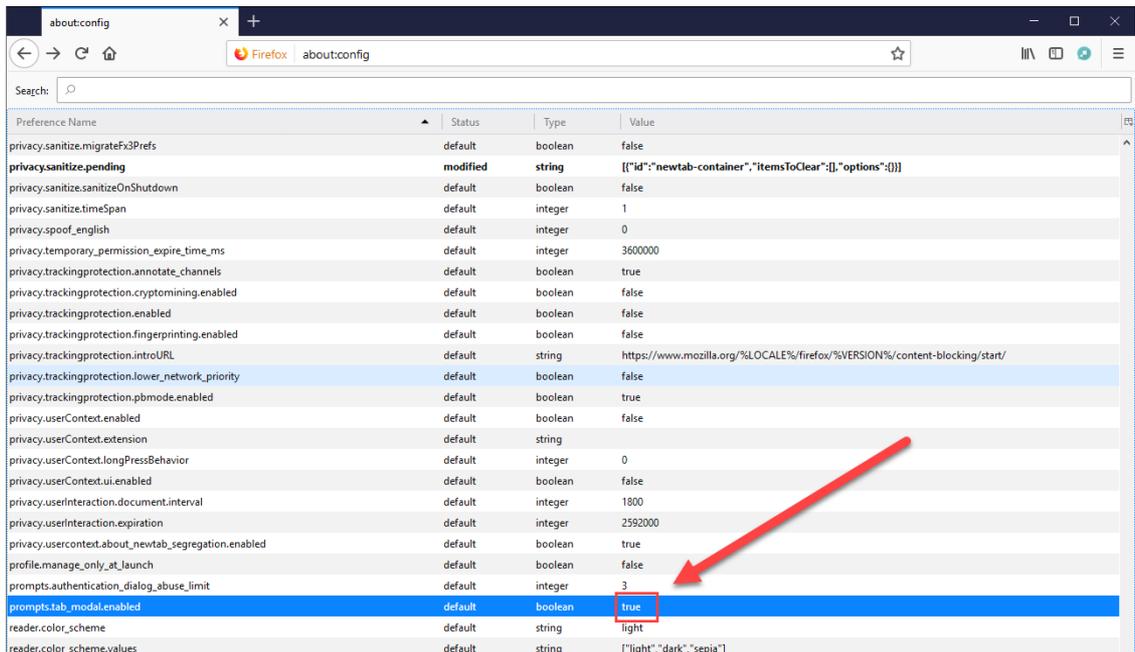
Alerts in "modal" dialog boxes seem to be enabled by default in Firefox. When these appear, they can freeze the browser and prevent the user from switching to other tabs or windows.

It is useful to know how to enable or disable this setting, as it can have an impact on the viability of automations. Follow these steps to disable the setting:

1. Enter the text "about:config" in the URL bar of a new tab on Firefox and press Go:

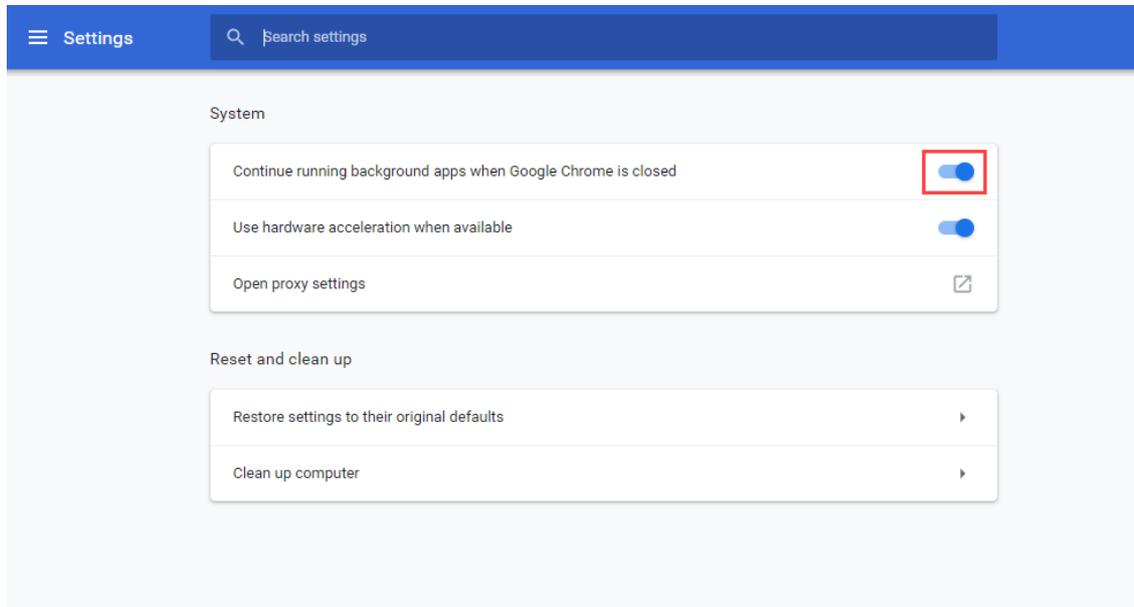


2. Search for the "prompts.tab_modal.enabled" preference name in the resulting list. Double-click on the row to change the value in the last column to "false":



For Chrome:

Disable running background apps when Google Chrome is closed. To do this, navigate to Settings > Settings Navicon > Advanced > System and click the slider icon next to "Continue running background apps when Google Chrome is closed".



3.8.9 Configure Internet Explorer for Servers

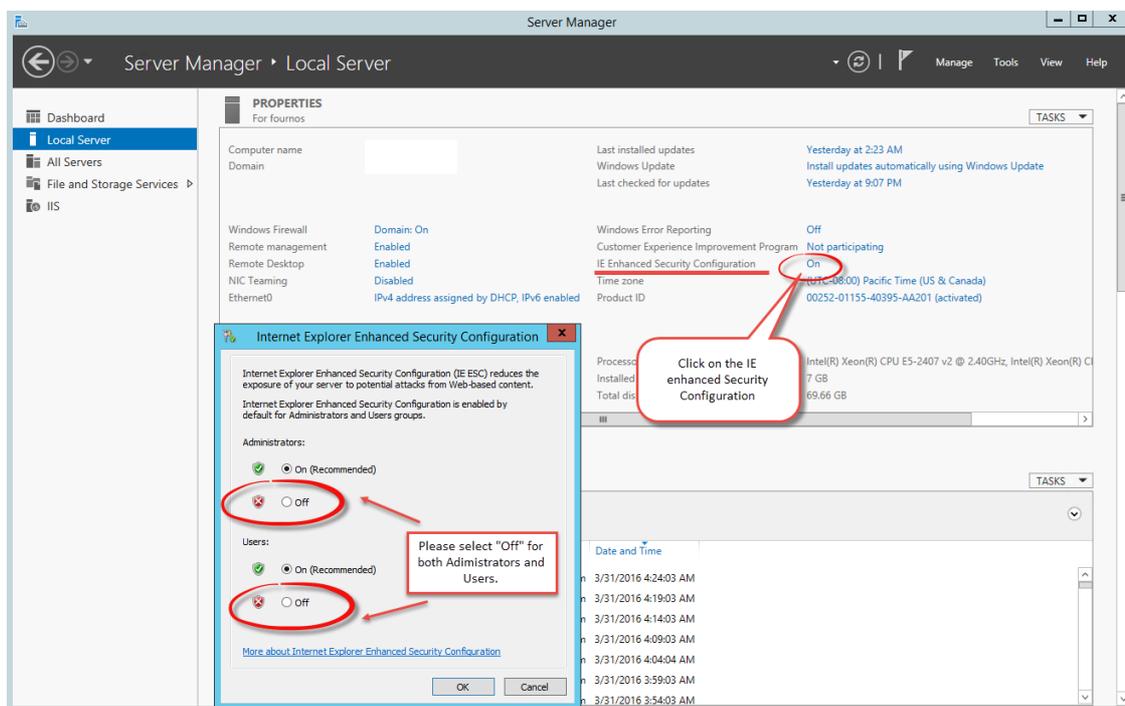
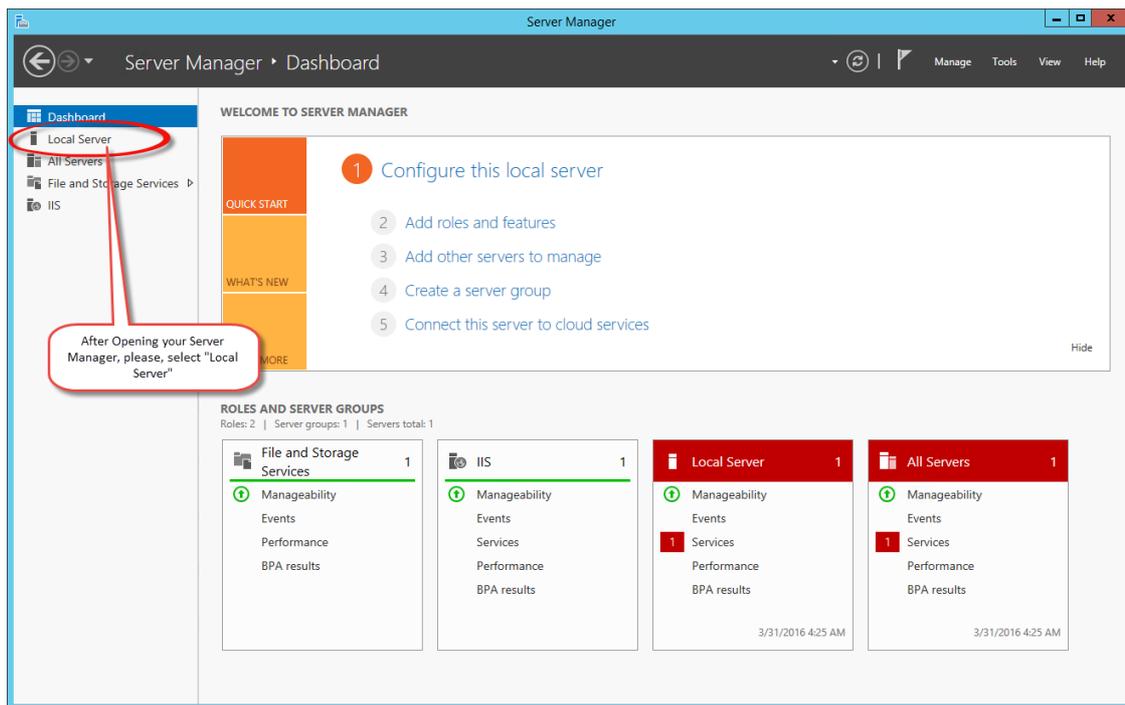
If you have ProcessRobot installed on a Server, for automating your web related tasks please note the following.

Windows Servers have the "Internet Explorer Enhanced Security Configuration" (IEESC for short) feature turned "On" by default, which prohibits any and all ProcessRobot Processes from properly launching an Internet Explorer or the Automation Browser via the "Launch New Internet Explorer" action.

Likewise, as long as the IEESC feature is "On" any and all web-automation actions will too fail to work as intended. Moreover, the design-time Web Helpers of the aforementioned actions will not work either as long as the IEESC feature is "On".

In order to be able to overcome the above, you would have to change the Security Configuration as per the screenshots below in 3 small steps.

1. Open the "Server Manager" and go to "Local Server"
2. Click on the "IE Enhanced Security Configuration"
3. Both for Administrators and Users select the "Off" radio button and click "OK".



You will now be able to control Internet Explorer through ProcessRobot for automating all kinds of web sites and web applications.

3.8.10 Actual Internet Explorer VS Automated Browser

Differences and limitations for using the ProcessRobot Runtime Browser versus the Actual Internet Explorer:

1. "Click Download link from Web Page" action:

This action only works with the Automated browser regardless the IE version, while in an actual IE, version 8 and below is a requirement.

2. For the actual IE to work reliably with the Web Automation Actions one needs to change its security options. ([Configure Internet Explorer for Web Automation](#)³⁸³)

Such tweaks are not necessary when using the ProcessRobot Runtime Browser. This can come in handy for users working in companies having certain standards when it comes to the security (thusly prohibiting users from changing the security settings of IE). For these users the ProcessRobot Runtime Browser is the way to go.

3. The ProcessRobot Automated Browser suppresses any and all message dialogs that may pop up unhindered in the actual IE.

When this feature of the ProcessRobot Runtime Browser is undesirable (e.g. when dealing with authentication dialogs) we apply the "(ShowDialogs)" suffix at the end of the url inside the Launch New Internet Explorer action, so as to temporarily allow such dialogs to pop up. Using such a workaround is not needed in the actual IE.

4. The ProcessRobot Runtime Browser does not support tabs or opening links in windows.

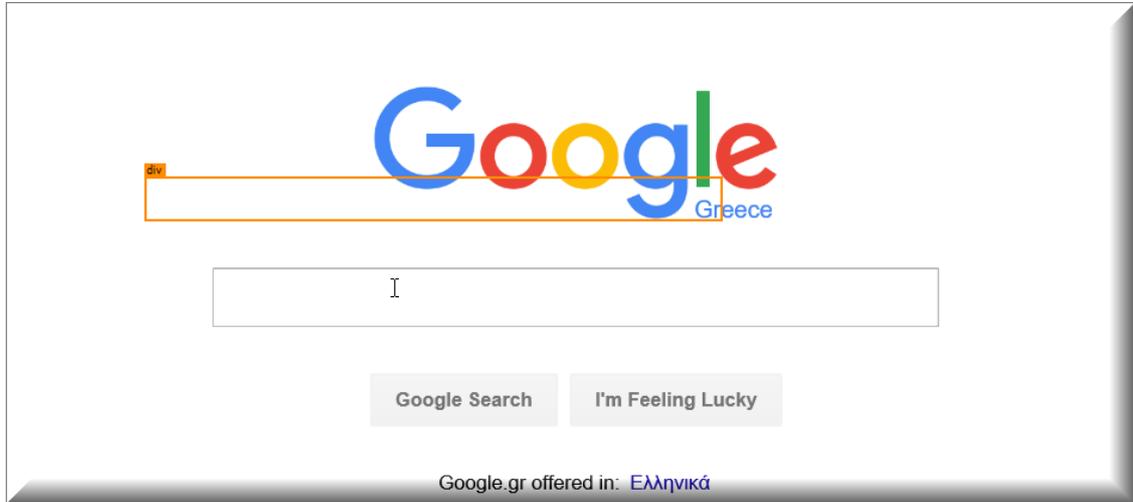
When the user clicks on a link in an actual IE, which would open in a new window, the Automated browser automatically opens the link it in the same window/instance (this discards all navigation history etc).

5. There is a small performance advantage with the Automated browser because it does not load unnecessary elements and ad-ons like the actual IE does.

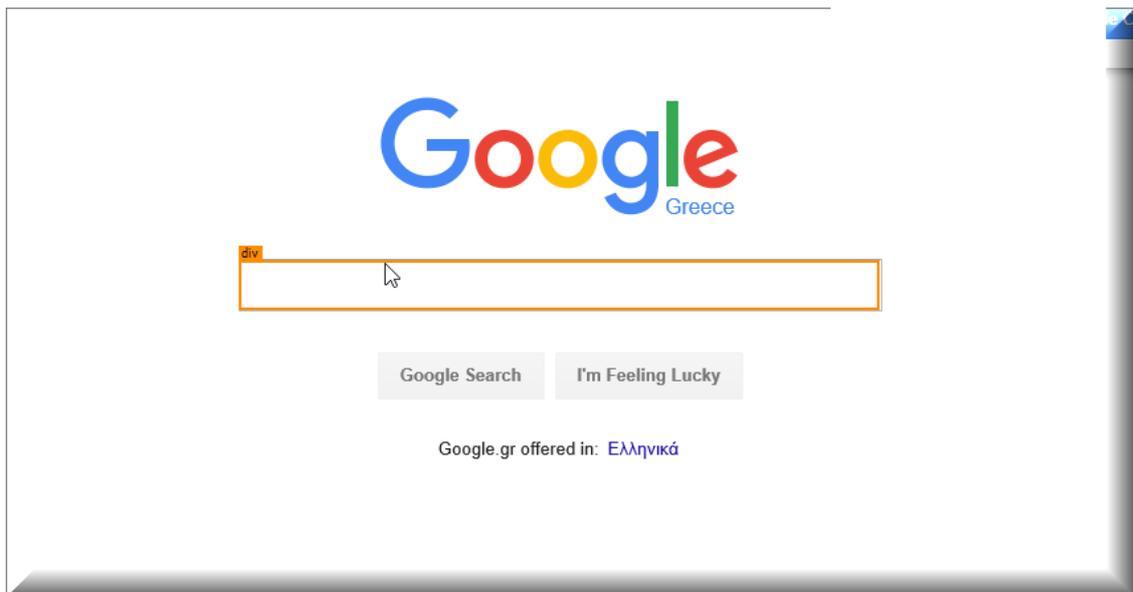
3.8.11 Web Helpers Offset

Sometimes, elements in a web page may be nested one into another, and still have the same size. In this case you can only highlight the deepest element and by moving the cursor out of it you also move it out of its parent elements (since they all have the same size). In this case, to select a parent element, move the mouse cursor over it and click Control-Shift to move the selection to its parent element instead of the deepest one. You can press Control-Shift repeatedly to walk up the elements hierarchy.

Also, there might be cases where the Web Helpers appear with an offset relative to the elements themselves. This is caused from the fact that the web page display is not at 100% and as a result the helpers are in a different position than the one actually expected. In order to overcome this issue, you can simple press **Ctrl+** (for zooming in) or **Ctrl-** (for zooming out) to make the Helpers circulate the elements correctly!



Helpers with offset.



Helpers with no offset.

3.9 User Libraries

3.9.1 Why Use User Libraries?

User Libraries allow you to significantly optimize your work flow and share your work with anyone you want through a number of options.

>> You can think of User Libraries as special Folders containing custom made Actions.

These will be available to you, or any other user that shares your PR installation, through the Actions pane of the *Process Designer* but you can also import/export User Libraries from other installations.

The idea behind User Actions is to minimize duplication while giving you the opportunity to produce much more readable and easy to maintain Processes.

If you:

- Have been writing repeatedly the same blocks of Actions from Process to Process,
- Wanted to be able to share readily a smart sequence of Actions with colleagues or the Process Robot community,
- Have been wishing that you could easily incorporate your colleagues' work on your scripts

...Then User Libraries are for you!

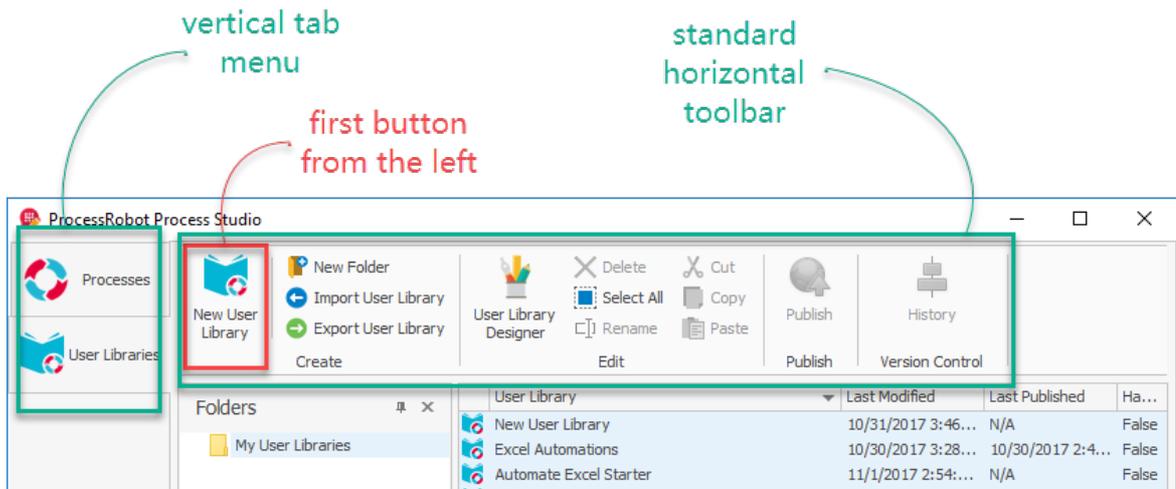
The concept of User Libraries is aligned with the following programming principles:

- [Code Reuse](#)
- [Don't Repeat Your Self](#)
- [Separation of Concerns](#)

It is also designed to enable partnership and script exchanges within the Process Robot Community, facilitating the swap of code components.

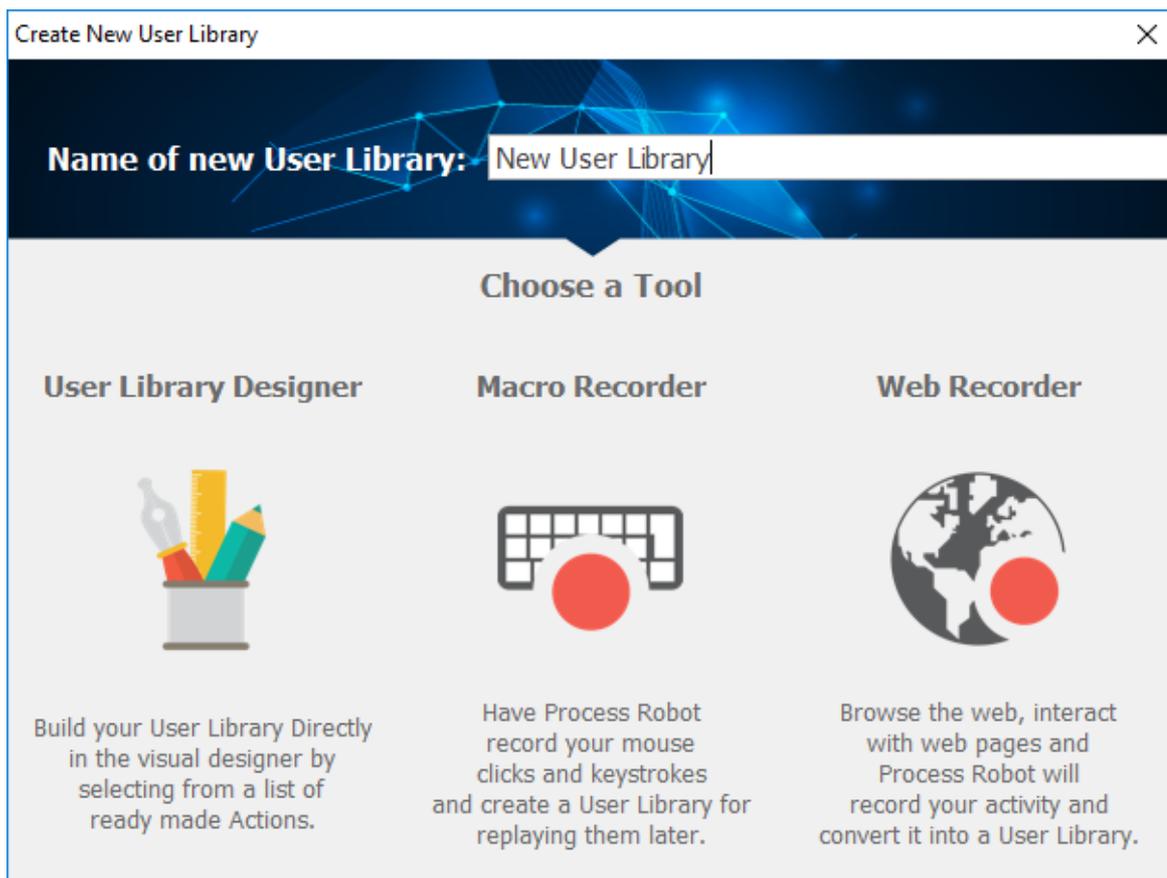
3.9.2 Accessing User Library Designer

You can access the User Library Designer (ULD) through Process Studio. Simply click the *User Libraries* tab from the vertical Tab Menu on the left of the screen and then click again on the *New User Library* button (first option from the left in the standard horizontal Toolbar).



This will result to a pop-up “Create New User Library” window asking you to define the name of the User Library and the tool you wish to use in order to start developing your library.

The *User Library Designer* will appear on your screen *directly* if you select the User Library Designer option.



If you select the Macro Recorder or the Web Recorder options, the User Library Designer will open after you click Finish on the respective Recording facilities. Just like in Process Designer, these options allow you to start your creative process through the recording and automated conversion of mouse clicks and keystrokes into Process Robot predefined Actions.

In any case, please know that whatever you choose now, you can invoke these Recorders at any time through their respective characteristic icon buttons in the main Toolbar of User Library Designer.

3.9.3 User Library Designer vs. Process Designer

The User Library Designer (ULD) has been built on top of the Process Designer (PD) and as a result the two interfaces share many design and operational similarities.

For this reason, this article will focus on their differences as these are fewer in number, more critical to know and thus more important to cover and appropriate.

If you feel that you need extra information on features the two interfaces share but are not covered in detail here, you should probably check the Process Designer [topic guide](#)^[285].

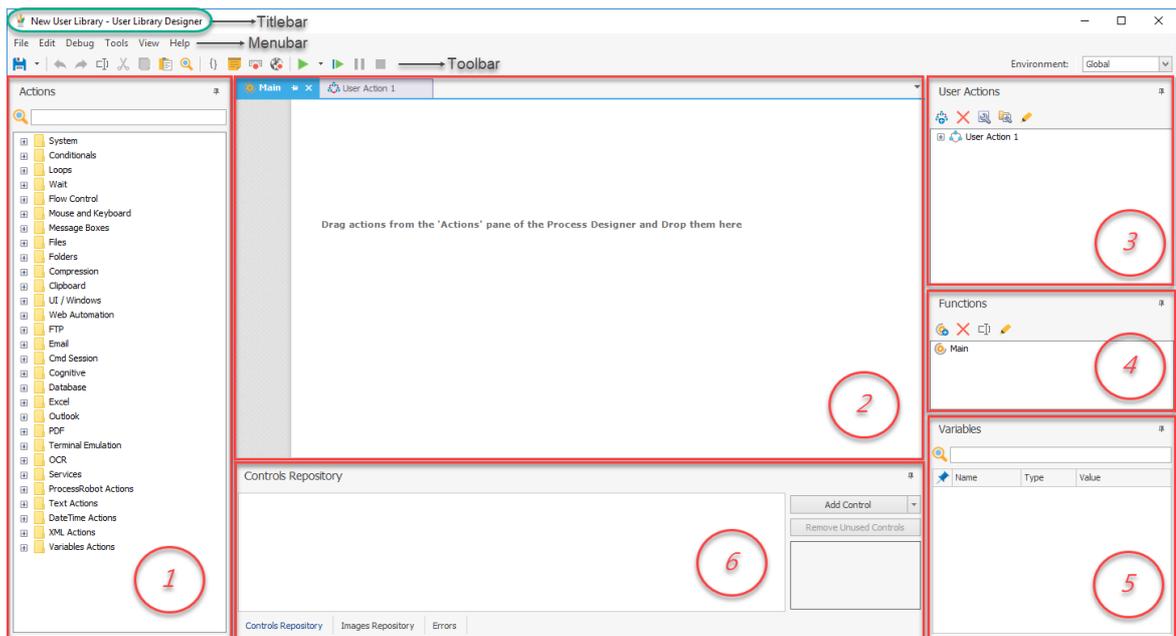
This is a list of the panes and tabs that work pretty much identically in the two interfaces:

- The Functions pane
- The Variables pane
- The two Interfaces' Main Menus and Main Toolbars
- The Controls Repository pane
- The Images Repository pane
- The Errors Pane
- The Environment Drop Down Menu (for setting the value of Global Variables according to the realities/scope of each Environment within a company)

For more info on the differences between the two components please continue reading on the [Structure of User Library Designer](#)

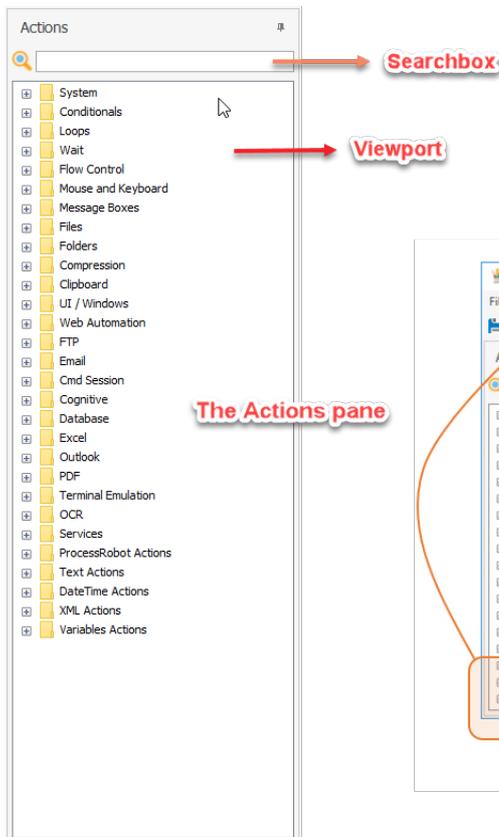
3.9.4 Structure of User Library Designer

The User Library Designer (ULD) consists of a Title Bar, a Menu Bar and a Tool Bar, along six panes that divide the contents of the window in a way designed to assist your development efforts.

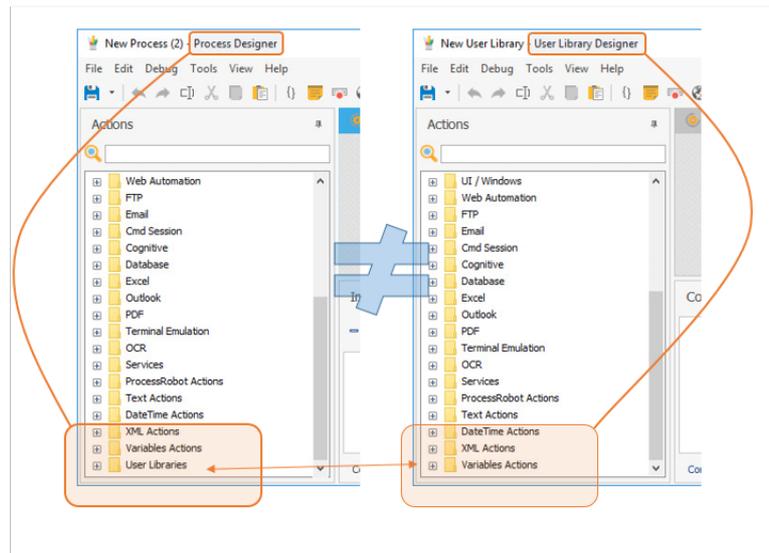


The Actions Pane

In the Actions pane you will find all the predefined Actions available to you for the development of *User Actions*, organized in 'Actions groups' through a folder structure.



Although the User Library Designer's (ULD) Actions pane is virtually identical to that of Process Designer (PD), the two panes differ in one important aspect. Unlike Process Designer's Action pane, the User Library Designer's Action pane does not contain any User Libraries, making it impossible for developers to use a User Action for the definition of another User Action *unless these come from the same Library*.



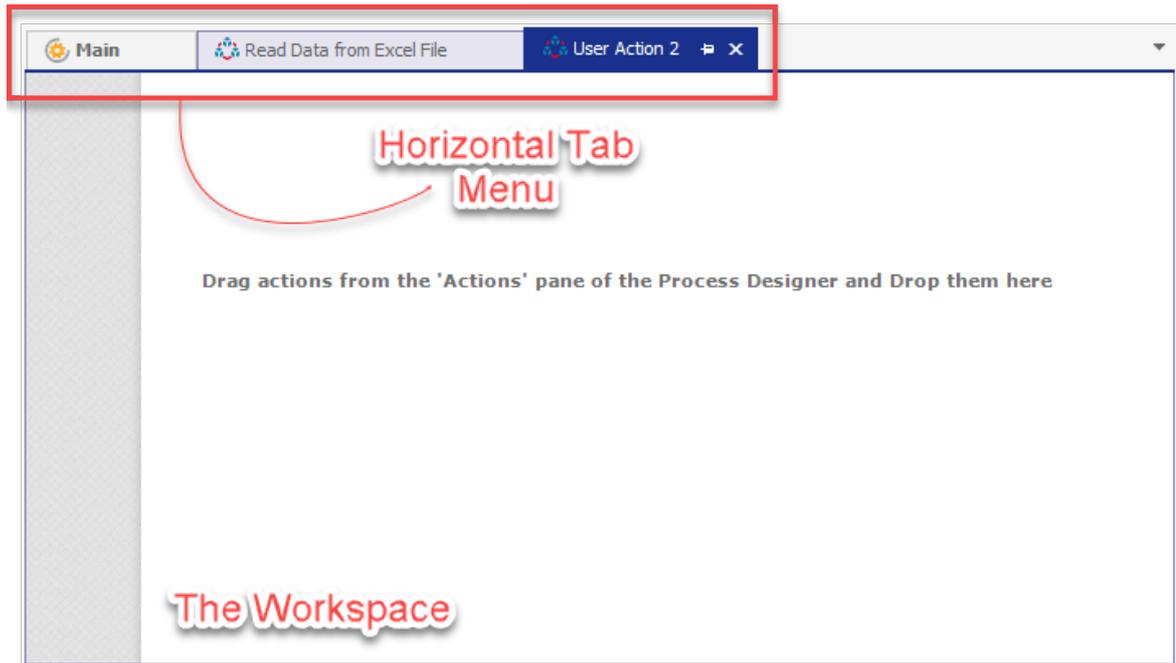
2

The Workspace

This pane displays the predefined Actions you have selected, in the order they will be executed.

User Actions can contain only the predefined set of Actions that one can find by default in the [Process Designer's](#) Action Pane and User Actions from the same User Library as this Action, but NOT User Actions from other Libraries.

The pane's Horizontal Tab Menu allows you to switch your attention from Main to any of the Library's User Actions.

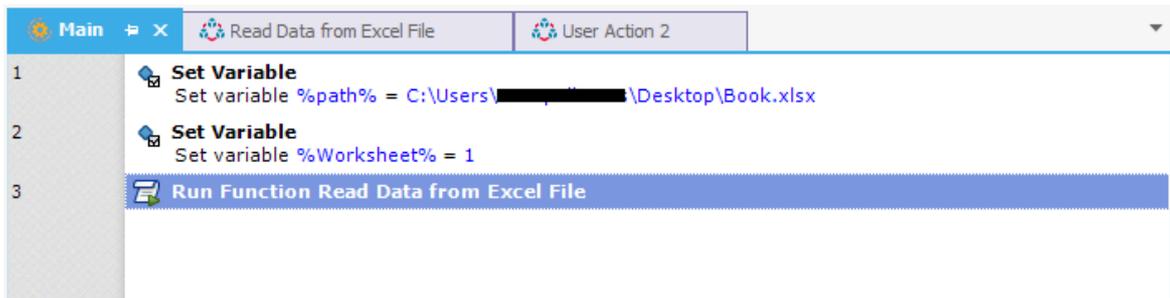


An important difference between the Process Designer's Workspace (PD) and the User Library Designer's (ULD) Workspace is the way in which you can test/run/debug your User Action.

Important! You can run/debug a User Action simply by running it as a Function in the Main Tab.

Step-by-step guide:

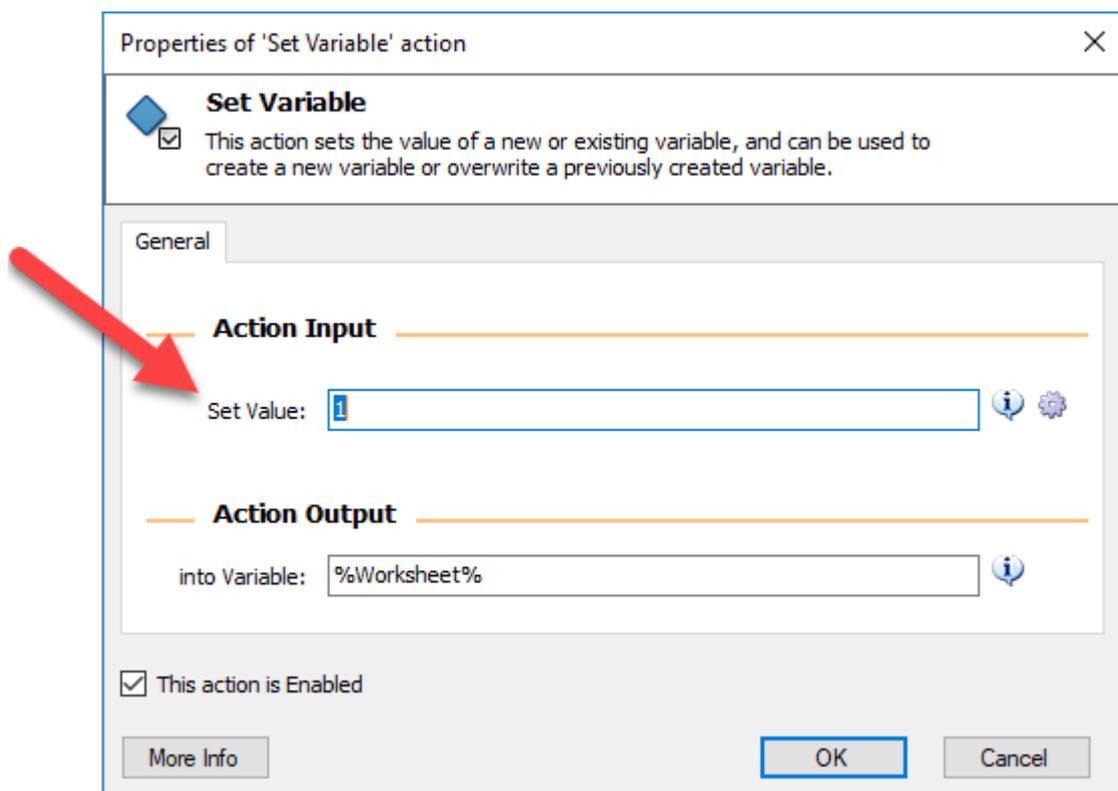
- a) Left-click on Main in the Workspace's horizontal Tab Menu
- b) Drag and Drop in it the Run Function ^[612] from the Actions pane's *Flow Control* Folder
- c) Set the name of the User Action you want to run/debug in the Run Function's Properties Dialog Box using the drop down menu options
- d) Use the Set Variable Action, as many times as it's necessary in order to provide the required input for the smooth execution of the User Action
- e) Press the Start Button (or F10) in the Designer's main Toolbar



In this diagram (from [An Example on User Libraries](#)^[423]), Read Data from Excel File is a User Action that 'compacts' four Actions used in a standard manner when we want to read a range of values from an Excel Spreadsheet.

What is needed as Input is the path leading to the xls of our choice and the index number denoting the Worksheet we wish to extract the data from.

We use the Set Variable Actions in order to set *values* into the *variables* that allow the User Action to interact with us:



Later on, when we are ready to take this User Action into Production, this step will be unnecessary as the User Action's Properties themselves will allow us to enter the necessary Input:

Properties of 'Read Data from Excel File' action

Read Data from Excel File

General Exception Handling

Action Input

Select the Worksheet: ⓘ ⚙

Path of Excel File: ⓘ ⚙

Action Output

Your Data: ⓘ

This action is Enabled

OK Cancel

3

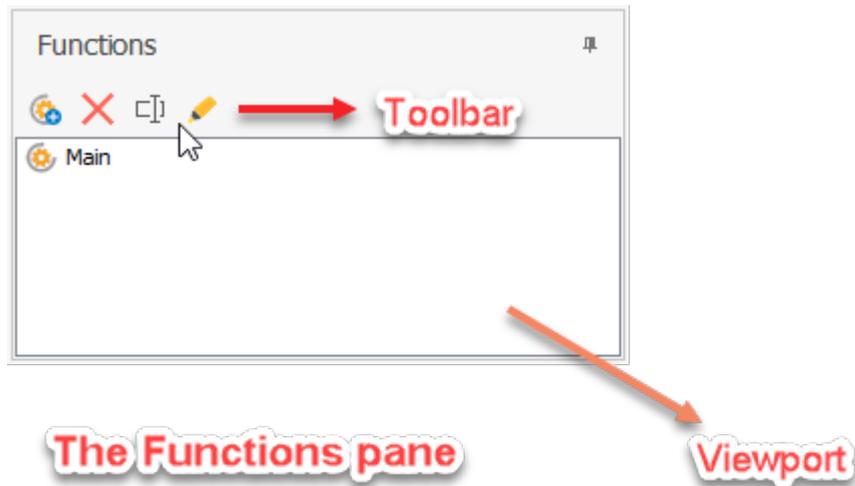
The User Actions Pane

The User Actions pane is probably the single most important difference between the User Library Designer and the Process Designer. For this reason is being analyzed as a separate [child topic](#)⁴⁰⁵.

4

The Functions Pane

The Functions pane in the User Library Designer is identical to that of [Process Designer](#) ^[285]. If you are struggling to understand the differences between Functions and User Libraries, we invite you to check the [User Actions and Functions](#) ^[413] topic of this guide.



5 The Variables Pane

The Variables pane in the User Library Designer is identical to that of [Process Designer](#) ^[286].

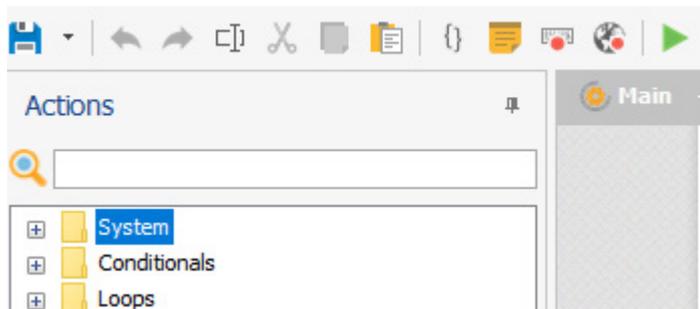
6 Controls Repository/Image Repository/Errors Pane

This tabular pane is identical to that of [Process Designer](#) ^[288].

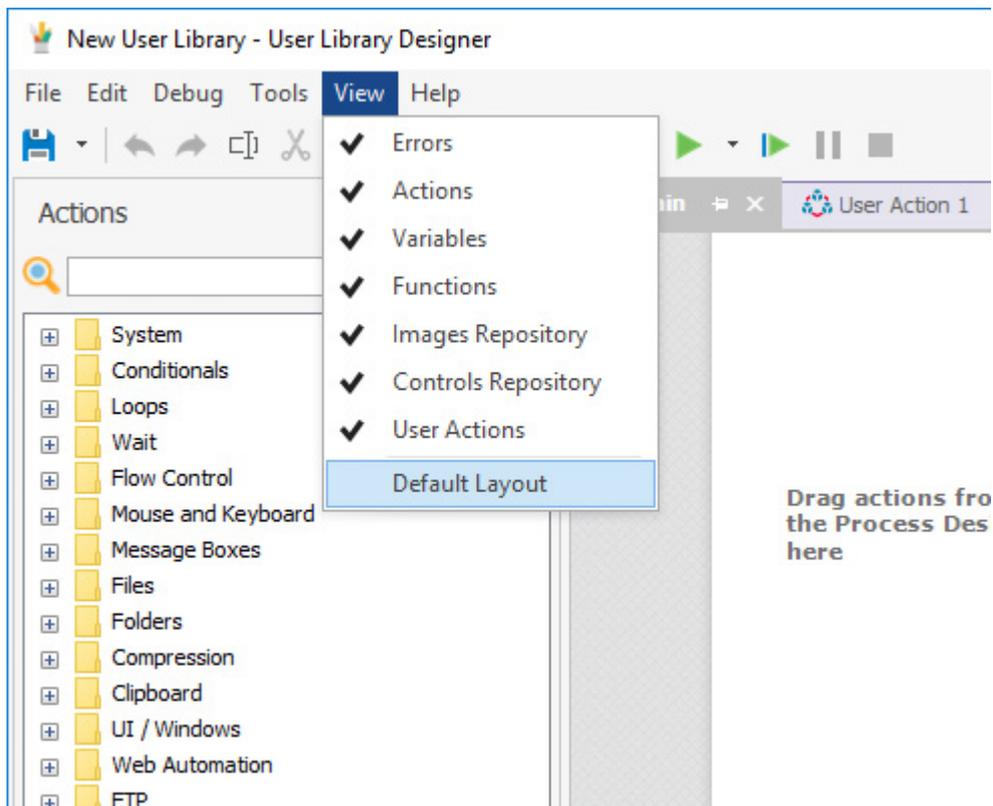
A Note on the Layout:

It is worth noting here that ULD's structural layout is flexible: not only you are able to adjust the size of the panes according to the needs of your work flow but *you can change the positioning* of the

panes using your mouse (drag and drop) or *auto hide buttons* by clicking the pin icon on the right of any of the panes' title.

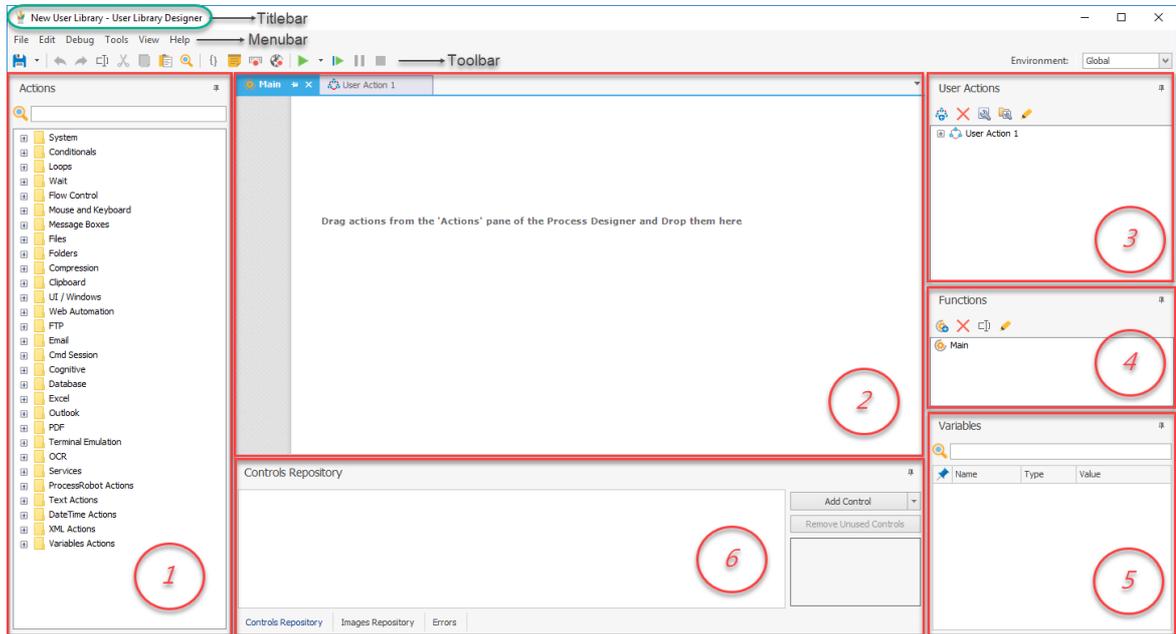


All changes are instantly reversible simply by right-clicking View from the main Toolbar and then selecting Default Layout from the resulting drop down menu.



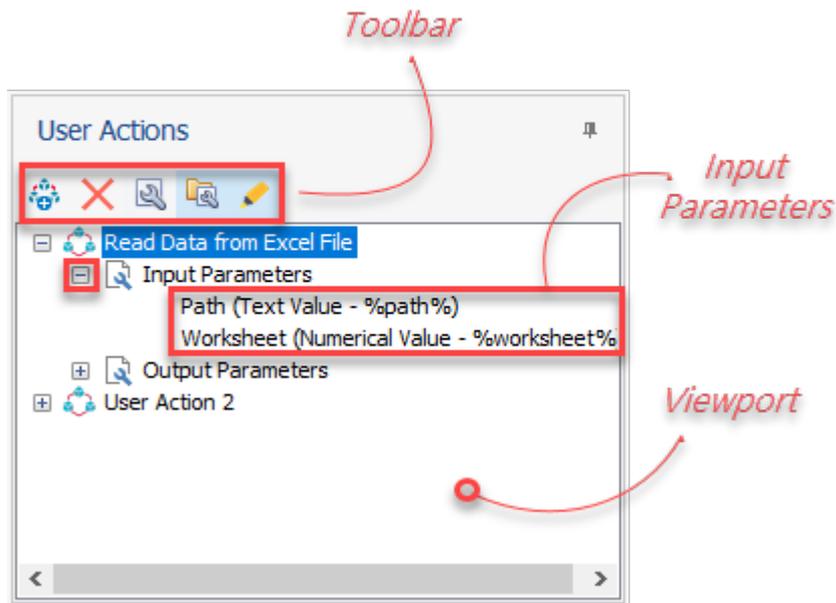
The structure of User Library Designer is *very similar to that of* [Process Designer \(PD\)](#)^[285] and that makes sense, since *User Libraries* are essentially Libraries of *portable Functions* that can work inside any Process, as custom-built predefined Actions.

3.9.4.1 The User Actions pane



3

Through this pane's toolbar you can *Create* User Actions, *Delete* existing User Actions, *Edit* a User Action's *Parameters*, *Edit the Name* and *Description of the User Action* itself or simply *Highlight* in the pane's treeview, whichever User Action is currently in development.



As you can see the pane's main viewport contains a treeview of all the Library's User Actions. Left-clicking the expand/collapse symbol next to the branches of the graphical element will allow you to access (via double-click) their Input or Output Parameters.

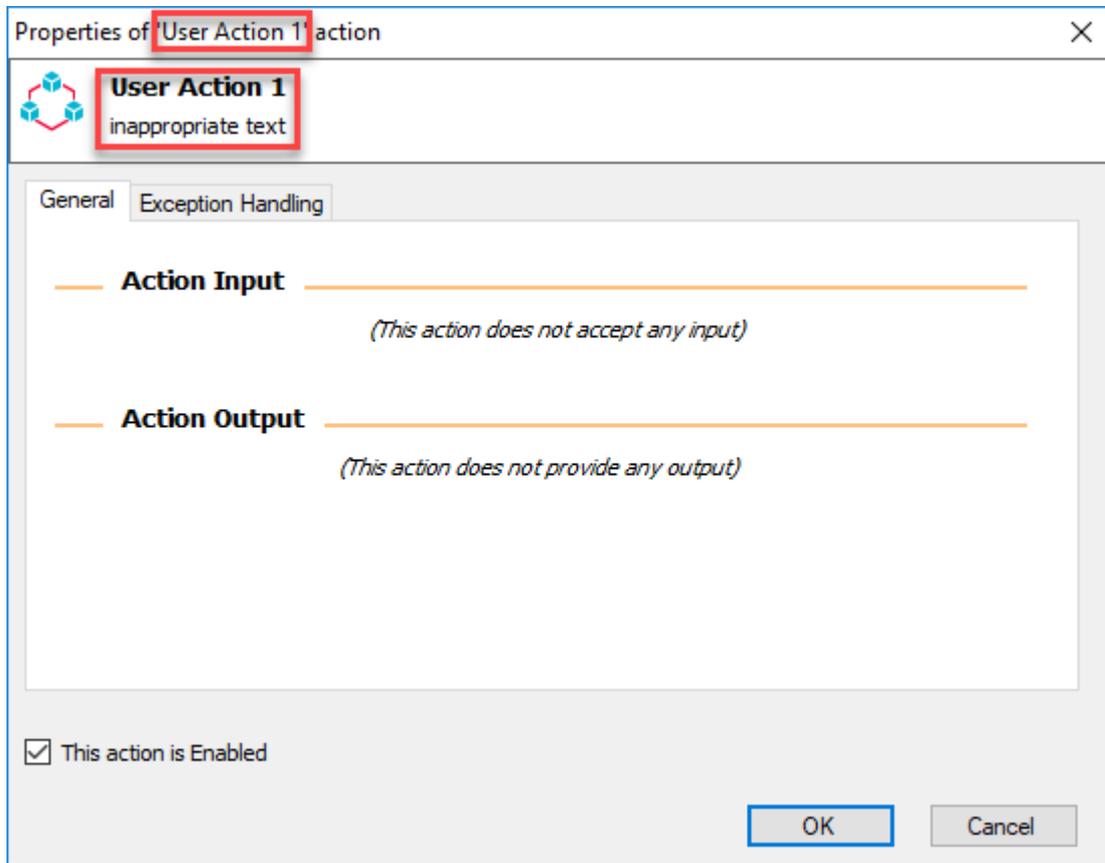
Clicking the Create New User Action



will open the Create New User Action Dialog Box.

This is comprised of two text fields that give you the opportunity to define the name of the Action along with a short description of its functionality.

It is important to understand that the information you 'll enter here will be displayed in the Action's Properties Dialog Box *every time* anyone is attempting to use this User Action.

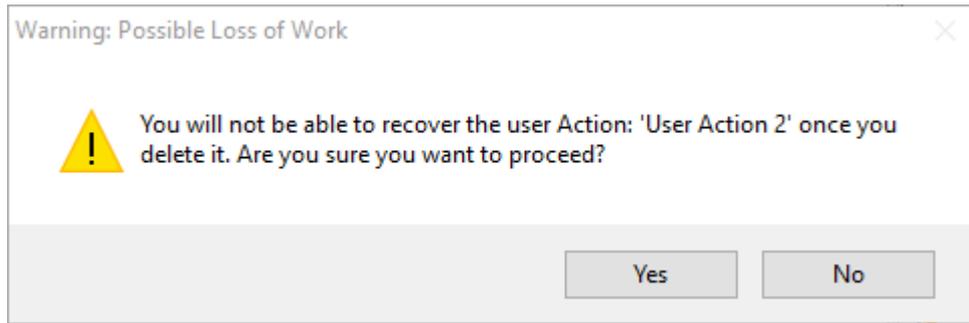


However you should also be aware that you can change this information anytime you want via the *Edit User Action Data* (fourth from the left on the toolbar).

Clicking the *Delete User Action* Button will result in the opening of a *Possible Loss of Work* alert box.

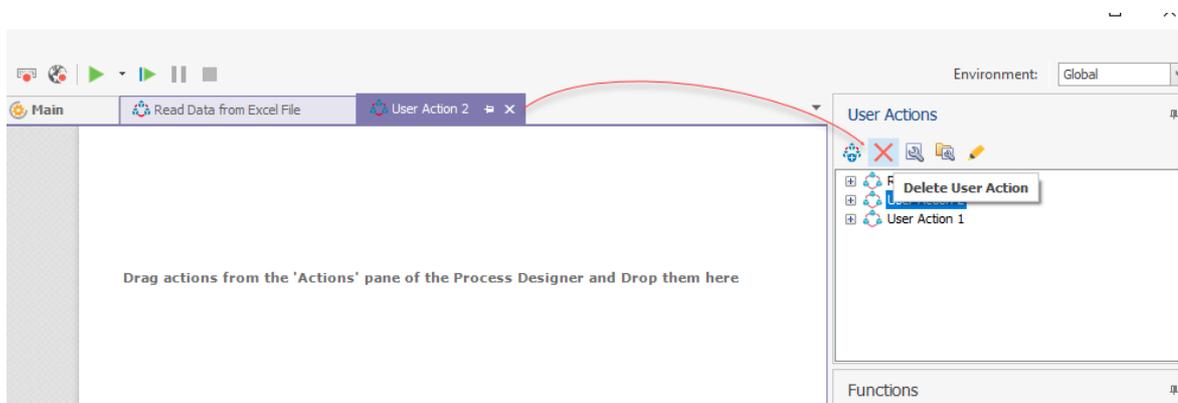


Button will result in the opening of a *Possible Loss of Work* alert box.



This is warning you against any action that might result in unintentional data loss, clearly specifying the name of the User Action you are about to delete.

The Action to be Deleted can be highlighted either through the respective Workspace Tab or the Tree View in the viewport.



Clicking the *Edit User Actions Parameter Data* 'Name-of-the User-Action' Parameters dialog box.

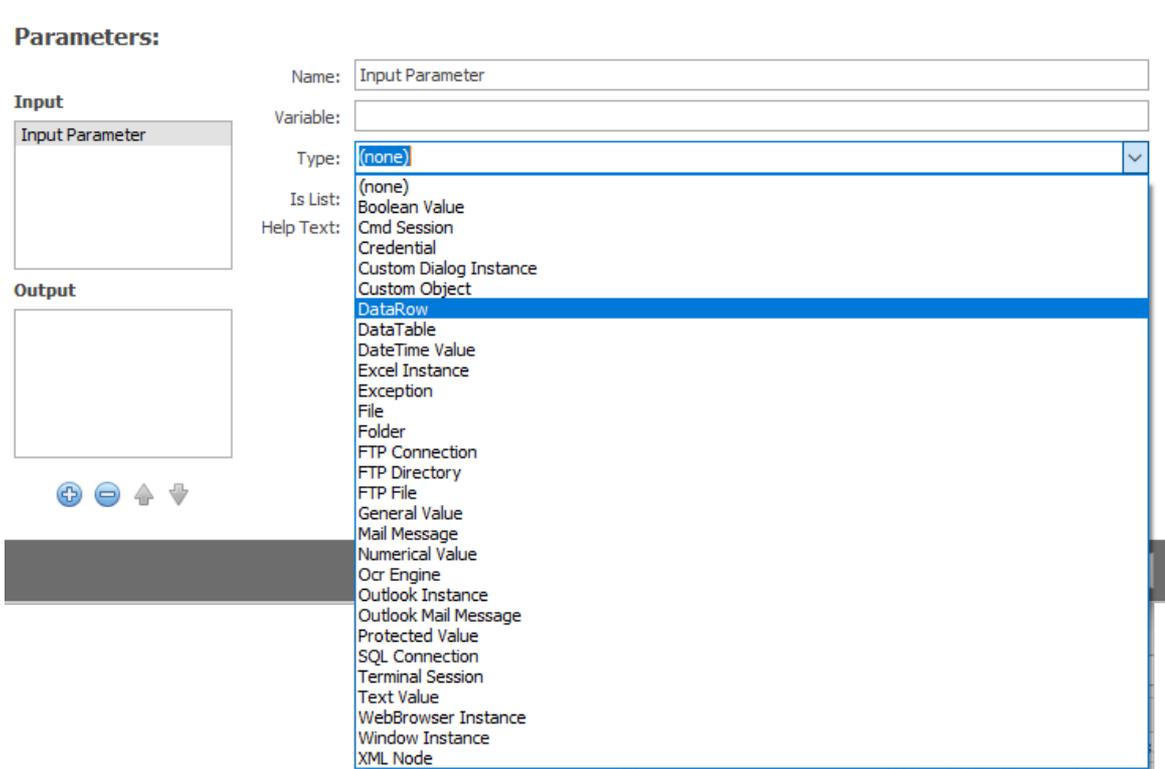


Button will result in the opening of the Edit

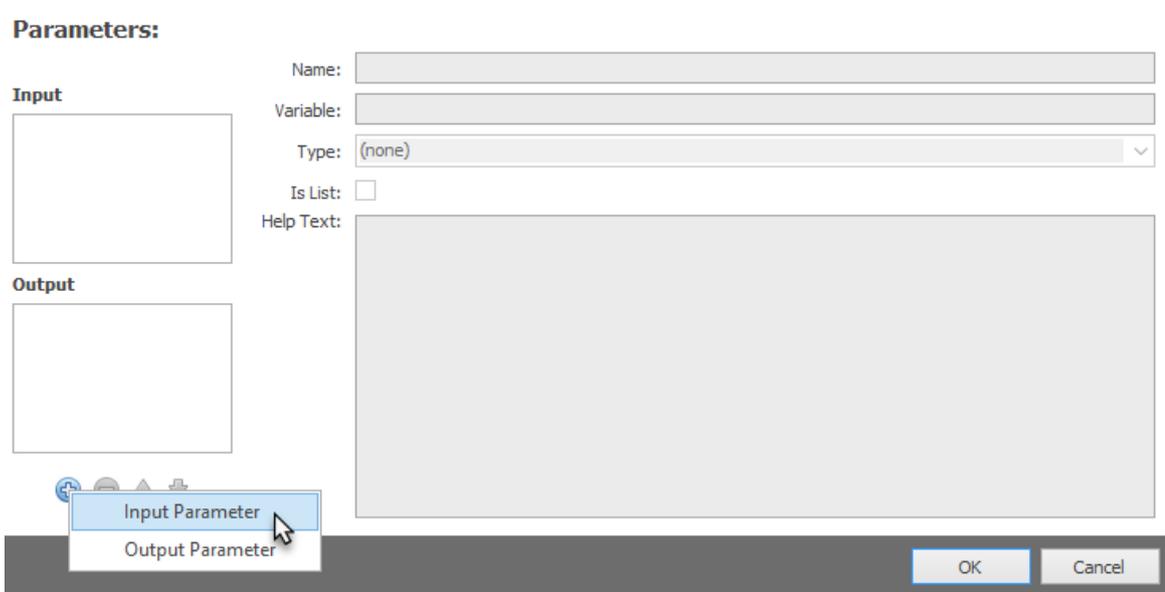
This box is very important as it is inviting you to define the input and output parameters of your Action.

Input and Output Parameters are what's allowing your User Action to communicate with anyone's using it. You can think of them as plugs, that allow your Action to accept or transmit data.

These Parameters can be of any type of data, from Boolean to a Folder but also do not have to be a single value of predefined type; it can be *a list of General Values*, Softomotive's keyword that allows variables to be defined as the program runs. However it should be clear that whether a single or a list, the value(s) have to be of the same type.



What is important to understand is how this interface allows you to enter or remove Input or Output Parameters:



To add an Input Parameter click the blue/white plus sign and then select the menu item of your choice in order for the rest of the form to become available for filling.

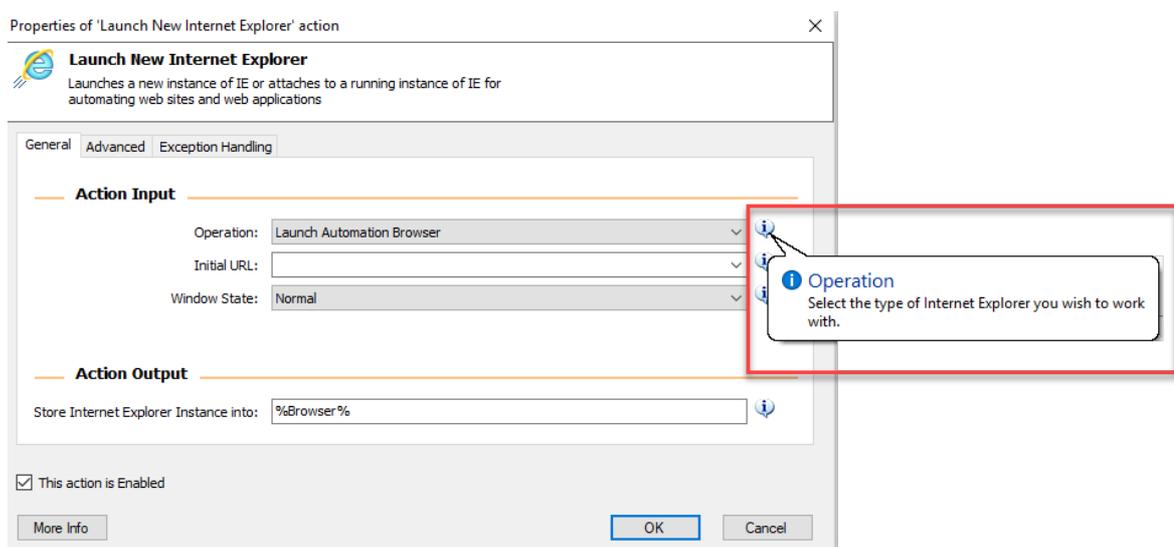
You can use the minus sign next to the plus in order to remove any of your UA's Parameters, *as long as your Library is not Published!*

In general, you should have in mind that adding or removing Parameters from a Library that is published is prohibited as this might interfere with the smooth operation of Processes that already use this specific User Action.

If you must add or remove Parameters in a published UA, you are well advised to create a new one since changes in published User Libraries take effect immediately after restarting (closing-reopening) your Process Studio facility.

Unless you use these little blue/white signs in order to add or remove Parameters, you'll be unable to complete the development of your User Library.

The Help Text you have the opportunity to set here, will be the content of the characteristic info-bubble by the side of any Action's Input or Output Parameter:



The Screenshot above is from the Properties of the predefined Launch New Internet Explorer Action. Let's see how the Message: "Wise men know themselves to be fools but fools think that they are wise" would be displayed on the Properties box of a custom made UA:

Edit 'User Action 3' Parameters

Parameters:

Input

Shakespear says

Name: Shakespear says

Variable: %wisewords%

Type: Boolean Value

Is List:

Help Text: Wise men know themselves to be fools but fools think that they are wise

Output

OK Cancel

Properties of 'User Action 3' action

User Action 3

General Exception Handling

Action Input

Shakespear says: [input field]

Action Output

(This action does not provide any output)

This action is Enabled

OK Cancel

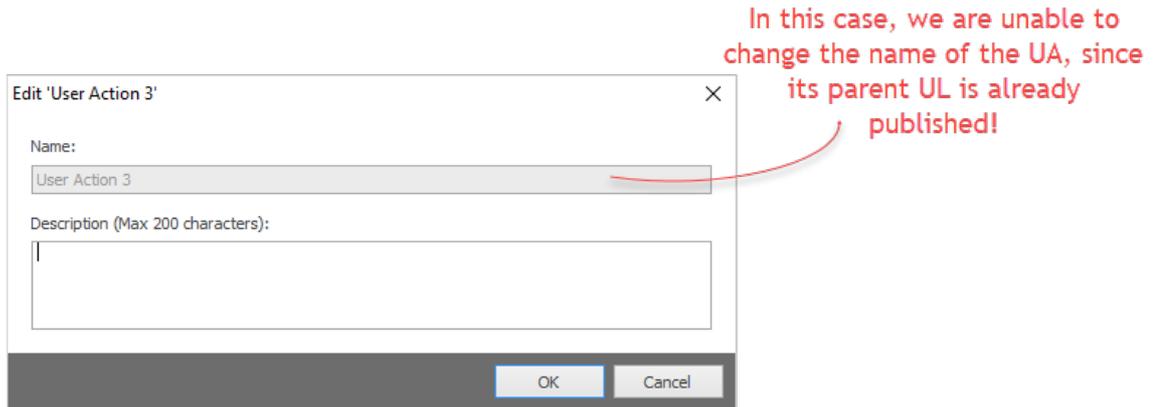
Shakespear says
Wise men know themselves to be fools but fools think that they are wise

You can find more info on how to do that along with many design tips and best practices on this guide's [An Example on User Libraries](#) ⁴²³.

Clicking the Edit User Action Data



Button will result in the opening of the Edit 'Name-of-the User-Action' dialog box.



This box is allowing you to edit the information you have previously entered with Create New User Button (first from the left on the same Toolbar).

Clicking the *Highlight Currently Active User Action*



Button will simply highlight the User Action we are currently working with in the Workspace, in the pane's treeview. This feature can be useful if the Library contains numerous Actions.

3.9.5 User Actions and Functions

A User Action can consist of as many Actions *and* Functions the user desires.

>>There are three important differences between Functions and User Actions:

1. A Function in Process Robot consists of predefined Actions grouped together so they can be invoked by a Function name.

A User Action in Process Robot consists of predefined Actions and Functions grouped together so they can be invoked by a User Action name.

2. A Function in Process Robot is available only to a single Process or a single User Library.

A User Action is readily available to ALL Processes in your Process Studio.

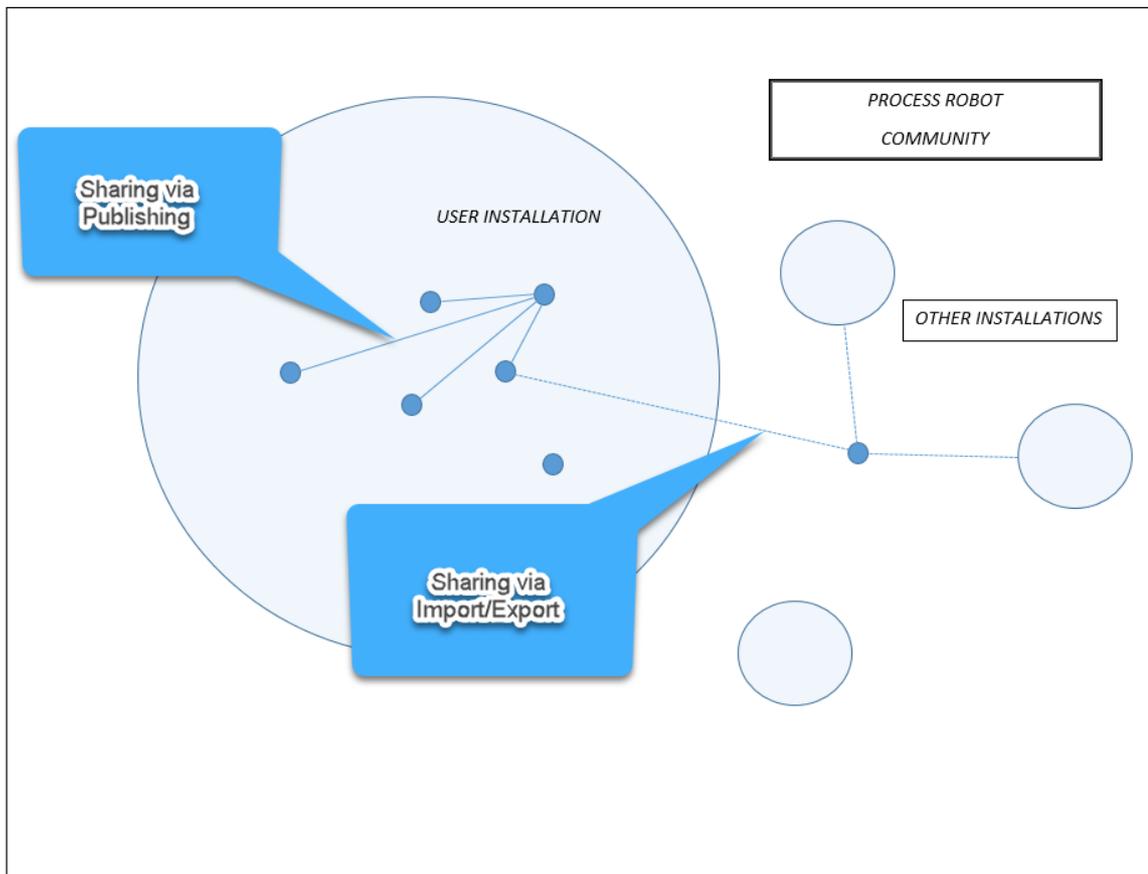
3. A User Action, unlike a Function, asks users for arguments through an interactive interface (the Action's Properties)

3.9.6 Sharing User Libraries

There are two ways in which you can share your User Library with other Process Robot users:

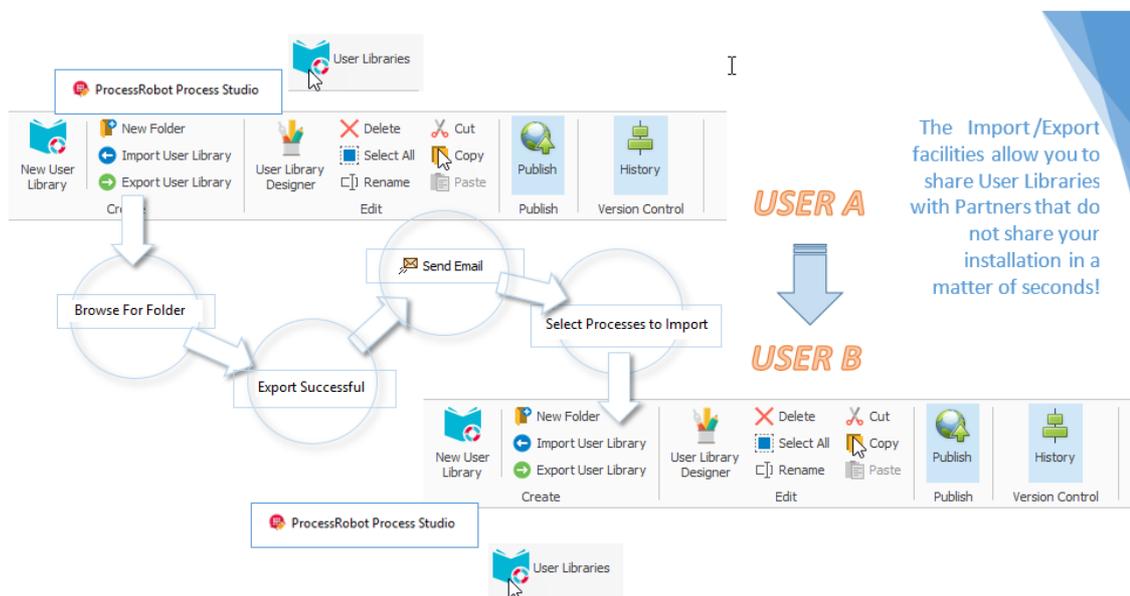
- Publishing User Libraries
- Import/Export User Libraries

What is meant by the term *Publishing* is that the User Library and all of its containing User Actions are becoming available to the Process Studio Designer of *the same installation*. To share User Libraries with users that operate on different installations, we can import/export our library as a *.pul* file through the Process Studio's Import/Export User Library button facilities (Process Studio>>User Libraries>>Import or Export User Library).



3.9.6.1 Importing/Exporting User Libraries

You can share your User Library with other Process Robot users outside your installation by exporting your Library as a pul file. Then you can treat this as an external file and send it via any conventional data-exchange scheme of your liking (usb, email, etc.) to any other ProcessRobot user that is operating outside your installation. Users on the receiving end of this transaction, can then import the Library in their topology through their Process Studio.



3.9.6.2 Publishing User Libraries

All code development on Process Robot is subject to a versioning Lifecycle that involves two steps:

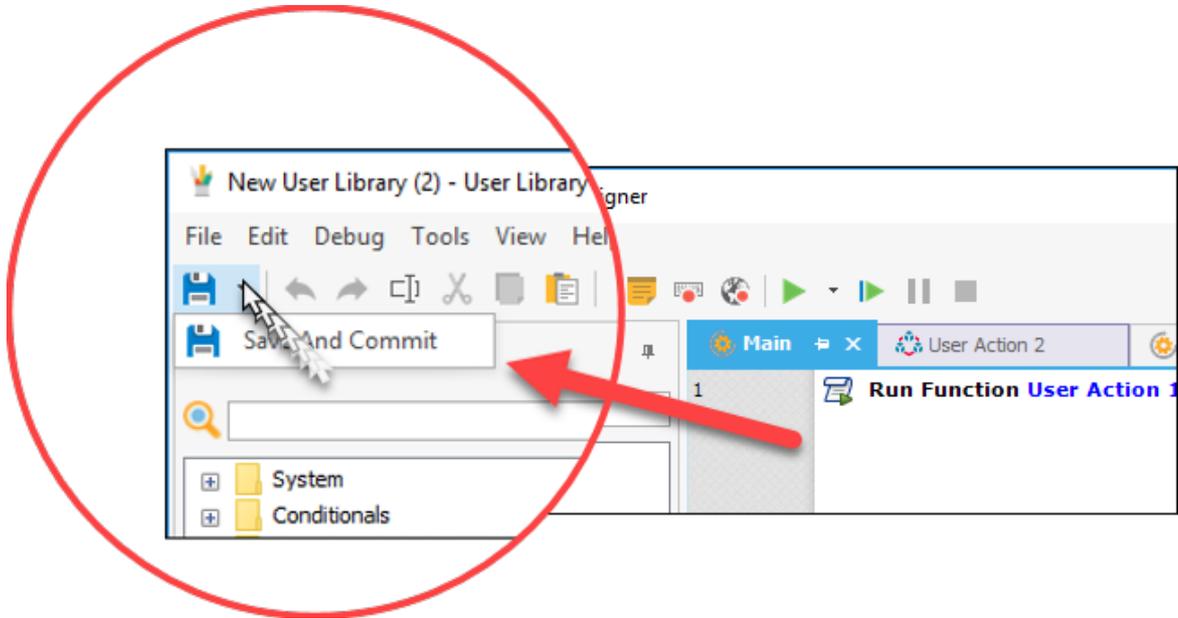
- Saving and Committing changes on a Designer
- Publishing a sequence of Actions via the Studio

User Libraries are not exempted of this reality. This article can be seen as an easy-to-follow guide on how to go through the Lifecycle of a User Library.

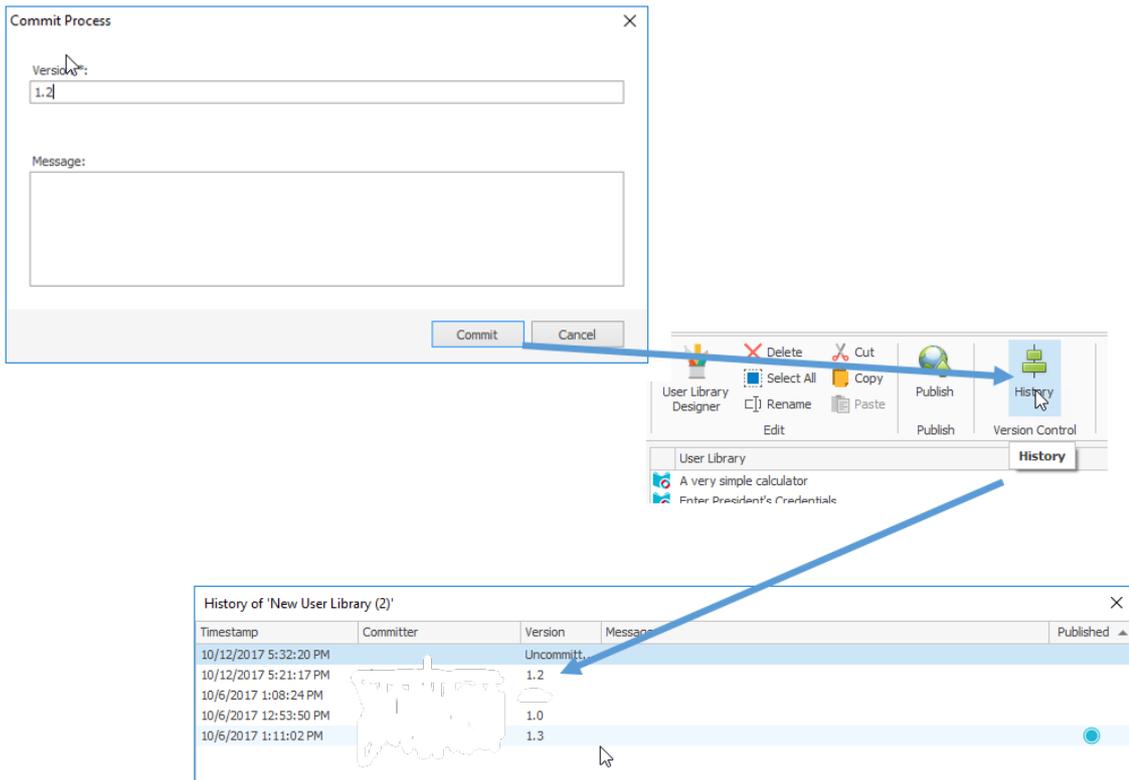
- Saving and Committing changes

Saving and committing changes in User Libraries is straightforward and simple.

After creating a User Library we can select to Save and Commit any changes through the Save Button's drop down "Save and Commit" option (Save can be found first from the left in the Designer's main Toolbar).



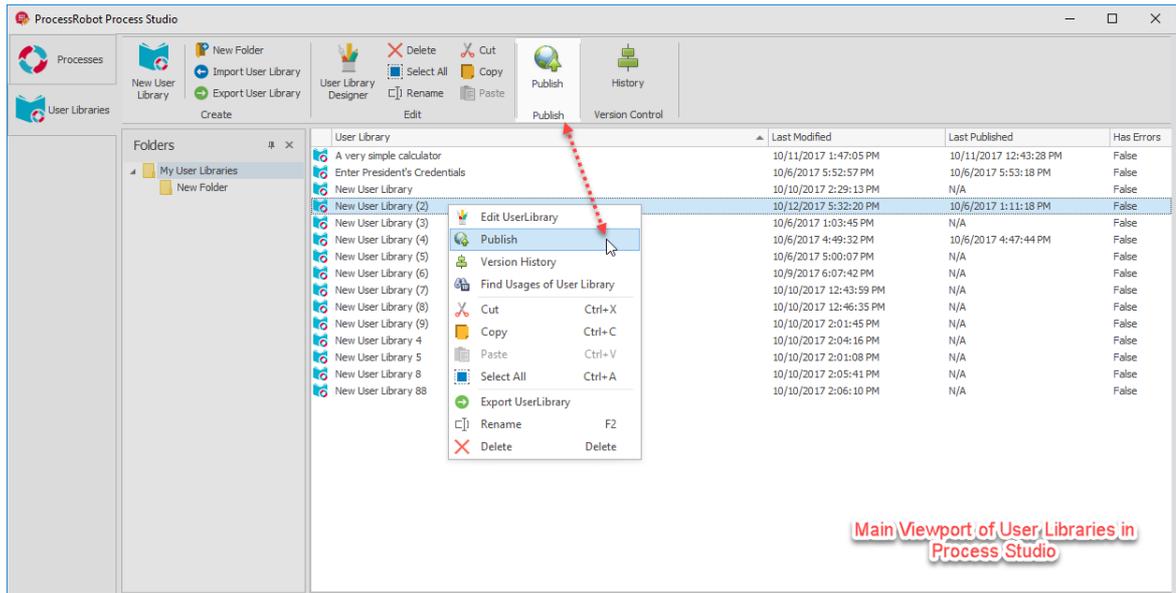
This will launch the Commit Process Dialog Box inviting us to set the name of the *Version* along with a message about this specific Version. Following this, the new commit will immediately appear in the History Dialog Box of the specific User Library available through the Version Control History Button of the main User Libraries Toolbar in Process Studio.



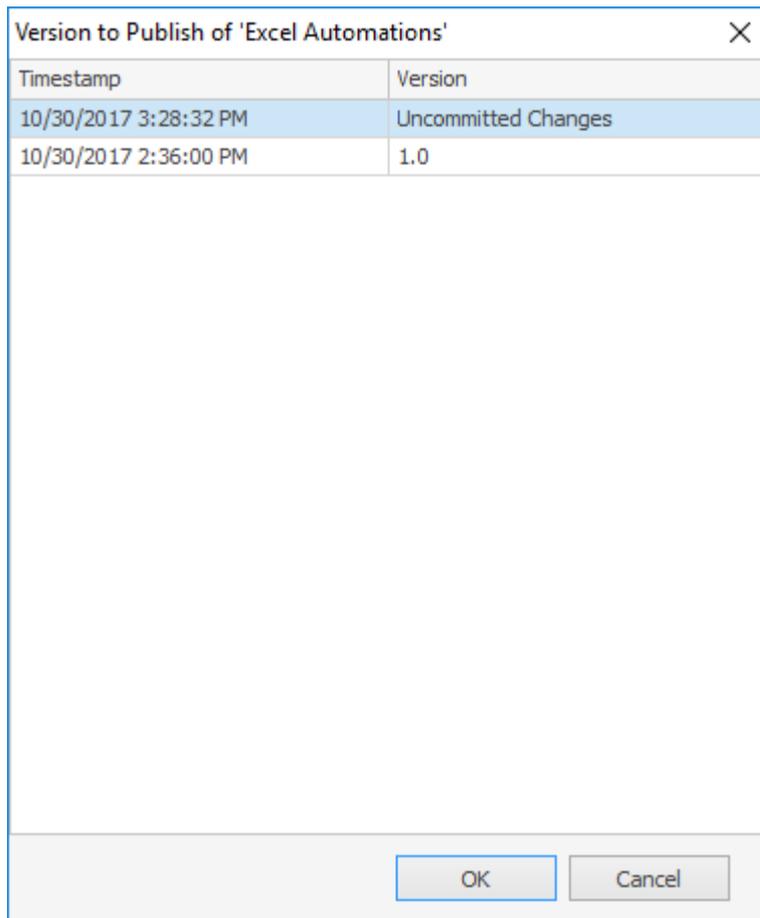
This Version of the Library however won't become Published unless we decide to do so, through an equally simple and straightforward publishing process.

b) Publishing a User Library

To publish any Version of a User Library simply right-click on the Library of your choice in the User Libraries main viewport and then select Publish (alternatively you can highlight the Library of your choice and then click on the main Toolbar's Publish Button).

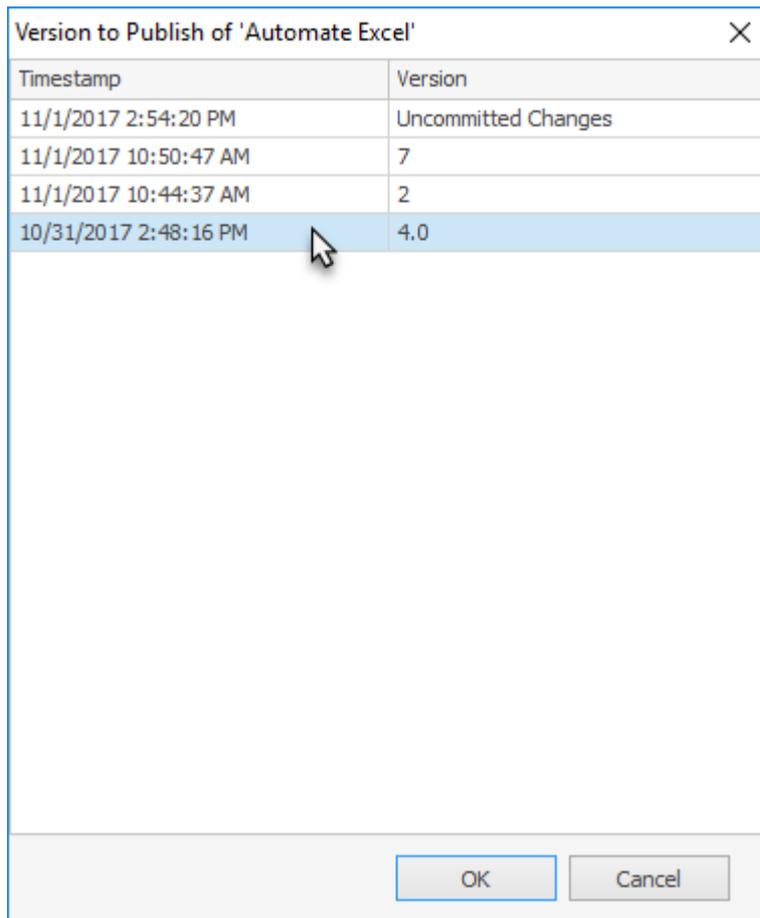


This will launch "The Version to Publish" dialog box that is giving you the option to *select* the Version of your Library you actually want published and then click OK (clicking Cancel will close the Dialog Box without saving your choice).



A key concept that you need to have in mind throughout the process of developing User Libraries is that there can only be one Version of a User Library published at any time.

You can change which Version of your Library is published whenever you want, by simply selecting the Version you wish to publish from the dialog box and then press OK.



The Publish Button in the Studio's main Toolbar allows you to quickly control which version of your User Library is published through the *Version to Publish* dialog box, while the *History Button* (Version Control) allows you to quickly review which Version is published through the History dialog box.

Timestamp	Committer	Version	Message	Published
10/30/2017 3:28:32 PM			Uncommitt...	
10/30/2017 2:41:13 PM	Alexandros Anapolitanos	2.1		<input checked="" type="radio"/>
10/30/2017 2:36:00 PM	Alexandros Anapolitanos	1.0		<input type="radio"/>

3.9.7 Editing User Libraries

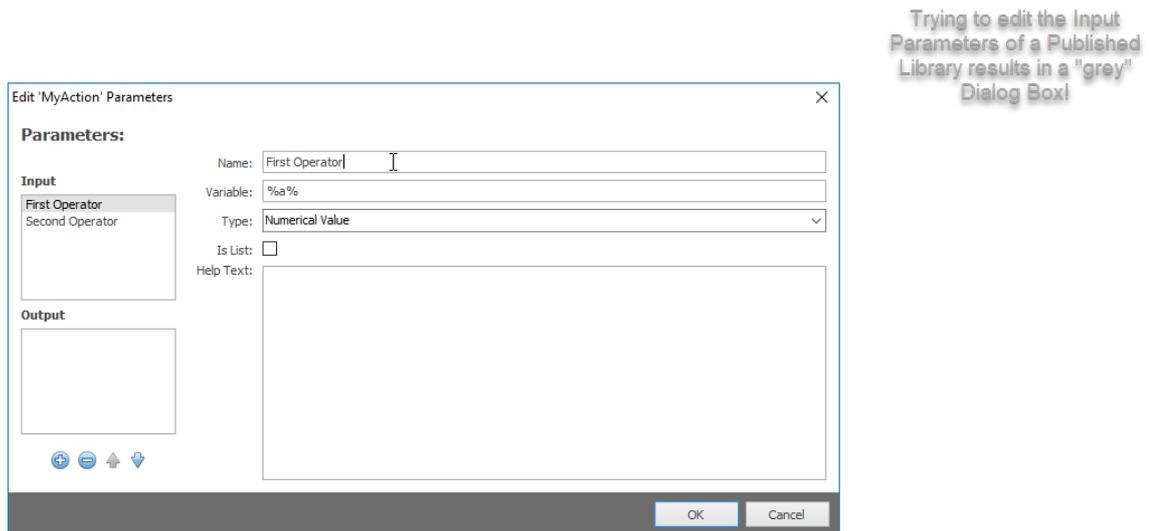
Right-clicking on any Version item of the History Dialog Box gives you the option to Checkout this Version. Checking-out a version will create an uncommitted instance of that Version available to you for editing the moment you will open your User Library with User Library Designer. (To open a Library simply double click on the User Library Icon on the left of its name in the central viewport.) This is also why the program requires you first to close any open Versions of this User Library in your machine before you are able to open a checked out version of any specific Library.

New User Library (2)	10/12/2017 5:32:20 PM	10/6/2017 1:11:18 PM	False
New User Library (3)	10/6/2017 1:03:45 PM	N/A	False
New User Library (4)	10/6/2017 4:49:32 PM	10/6/2017 4:47:44 PM	False
New User Library (5)	10/6/2017 5:00:07 PM	N/A	False
New User Library (6)	10/9/2017 6:07:42 PM	N/A	False
New User Library (7)	10/10/2017 12:43:59 PM	N/A	False
New User Library (8)	10/10/2017 12:46:35 PM	N/A	False
New User Library (9)	10/10/2017 2:01:45 PM	N/A	False
New User Library 4	10/10/2017 2:04:16 PM	N/A	False
New User Library 5	10/10/2017 2:01:08 PM	N/A	False
New User Library 8	10/10/2017 2:05:41 PM	N/A	False
New User Library 8	10/10/2017 2:05:41 PM	N/A	False

If you are experiencing difficulties opening a library, simply try double-clicking on the icon in its left side

Users should be aware that editing already published libraries has its limits. You cannot add, remove or change the name of Input or Output parameters, you can simply tamper with the Library's logic (add or remove actions in the Workspace, add or remove controls etc).

This is essentially a form of protection of one's own self, as changes in Published Libraries are taking effect immediately, influencing all Processes in which they are playing an active role. If you feel it absolutely necessary to add or remove Parameters from your Action then you are advised to create a new User Action within the same Library (all Functions of a User Library are accessible from all of its User Actions).



3.9.8 Permissions and User Libraries

From The Control Desk you can Allow or Deny or Deny permissions to users who play different Roles within your Company.

One of these fields is related to User Libraries and more specifically with the:

- Create/Edit User Libraries
- Deletion of User Libraries
- Publishing of User Libraries
- and the Import/Export of User Libraries

This last field is quite important as it gives the permission to people within your company to easily import or export User Libraries *outside* your installation.

Another noteworthy aspect of these permissions is that allow to all the usage of User Libraries.

Finally the user should be aware that these permissions affect *the whole content of a User Library*, that means all of the User Actions inside a User Library.

Edit Role [X]

Role Name:

Permissions:

Permission	Allow
Process	
Create/Edit Process	<input type="checkbox"/>
Delete Process	<input type="checkbox"/>
Execute Process	<input type="checkbox"/>
Import/Export Processes	<input type="checkbox"/>
User Libraries	
Create/Edit User Libraries	<input checked="" type="checkbox"/>
Delete User Libraries	<input type="checkbox"/>
Publish User Libraries	<input type="checkbox"/>
Import/Export User Libraries	<input type="checkbox"/>
Folder	
Create Folder	<input type="checkbox"/>
Rename Folder	<input type="checkbox"/>
View Content	<input type="checkbox"/>
Delete Folder	<input type="checkbox"/>
Trigger	
Create/Edit Trigger	<input type="checkbox"/>
Delete Trigger	<input type="checkbox"/>
Schedule	
Create/Edit Schedule	<input type="checkbox"/>

3.9.9 An Example on User Libraries

Read Data from Excel File

Reading Data from Excel Files involves four standard steps that you are very likely to repeat *a lot* while you are working with Softomotive products. At this example we are going to show you how User Actions can help you avoid repeating yourself while using your favorite code components across all your Processes in an instance!

Identifying the opportunity:

Identifying the chance to use a User Action is important as it will give you the opportunity to make your life easier and productive. Reading data from an Excel File can be a good Case Study as it makes use of four standard Actions:

- Launch an Excel Instance using a specified Path,
- Set an Active Excel Worksheet using a specified index,
- Get First Free Column/Row from the specified Worksheet and
- Read & Save your Data!

Thinking in terms of Input & Output:

Transforming these four Actions into one means that the User that will use this compacted script, should be able to enter the same amount of Input it would have been otherwise asked to provide for the smooth execution of this sequence of Actions. In other words, in order to compact our 4 Actions script into a single Action, we need to make sure that we provide these interactive "slots" in this new User Action's interface so everything can fall into place harmoniously.

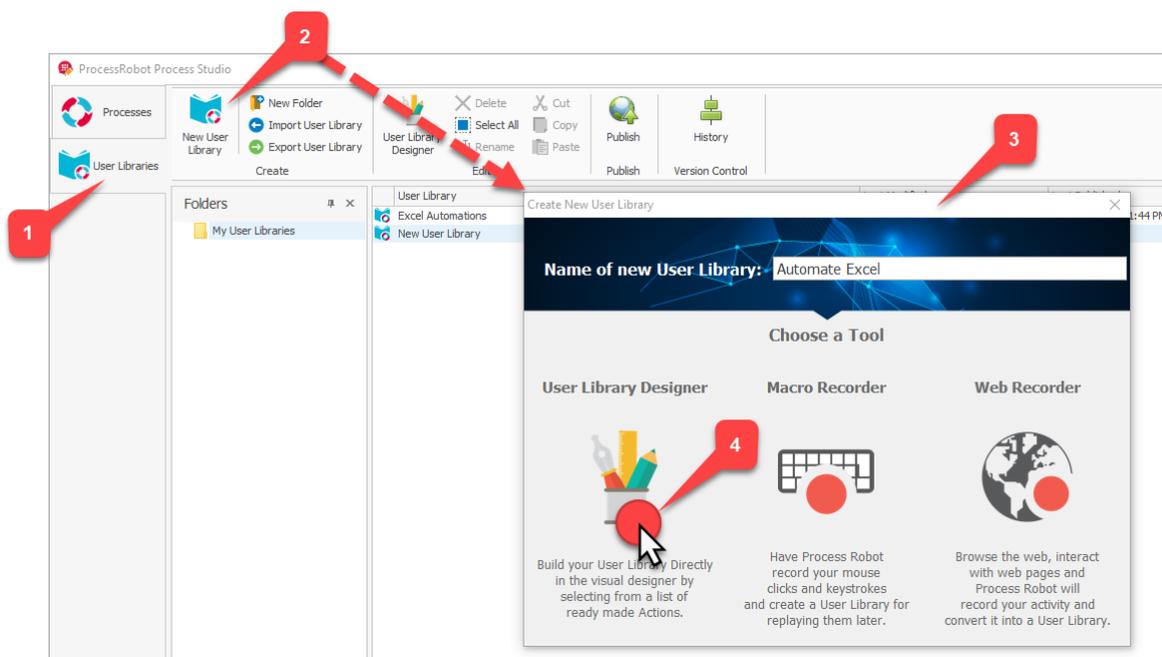
In the same spirit we need to ensure that the output of our Action is available somehow to other segments of our code. User Actions have been designed to allow you to compartmentalize your code in a manner that is handy and functional while offering you the full interactive interface of a predefined Action.

A 3 steps procedure:

Having said that, we are ready to jump on how to create, commit and publish our User Action!

1. Open the User Library Designer

Opening the User Library Designer (or ULD) is the first step in order to create and share your User Action. The following diagram shows you in detail how you can access the ULD from Process Studio.

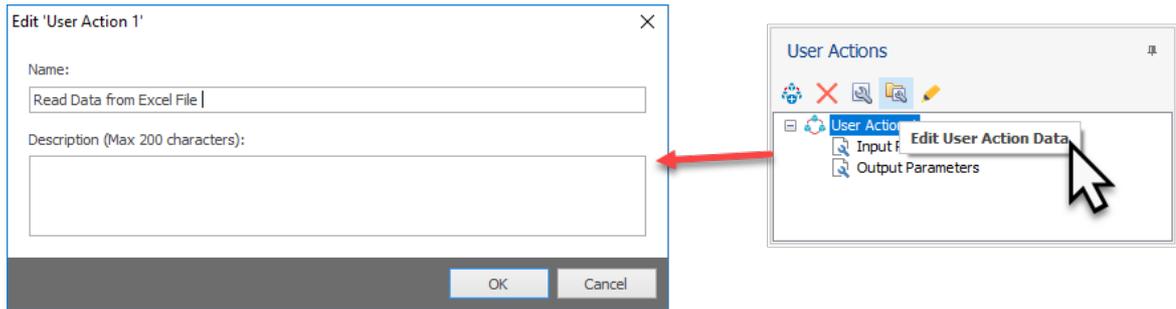


As you can see we have decided to name our Library "Automate Excel". User Libraries are containers of User Actions, so you might find it convenient to give them descriptive names that make the grouping and allocation of User Actions intuitive and easy to work with.

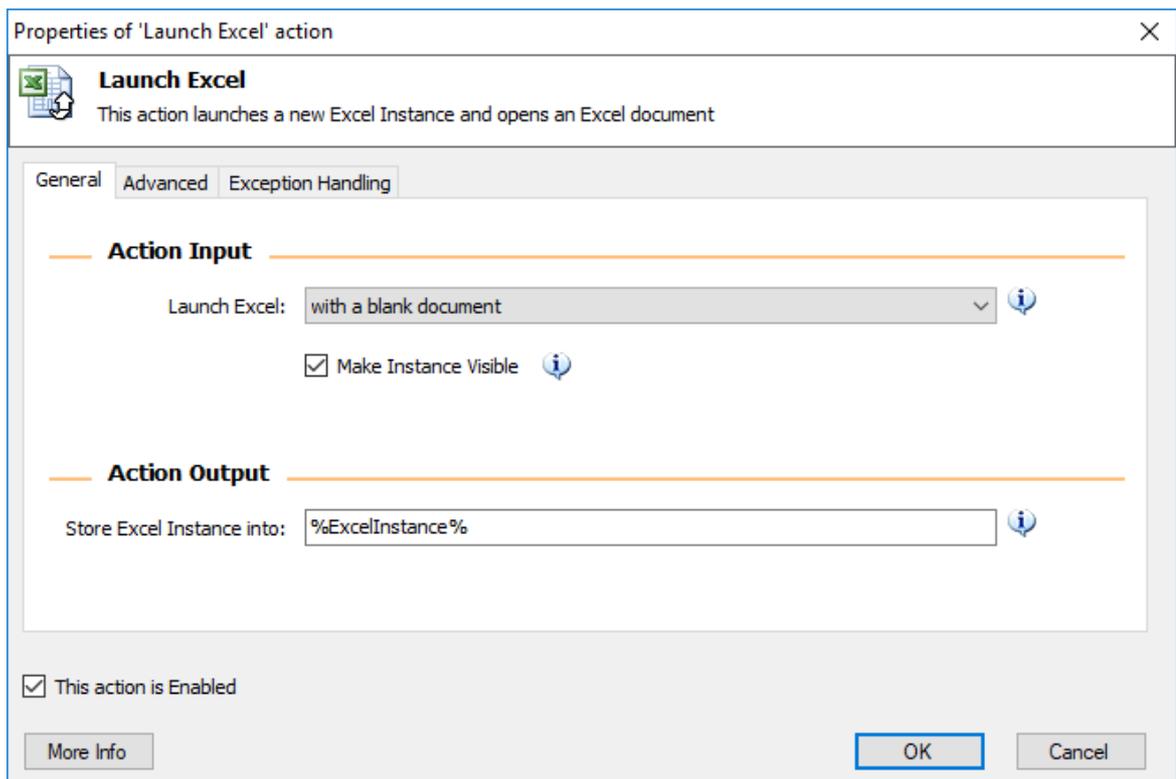
2. Developing your User Action

In order to start working on our first User Action - in this case "Read Data from Excel File" - you need to direct your attention on the User Actions pane.

As you can see, the program has been anticipating your intentions and has already created a User Action by the name *User Action 1*. However by pressing the fourth button from the left in the pane's toolbar (Edit User Action Data) we can rename whichever User Action is highlighted in the tree-view. So, by left-clicking once on User Action 1 and then pressing the aforementioned button, we can observe the Edit 'User Action 1' dialog box popping in our screen, giving us the opportunity to rename the specific User Action into "Read Data from Excel File".



Left-clicking on the expand (+) of the User Action in the tree-view will reveal two branches: (a) the Input Parameters and (b) Output Parameters sub-items. Input and Output Parameters are the terms we use in order to describe the interactive slots that helps us define the structure of what should be familiar by now, Action's Properties dialog box. If you are not familiar with the Properties dialog box you are strongly advised to spend some additional time reading our documentation on Process Designer before venturing into User Libraries.



In order to add or remove Parameters, we need but to double-click on the appropriate branch (Input when we desire to construct an interactive box through which the User will be able to *insert*

information and output when we wish to create an outlet that will give us the opportunity to export the outcome of our Action inside the scope of a larger script)

Double-clicking on Input Parameters will result into the Edit 'User Action 1' Parameters dialog box:

Parameters:

Input

Output

Name:

Variable:

Type: (none)

Is List:

Help Text:

OK Cancel

For the purposes of our example, we need to create at least two Input Parameters in order to allow users to state **which file and Worksheet they want to extract data from** (the xls's path and the index number or name of the Worksheet).

To do so, as it has been explained in detail before, we'll press the blue/white plus icon at the bottom-left and then clarify that the addition concerns an Input Parameter. At this point, the right hand-side contents of the box will become accessible (de-greayed) and we'll be free to enter the name of the Parameter, the Variable that will hold its value, its Type and specify whether it is a List or not, along with some optional Help Text.

Edit 'Read Data from Excel File' Parameters

Parameters:

Input

Path

Output

Name: Path

Variable: %path%

Type: Text Value

Is List:

Help Text: This is the path that should lead to the xls of our choice

OK Cancel

After pressing OK, we'll repeat the exact same process in order to add another Input Parameter that will define the index number of the Worksheet we wish to extract data from.

Edit 'Read Data from Excel File' Parameters

Parameters:

Input

Worksheet

Output

Name: Worksheet

Variable: %worksheet%

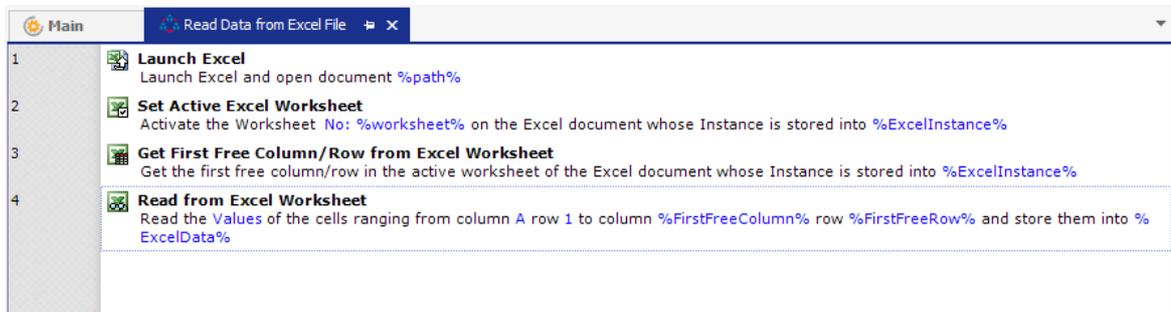
Type: Numerical Value

Is List:

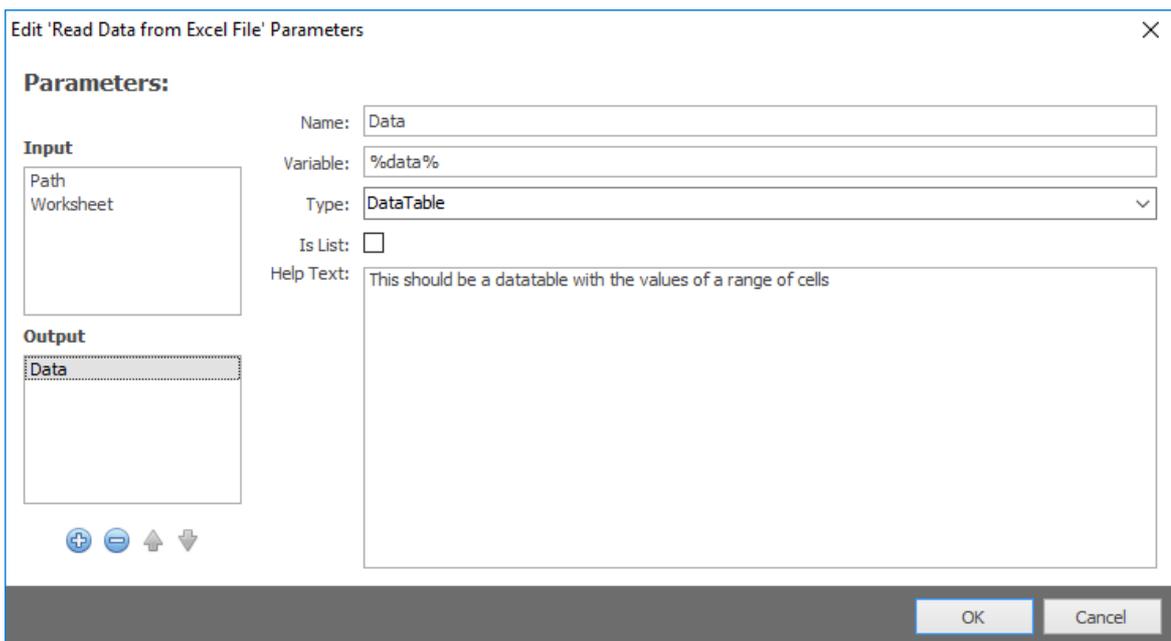
Help Text: This is the index number that should point to the Worksheet of our choice

OK Cancel

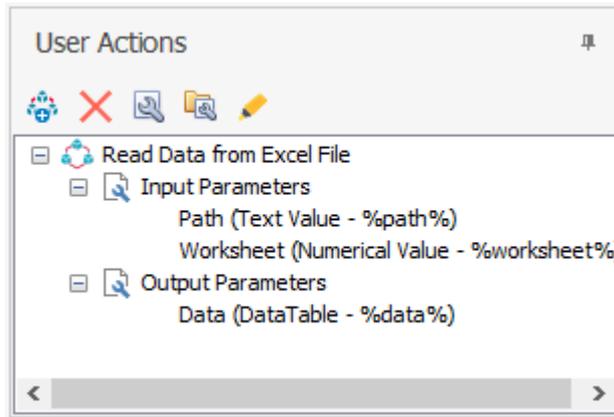
Now, we are ready to create the sequence of Actions in our Workspace (we won't go to any details here since we are assuming that the reader is already familiar with the basics of creating a simple process). The end result should look something like that:



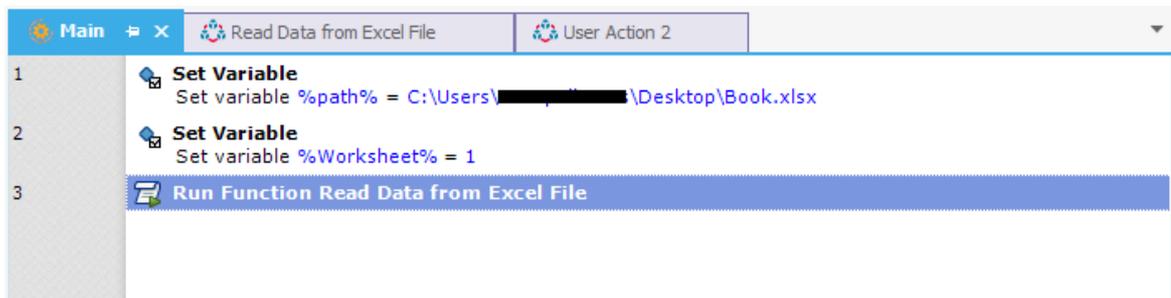
At this point we are ready to introduce an Output Parameter following the exact same process we did for the Input Parameters except clicking the Output option instead of the Input option.



At this point the Treeview should look something like that:



Now, in order to test our User Action (or UA) we can Run it as Function from Main, after having set appropriate values for the variables representing our Input Parameters (for that purpose we can use Set Variable Actions).

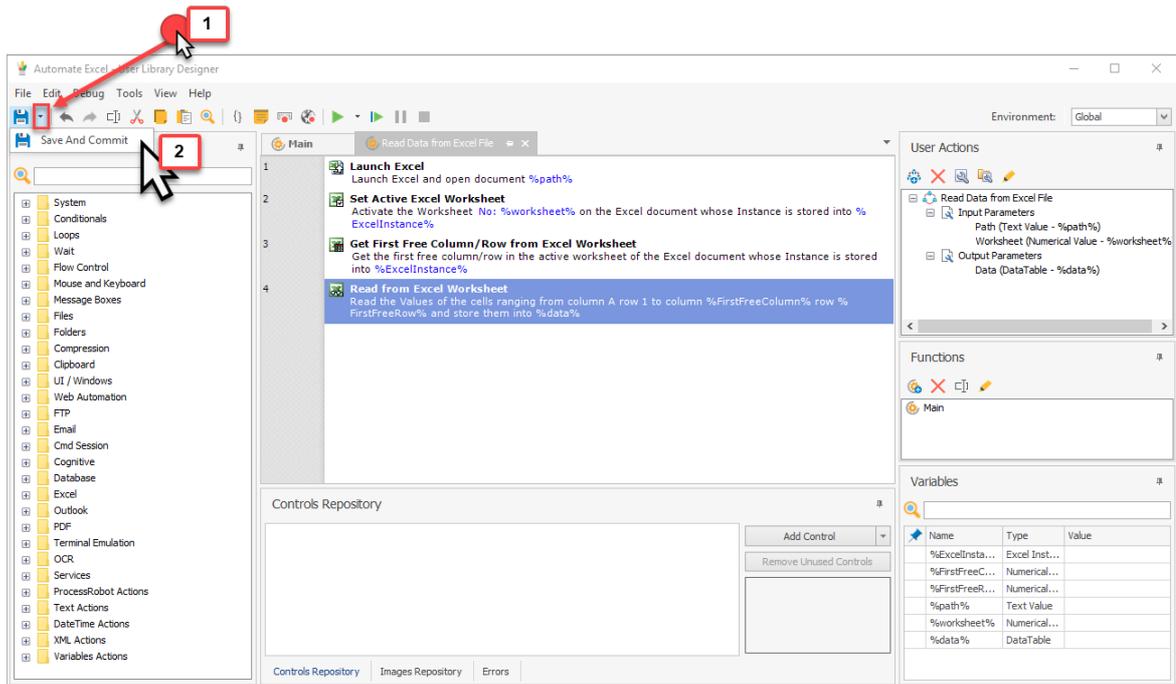


3. Publishing your User Library

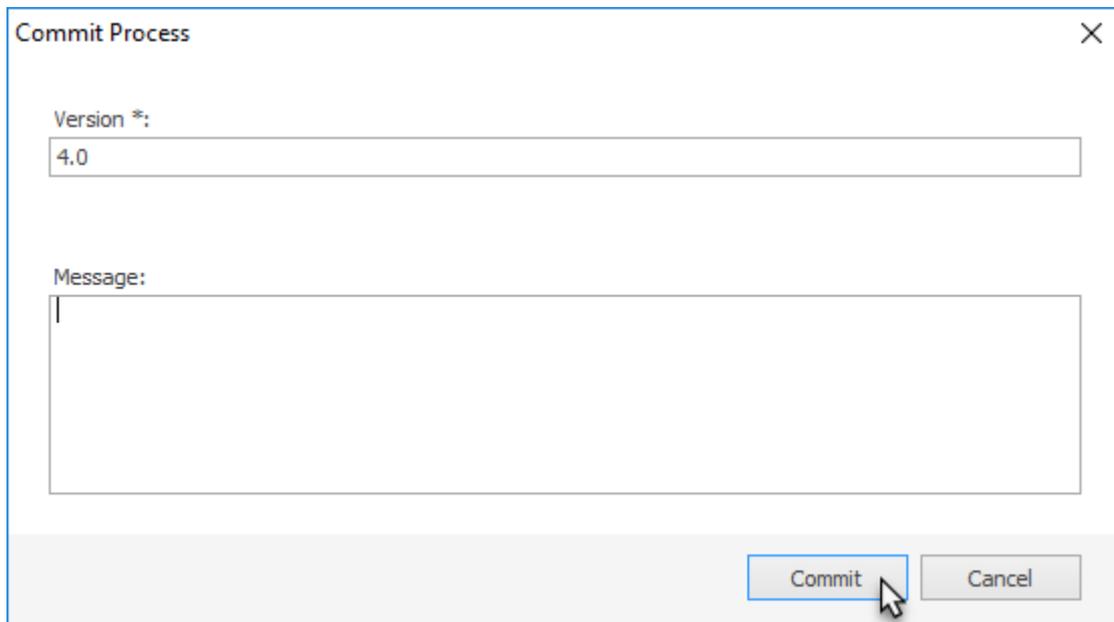
Before you are ready to share your Action with your colleagues or the PR community, you first need to save and commit your changes as a new Version.

User Actions follow as all script development in PR, a Lifecycle of development, approval and publishing you must surely know about by now through Processes.

So to Save and Commit the changes on your Process Designer simply click first on the little menu arrow by the Save option on ULD and then the Save And Commit item.



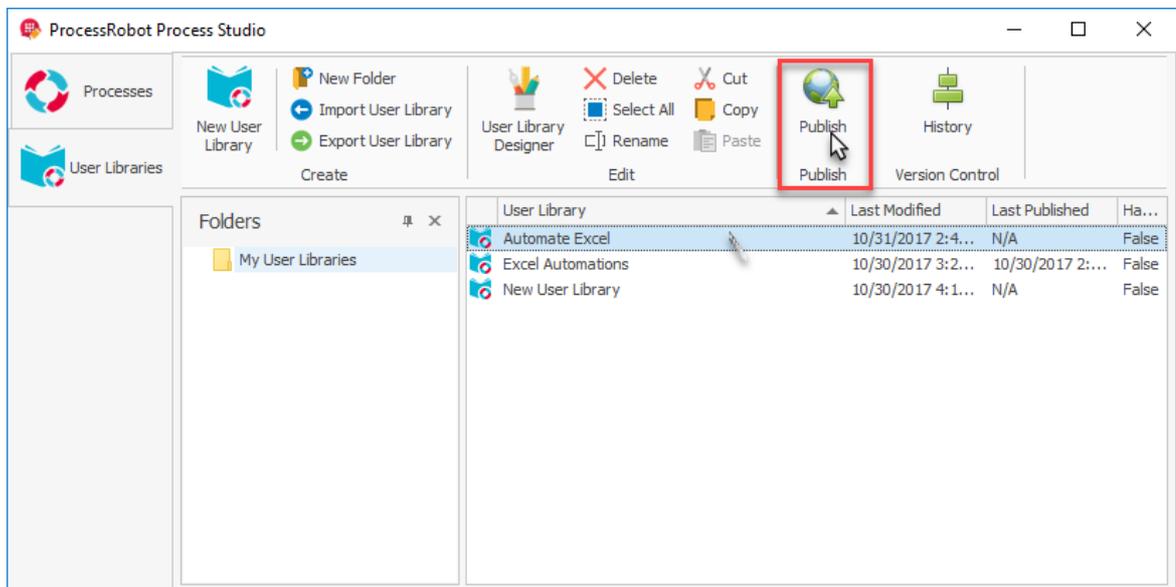
This will result to a Commit Process dialog box, that we'll fill out in the following manner:



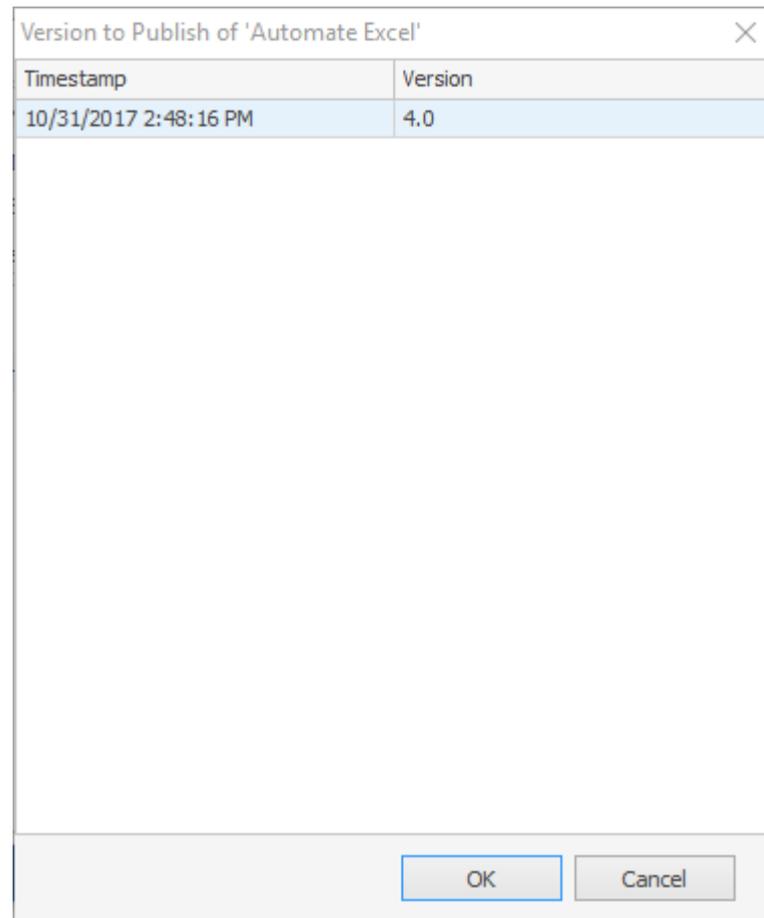
Notice that we are about to click Commit without leaving a Message, since this is optional and meant to help you and your colleagues keep track of your development efforts.

Now that we have committed our changes, it is time to close the ULD and finally Publish *Automate Excel*, the User Library that holds our beloved first User Action.

To do so, we'll head to Process Studio, highlight the Library of our choice (Automate Excel) on the main Viewport and then click on the Publish button.



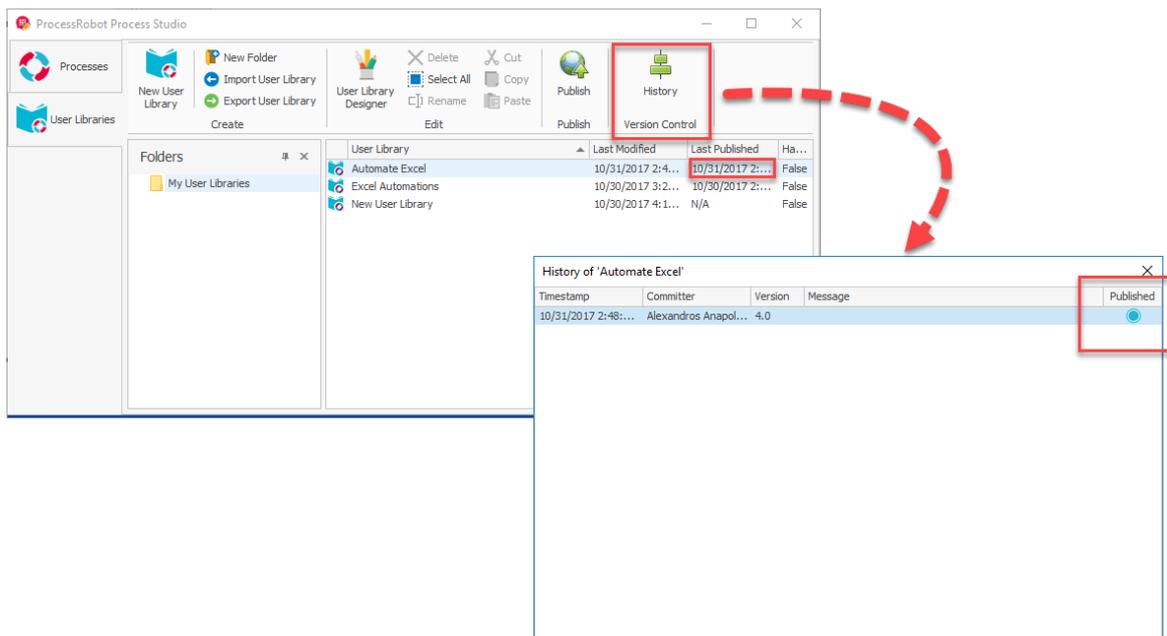
This will result on the Version to Publish dialog box that will finally publish our Library, making it instantly available to all Processes of our installation.



The image shows a dialog box titled "Version to Publish of 'Automate Excel'". It contains a table with two columns: "Timestamp" and "Version". The first row of the table shows a timestamp of "10/31/2017 2:48:16 PM" and a version of "4.0". Below the table, there are two buttons: "OK" and "Cancel".

Timestamp	Version
10/31/2017 2:48:16 PM	4.0

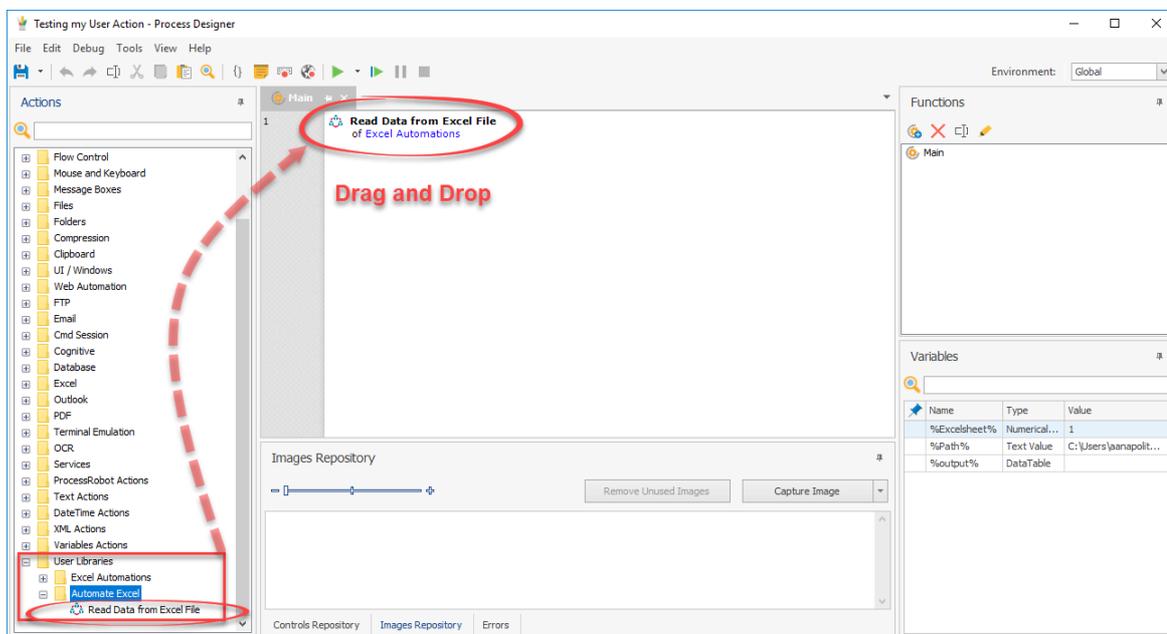
We can verify that our Library has been published by either clicking on the Version Control button that will allow us to check the History of our Library, or simply check the Last Published column on the central Viewport:



The Ultimate Test: Using our User Action

Now we are ready to use our User Action into any Process we want. To do so in our PR installation simply open up a new Process on Process Designer (we've done so through the creation of a Testing my User Action).

Having done so, simply check the very last Folder at the bottom of the Actions pane:



As the diagram shows having located the UA, what is left to happen is simply to Drag and Drop the Action to your Workspace and pass the values of Path and Worksheet into its Properties before you are 100% set to use your Action.

3.10 Advanced Topics

3.10.1 Variables

Throughout your Process, you will need to store useful information for passing them on from one action to another: the current date, the file that the user selected, the contents of the clipboard, etc. ProcessRobot allows you to store this kind of data, for later use, through variables. You can think of a variable like a storage bin that saves critical information for the smooth execution of a Process. You can save all kind of different types of data into variables, such as text or numbers. For more information about Data Types please have a look [here](#)⁴³⁶.

In order to use variables, you will have to use the specific notation associated with them: the variable's name should be enclosed in percentage characters. For example, in order to use a variable called "ClipboardText" as input for "Set Clipboard Text" action, you must write: %ClipboardText% in the "Text to store into Clipboard" field of the specific action's properties dialog box.

Creating Variables

There are 2 main ways to create a new variable:

1. Through the "Variables Manager Window". For more information about where to find and how to use this method you can see: "[Variables Manager Window](#)"³¹¹.
2. Variables can also be created by Actions. Most actions generate output based on the input that has been provided to them. For example the "Get Text Length" action under "Text Actions" category, will accept the text that you want to measure as an input, and provide the number of chars that this text contains as an output. This output will be stored in a variable for later use.

Using Variables

Variables can be used to store any specific type of information that may need to be utilized throughout the Process. Most of the variables that you will use are the "product of a specific action's labor": the current date, the files contained in a specific folder, the text that is stored in the clipboard, is information that you can retrieve and store in variables through specific actions.

ProcessRobot provides you with an easy and straightforward way to use variables that have been previously created in your Process: there is a special "gear" icon  next to every input that accepts variables in the Action Properties Dialog box. By clicking on that icon, you will be able to directly select any variable or variable property (for information about properties you can check "[Data type Properties](#)"⁴³⁹) that you want to use as an input.

Notice that the gear icon is placed only next to the "Action's Inputs" (there are no gear icons

next to “Outputs”). This does not mean that you cannot use a variable that already exists, as an output, thus overwriting its contents and, possibly, changing the type of data that the variable contains. However, it is suggested that you use a new variable to store a different action’s output: it will help you to keep your Process clear and have a full understanding of what kind of data each variable contains at any given time of your Process’s execution.

3.10.1.1 Data Types

Every piece of information that you store in your variables, has a specific type. There are **18** main data types that ProcessRobot recognizes:

- **Text:**

A “Text” variable can contain any kind of text, from your email address to the text contents of txt file. Notice that the user’s input through the “[Display Input Dialog](#)” action will always be stored as text (even if the user enters a number). In order to convert text into number you will have to use the “[Convert Text to Number](#)” action.

- **Number:**

This is the data type of a variable that holds a number.

- **Boolean:**

Can be either True or False. For example the attributes of a file (such as “Hidden”) are of type Boolean. If a file is stored in a variable called %MyFile%, %MyFile.IsHidden% will be True if the file is hidden and False if it is not (IsHidden is a property of the File Data Type; for more information about the data type properties see “[Data Type Properties](#)”).

- **Credential:**

This is a Data Type that allows you to store and access sensitive information like passwords and addresses in Process Robot. The data are stored as lists of property-value pairs in which the Values can be encrypted through the [Master Key](#) of your Process Robot. These values can be accessed within your Processes via their corresponding Credential and Property Names using dot (.) notation, through the “[Get Credential](#)” and “[Update Credential](#)” ProcessRobot Actions.

- **Custom Dialog Instance:**

A variable of that type is created through the “[Display Custom Dialog](#)” action. It is used to uniquely identify the specific custom dialog created by the action. Every Custom Dialog Instance has properties (e.g. LastButtonPressed). For more information about data type properties see “[Data Type Properties](#)”. Variables of the “Custom Dialog Instance” type can be used in place of a UI Automation “Window Instance” in UI and Windows actions.

- **Custom Object:**

Custom Objects are a type of Variable in ProcessRobot that allows users to generate and/or translate (“[Convert Custom Object To Json](#)”) data in JSON format. Custom Objects are

essentially collections of pairs of Property Names and corresponding Values. You can access these values using dot (.) notation or bracket ([]) notation since any two different Properties of a Custom Object may have the same Value but not the same Name.

- **List:**

A list contains a collection of values (it is the equivalent of single-dimensional arrays in programming). Depending on the type of the list items we can have List of Texts, List of Numbers, List of Files etc. For more information about lists, see "[Lists](#)".

- **DataRow:**

A data row is an iterable collection that contains all values corresponding to the columns of a [data table](#) for a given row. It is the product of an iteration of a data table.

- **DataTable:**

A data table contains data in a tabular form (it is the equivalent of two-dimensional arrays in programming). For more information about data tables, see "[Data Tables](#)".

- **Date Time:**

Contains basic date and time information. The form that will be used to print a "date time" variable (e.g. through a "Display message" action) depends on the machine's regional settings. To retrieve a special textual representation of a Date Time value you need to convert the value into text with the "[Convert DateTime to Text](#)" action.

- **Excel Instance:**

A variable of that type can be created through the "Launch Excel" action. It is used to uniquely identify the specific instance of Excel that was launched. Every other action under "Excel" category (e.g. "Read from Excel") will require from you to specify an existing Excel instance (i.e. the specific excel file whose data will be read). Variables of the "Excel Instance" type can be used in place of a UI Automation "Window Instance" in UI and Windows actions.

- **FTP Connection:**

A variable of that type can be created through the "Open FTP Connection" and "Open Secure FTP Connection" actions. It is used to uniquely identify the FTP connection that was opened. Every other action under "FTP" category (e.g. "Download File(s) from FTP") will require from you to specify an existing FTP connection (i.e. the specific FTP site where the file(s) will be downloaded from).

- **File:**

Represents a file. Files can be used as input in any of the actions under "Files" category. Every file has properties (e.g. Size). You can retrieve a file's properties by using the following notation: %VariableName.PropertyName%. For more information about data type properties see "[Data Type Properties](#)".

- **Folder:**

Represents a folder. Folders can be used as input in any of the actions under "Folders" category. Every folder has properties (e.g. CreationTime). You can retrieve a folder's

properties by using the following notation: %VariableName.PropertyName%. For more information about data type properties see "[Data Type Properties](#)".

- **Browser Instance:**

A value of that type can be created through the "[Launch New Internet Explorer](#)", "[Launch New Firefox](#)" or "[Launch New Chrome](#)" actions. It is used to uniquely identify the specific instance of a Web Browser that was launched or attached to. Every other action under "Web Automation" category (e.g. "[Click Link on Web Page](#)") will require you to specify an existing Web Browser instance (i.e. the specific browser window that contains the link to be clicked). Values of the "Web Browser Instance" type can be used in place of a UI Automation "Window Instance" in UI and Windows actions.

- **Mail Message:**

Represents a email. Emails variables are populated by the "[Retrieve Emails](#)" action. Every email has properties (e.g. Subject). You can retrieve an email's properties by using the following notation: %VariableName.PropertyName%. For more information about data type properties see "[Data Type Properties](#)".

- **Ocr Engine:**

A variable of this type can be created through the "Create OCR Engine" group of actions. It will hold the OCR's engine's instance created in a variable, in order to be used into further OCR related actions that need the engine to be initialized, for them to work. Such actions are: "[Extract Text With OCR](#)", "[Extract text From PDF With OCR](#)" and "".

- **Window Instance:**

A variable of that type can be created through the "[Get Window](#)" action. It is used to uniquely identify an application window on the user's desktop. UI and Windows actions will require you to specify a target Window instance (i.e. the specific window which will be acted upon / contains the target element(s)).

3.10.1.2 Lists

Lists are collections of items of not necessarily the same type. You can create a list through the "Create New List" action (under "Variables" category) and add an item to that list through the "Add Item to List" action. You can also create a list through actions that generate lists as output (e.g. "Read text from file" will return a list if you select to store the file's contents as a list and "Get Files in Folder" action will return a List of Files).

In order to retrieve a specific item of a list, you will have to use the following notation:

%VariableName[ItemNumber]%

Keep in mind that the "ItemNumber" should be 0 for the first item of the list, 1 for the second and so on.

For example, if you have a list that is stored in the variable `%MyList%` and contains 10 items, you can retrieve the first item with: `%MyList[0]%` and the last item with `%MyList[9]%`.

Very often the "For Each" loop is used for iterating through the items of a list.

3.10.1.3 Data Tables

Data tables contain data in a tabular form (it is the equivalent of two-dimensional arrays in programming). A data table contains rows and columns and each item stored in the data table can be retrieved through its unique row and column number. You can think of data tables as lists: each item of the list is also a list.

There is no direct way to create a data table but there are 2 actions that generate a data table as an output: the "Read from Excel" action and the "Execute SQL Statement" action.

In order to retrieve a specific item of a data table, you will have to use the following notation:

`%VariableName[RowNumber][ColumnNumber]%`

Keep in mind that "RowNumber" and "ColumnNumber" should be 0 for the first item (row or column), 1 for the second and so on.

For example, if you have a data table that is stored in the variable `%MyDataTable%`, you can retrieve the first item with: `%MyDataTable[0][0]%`.

Data Rows

If you loop through a data table with a "For Each" action, the variable that will contain the current iteration's data, is considered to be a "Data Row". Data rows are basically lists: they are no different than regular lists (they contain several items, their items can be retrieved through the regular list notation), except for the part that they represent a row in the data table's scheme. Remember that you can still treat them as if they were lists without any problem at all.

3.10.1.4 Data type Properties

Some of the built in ProcessRobot data types have properties, which are values associated to the value stored in the variable. The value of these properties can be accessed directly through the following notation: `%VariableName.PropertyName%`.

Below you can find the available properties for the different built-in ProcessRobot data types:

Data type Text has the following property:

- **Length:** the length (in characters) of the text stored into the variable.

Data type DateTime has the following properties:

- **Year:** The year part of the DateTime value.
- **Month:** The month part of the DateTime value.
- **Day:** The day part of the DateTime value.
- **DayOfWeek:** The name of the day (e.g. Sunday, Monday etc)
- **DayOfYear:** The day of the year part of the DateTime value (1-365/6)
- **Hour:** The hour part of the DateTime value.
- **Minute:** The minute part of the DateTime value.
- **Second:** The seconds part of the DateTime value.

Data type List has the following property:

- **Count:** the number of items stored into the list.

Data type File has the following properties:

- **FullName:** The full path to the file.
- **RootPath:** The root path of the file (e.g. C:\).
- **Directory:** The directory where the file is stored.
- **Name:** The name of the file, including the extension (e.g. MyFile.txt).
- **NameWithoutExtension:** The name of the file, without its extension.
- **Extension:** The extension of the file.
- **Size:** The size of the file (in bytes).
- **CreationTime:** The date when the file was created.
- **LastAccessed:** The date when the file was last accessed.
- **LastModified:** The date when the file was last modified.
- **IsHidden:** This property is true if the file is hidden or false if the file is visible.
- **IsSystem:** This property is true if the file is a system file or false if it is not.
- **IsReadOnly:** This property is true if the file is read only or false if it is not.
- **IsArchive:** This property is true if the file is an archive or false if it is not.
- **Exists:** This property is true if the file exists or false if the file does not exist.

Data type Folder has the following properties:

- **FullName:** The full path to the folder.

- **RootPath:** The root path of the folder (e.g. C:\).
- **Parent:** The parent directory of the folder.
- **Name:** The name of the folder.
- **CreationTime:** The date when the folder was created.
- **LastModified:** The date when the folder was last modified.
- **IsHidden:** This property is true if the folder is hidden or false if the folder is visible.
- **Exists:** This property is true if the folder exists or false if the folder does not exist.

Data type Mail Message has the following properties:

- **MailFolder:** the name folder the email message is retrieved from
- **Uid:** the unique identifier of the message
- **From:** the sender of the email message
- **To:** a list of values containing the recipients of the message
- **Cc:** a list of values containing additional recipients for the message (carbon copy)
- **Date:** the date and time in which the message was sent
- **Subject:** the subject of the message
- **Body:** the body of the message. This can be in plain text or in HTML form
- **BodyText:** if the previous property contains HTML, this property contains the body of the message in plain text form.
- **Attachments:** a list of [File](#) data types that represent the attachments of the email message (if any) after they have saved locally.

Data type FTPFile has the following properties:

- **FullName:** The full path to the file.
- **Directory:** The directory where the file is stored on the FTP Server.
- **Name:** The name of the file, including the extension (e.g. MyFile.txt).
- **NameWithoutExtension:** The name of the file, without its extension.
- **Extension:** The extension of the file.
- **Size:** The size of the file (in bytes).
- **LastModified:** The date when the file was last modified.

Data type FTPFolder has the following properties:

- **FullName:** The full path to the folder.
- **Parent:** The parent directory of the folder.
- **Name:** The name of the folder.

- **LastModified:** The date when the folder was last modified.

Data type DataTable has the following properties:

- **RowCount:** The number of rows of the DataTable
- **Columns:** A list that contains the names of the columns of the DataTable

Data type DataRow has the following property:

- **ColumnsCount:** The number of columns that the DataRow holds.

Data type Custom Dialog Instance has the following property:

- **LastButtonPressed:** The name of the button that was last pressed on the dialog.
- **DialogWasForciblyClosed:** This property is true if the dialog was closed by a close window command or by pressing the "Cancel Button", false otherwise.
- **Handle:** The handle of the Window belonging to this specific Custom Dialog.

3.10.1.5 Persistent Variables

What Persistent Variables are and why do I need them?

When you are creating a Process with ProcessRobot, you may sometimes want to keep the value of a specific variable available for the future executions of the Process. Persistent Variables allow you to do exactly that: keep a variable's value available for the next time that the Process executes.

Before this feature was available, the standard way to keep values was to store them into external files, a process which sometimes could be complicated. Now by simply marking a variable as persistent we can be sure that its value will be kept between the executions of the Process.

You can set a variable as persistent through the Tools Menu option > [Variables Manager](#)^[311] window while in the Process Designer.

When you first create the persistent variable it will not have any stored value. However, if you run the Process and give the variable a specific value, you will be able to see the currently stored value from the "Edit Variable" window in the Process Designer's Variable Manager.

3.10.1.6 External Variables

If from a Process (Process A) you run a different Process (Process B) by using the "Start Process" action, all variables of Process A are available to Process B. These variables in Process B are considered "External Variables" and can be used for passing data from one Process to another.

For Process B to access any of the external variables, the "ext:" prefix must be included before the variable name. For example, if Process A defines a variable **%MyData%**, Process B can access the value of that variable using the notation: **%ext:MyData%**

3.10.1.7 Custom Date Formats

You can customize the way that the date is represented in the [Convert Date Time to Text Action](#) by entering characters that represent various parts of the Date/Time along with any other characters that can be used to link the Date/Time parts.

Various characters and combinations can be used, some of which are displayed below.

When displayed on their own these characters produce the following formats:

- d: The Short Date (04/02/2010)
- D: The Long Date (Friday, April 02, 2010)
- f: The Full Date Time - short Time (Friday, April 02, 2010 10:00 AM)

- F: The Full Date Time (Friday, April 02, 2010 10:00:46 AM)
- g: General Date Time - short Time (04/02/2010 10:00 AM)
- G: The General Date Time - long Time: (04/02/2010 10:00:46 AM)
- M: The Month and day of the month: (April 02)
- m: The Month and day of the month: (April 02)
- r: A Shorter Full date (Fri, 02 Apr 2010 10:00:46 GMT)
- R: A Shorter Full date (Fri, 02 Apr 2010 10:00:46 GMT)
- s: The Sortable Date Time (2010-04-02T10:00:46)
- t: The short time (10:00 AM)
- T: The Long Time (10:00:46 AM)
- y: The Month and Year: (April, 2010)

When used as a combination the characters have the following representation:

- d: The day of the month as a number from 1 to 31 (2)
- dd: The day of the month as a number from 01 to 31 (02)
- ddd: The abbreviated day of the week (Fri)
- dddd: The full day of the week (Friday)
- gg: The period era (A.D.)
- h: The hour as a number from 1 to 12 (10)
- hh: The hour as a number from 01 to 12 (10)
- HH: The hour as a number from 00 to 23 (10)
- m: The minutes as a number from 0 to 59 (0)
- mm: The minutes as a number from 00 to 59 (00)
- M: The month as a number from 1 to 12: (4)
- MM: The month as a number from 01 to 12 (04)
- MMM: The abbreviated month (Apr)
- MMMM: The month (April)
- s: The seconds as a number from 0 to 59 (46)
- ss: The seconds as a number from 00 to 59 (46)
- tt: The AM/PM designator (AM)
- y: The last digit of the year (0)
- yy: The last two digits of the year (10)
- yyyy: The year (2010)
- zz: The time zone (+02)
- zzz: The time zone in full format (+02:00)

You can mix and match these special characters in the Custom Date Time Format and add any other characters you wish to connect them. If you wish to use any of the characters seen in the above list as a connection character you will need to use the backslash(\) in front of it. For example if in your custom date you need to represent the phrase 'The date is ' followed by the Date you will need to enter the following in the Custom Date field:

```
T\he \da\te i\s: d/M/yy
```

3.10.1.8 Custom Objects

Custom Objects are a type of Variable in ProcessRobot that allows users to generate and translate data in JSON format.

This topic consists of the following sections:

- a) What is JSON and why is it important?
- b) The JSON format
- c) The Custom Object format

If you already know what JSON is please feel free to jump straight to c) The Custom Object format.

- a) What is JSON and why is it important?

JSON is a text format that is:

- a) easy for humans to read and write,
- b) easy for machines to parse and generate and
- c) completely language independent but uses conventions that are familiar to programmers.

These properties make JSON an ideal data-interchange language that is very popular among web developers.

JSON stands for Javascript Object Notation and it is based on a subset of the [JavaScript Programming Language](#). You can read more about JSON on the Standard's [official page](#).

- b) The JSON format

Here is an example of JSON data:

```

{
  "firstName": "John",
  "lastName": "Locke",
  "address": {
    "streetAddress": "24 2nd Street",
    "city": "Somerset",
    "kingdom": "England",
    "postalCode": 10210
  },
  "phoneNumbers": [
    "212 555-1294",
    "646 555-6567"
  ]
}

```

As you can see, the JSON format is inspired after two structures:

a) an Object, which can be seen as an unordered set of name/value pairs, like { "firstName": "John", "lastName": "Michael" } inside curly braces and separated by commas.

All names has to be strings (i.e. text values), while values can be almost anything, from Boolean to another Object!

Another thing to note is the colon (:) used to separate the name from its corresponding value.

b) an Array, which is an ordered collection of values, like [1,2,3,4] inside brackets and separated by commas.

Same as before, values can be almost anything, from numbers to strings!

The combination of these two structures is allowing the creation of ordered lists of name/value pairs. You can access individual object values by using either the **dot (.) notation** in the following manner:

```

myObj = { "name": "John", "age": 378, "car": null };
x = myObj.name;

```

or by using the **bracket ([]) notation**:

```

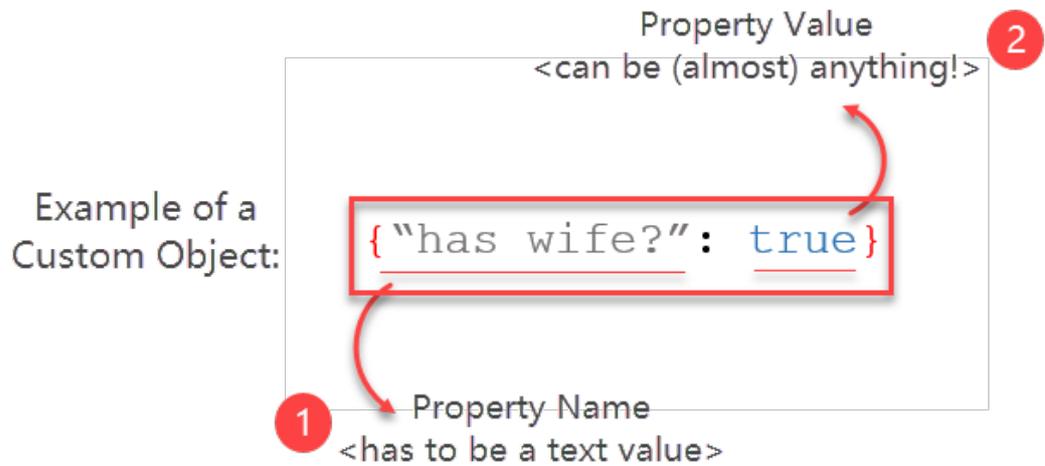
myObj = { "name": "John", "age": 378, "car": null };
x = myObj["name"];

```

c) The Custom Object format

Understanding the JSON format is important, because Custom Objects were created in order to allow us to build and use data in that format within Process Robot.

Just like JSON, custom objects in Process Robot consist of a name and a value; they are essentially a pair of two data:



1

A Property Name:

A Property name will be by default a Text Value. Given that fact it can consist of almost any combination of characters.

However:

-We can use bracket notation in order to access the value corresponding to the Property Name of the example Custom Object in the following manner.

Setting %BooleanTrue% as `true` via Tools>Variables Manager and 'writing' the following sequence of actions:

- 1  **Create New Custom Object**
Create a new Custom Object and store it into %NewCustomObject%
- 2  **Set Variable**
Set variable %NewVar% = %BooleanTrue%
- 3  **Add Property To Custom Object**
Add Property *has wife?* with value %NewVar% into Custom Object %NewCustomObject%
- 4  **Set Variable**
Set variable %NewVar2% = %NewCustomObject["has wife?"]%

If we test this on Process Designer, it will produce the following Variable values:

Variables			
Name	Type	Value	
%NewCustomObject%	Custom Object	{ <i>has wife?</i> :true}	
%NewVar2%	Boolean Value	true	
%NewVar%	Boolean Value	true	
%BooleanTrue%	Boolean Value	true	

-if we try to access the same value, **using dot notation**, we will get a Design error on action 4, Set Variable %Newwar2%.

Properties of 'Set Variable' action

Set Variable
 This action sets the value of a new or existing variable, and can be used to create a new variable or overwrite a previously created variable.

General

Action Input

Set Value:

Action Output

into Variable:

This action is Enabled

More Info OK Cancel

This means that in order to be able to access a value using dot notation, **we need to be careful to avoid producing syntax errors.**

It also should be noted that two different Properties of a Custom Object may have the same Value but not the same Name! (This will produce a Runtime Error informing us that Property Name exists.)

2

A Property Value:

Just like in JSON, property values can be of almost any kind, from a Data Table to a List.

This versatility allows you to use Custom Objects either as a medium that allows you to translate data into JSON within PR or as a custom variable type that can increase your own convenience and capabilities in Process Robot.

For example, let us consider another example:

1	 Create New Custom Object Create a new Custom Object and store it into %NewCustomObject%
2	 Create New List Create a new empty list and store it into: %NewListVar%
3	 Add Item to List Add item Tania into list %NewListVar%
4	 Add Item to List Add item Amanda into list %NewListVar%
5	 Add Item to List Add item Korinna into list %NewListVar%
6	 Add Property To Custom Object Add Property wife with value %NewListVar% into Custom Object %NewCustomObject%
7	 Set Variable Set variable %NewVar2% = %NewCustomObject.wife[1]%

If we test this on Process Designer, it will produce the following Variable values:

Variables			
Name	Type	Value	
%NewVar2%	Text Value	Amanda	
%NewListVar%	List of General Values	List of General Values containing 3 i...	
%NewCustomObject%	Custom Object	{"wife":["Tania","Amanda","Korinn...	

What is noteworthy here is the **%NewCustomObject.wife[1]%** dot notation syntax we used in order to access the Text Value "Amanda".

This combination of dot and bracket notation is super convenient for accessing list items within the Property Value of a Custom Object and is definitely something you should know in order to make the most of your custom object adventures.

It also should be noted that two different Properties of a Custom Object may have the same Value but not the same Name! (This will produce a Runtime Error informing us that Property Name exists.)

3.10.2 Loops

ProcessRobot allows you to use “Loops”, a concept that is really important in programming and proves to be invaluable when combined with ProcessRobot’s actions. The idea behind a loop is to make ProcessRobot repeat one or more actions multiple times. Loops in ProcessRobot come in 3 flavors:

- Loop: Repeats one or more actions a specific number of times. For more information see [“Simple Loops”](#).
- Loop Condition: Repeats one or more actions until a specific condition is met. For more information see [“Loop Condition”](#).
- For each: Repeats one or more actions for every item contained in a list or data table. A different item of the list (or data row of the data table) will be saved in a variable during each iteration, allowing you to use it inside the actions of the loop. For more information see [“For Each”](#).

Note: ProcessRobot will automatically indent the block of actions that will be repeated by a loop.

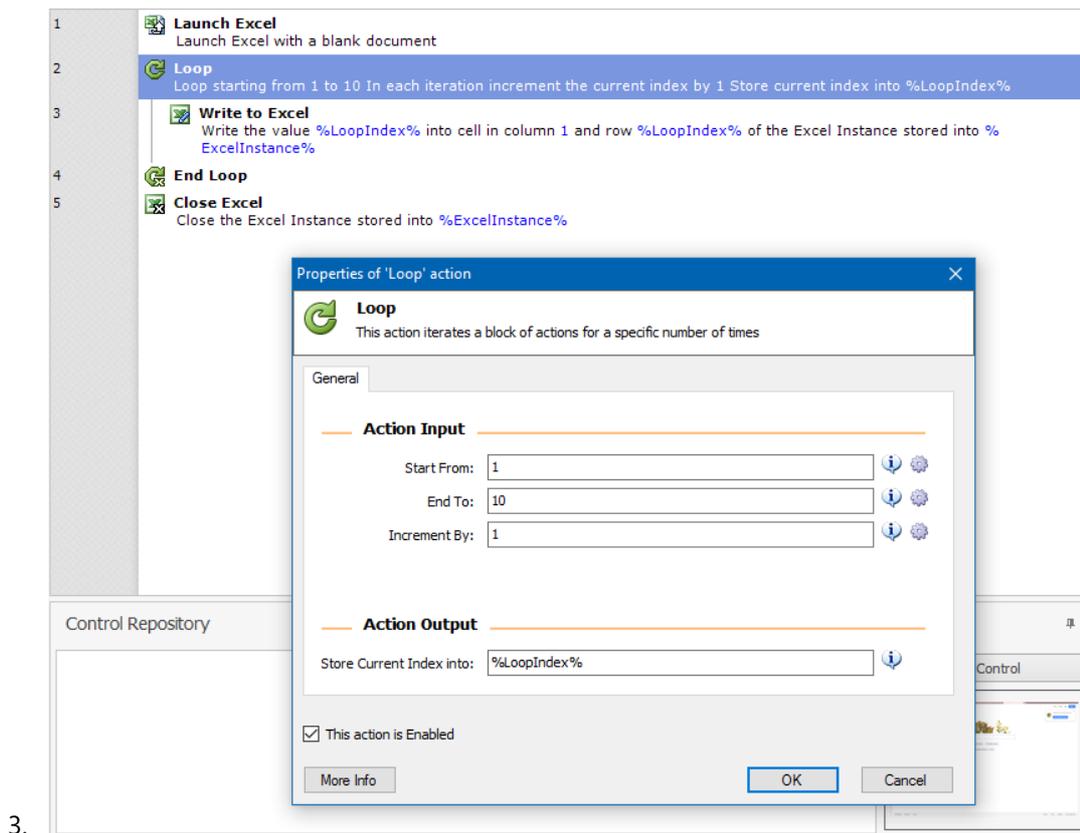
3.10.2.1 Simple Loops

ProcessRobot implements simple loops with the [“Loop”](#) action. Basically, a simple loop will repeat the block of actions that it contains, for a set number of times. Note that a specific variable can be created to keep track of the current iteration’s number.

Use

A simple loop is ideal to use in 2 cases:

1. If you know the exact number of times that action(s) should be repeated
2. If the %LoopIndex% variable must be used somewhere inside the loop (“LoopIndex” is the default variable’s name that ProcessRobot uses to store the current iteration’s number).



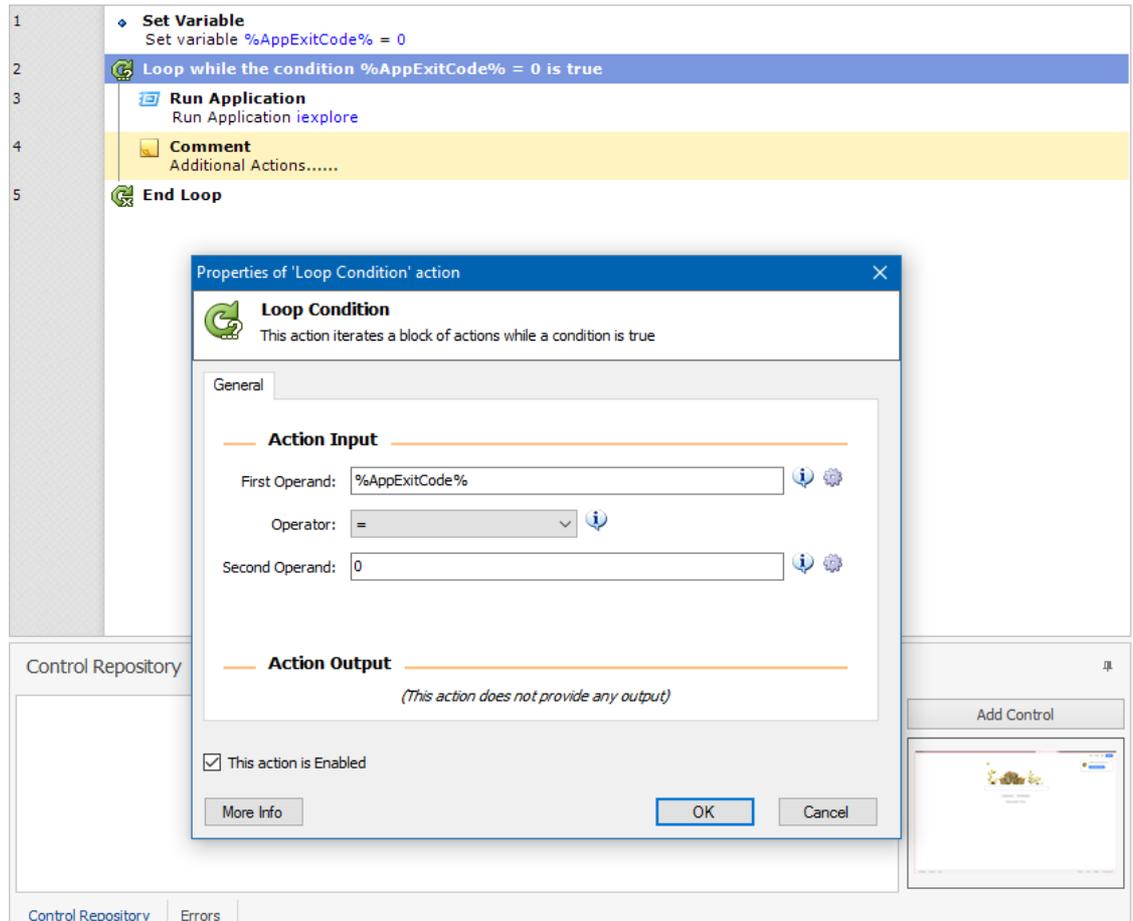
You can see an example of a simple loop usage in "Advanced" category of ProcessRobot's examples: "8 - Rotate Excel Data"

3.10.2.2 Loop Condition

A "Loop Condition" is implemented in ProcessRobot through the homonym [action](#)^[586]. Basically, Loop Condition will repeat one or more actions while a certain condition is true (for more information about conditions check "[Conditionals](#)"^[474]). Keep in mind that at some point, inside the loop, you will have to make the loop's condition false otherwise it will keep on repeating and your Process will never end (this is called an endless loop).

Example:

Suppose that you have an executable that you want to keep executing until it returns a non-zero error code. All you have to do is create a Loop Condition that will repeat a "Run Application" action while the %AppExitCode% equals to zero.



3.10.2.3 For Each

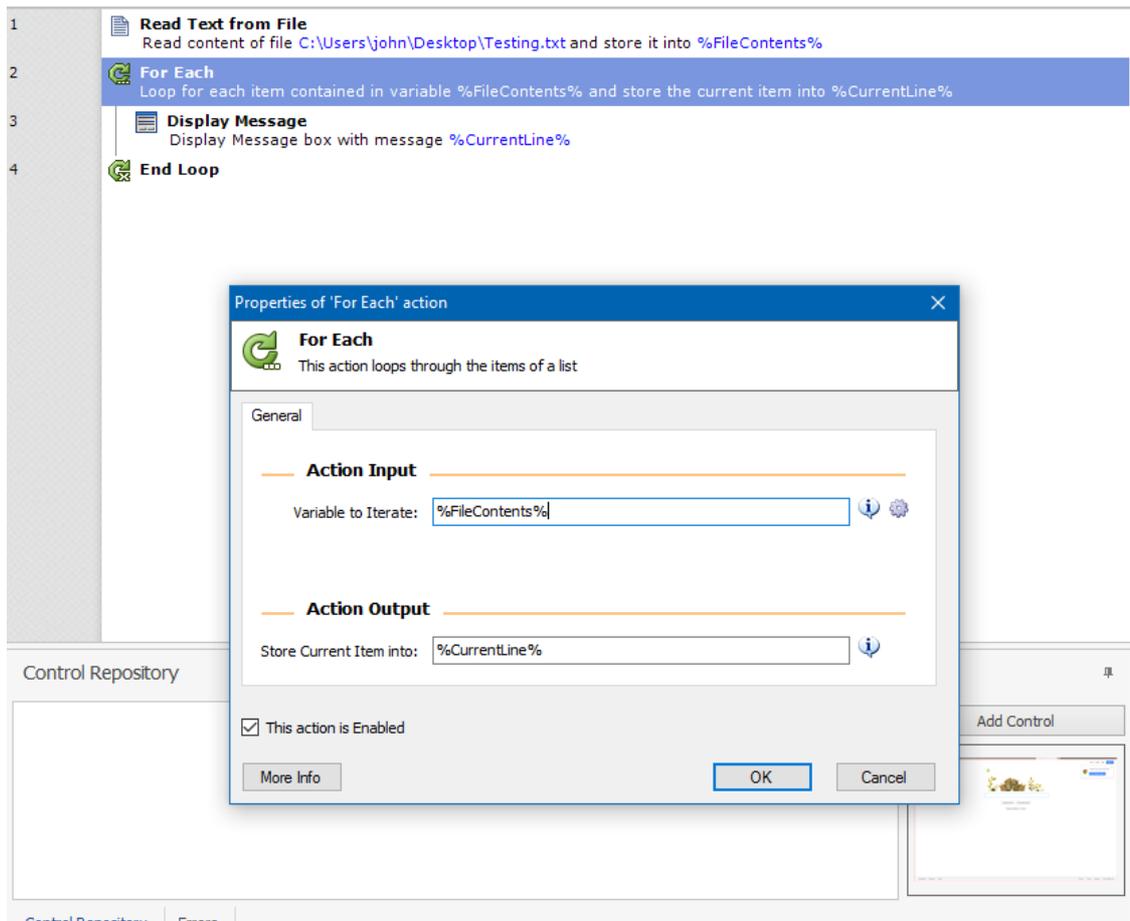
A "[For Each](#)⁵⁸⁷" loop, iterates through every item of a list or data table, and stores the current item in a variable. This kind of loop is extremely useful when you do not know or you do not need to know the number of the iterations that will be needed, but you do want to iterate through every item of a list (or or row of a data table) and use each one of them in an action (or a series of actions).

Usage Examples

There are plenty of situations that you may need to use a "For Each" loop: i

If you want to check which of the text files that are contained in a list have a specific phrase in it, or if you need to use each data row's contents to fill in a web form, you will have to use it. Basically, if you use the words "For Each" when you are describing the task that you need to automate, you will most likely have to use this action.

Also if you choose to read a text file as a list where each item of the list will be an item, for you can use a "For Each" loop in order to iterate through each line and apply some actions on it.

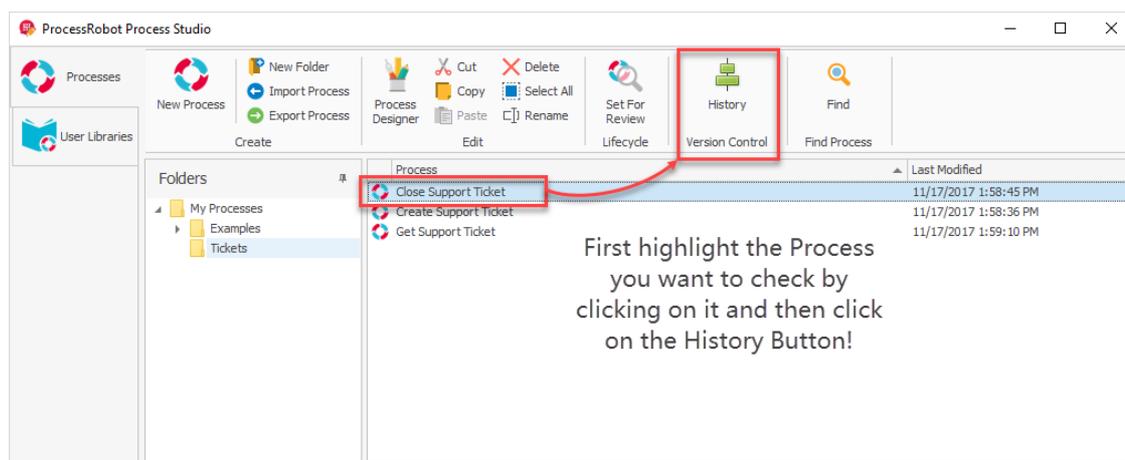


3.10.3 Version Control

3.10.3.1 Process Lifecycle

The lifecycle of a Process consists of three stages: Development, Review and Production.

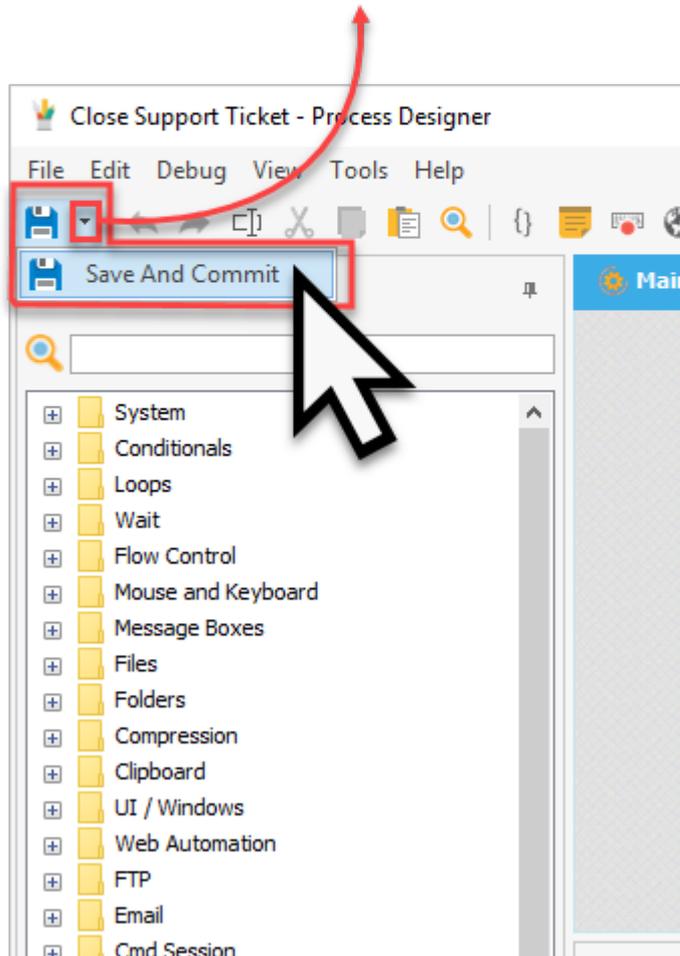
The stages that a Process has been throughout its 'life', can be seen by clicking on "History" in the "Version Control" section, either from the Control Desk or the Process Studio console.



Development

Since we have created a Process that does something, we have the option to Save or [Save and Commit](#) the Process.

In order for the 'Save and Commit' menu item to appear you need to click once on the little drop down arrow on the left of the characteristic Save Icon 



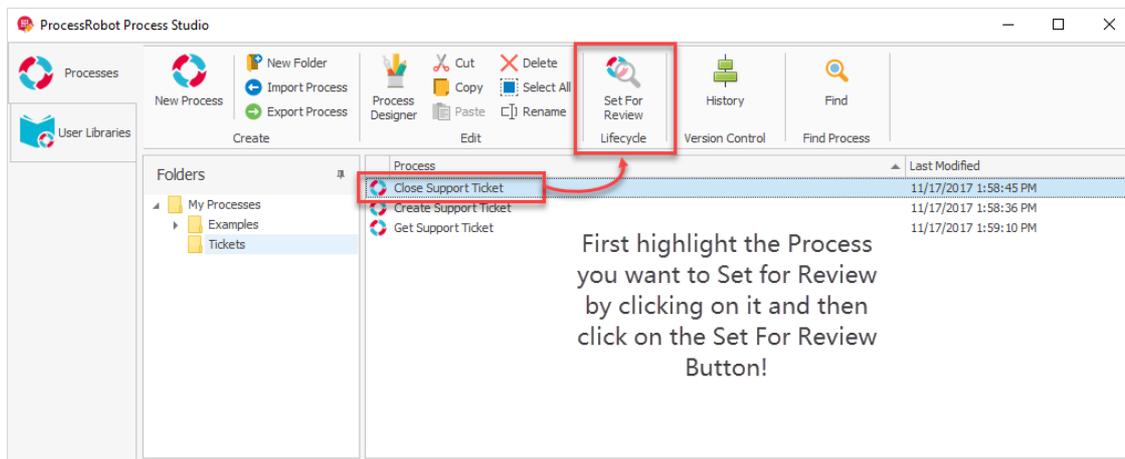
Should we choose the latter option, a version of the Process is created. This version is automatically placed *in Development*. Practically, this means that the Process has been saved, a version has been assigned to the saved Process and that this version is ready to migrate to *Review*.

Review

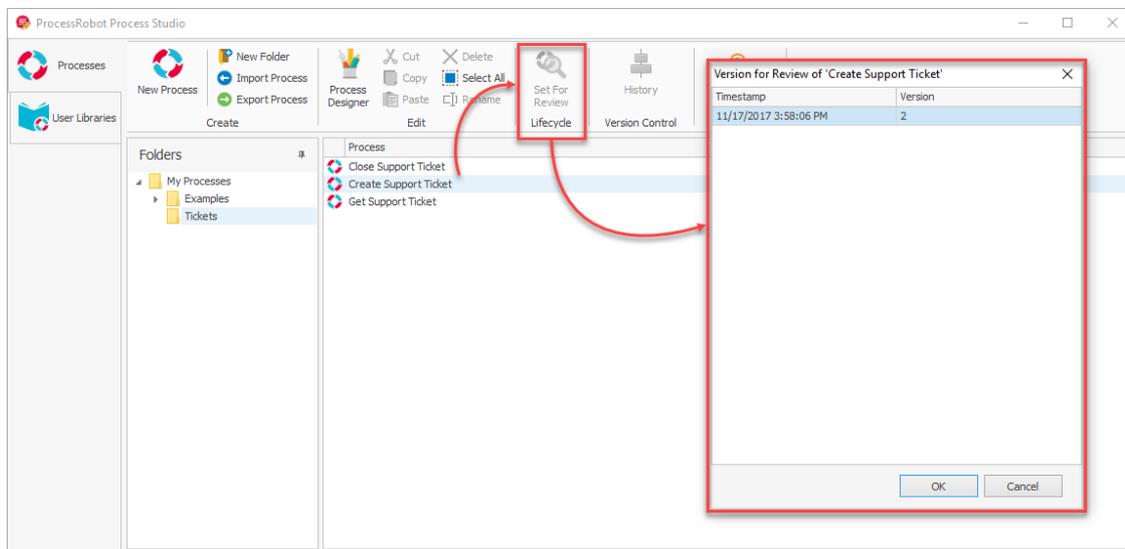
Once a version has been transferred to the Review stage, the user can test the execution of the Process to different Robots or [Robot Pools](#)^[217]. At this point, the user is not able to further edit the

Process and he is only able to review the way it performs in many stage, in order to ensure that it meets his expectations. Should any defects be found, then the Process should be Discarded from this stage and move back to Development. If everything works as expected, it is ready to be transferred to the final stage, which is Production.

In order to set a Process for Review, we click on the Process from the Process Studio console and select the Set for Review option.



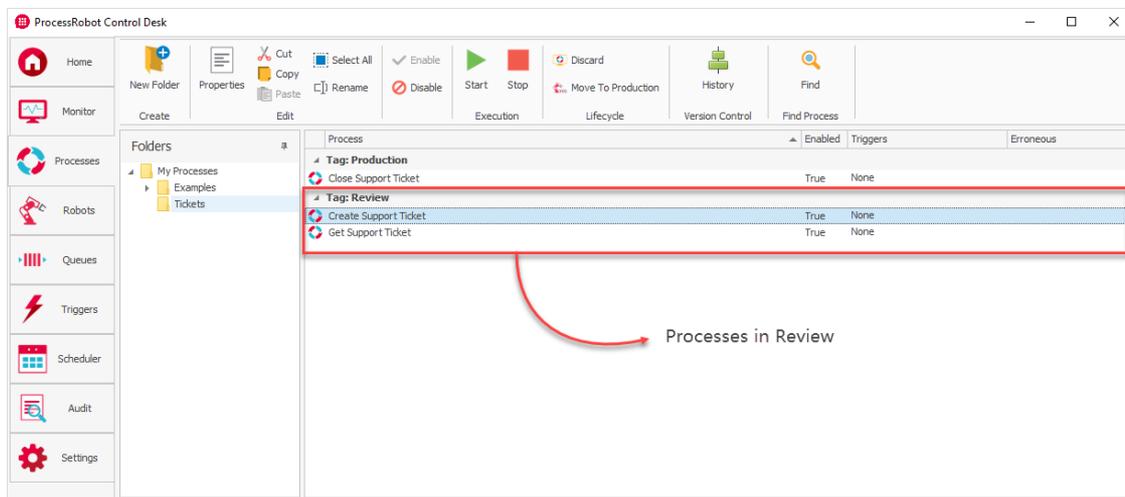
At this point, a pop up window will prompt the user to select the version of the Process which will be set for Review.



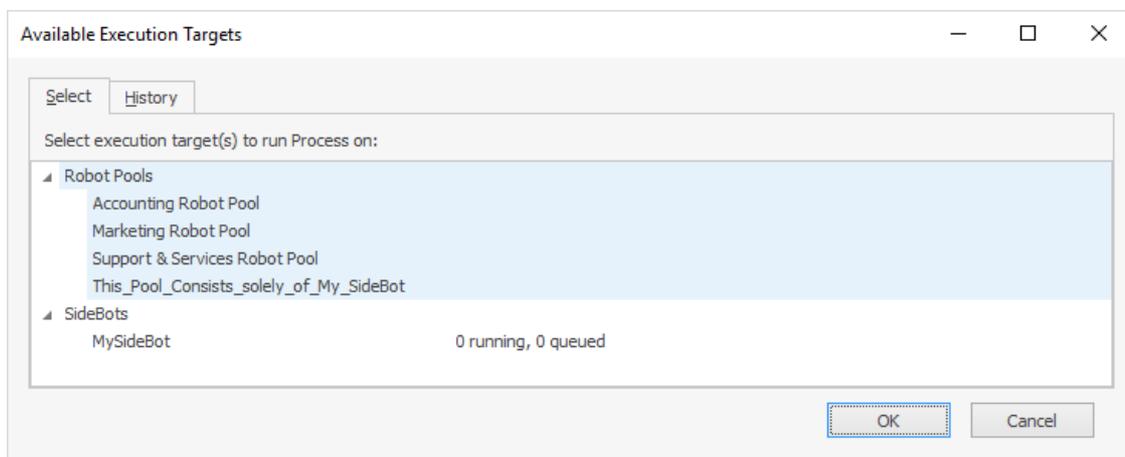
Important Note: The versions that are available to Set for Review, are only the ones that have been created later than the latest version which has already been Set for Review. For example, if version 3.0 has already been Set for Review, then only versions 4.0 and 5.0 will be available in the

future. Versions 1.0 and 2.0 cannot be Set for Review anymore. Additionally, if version 3.0 which has already been Set for Review is Discarded, then the available versions will be 3.0, 4.0 and 5.0.

The moment that a Process is being transferred to the Review stage, it appears in the Control Desk, in the same folder as the one that it is in the Process Studio and under the Tag: Review.



When we click to run a Process that is under Review from the Control Desk, we will be prompted to set the available Execution Targets that the Process will run on.



Only one version of a Process can exist at the Review stage at any given time.

When a newer version is Set for Review, then the older one is no longer in this stage. For example, in the Process "New Process 4" we have transferred version 3.0 to the Review stage. Should we later want to transfer version 4.0, then the "History" of "New Process 4" will look like the screenshot below:

Timestamp	Committer	Version	Message	Develop...	Review	Production
5/3/2017 1:54:08 PM	Peter Feleskouras	5.0	This is version 5.0	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
5/3/2017 1:54:00 PM	Peter Feleskouras	4.0	This is version 4.0	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
5/3/2017 1:53:51 PM	Peter Feleskouras	3.0	This is version 3.0	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
5/3/2017 1:53:40 PM	Peter Feleskouras	2.0	This is version 2.0	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
5/3/2017 1:53:29 PM	Peter Feleskouras	1.0	This is version 1.0	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>

Since a Process has been Reviewed, there is the option to either Move it to Production (the final stage) or Discard it.

The screenshot shows the ProcessRobot Control Desk interface. The top toolbar includes buttons for Home, Monitor, Processes, Robots, Queues, Triggers, and Scheduler. A central toolbar contains actions like New Folder, Properties, Cut, Copy, Paste, Select All, Rename, Enable, Disable, Start, Stop, Discard, Move To Production, and History. The main area displays a list of processes under the 'Tag: Review' filter:

Process	Enabled	Triggers	Erroneous
New Process 1	True	None	
New Process 2	True	None	
New Process 4	True	None	

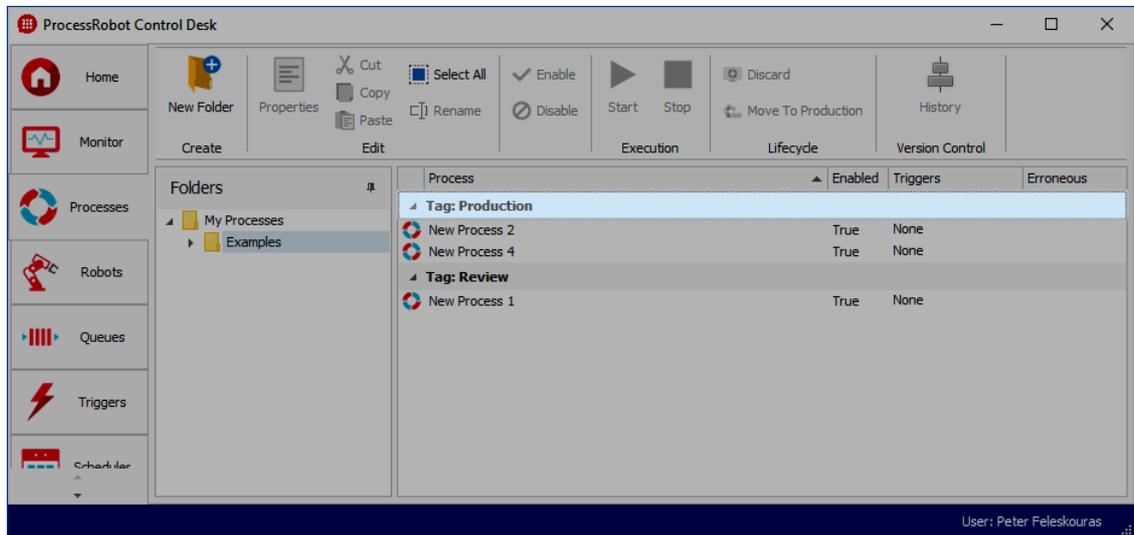
The user is identified as Peter Feleskouras.

If a Process is Discarded from Review, it will automatically go back to Development. At this point, the same version or an earlier one can be set again for Review from the Process Studio console.

Production

The final stage of a Process Lifecycle is that of Production. When a Process is in this stage, it has been developed, it has been tested and it is ready to be deployed to an Execution Target.

A Process can pass from Review to Production only from the Control Desk, where it is placed under the Tag: Production.



Should we click to Start the Process from the Control Desk, it is necessary that we define the Execution Target (the Robots) it will run on.

Discard from Production

A Process can be Discarded from Production and be transferred back to the Development stage. Note that the Process will NOT go back to Review, but it will migrate directly to Development. From this stage, we are able to set again for Review the same or a newer version of the Process from the Process Studio console.

When a Process is Discarded from Production (or Review) a circle with a red mark appears in the Process's History window.

History of 'New Process 4'						
Timestamp	Committer	Version	Message	Develop...	Review	Production
5/3/2017 1:54:08 PM	Peter Feleskouras	5.0	This is version 5.0			
5/3/2017 1:54:00 PM	Peter Feleskouras	4.0	This is version 4.0			
5/3/2017 1:53:51 PM	Peter Feleskouras	3.0	This is version 3.0			
5/3/2017 1:53:40 PM	Peter Feleskouras	2.0	This is version 2.0			
5/3/2017 1:53:29 PM	Peter Feleskouras	1.0	This is version 1.0			

If version 4.0 in the above example is Discarded from Production, then ONLY version 5.0 will be available to Set for Review from the Process Studio. All the other versions will NOT be available any more.

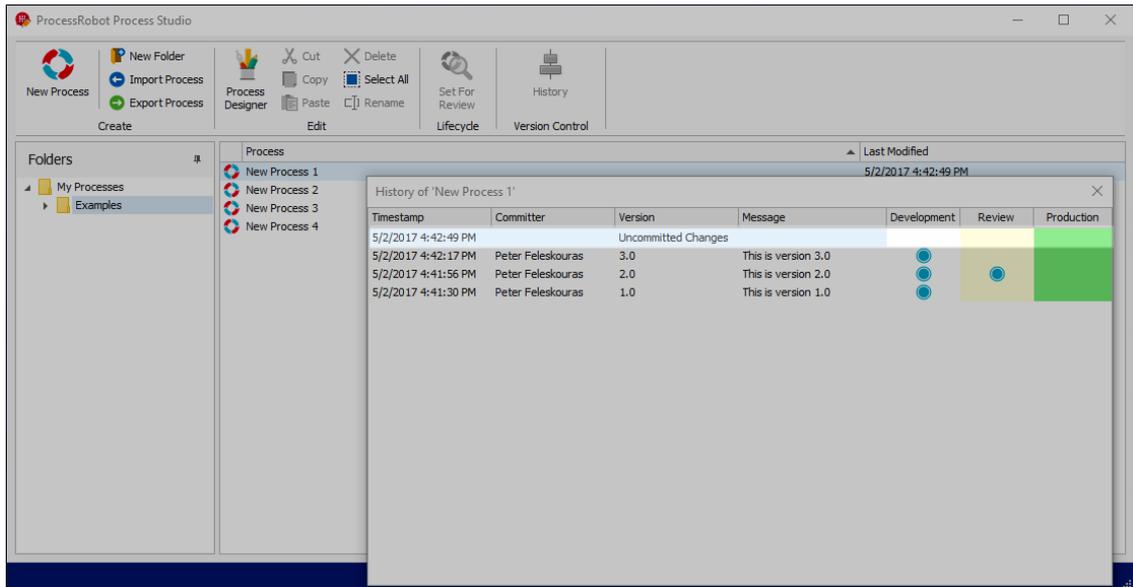
Note: Every time that a Process is transferred from one stage to another (forward or backwards) we are prompted to write a Message, in order to keep on record the reason of the transfer.

3.10.3.2 Manage Process Versions

In the Version Control section, the version History of a Process is shown, either in the Control Desk or in the Process Studio.

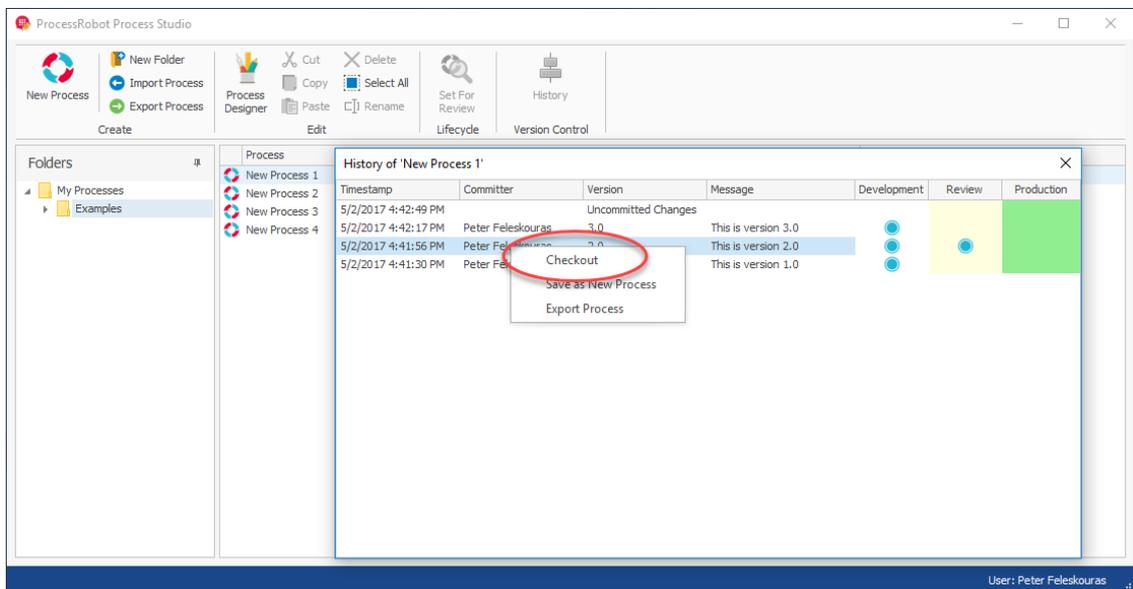
When a process is edited, then there is either the option to Save it or [Save and Commit](#)⁴⁶⁵. Should you choose to commit any changes that you make to the Process, a new Version of the Process is created and you have to define the Version of the committed changes. When a version is created, it is automatically placed at the Development stage of the Process's lifecycle.

If we edit a process and we just Save it, then an "Uncommitted Changes" version of the Process is created. Such version does not belong to any of the three stages of the product lifecycle (Development, Review, Production).

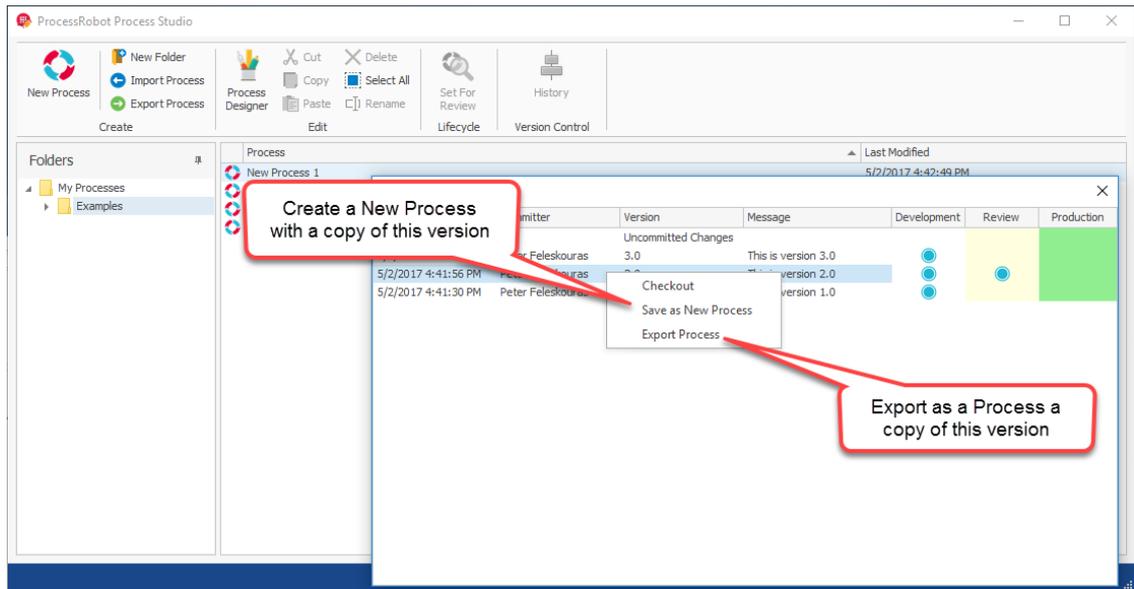


Each Process can have only one “Uncommitted Changes” version and is the only version that we are able to edit.

When we edit a Process, in reality, we edit a copy of the Process’s latest version. We can also edit a copy of another version, should we right click on the version “Checkout”. In this way, the copy of that version becomes the “Uncommitted Changes” version and we are prompted to commit the current History changes (previous Uncommitted Changes), if there is already such version in the Process’s History.



Instead of Checkout, there is also the choice to “Save as New Process” or “Export Process” when you right click on a version.



If we double click on a version, the “Commit Details” window will appear. In this window, we can find information about the version:

- Commit timestamp: Date and Time of the Commit.
- Committer: The User who made the Commit.
- Version: The version number.
- Commit Message: The message written when the Commit was made.
- The Transition History of the specific version through its lifecycle.

Commit Details
✕

Commit timestamp:
5/2/2017 5:54:46 PM

Committer:
Peter Feleskouras

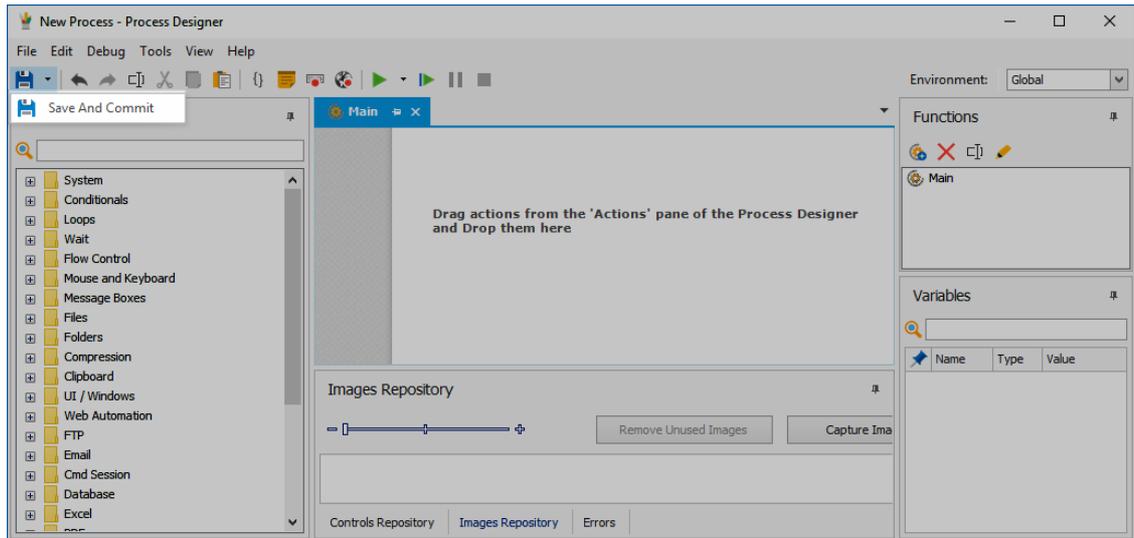
Version:
2.0

Commit Message:
This is version 2.0

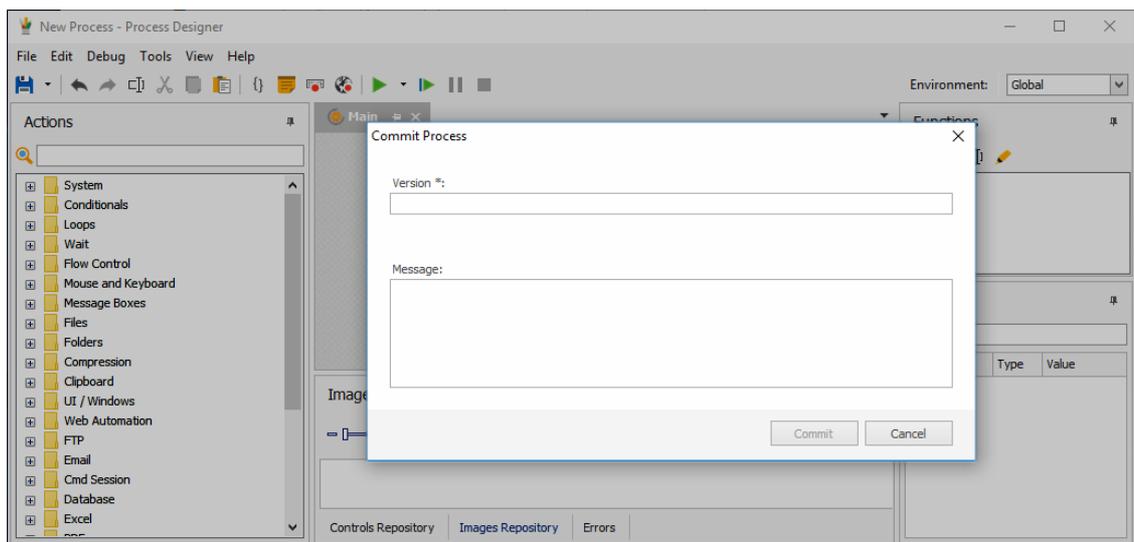
Timestamp	Username	Transition Type	Message
5/2/2017 5:56:54 PM	Peter Feleskouras	Production Version Discarded	Discard from Production
5/2/2017 5:55:22 PM	Peter Feleskouras	Moved to Production	Move version 2.0 to Production
5/2/2017 5:55:01 PM	Peter Feleskouras	Set for Review	Review version 2.0

3.10.3.3 Save and Commit

In the [Process Designer](#) ²⁸⁵ window, there is the option to either Save or Save and Commit a Process from the drop down list of the floppy disk icon. The Save and Commit option is used when the development of the Process has been completed and we are not willing to make any further changes in the future.

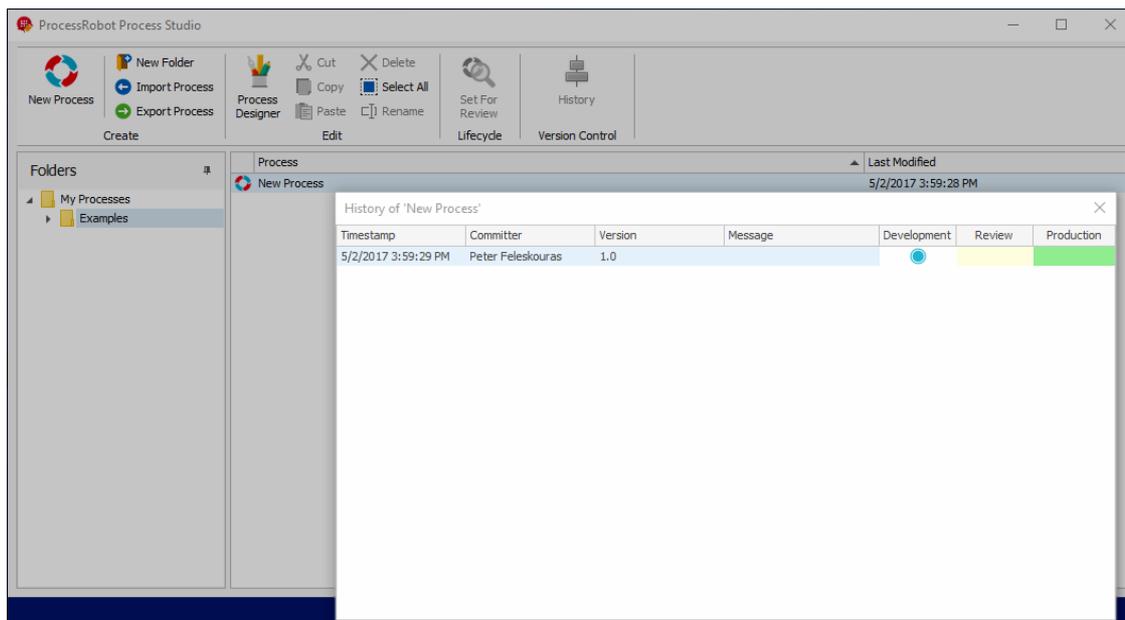


By clicking on this option a Version has to be assigned to the Process along with a corresponding message.



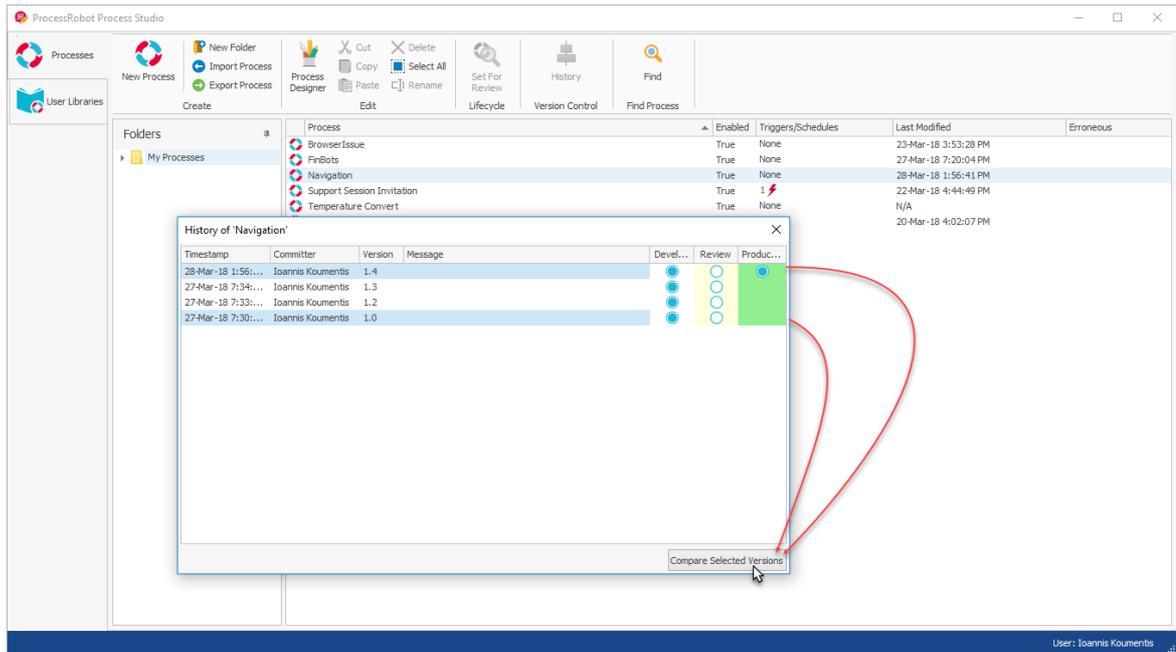
When a Process is committed, the Version of the Process which is being created can be neither modified nor deleted. Should you want to delete a Version, it is necessary that the whole Process is deleted.

A committed Process is at the stage of Development, as we can see from the Process Studio, by highlighting the Process and clicking on the History tab.



3.10.3.4 Compare two Process Versions

When the History window of a specific Process is open, you are able to observe the differences between two different versions by selecting them and clicking on the 'Compare Selected Versions' button.



Compare two different versions

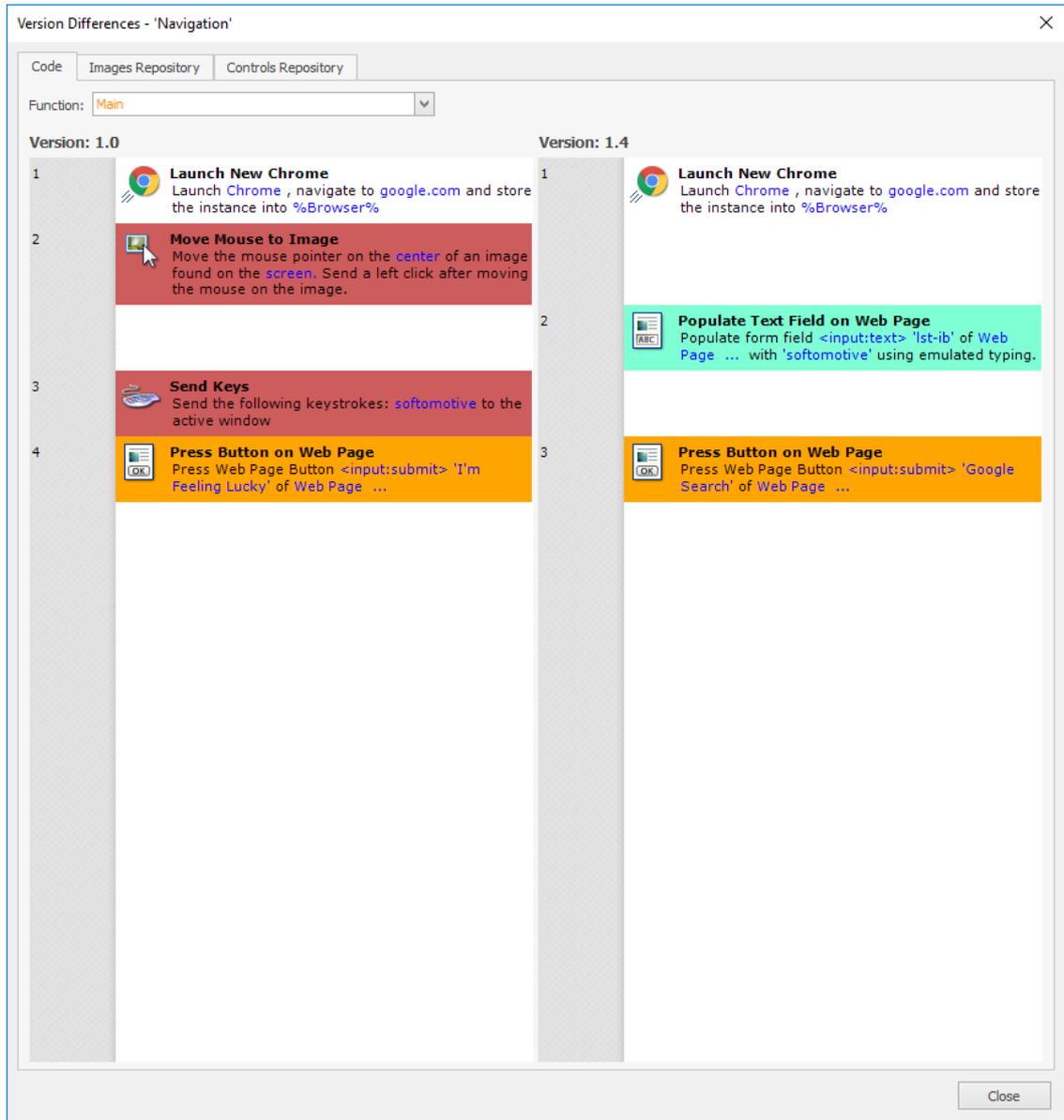
The Version Differences window opens, where the user can view the differences on the:

- Code (Process Designer)
- Images Repository
- Controls Repository

of the two selected Process versions.

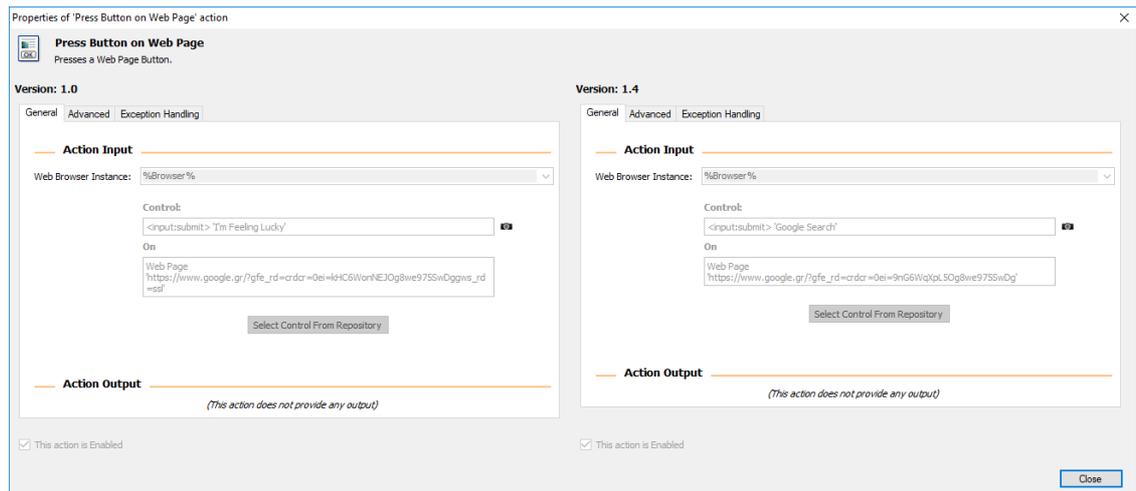
In the Code tab, you can review each version's actions, side by side, on all the available functions. The actions that have been removed from the newer version are highlighted in **red** color, those that existed in the older version and have been modified are highlighted in **orange** and the ones that have been added in the newer version are highlighted in **green**.

The font colors of the Functions' names follow the same pattern, accordingly.



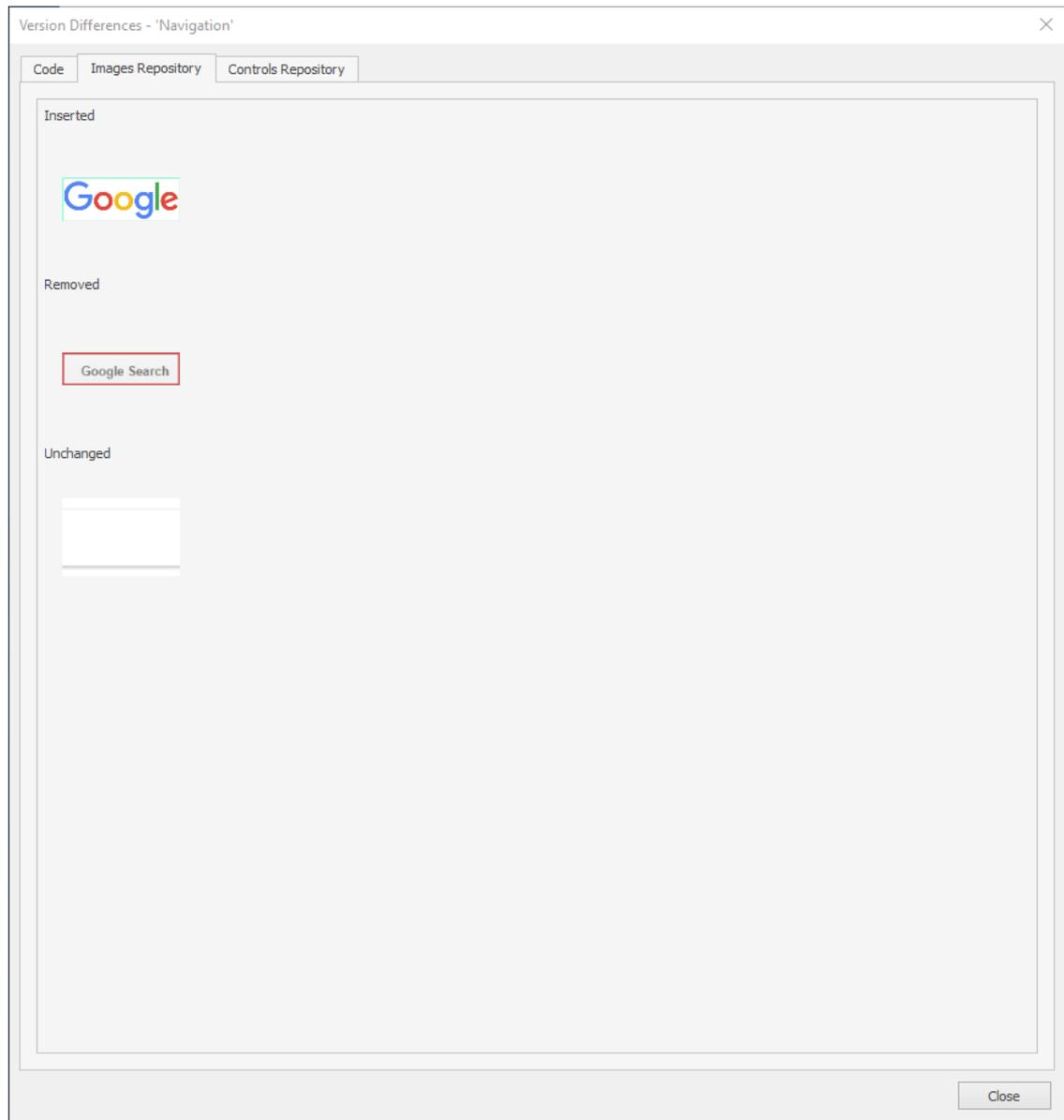
Code differences in Process Designer

By double-clicking on a specific action, a review of its Properties window opens, so that you can review the applied configuration. In case the action exists in both versions, its Properties windows are depicted side by side.



Differences in Action Properties

Any changes occurred in the Images Repository, can be reviewed on the corresponding tab. Please note that the Inserted images are depicted in **green** outline and the Removed images in **red** outline.



Differences in Images Repository

The same logic is applied in the Controls Repository tab where the controls that have been removed from the newer version are depicted in **red** font, those that existed in the older version and have been modified are depicted in **orange** font and the ones that have been added in the newer version are depicted in **green** font.

```

Version Differences - 'Navigation'
Code Images Repository Controls Repository
├ Web Page 'https://www.google.gr/?gfe_rd=cr&dc=0&ei=VXi7WqjxO4zKpy2spAP'
├   <input:submit> 'Google Search'
├     form[id="tsf"] > div[class="tsf-p"] > div[class="jsb"] > center > input[name="btnK"][type="submit"]
├     body > div > div:eq(2) > form > div:eq(1) > div:eq(2) > center > input[type="submit"]:eq(0)
├ Web Page 'https://www.google.gr/?gfe_rd=cr&dc=0&ei=kHC6WontNEJOg8we975SwDg&gws_rd=ssl'
├   <input:submit> 'I'm Feeling Lucky'
├     body > div > div:eq(2) > form > div:eq(1) > div:eq(2) > center > input[type="submit"]:eq(1)
├ Web Page 'https://www.google.gr/?gfe_rd=cr&dc=0&ei=9nG6WqXpL5Og8we975SwDg'
├   <input:text> 'Ist-ib'
├     input[id="Ist-ib"]
├     body > div > div:eq(2) > form > div:eq(1) > div:eq(1) > div:eq(0) > div:eq(0) > div:eq(1) > div > div > div:eq(1) > div > input[type="text"]:eq(0)
├   <input:submit> 'Google Search'
├     form[id="tsf"] > div[class="tsf-p"] > div[class="jsb"] > center > input[name="btnK"][type="submit"]
├     body > div > div:eq(2) > form > div:eq(1) > div:eq(2) > center > input[type="submit"]:eq(0)
Close

```

Differences in Controls Repository

3.10.4 Connecting to a Database

ProcessRobot can connect to any Database and apply SQL queries to it. All you have to do is find the correct connection string and pass it to the "[Execute SQL Statement](#)⁸⁷⁶" action.

In computing, a connection string is a string that specifies information about a data source and the means of connecting to it. It is passed in code to an underlying driver or provider in order to initiate the connection.

The connection string may include attributes such as the name of the driver, server and database, as well as security information such as user name and password.

3.10.4.1 Connect to Excel or Access Database

Connect to an Excel Database:

To connect to an Excel database the most common connection string that one can use is:

Provider=Microsoft.ACE.OLEDB.12.0;Data Source=C:\myFolder\myExcelFile.xlsx;Extended Properties="Excel 12.0 Xml;HDR=YES";

Where Data Source is the full path of the Excel File.

Connect to an Access Database:

To connect to an Access Database the most common connection string is the following:

Provider=Microsoft.ACE.OLEDB.12.0;Data Source=C:\myFolder\myAccessFile.accdb; Persist Security Info=False;

Where Data Source is the full path of your Access Database

For other Databases like SQL Server, Oracle, My SQL please, have a look at the [Connection strings](#) section.

3.10.4.2 Connection Strings and Drivers

More connection strings for any kind of Database can be found in the link below:

www.connectionstrings.com

Of course you are free to use variables in any connection string like for example:

Server=%ServerAddress%;Database=%myDataBase%;User Id=%Username%;Password=%Password%;

or

Provider=Microsoft.ACE.OLEDB.12.0;Data Source=%ExcelFile%;Extended Properties="Excel 12.0 Xml;HDR=YES";

Most common Drivers List:

You must download and install the appropriate drivers from the list below according to the database that you want to connect to:

<https://www.microsoft.com/en-us/download/details.aspx?id=5793>

<https://www.microsoft.com/en-us/download/details.aspx?id=13255>

<https://www.microsoft.com/en-us/download/details.aspx?id=27835>

<http://www.ch-werner.de/sqliteodbc/>

<https://dev.mysql.com/downloads/connector/odbc/>

<https://dev.mysql.com/downloads/connector/odbc/5.1.html> (For previous generally available ODBC connectors. Make sure that you download the correct one according to your machine characteristics 32-bit or 64-bit)

For Process Development tips regarding SQL queries and Connection Strings, click on the following link:

<https://support.softomotive.com/support/solutions/articles/35000106850-tips-on-sql-queries-and-connection-strings>

3.10.5 Conditionals

Conditionals are very important concept ProcessRobot, since they allow you to execute specific actions or blocks of actions only if a certain condition is met. If the condition is not met, the action(s) that the conditional structure contains will be skipped.

Usage Examples

There are tons of examples about where you might want to use a conditional: if you want to delete only files that have not been modified during the last week, if you want to know whether a specific process is running before launching an application, if you want to verify that a file does not exist before creating it and so on. Practically, conditionals are the first thing that you will have to use in order to automate tasks through ProcessRobot, since they give you the ability to choose whether some part of your Process will be executed or not.

Note: ProcessRobot will automatically indent the block of actions that is contained in a conditional structure.

3.10.5.1 If/Else/EndIf

The most common way to control which actions of your Process will be executed is to use the "If" action. This action checks whether or not a certain condition is true. If so, the action or block of actions, that are included between "If" and "End If", will be executed.

An "If" statement consists of 2 operands (the operands are the pieces of information that will be compared) and an operator. The operator can be:

- "is equal to" (=),
- "is not equal to" (<> or !=),
- "is greater than" (>),
- "is greater or equal to" (>=),
- "is smaller than" (<)
- "is smaller or equal to" (<=)
- Starts with
- Does not Start with
- Contains
- Does not contain
- Ends with

- **Does not End with**
- **Is Empty**
- **Is not Empty**

and it defines how the first operand should relate with the second operand in order for the condition to be met.

An "If" statement may optionally contain an "Else" action. This action should always be placed between an "If" and an "End If" action, and it defines the action or set of actions that will be executed if the condition that is defined by the "If" statement is not met. Basically, you will have to place the actions that will be executed if the condition is met between "If" and "Else", and the actions that will be executed if the condition is not met between "Else" and "End If".

You can also use the "Else If" action (which can be placed within the If/End If block and always before the Else action) to set mutually excluded conditions for which different blocks of actions need to be executed.

Note: If the first and the second operands are variables, you will have to make sure that they have the same data type or the condition will, most likely, not be met (or it will always be met if you used the <> operator). For example, if your conditional checks whether a variable of Date Time type is equal to a variable that contains a file, this condition will never be met, hence the set of actions contained in the conditional will never be executed.

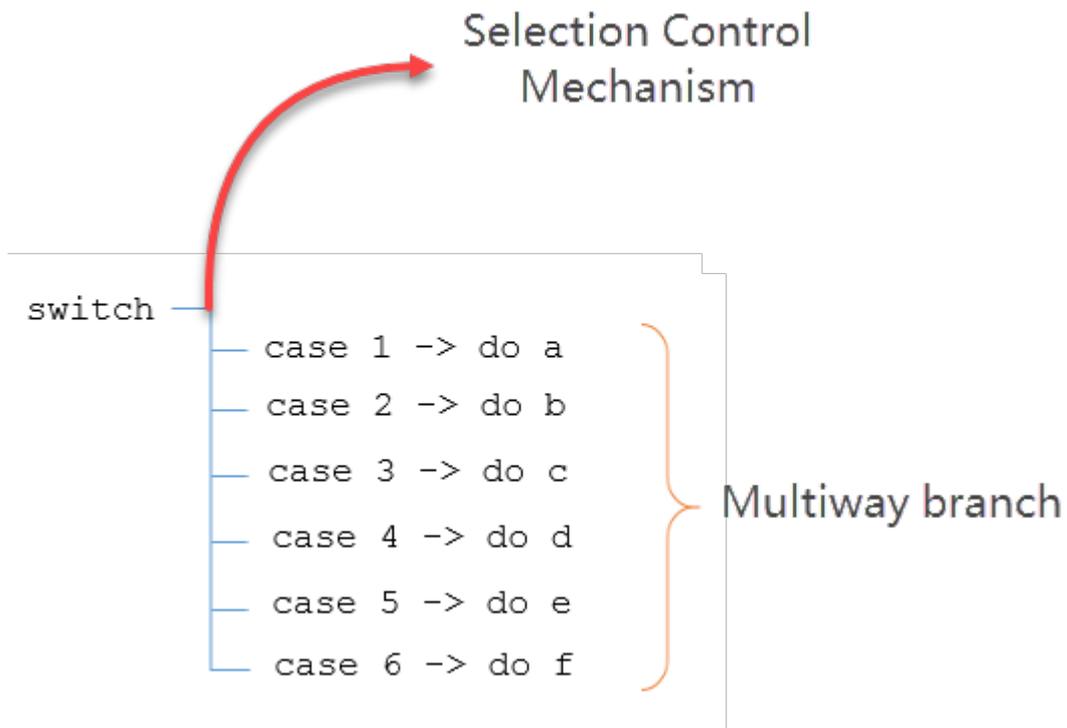
3.10.5.2 If Variations

Apart from the standard If action, there are 5 additional conditional actions:

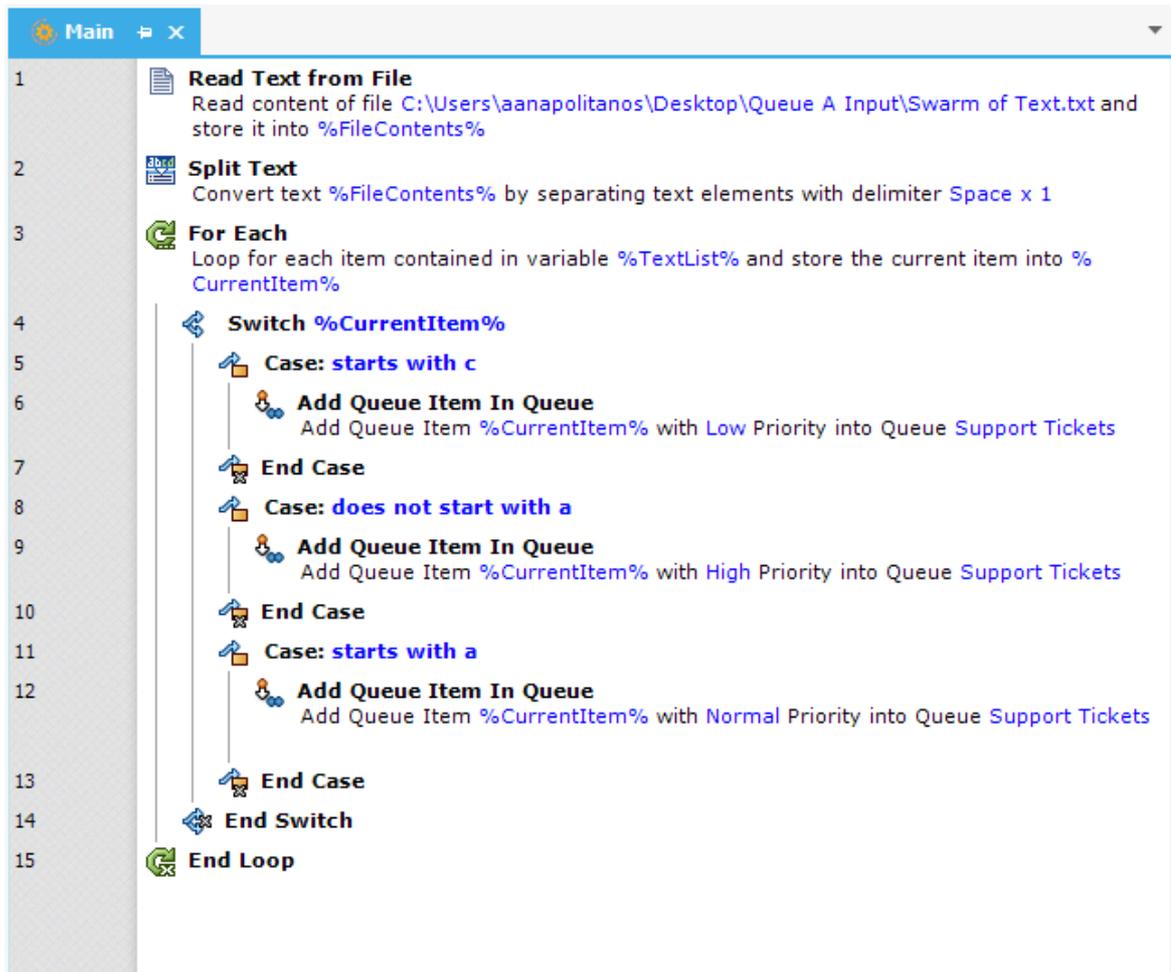
1. [If File Exists](#)^[559]: Allows you to check if a file exists (or does not exist) before executing an action (or a block of actions). This action can prove very useful if, for example, you want to know if a specific file exists before performing an action on it.
2. [If Folder Exists](#)^[560]: Allows you to check if a folder exists (or does not exist) before executing an action (or a block of actions).
3. [If Service](#)^[561]: Allows you to perform an action (or a block of actions) if a specific service is running (or is paused or is stopped).
4. [If Process](#)^[562]: Allows you to perform an action (or a block of actions) if a specific process is running (or is not running).
5. [If Window](#)^[562]: Allows you to perform an action (or a block of actions) if a specific window is open (or is not open).
6. [If Image](#)^[566]: Allows you to perform an action (or a block of actions) if a specific image is found/is not found on the screen.
7. [If Web Page Contains](#)^[573]: Allows you to perform an action (of a block of actions) if a web page contains/does not contain a specific element or some text.
8. [If text on Screen \(OCR\)](#)^[567]: Allows you to check if a given text is found on the screen or not, using an OCR Engine of your choice.

3.10.5.3 Switch-Case

[Switch](#)^[578] marks the beginning of a switch block. Unlike if-else or else-if conditional statements, a switch block can address a number of possible execution paths. A switch statement is a type of selection control mechanism used to allow the value of a variable or expression to change the control flow of your scripts via a multiway branch:



[Case](#)^[580] specifies a condition on an existing [Switch](#)^[578] block and marks the beginning of a sequence of actions that will be executed if the conditional evaluates to true.



An example of a very simple process using Switch-Case

The end of this sequence of actions is marked by the complimentary to [Case](#)^[580], [End Case](#)^[583] action.

[End Switch](#)^[580] marks the end of a [Switch](#)^[578] block.

3.10.6 Enate Actions

Enate is a Robotic Service Orchestration (RSO) platform. Through this platform, Users can break down business processes, called "Cases" on Enate, into smaller tasks, called "Actions" and "Tickets". Users can then allocate these Actions and Tickets so that each can be performed by a human, RPA software, or artificial intelligence (AI) technologies and digital agents.

ProcessRobot is integrated with Enate so that its Enate Actions enable Users to:

- Log in and out of Enate
- Check for and Get Available Work on Enate
- Create Cases, Actions and Tickets on Enate

- Update Cases, Actions, Tickets and Checklists
- Add Files to and Save Files from Enate Items

Read more about Enate on their official website: <http://www.enate.net>

3.10.6.1 Authentication on Enate

Logging into Enate

The first step for all interactions with the Enate platform is to use the [Login to Enate](#)^[951] Action. This is the only Action that creates the [Enate Instance](#)^[436], which is required by the majority of subsequent Enate Actions. This Action requires the following Input:

- Enate URL
- Username
- Password

The Username and Password can be stored as a Credential in the [Credential Manager](#)^[241] and retrieved using the [Get Credential](#)^[1015] Action. These can then be entered into the Login to Enate Action using the format:

- %Credential.Username%
- %Credential.Password%

✕
Properties of 'Login to Enate' action


Login to Enate
 This Action Logs in to Enate and returns an Enate Instance

General
Advanced
Exception Handling

Action Input

Enate URL: i ⚙

Username: i ⚙

Enter Password: As Variable i

Password: ⚙ i

Action Output

Enate Instance: i

Status Code: i

This action is Enabled

More Info
OK
Cancel

Logging out of Enate

When interaction with Enate with a particular User has completed, the [Logout from Enate](#)⁹⁶⁶ Action will be used to close the session. This Action requires as Input only the Enate Instance.

3.10.6.2 Retrieving Work from Enate

To retrieve work from Enate, it is first necessary to create the [Enate Instance](#)^[436] by using the [Login to Enate](#)^[957] Action. The credentials of an Enate Robot -not to be confused with a ProcessRobot Robot- with the appropriate permissions will be used to log in.

Having logged in to Enate, the next step is to check whether there is available work to be performed. For this, the [Is Work Available on Enate](#)^[952] Action is used. This Action requires as Input the Enate Instance created with the Login to Enate Action. It produces an Output Variable that stores a [Boolean](#)^[436] (true or false) value regarding whether or not there is available work.

If there is available work on Enate, the value of this Output Variable will be "true". A [Loop Condition](#)^[586] can be used to repeat a set of Actions for as long as the value remains "true".

The first Action in this set will be the [Get More Work from Enate](#)^[953] Action. This Action produces as Output a [Custom Object](#)^[436].

Next, the [Get Action from Enate](#)^[961] Action will be used to obtain the Enate Action, not to be confused with a ProcessRobot Action. As Input, this Action requires:

- the Enate Instance
- the Action GUID (Global Unique Identifier); this GUID is an attribute of the Enate Work Custom Object and can be entered in the format %EnateWork["GUID"]%

Properties of 'Get Action from Enate' action

Get Action from Enate
This Action retrieves an Action from Enate

General | Advanced | Exception Handling

Action Input

Enate Instance: %EnateInstance%

Action GUID: %EnateWork["GUID"]%

Existing Packet Activity GUID:

Action Output

Action: %EnateAction%

Status Code: %StatusCode5%

This action is Enabled

More Info OK Cancel

This Action produces as Output a Custom Object. This Custom Object has several attributes, some of which are in turn also Custom Objects.

See the [Using Enate Actions in Processes](#)^[483] article for more details about how these attributes can be used to determine a Process' flow.

3.10.6.3 Creating Cases on Enate

Once logged into Enate with a User that has the appropriate permissions, the [Create Case on Enate](#)^[953] Action can be used.

This Action accepts multiple Input values:

- [Enate Instance](#)^[436]; this has been created by the [Login to Enate](#)^[951] Action
- Customer
- Contract
- Service
- Process

The above four values must be known to the User and must correspond to an existing Customer, Contract, Service and Process in the User's Enate account.

- Case Title; a User-defined value to name the new Case.
- Email Address; must correspond to an existing contact email in the User's account.

The Action will store the created case as a [Custom Object](#)^[436]. This is important, as this Custom Object is required to perform subsequent operations on the Case.

To add files or update the Case on Enate, the User must first retrieve it using the [Get Case from Enate](#)^[955] Action.

This Action requires the Case GUID to be entered as Input. The GUID is an attribute of the Custom Object created by the Create Case on Enate Action and can be entered as % EnateCreatedCase.GUID%.

When the Get Case from Enate Action runs successfully, it will create another Custom Object for the Enate Case.

The [Add File to Enate Item](#)^[964] Action can be used to add a file to an Enate Case, Action or Ticket. The Enate Item is specified by entering the Custom Object in the appropriate field in the Action's Input.

Additionally, this Action requires these Input values:

- the filepath of the file to upload
- a note describing the uploaded file

The [Update Case on Enate](#)^[956] Action is used to update the Case once the file has been added. This Action accepts two Input values:

- the Enate Instance
- the Enate Case Custom Object

Having created the Case, added a file to it and updated it, the User can end the session. To achieve this, the [Logout from Enate](#)^[966] Action is used. This Action requires as Input only the Enate Instance. (See the [Authentication on Enate](#)^[478] article for more details.)

3.10.6.4 Using Enate Actions in Processes

The [Authentication on Enate](#)^[478] and [Retrieving Work from Enate](#)^[480] articles cover how to log into Enate and retrieve an Enate Action.

Assuming an Enate Action has been retrieved in a Process using the [Get Action from Enate](#)^[961] Action, the Enate Action will be stored in a [Custom Object](#)^[436] Output Variable.

The values of the EnateAction Custom Object's attributes can be used to determine how a Process should proceed.

For example, one of the attributes of the EnateAction Custom Object is CurrentStep, which is in turn also a Custom Object. One of the attributes of the CurrentStep Custom Object is StepName. The name of the current step in the Enate Action is stored in this attribute.

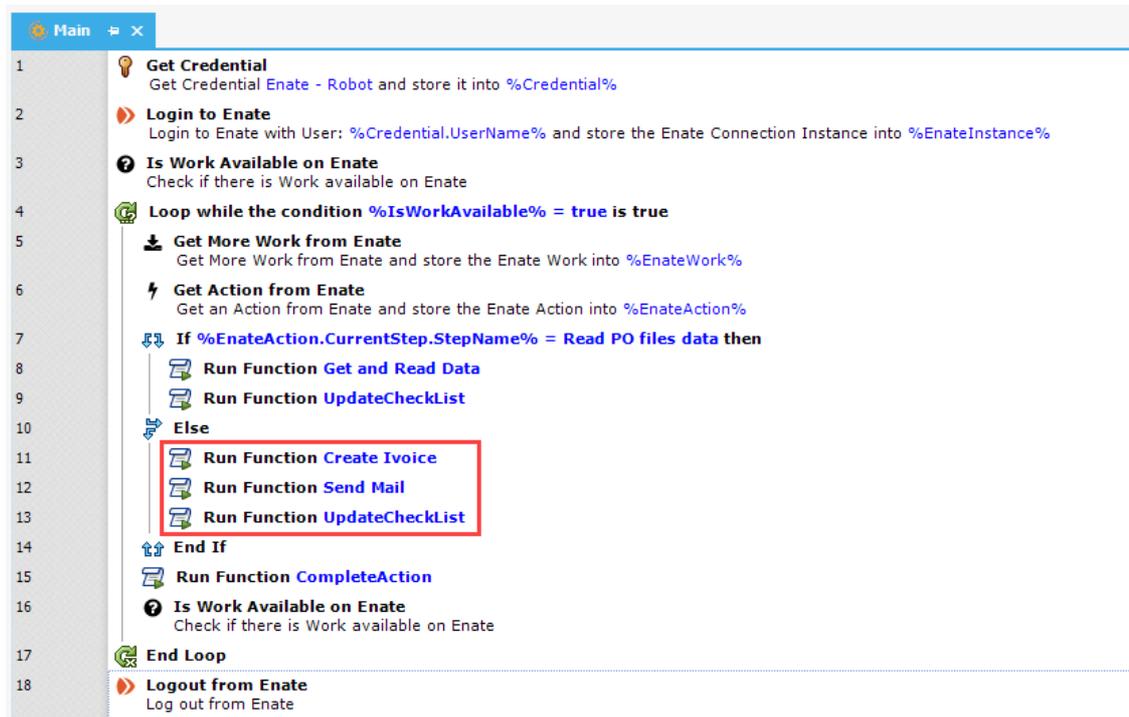
Assuming the User knows the names of the steps included in the Enate Action, the value of this attribute can be used to further determine the direction of the Process.

For example, an [If](#)^[557] Action can be used to check whether the current step name is a specific value, such as "Read PO files data". If this condition is true, then a certain set of Actions or Functions can be executed.

```

1 Get Credential
  Get Credential Enate - Robot and store it into %Credential%
2 Login to Enate
  Login to Enate with User: %Credential.UserName% and store the Enate Connection Instance into %EnateInstance%
3 Is Work Available on Enate
  Check if there is Work available on Enate
4 Loop while the condition %IsWorkAvailable% = true is true
5 Get More Work from Enate
  Get More Work from Enate and store the Enate Work into %EnateWork%
6 Get Action from Enate
  Get an Action from Enate and store the Enate Action into %EnateAction%
7 If %EnateAction.CurrentStep.StepName% = Read PO files data then
8   Run Function Get and Read Data
9   Run Function UpdateCheckList
10 Else
11   Run Function Create Invoice
12   Run Function Send Mail
13   Run Function UpdateCheckList
14 End If
15 Run Function CompleteAction
16 Is Work Available on Enate
  Check if there is Work available on Enate
17 End Loop
18 Logout from Enate
  Log out from Enate
  
```

Alternatively, if the current step name is any value other than the one specified, such as "Read PO files data", then a different set of Actions or Functions can be executed, using an [Else](#)^[576] Action.



3.10.7 CSS Selectors

Web Automation functionality is one of the most important and commonly used features of ProcessRobot. Web-related actions will allow you to automatically click on links, fill and submit web forms, as well as scrape data from web pages. You can build Web Automation Processes either by combining the appropriate Actions in the Process designer, or by recording your interaction with a webpage and automatically converting it into Process actions through the Web Recorder.

While ProcessRobot provides you with quite a specific visual feedback (through the Web Helper or the Extract Preview window) about the webpage elements you have selected, it can also provide you with detailed information about the values used to point to the position of these elements in the HTML page, found in the Control Repository.

CSS (standing for Cascading Style Sheets) was introduced together with HTML 4, to denote the presentation semantics (the look and formatting) of a document written in a markup language. A CSS rule (essentially the statement that tells browsers how to render particular elements on an HTML page) consists of two parts, the selector (which "selects" the elements on an HTML page that are affected by the rule set) and the declaration block (which defines the style to be applied to those elements). You will find links to several external resources focusing on CSS, at the bottom of this article.

In ProcessRobot we use the CSS selectors to pinpoint the web page elements we want to interact with. The information that is generated and displayed in the CSS Selector field is based on the underlying tree structure of the HTML page.

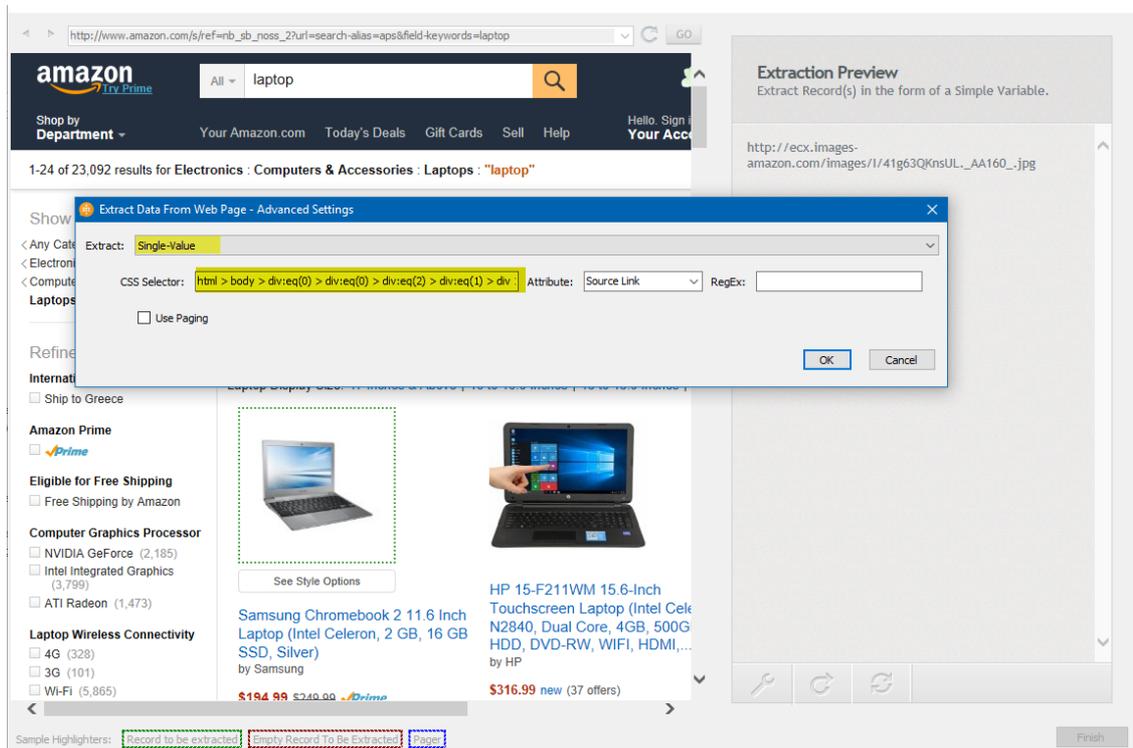
For most of the cases, defining the desired the webpage elements through the Web Helper proves to be sufficient but there are exceptions for which the user has to manually point to the html element. This may be necessary either due to specific design characteristics of the HTML page structure or to subsequent -to our Process design- changes of it (although ProcessRobot is adaptable to some of these variations and can still track the initially user-defined elements).

To overcome this inconsistency, user can manually include or remove attributes for a selector, or build one from scratch that will uniquely and undoubtedly define the desired element.

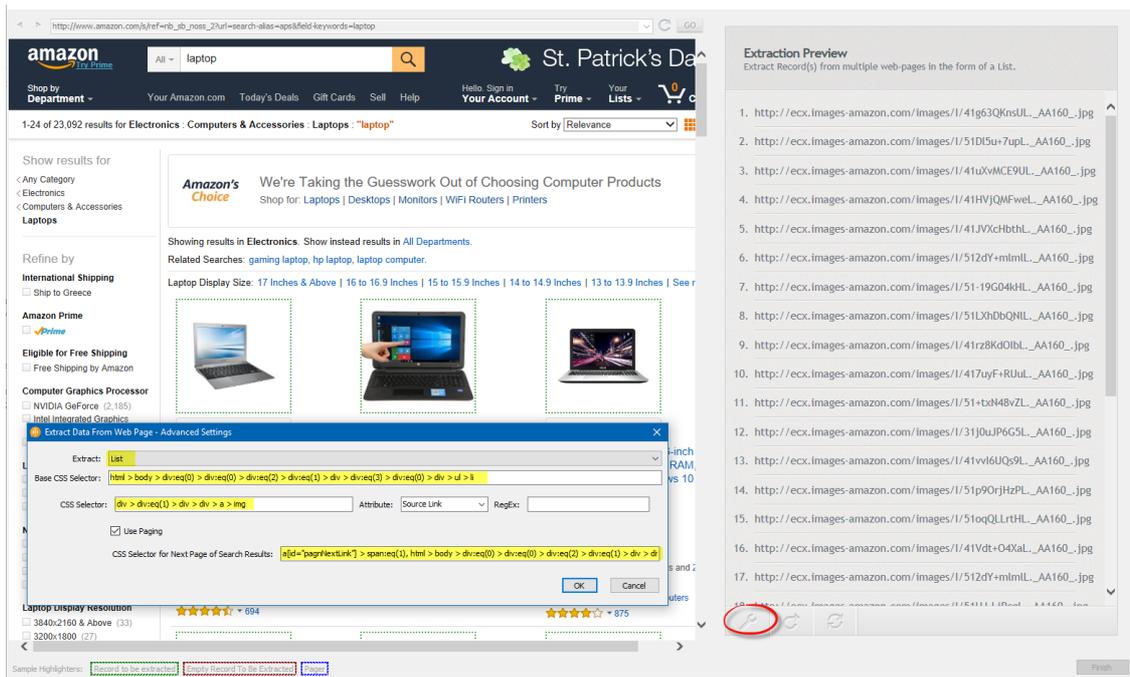
Just click on the control whose selector you want to edit, click on the selector, click on "Edit", select the "Custom" option from the top left part of the Selector Builder window and build your selector in the CSS Selector field. The steps are explained in detail in the [Building a WebAutomation Process](#) ^[353] topic.

Selecting controls - HTML elements with CSS Selectors, can be achieved relatively easy within ProcessRobot (with add control and a LeftShift+click), but it requires a more advanced knowledge of HTML and CSS when you want to build your own selectors.

Especially for the Extract Data from Web Page action, the CSS Selector can be edited through the Advanced Settings Icon of the Web Helper window (opened by the Specify Web Data to Extract button of the action's Property Window).



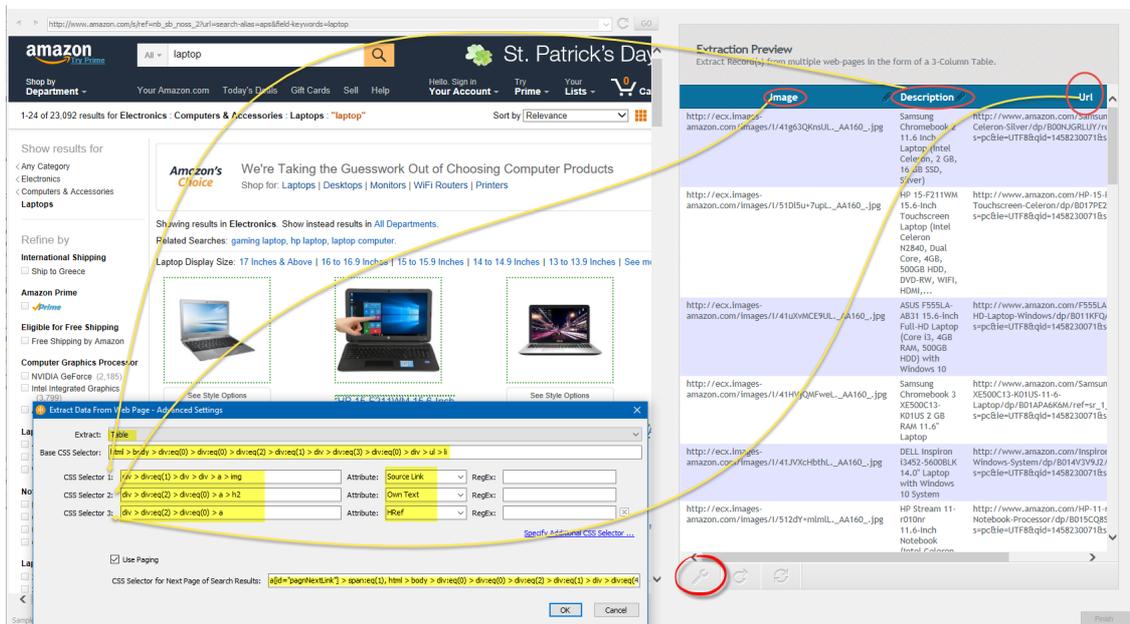
Moreover, when extracting list-type data from a web-page, the CSS Selector has the following structure:



where the "Base CSS Selector" represents the route on the HTML tree that points to the level of elements that constitute the list, while the "CSS Selector" part points to the specific attribute of each element (or each descendent) we want to extract.

The same logic applies also when extracting data in table format, as also explained in detailed in the Using ProcessRobot > Web Automation ([Web Data Extraction Part I](#)³⁷³, [Web Data Extraction Part II](#)³⁷⁸, [Web Data Extraction Part III](#))³⁸² topic.

In this case we have to specify as many CSS Selectors as the columns of the table (the categories of data we want to retrieve for each item).



Similar to the inconsistencies that may appear when defining the desired elements through the Web Helper, we may also have to manually input the "CSS Selector for Next Page of Search Results" (used for cases of extracting records in forms other than Single Variables).

Using manually inputted CSS selectors can be a powerful alternative in pointing the desired elements for the minority of cases that webpage 's structure hinders ProcessRobot's ability to automatically spot and interact with them.

Using classes (.) and ids (#) in the CSS Selectors:

To View the source HTML code of a web page, press F12. There you can see how the page is structured and how each element is represented.

If the element you are trying to access contains a text i.e. it is a div and the html shows the below:

```
<div class="someclass" id="someid">text</div>
```

Should you wish to use its id or class i.e. then you should have something like

```
...>...> div.someclass or ...>...> div[class="someclass"] if you want to use its class or,
```

```
...> ...> div#someid or ...> ...> div[id="someid"] if you want to use its id. Note, that the id attribute is unique in the HTML for each element, so if you know the id then you can ignore the previous route and simply go with "div#someid" !
```

A very useful attribute that you can use is the "[:contains\("sometext"\)](#)". If you know that the element you are trying to access contains a specific string then you can use this attribute to access it, i.e. for the example above:

```
...>...> div:contains("text")
```

External Links and Helpful Tutorials

http://www.w3schools.com/css/css_intro.asp

http://www.w3schools.com/cssref/css_selectors.asp

<http://css.maxdesign.com.au/selectutorial>

<http://benhowdle.im/cssselectors>

<https://api.jquery.com/category/selectors/>

<http://www.ProcessRobot.com/community/topic/projeto-web-recorder-running-extremely-slow/#post-62796>

3.10.8 Email Interaction with Two Step Verification

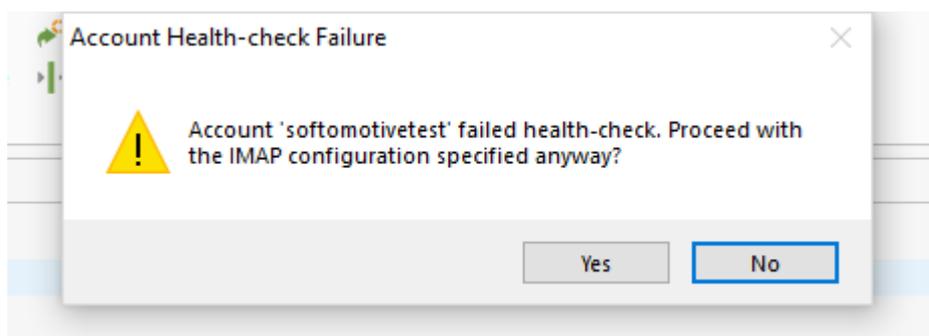
Two-Step Verification

Most email providers offer an extra layer of security for their accounts in addition to the account's password. This security type is most commonly known as two-step verification. This extra security layer is enforced by sending a confirmation request through an SMS or phone call to an account holder's phone number, a push notification to a trusted device or an email to a secondary email address.

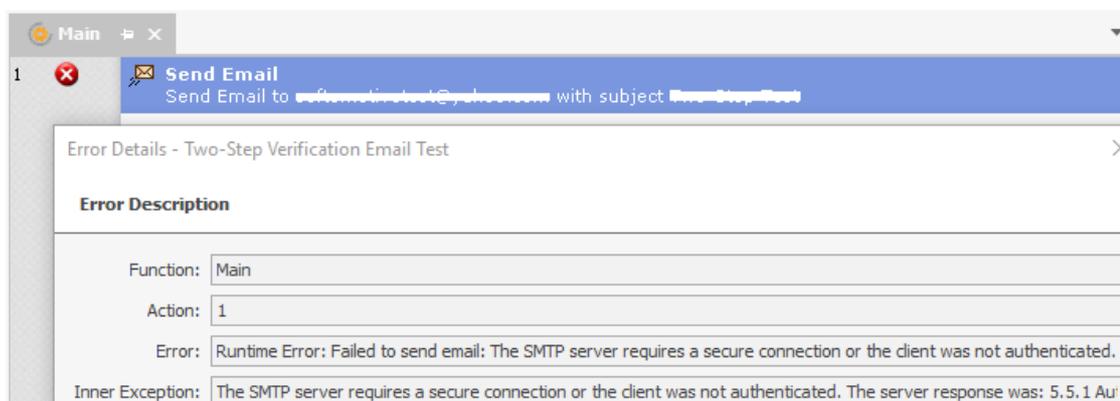
If you have enabled the two-step verification on your email account, you will find that if you try to use this account in ProcessRobot, you will receive an error unless an app-specific password is created through the email account.

Errors

When ProcessRobot tries to access an email account that has the two-step verification enabled but does not have an app-specific password created for ProcessRobot, this will result in errors. In the example below, an Email Monitor Trigger is configured to check the inbox of an email account with two-step verification enabled. The correct username, password and IMAP (incoming mail server) settings are entered. ProcessRobot attempts to connect to the email account and returns the following error: "Account failed health-check. Proceed with the IMAP configuration specified anyway?"



In the Process Studio, an error sending an email because ProcessRobot was unable to access the email account due to two-step verification would look like this:



Another example of where a relevant error could occur is in the Audit tab of the ProcessRobot Control Desk:

Date & Time	Source	Message	Screenshot
8/9/2018 11:59:56 AM	Two-Ste...	Process 'Two-Step Verification Email Test' completed with result: Error in Function 'Mai...	

In the Details area of the Audit tab is a more detailed description of the error shown above:

Details

Selected Event:

Message: Process 'Two-Step Verification Email Test' completed with result: Error in Function 'Main' at Action#1 'Send Email'

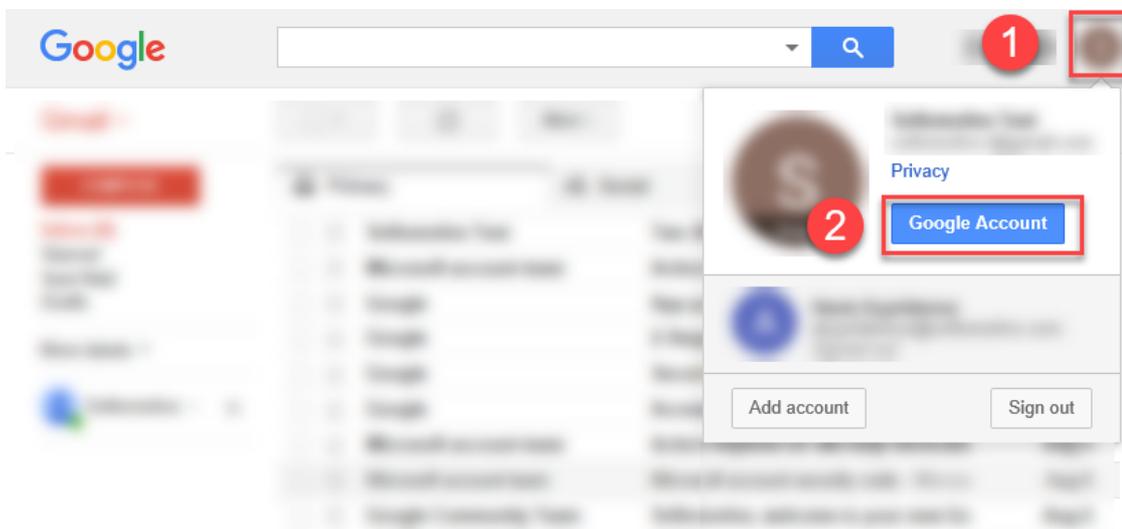
Additional Details: Failed to send email: The SMTP server requires a secure connection or the client was not authenticated. The server response was: 5.7.57 SMTP; Client was not authenticated to send anonymous mail during MAIL FROM [HE1PR0102CA0065.eurprd01.prod.exchangelabs.com]

How to Create an App-Specific Password:

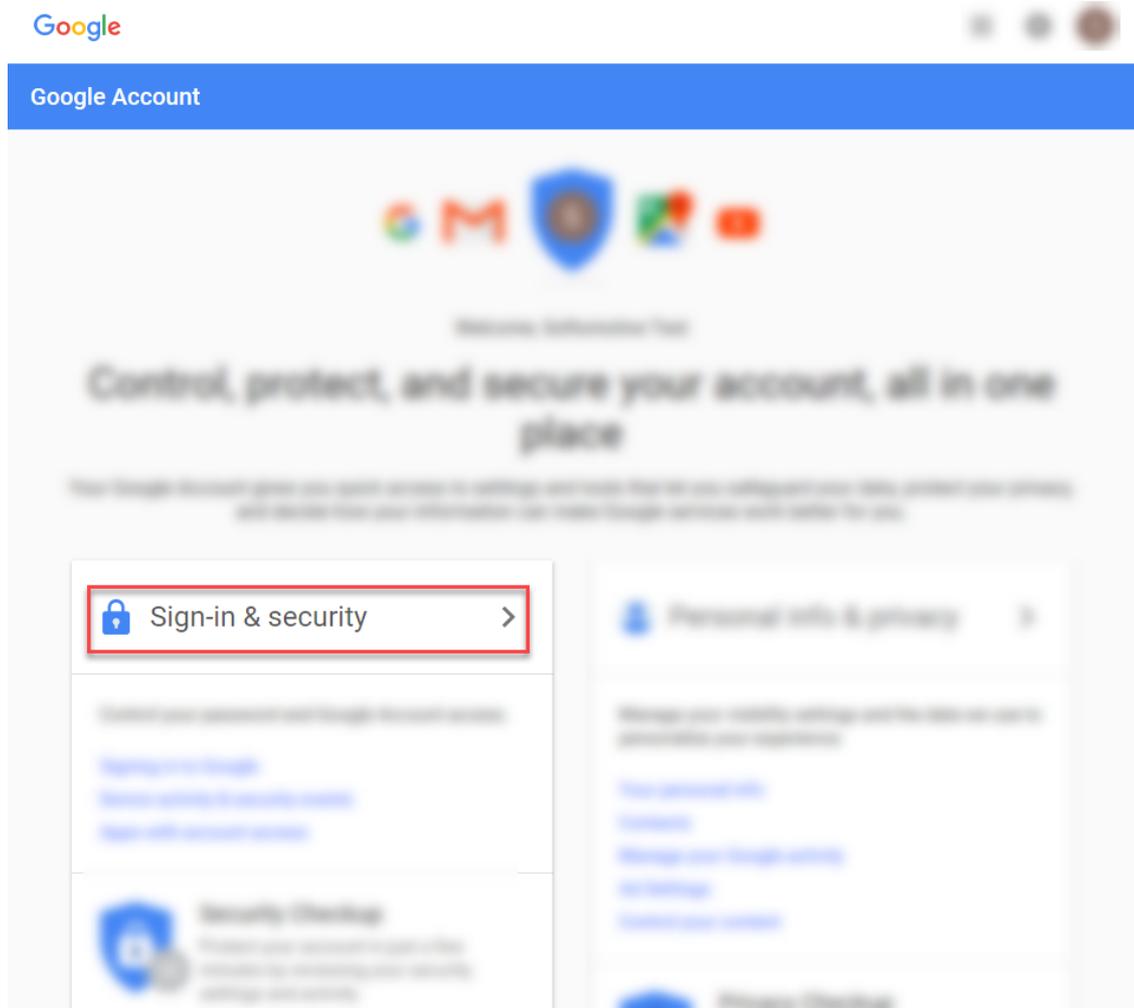
The solution for errors like the ones in the examples above is to create an app-specific password for ProcessRobot, so that it can access and use the email account despite the two-step verification. Below is a step-by-step guide to creating an app-specific password for ProcessRobot with three popular email providers: Gmail, Outlook and Yahoo.

Gmail

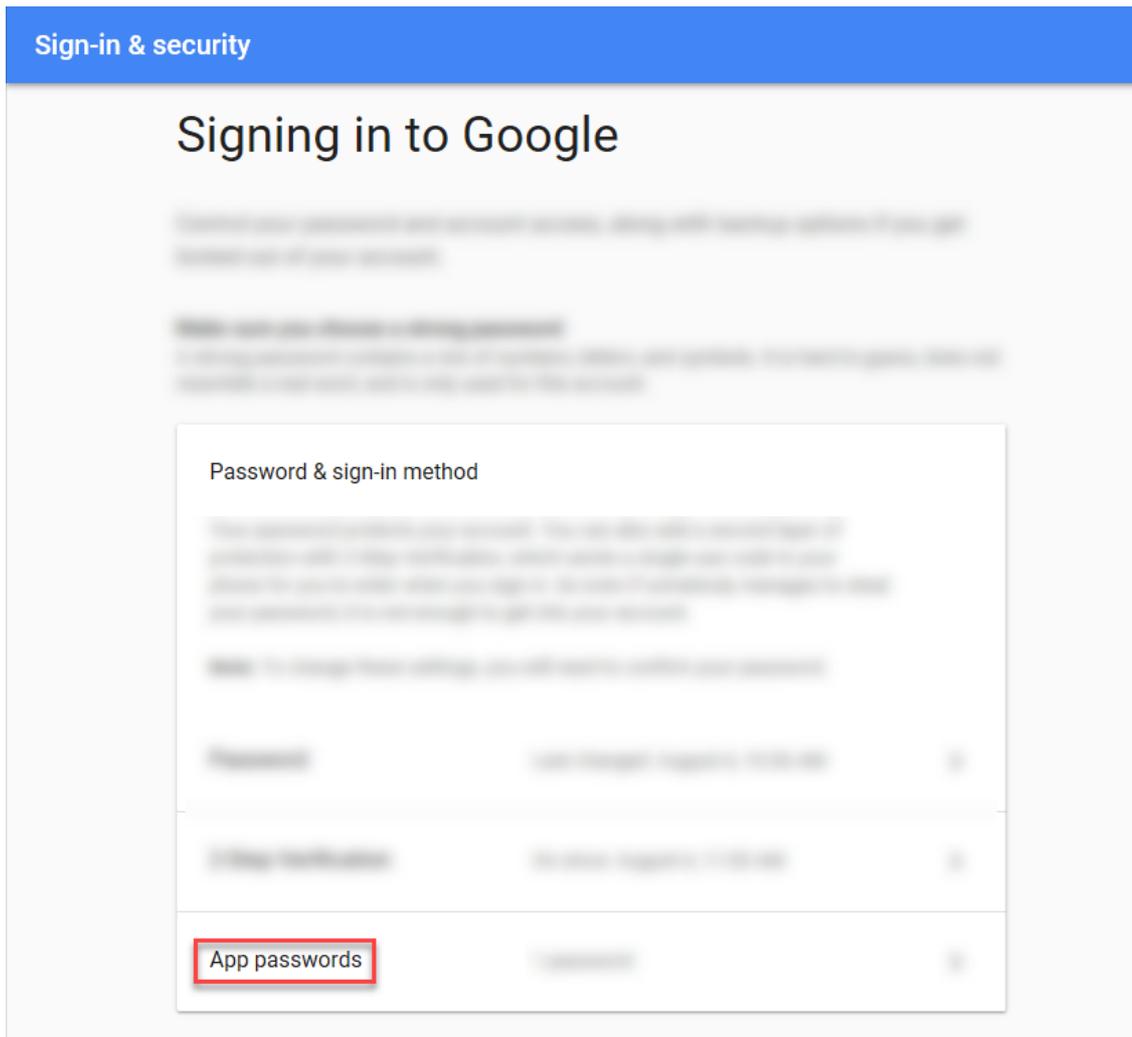
Sign in to the Gmail account. Click on the icon in the top right and select "Google Account".



From the Google account page, select "Sign-in & security".

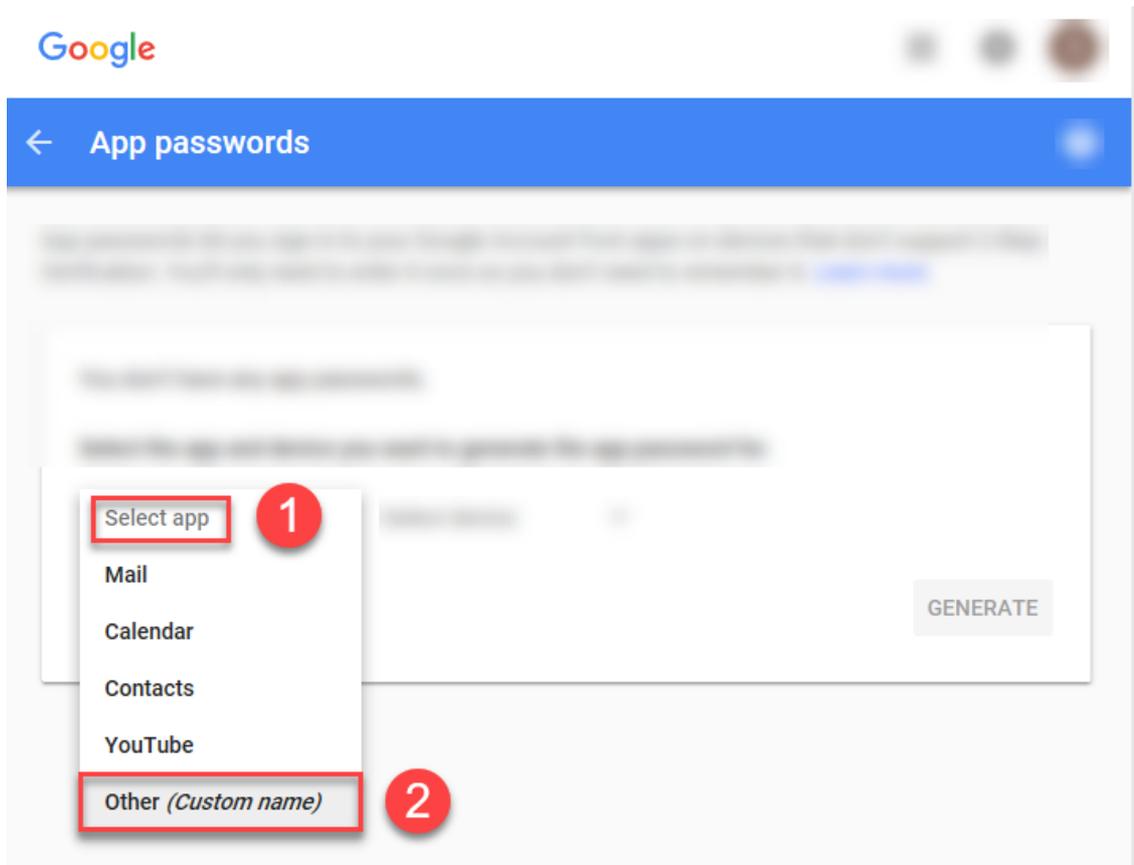


On the Sign-in & security screen, scroll down to "Signing in to Google" and the "Password & sign-in method" box. If 2-Step Verification is enabled, the last section in this box will be "App passwords". Click on App passwords.

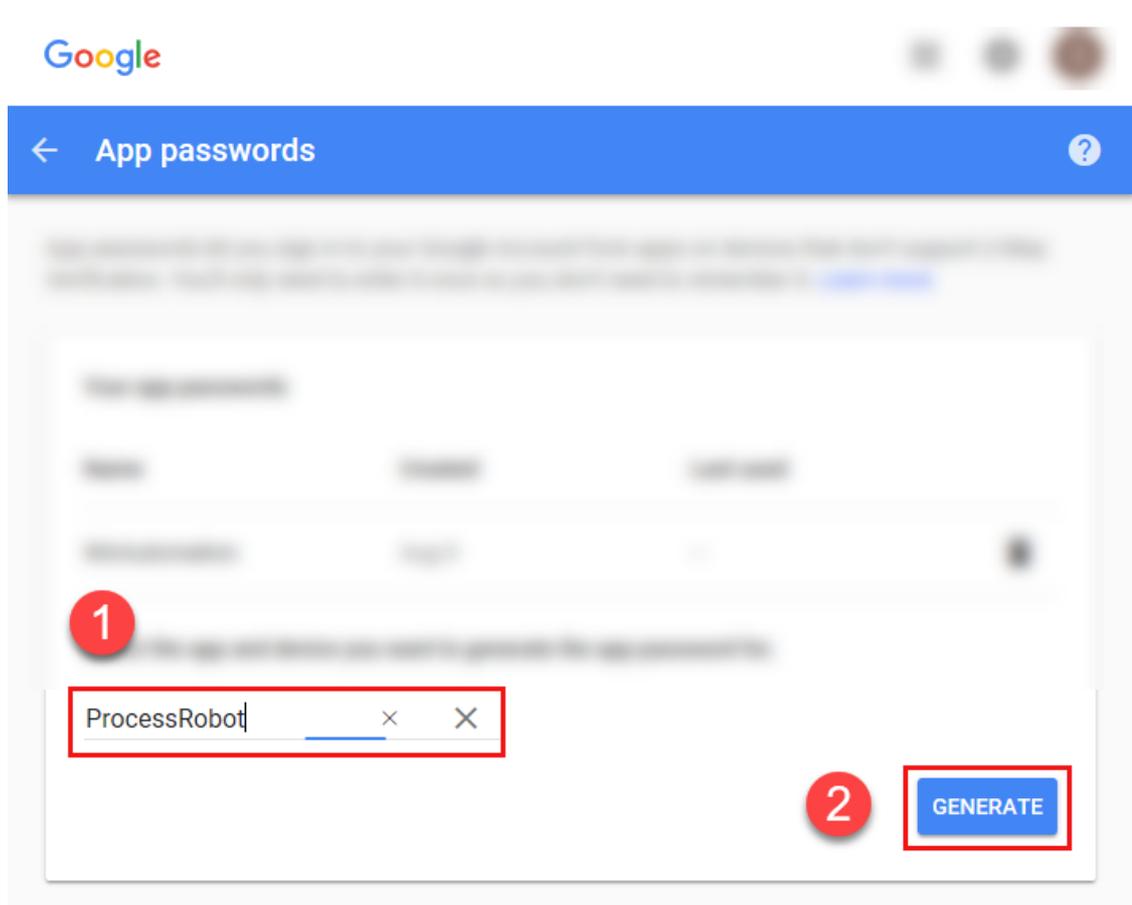


To display the "App passwords" screen, Gmail will require the account password to be re-entered.

On the "App passwords" screen, click "Select app" and choose the last option, "Other (Custom name)".



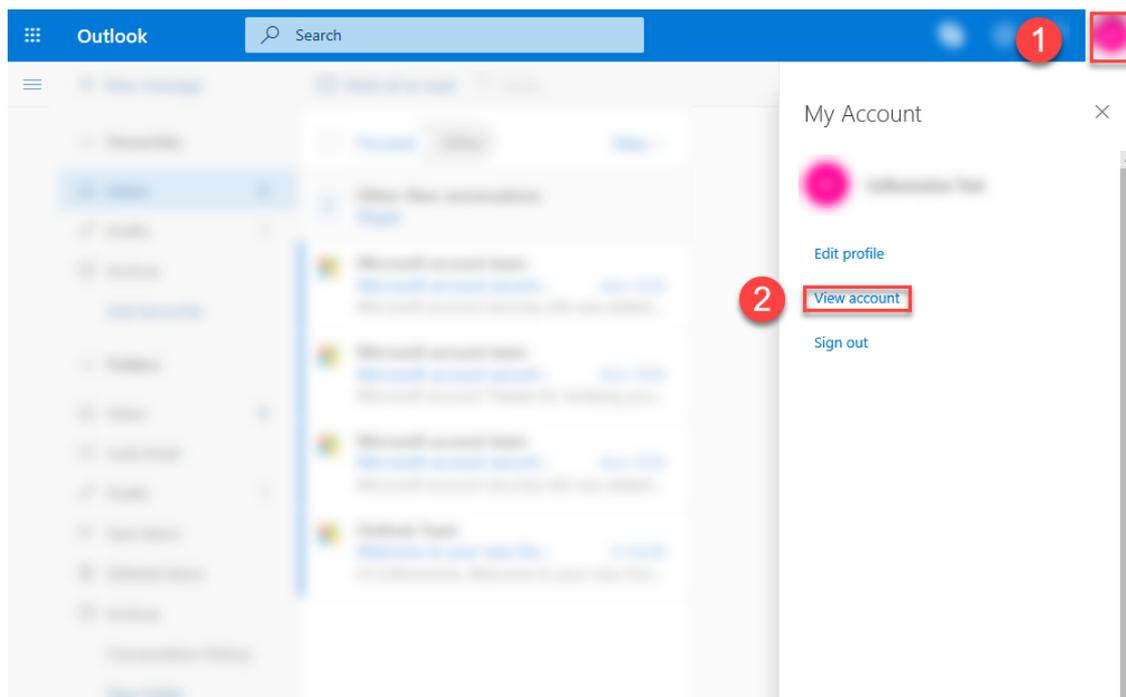
Next, type the name of the app for which you want to create the password, in this case "ProcessRobot", and click "Generate".



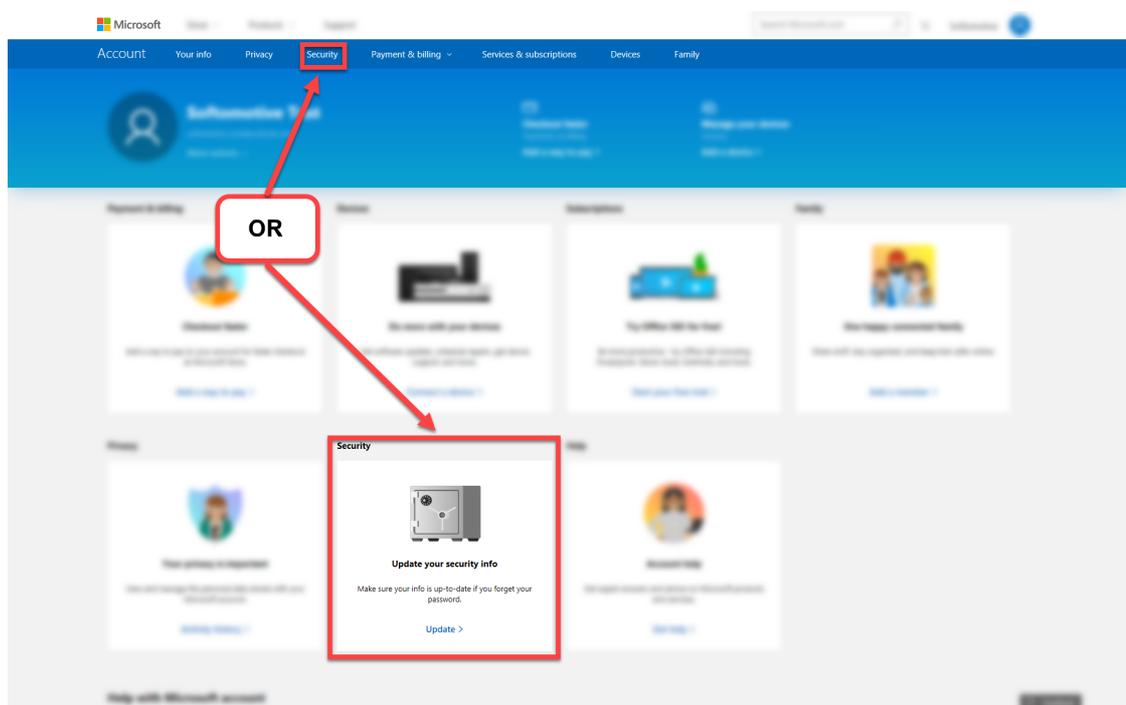
The next box that appears will display the generated password, which you can copy and paste into the screen of ProcessRobot that requires the email account's password.

Outlook

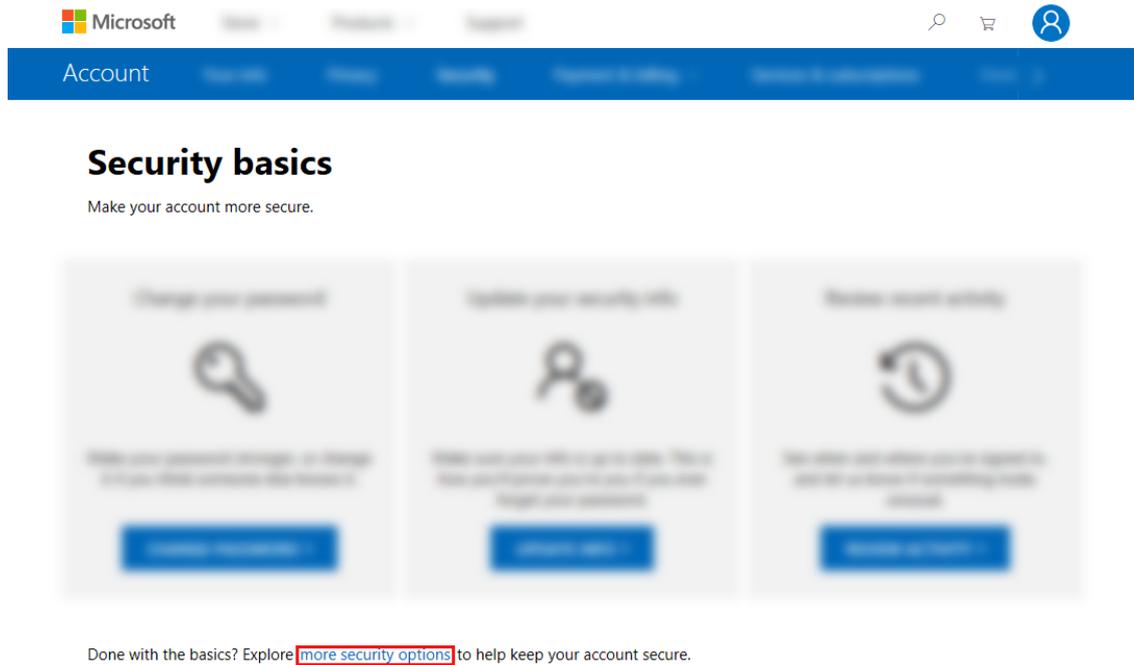
Sign into the Outlook email account. Click on the icon in the top right and select "View account".



On the account page, select "Security" from the top bar or the boxes in the middle of the screen.



On the next screen, with the heading "Security basics", click on the link at the very bottom, the highlighted part of the sentence: "Explore more security options to help keep your account secure."



The screenshot shows the Microsoft Account Security Basics page. At the top, there is a Microsoft logo and navigation links for Home, Outlook, and Teams. On the right, there are icons for search, shopping cart, and a user profile. Below this is a blue navigation bar with the word "Account" and several menu items. The main heading is "Security basics" with the subtext "Make your account more secure." Below this are three cards: "Change your password" (with a key icon), "Update your security info" (with a person icon), and "Review recent activity" (with a clock icon). Each card has a blue button at the bottom. At the bottom of the page, there is a link: "Done with the basics? Explore [more security options](#) to help keep your account secure."

Outlook will require the account password to be re-entered.

On the next screen, with the heading "Additional security options", scroll down to the "App passwords" section and click "Create a new app password".



Additional security options

Manage how you sign in to Microsoft

How you sign in to your account is what you use to get into your account. You can sign in with a password or use a security key to sign in.

[Manage sign-in options](#)

Two-step verification

Two-step verification is a security feature that adds an extra layer of protection to your account.

[Turn on two-step verification](#)

Identify verification apps

A verification app is the best way to verify your identity. [Learn more](#)

When you use an app or device that doesn't support security codes, you need to use an app password to sign in. [Learn more about app passwords](#)

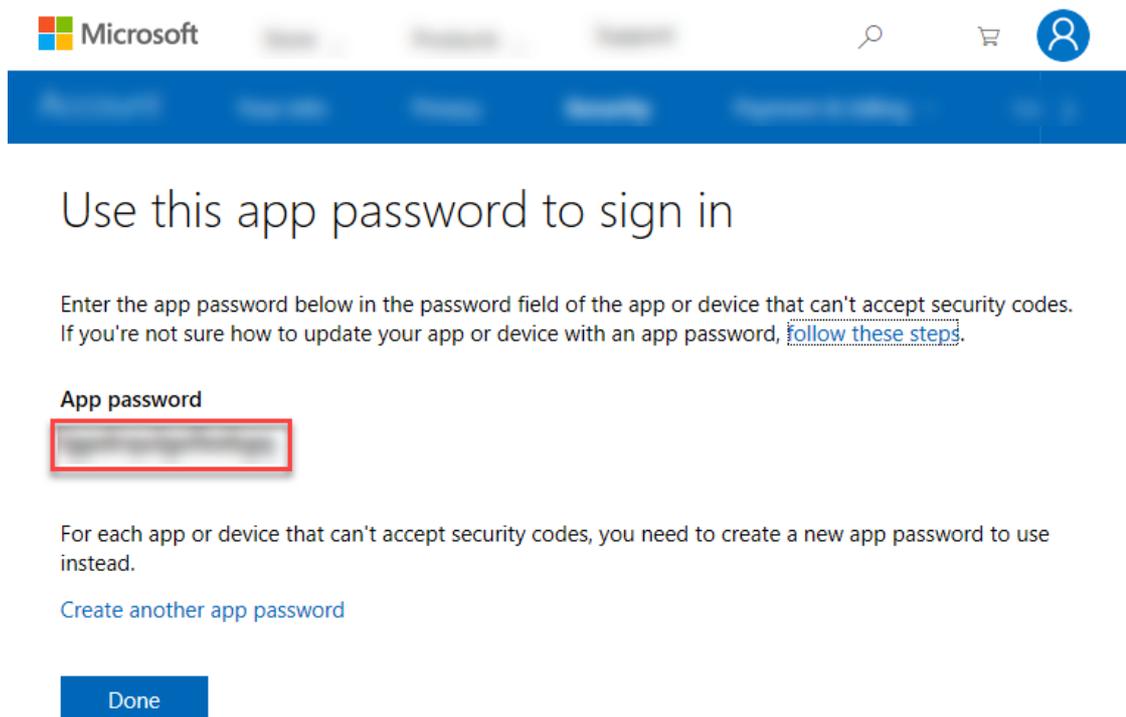
[Create a new app password](#)

App passwords

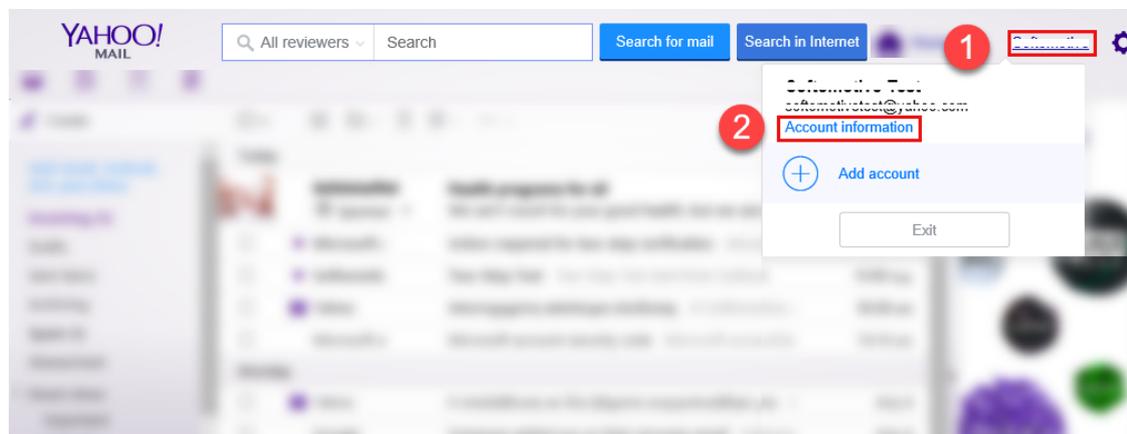
Some apps and devices (such as Xbox 360, Windows Phone, or mail apps on your other devices) don't support security codes for two-step verification. In these cases, you need to create an app password to sign in. [Learn more about app passwords](#).

[Create a new app password](#)

The next screen, with the heading "Use this app password to sign in" will display the app-specific password.



Copy and paste this password into the screen of ProcessRobot which requires it.



Yahoo

Sign into the Yahoo email account. Click on the icon with the account name in the top right and select "Account information". This button can also appear as "Account info" or "Personal info" depending on your location and language settings.

The screenshot shows the Yahoo! account settings interface. At the top, there is a purple header with the 'YAHOO!' logo on the left and a search bar, a grid icon, and an envelope icon on the right. Below the header, the page is titled 'Personal information' with an 'Edit' link on the right. On the left side, there is a vertical sidebar with several menu items: 'Personal information' (with a person icon), 'Account security' (with a shield icon and highlighted by a red box), 'Recent activity' (with a clock icon), 'Preferences' (with a list icon), and a 'Help' button. The main content area on the right shows a large circular profile picture placeholder with a camera icon, and below it, several rows of blurred text representing account details.

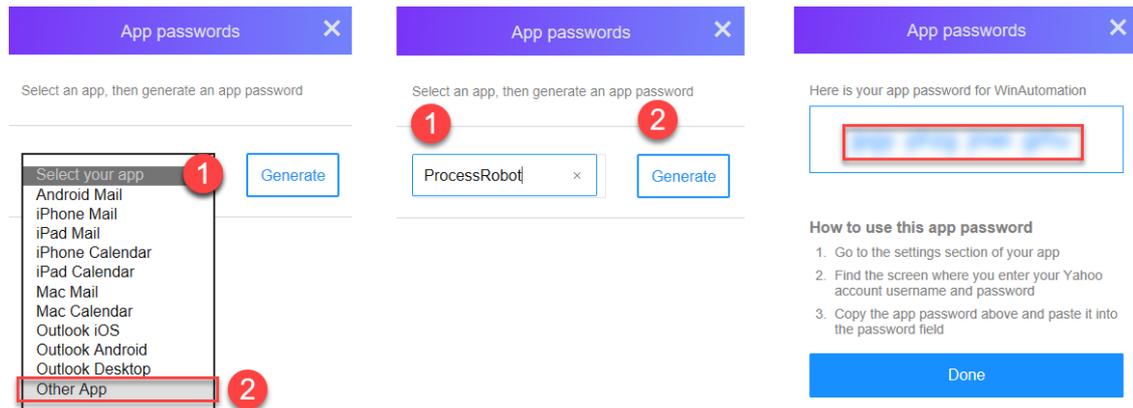
On the "Personal info" screen, click on "Account security" on the left.

The screenshot shows the Yahoo! Account Security settings page. The top navigation bar is purple with the Yahoo! logo on the left and a user profile icon, a grid icon, and a mail icon on the right. The left sidebar contains a menu with the following items: Personal info, Account security (selected), Recent activity, Preferences, and a Help button. The main content area is titled "Account security" and is divided into several sections: "How you sign in" (Password is enabled, with a "Change password" link), "Secure your account with Yahoo Account Key" (Use your phone to sign in, with a "See how it works" button), "Phone numbers" (blurred), "Add recovery email address" (link), "Two-step verification" (Additional security, with a toggle switch turned on), and "Generate app password" (highlighted with a red box, with a sub-note: "Apps like Outlook for desktop not working? Generate an app password to reconnect them.").

Yahoo will require the account password to be re-entered before displaying the "Account security" screen.

On the "Account security" screen, click on the last option, "Generate app password".

In the "App passwords" box that appears, click on "Select your app" and select the last option, "Other App". Type the name of the app in the "Enter custom name" box, in this case "ProcessRobot", then click "Generate". The next box will display the generated app-specific password. Copy this and paste it into the screen of ProcessRobot that requires it.



Two-Step Verification and Other Email Providers

In this article disabling two-step verification was described for only three types of account; Gmail, Outlook and Yahoo, but many other email providers offer it as well.

3.10.9 The use of the % sign

In ProcessRobot, the % sign is used as brackets around a Variable. For example, "%name%" is the Variable 'name'. When you use %name% as an input, the Action will use the value that has previously been stored in that Variable, so you should have an earlier Action that assigned a value to %name% as an output. By assigning output values to Variables, and then using them as input in later steps, you can pass information through a ProcessRobot Process.

All Variables are cleared at the end of a Process, so Variables can only run from one Process to another if they run at the same time - see [External Variables](#)⁴⁴⁴.

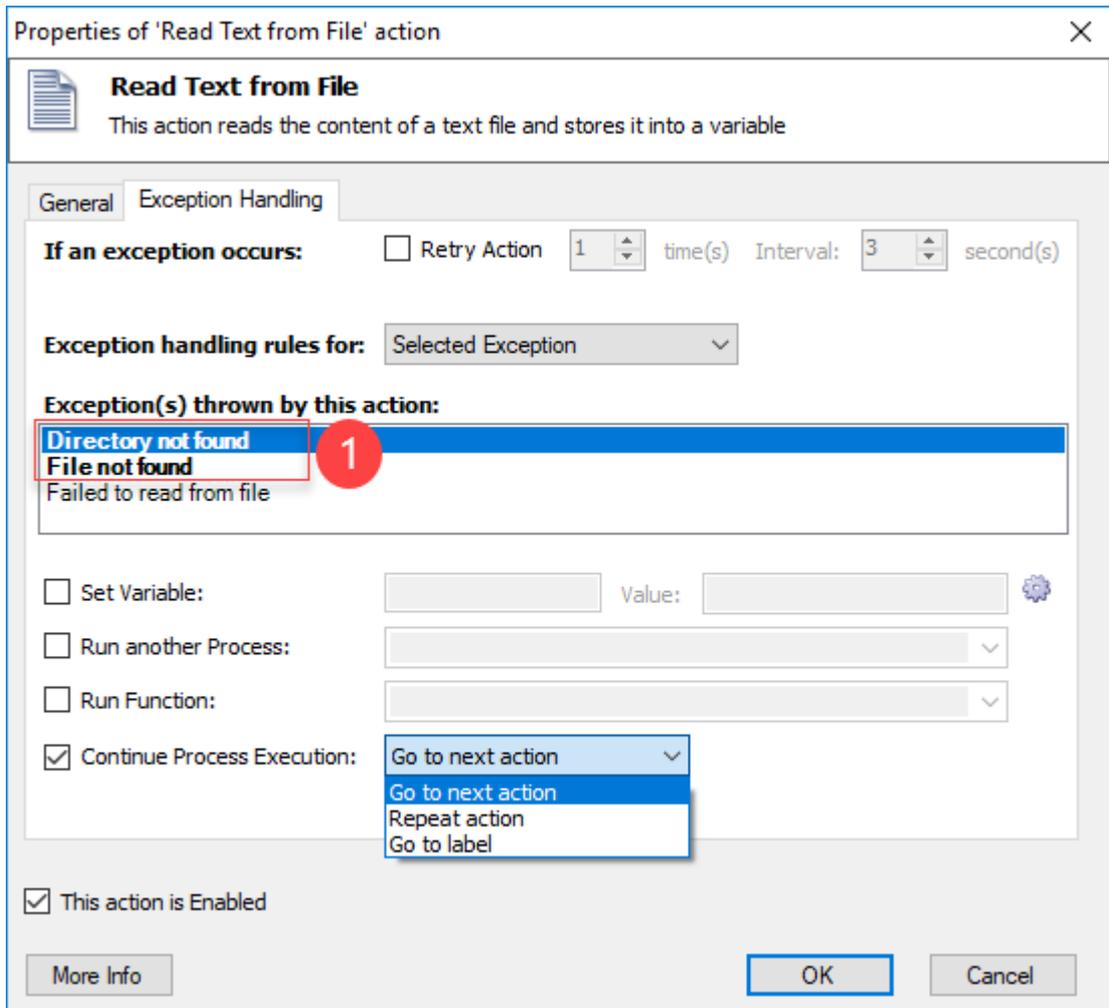
There are two special situations to this notation. The first, if you want to use the percent sign as a percent sign, special notation is needed so as not to call a Variable. Therefore, write a backslash first to denote 'this is not a Variable'. Thus if you want to, say, display the text "5%" you should write "5\%" instead.

If you want to access a file named by a Variable, you might write "C:\folder\%filename%". This won't work, **as "\%" means it isn't a Variable**. Therefore, in this special case, you would need to write, "C:\folder\\%filename%", as the double backslashes mean the Variable is a Variable. Just remember:

- %filename% is the Variable 'filename'
- \%filename is the text %filename
- C:\folder\\%filename% is the path to a file named in the Variable 'filename' in the folder C:\folder\

3.10.10 Exception Handling

Exceptions occur when a Process finds itself in an unexpected situation. For example a "Read Text from File" Action may attempt to read from a file that does not exist while the Robot is running. It is also known as a Run-Time Error. The default response of ProcessRobot to Exceptions is to fail that Process. However, other options are also allowed. By going into the Exception Handling tab of an Action's Properties, you can specify how the Action will handle an exception.



The Exception Handling Tab of an Action for a Selected Exception

First, by default, it will fail the Action and thus the Process. Next, you could tell it to retry the Action after a specified delay, as many times as you want. For example, you could set the Exception Handling to retry 8 times, once every 5 seconds.

Then all the exception that this specific action can throw are listed in the "Exception(s) thrown by

this action" list. By clicking on a specific exception, you can specify how this exception will be handled.

In general, there are four responses an Action can have to a specific exception.

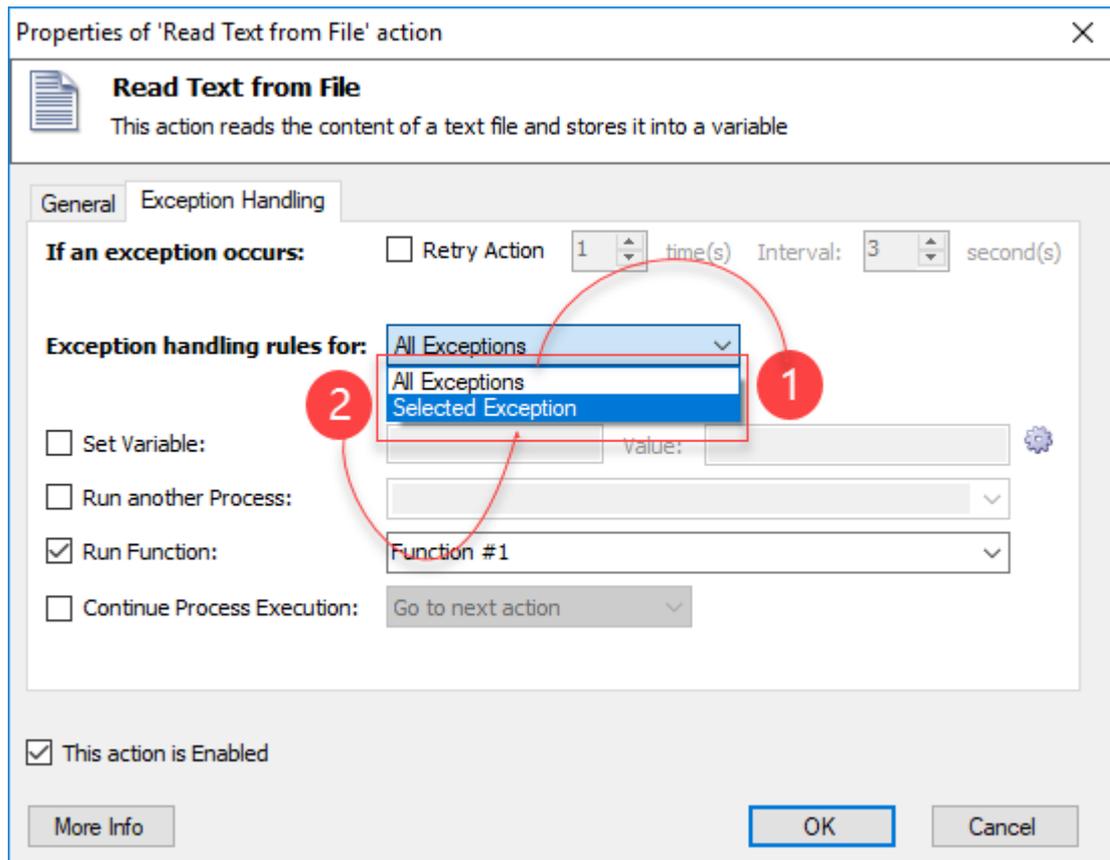
The Action could run a different Process and/or continue. If you choose to continue, you can then tell the Action to go to the next action, a specific Label or Function (set to create a certain response to the exception), to set a Variable, that can give you a record of the exception for later Actions or even repeat that action .

Once you have set a behavior to one specific exception, you can click on another one to select a different behavior. Thus, the Action could handle three different exceptions in three different ways after retrying X times.

It is very important, that you can select from a variety of options in the exception handling. First, you can set another value to a variable and retry the action using the new values of the variable. One more option that the user has, is to run another Process upon the exception. For example, a Process could be triggered that would send an email to a contact about the Processes' failure.

Moreover, the user can set the Process to run a function after the exception occurs in order to complete part of the task or to notify an interested party. Also, the user can set the Process to proceed with the execution from another point of the Process.

Please note, that you can do the above for 'All Exceptions' [2] or set them in any way you want according to a 'Selected Exception' [3] thrown by this action.



The Exception Handling Tab of an Action for All Exceptions

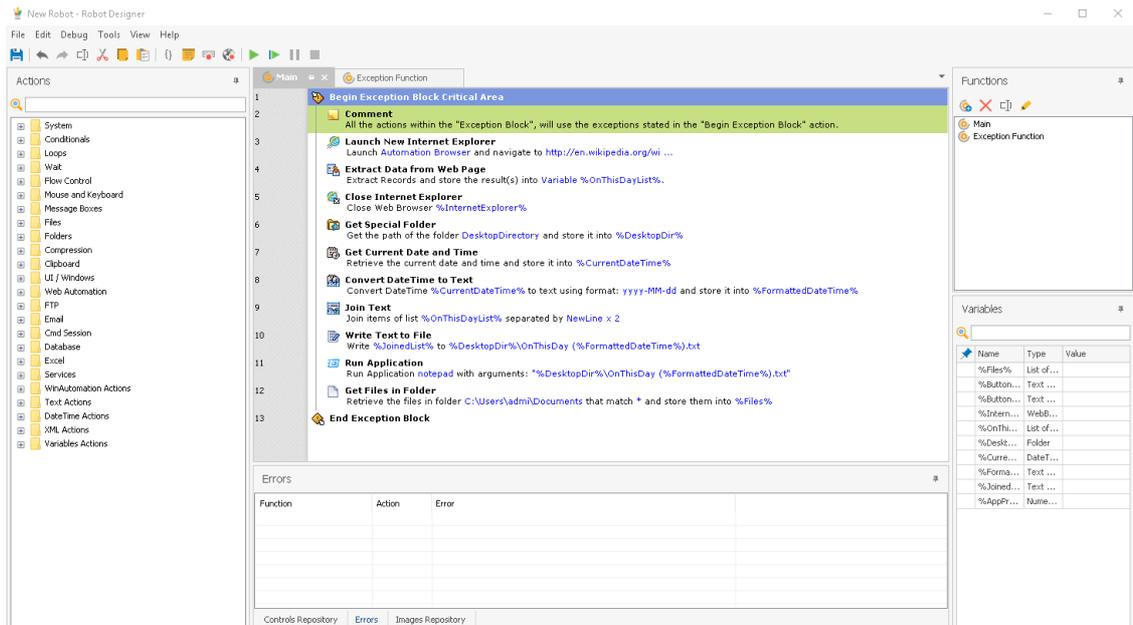
Notice that all the exceptions that have some custom Exception handling rules assigned, are displayed in bold (as you can see in the first screen-shot above [1]). Exceptions that will immediately cause the Process to fail (the default behavior) are displayed in normal text.

Exception Handling is different from the [Error Handling Tab](#) [139] in the Console as Error Handling is what ProcessRobot should do in the event that a Process failed, while Exception Handling is what to do when something unexpected occurs in a specific action so that the Process does not fail.

3.10.11 Exception Block

As mentioned previously, the [Exception Handling](#) [502], is being used when a Process finds itself in an unexpected situation. It is also known as a Run-Time Error.

The exception handling, can be placed either for an action, or for a group of actions that are contained between the "Begin Exception Block" and "End Exception Block" actions.



Block exception

As in the single action's exception handling, the "Exception Block" provide the users with three responses for the robot to have upon an exception.

First, upon an exception, the robot is able to set a value to a variable. This for example could be prior to sending an email and the variable could state that the execution failed.

Secondly, in case an exception occurs within the block, another function could be triggered.

Lastly, the "Exception Block" has the option to continue the robot execution, either with the next action, the first action after the block, repeating the action, or even repeating the whole block execution.

In the above example, in case of an exception in any of the actions, the "Exception Function" is going to be triggered.

Properties of 'Begin Exception Block' action

Begin Exception Block
This action sets the beginning of an exception handling region.

General

Action Input

Name:

Set Variable:

Run Function:

Continue Execution:

Action Output

(This action does not provide any output)

This action is Enabled

Please keep in mind the fact, that in case an action that is contained in an "Exception Block" already has an exception handling rule set and fails, both rules will apply, first the actions' and second the exceptions'.

3.10.12 Image Recognition

There are three actions based on Image Recognition: "Wait for Image", "If Image" and Move Mouse to Image. Both of these Actions require the computer to recognize an image on the screen.

How the Image Recognition works?

Image Recognition works by checking one pixel (in your image) to another (starting in the top left corner and working across then down the screen). If that pixel matches, it checks the second. If the first seven match, but the eighth doesn't, it checks the next pixel against the first in its image.

To use Image Recognition, first you take a screenshot of the part of the screen that you want. Once that image is stored, when the Process gets to that Action, it will start working on finding the stored image. Move Mouse to Image is useful if an image (like a button you need to click) could move.

Since Image Recognition works by comparing one pixel to another, if your saved image and most of the screen has a certain color in the upper left corner, it can take some time to find the right

image. To make this more efficient, start with a less-common color in the upper left corner. For example, with a white background for most of the screen, this image



will take less time to find than this image.



Both will find the image, but the first will be quicker. The less of a color above and to the left of your image's first pixel color, the faster the Action will work.

NOTE: Image recognition actions is unfortunately not applicable across different screen resolutions. This is why its use, is suggested as a last resort, while at the same time taking under consideration that the script will run on a certain screen analysis. For, example should you wish to deploy a Process to a Solobot, it is suggested that you refrain from using Image recognition actions, unless you are 100% sure that the resolution will be the same across the target machine that will run the application.

3.10.13 Interactive/Non Interactive Processes

Any time you use the keyboard or mouse, you are creating input for your computer; and any time something shows on the monitor, your computer is showing output. These are called interactive functions because the computer interacts with you.

Non-interactive functions are those that don't require input (keyboard & mouse) or create output on your monitor. For example, if your computer is set up to automatically connect to the Internet on start-up, this is non-interactive as you don't have to do anything, and nothing shows up on the monitor. Likewise, computers will check email, install updates, move files, etc - all without any action or knowledge on your part.

Since ProcessRobot can make changes automatically, without input or output, a Process can be interactive (needing keystrokes or mouse events with or without Macros) or non-interactive depending on the actions it contains.

As interactive functions can only work when a user is logged in, only non-interactive Processes can be run when all users are logged out.

3.10.14 Valid Key Codes for the Send Keys Action

Consider the following scenario: You need to increase the font-size in a Word document by pressing **CTRL + SHIFT + '.'**, using the Send Keys Action.

In this case, you would want to simulate the dot key press, which is not the same as sending the dot character. To simulate key-presses inside a Send Keys Action, use the following notation:

{KeyCode}

where KeyCode is the capital form of the letter itself for letters A-Z, OemSemicolon for ';', Oemplus for '+', Oemcomma for ',', OemMinus for '-', OemPeriod for '.', OemQuestion for '?', Oemtilde for '~', OemOpenBrackets for '[', OemPipe for '|', etc. You will find the full list of the valid Key Codes at the bottom of this help topic.

To return back to the Word example, the font-size increase should be written as:

{Control}({Shift}({OemPeriod}))

To produce a colon ':' key press, you may use

{Shift}({OemSemicolon})

The same rule applies for other characters produced by pressing Shift + another key.

Full List of Valid Key Codes

LButton	BrowserS	D0	Add
RButton	earch	D1	Separator
Cancel	BrowserF	D2	Subtract
MButton	avorites	D3	Decimal
XButton	BrowserH	D4	Divide
1	ome	D5	F1
XButton	VolumeMu	D6	F2
2	te	D7	F3
Back	VolumeDo	D8	F4
Tab	wn	D9	F5
LineFeed	VolumeUp	A	F6
MediaNext	Track	B	F7
Clear	MediaPrev	C	F8
Enter	iousTrack	D	F9
Return	MediaStop	E	F10
ShiftKey	MediaPlay	F	F11
y	Pause		
	LaunchMai		
	l		

Control Key	SelectMedia	G	F12
		H	F13
Menu	LaunchApplication1	I	F14
Pause		J	F15
CapsLock	LaunchApplication2	K	F16
Capital	OemSemicolon	L	F17
		M	F18
Hangul Mode	Oem1	N	F19
	Oemplus	O	F20
Hangul Mode	Oemcomma	P	F21
KanaMode	OemMinus	Q	F22
		R	F23
JunjaMode	OemPeriod	S	F24
		T	NumLock
FinalMode	Oem2	U	Scroll
KanjiMode	OemQuestion	V	LShiftKey
HanjaMode	Oem3	W	RShiftKey
	Oemtilde	X	LControlKey
Escape	Oem4	Y	RControlKey
IMEConvert	OemOpenBrackets	Z	LMenu
		LWin	RMenu
IMENonconvert	OemPipe	RWin	BrowserBack
IMEAccept	Oem5	Apps	BrowserForward
	OemCloseBrackets	Sleep	BrowserRefresh
IMEAccept	Oem6	NumPad0	BrowserStop
IMEChange	OemQuotes	NumPad1	
		NumPad2	
Space	Oem7	NumPad3	
Prior	Oem8	NumPad4	
PageUp	Oem102	NumPad5	

PageDoOemBack	NumPad6
wn	slash
	NumPad7
Next	ProcessKey
	NumPad8
End	Packet
	NumPad9
Home	Attn
	Multiply
Left	Crsel
Up	Exsel
Right	EraseEof
Down	Play
Select	Zoom
Print	NoName
Execute	Pa1
Snapsh	OemClear
ot	
PrintScr	KeyCode
een	Shift
Insert	Control
Delete	Alt
Help	Modifiers

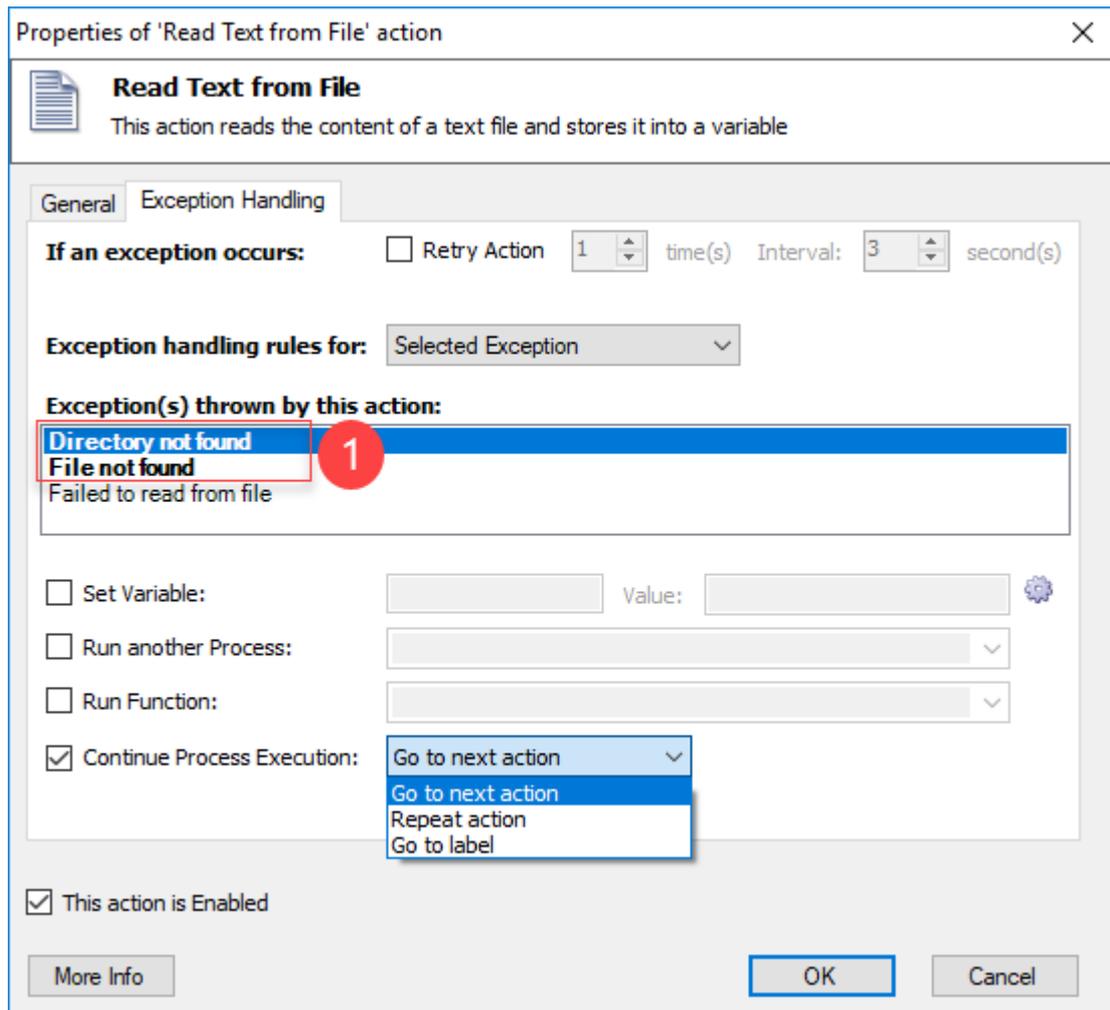
3.10.15 Error-Exception Handling Levels

Error and Exception Handling in Process Robot is one of the most advanced in the market. It consists of several levels as they are mentioned below from lowest to highest:

1st Level

The first level refers to the exception Handling that the user can set in a specific action. For every action that you believe based on your tests that is most likely to fail, you can set your exceptions here.

[Action's Exception Handling](#) ⁵⁰²



Action: 1st Level Of Exception Handling

2nd Level

The second level refers to the exception Handling that the user can set in a block of actions within a Process. In case any action within the block throws an exception it can be handled!

[Exception Block](#)⁵⁰⁴

Properties of 'Begin Exception Block' action

Begin Exception Block
This action sets the beginning of an exception handling region.

General

Action Input

Name: ⓘ

Set Variable: ⓘ

Run Function: ⓘ

Continue Execution: ⓘ

Action Output

(This action does not provide any output)

This action is Enabled

Block: 2nd Level Of Exception Handling

3rd Level

The third level refers to the Process as a whole. This means that if any action within the Process fails (even if there is exception handling in its actions or no), then in the Process Error Handling in the Process's Properties you can specify the behavior you wish.

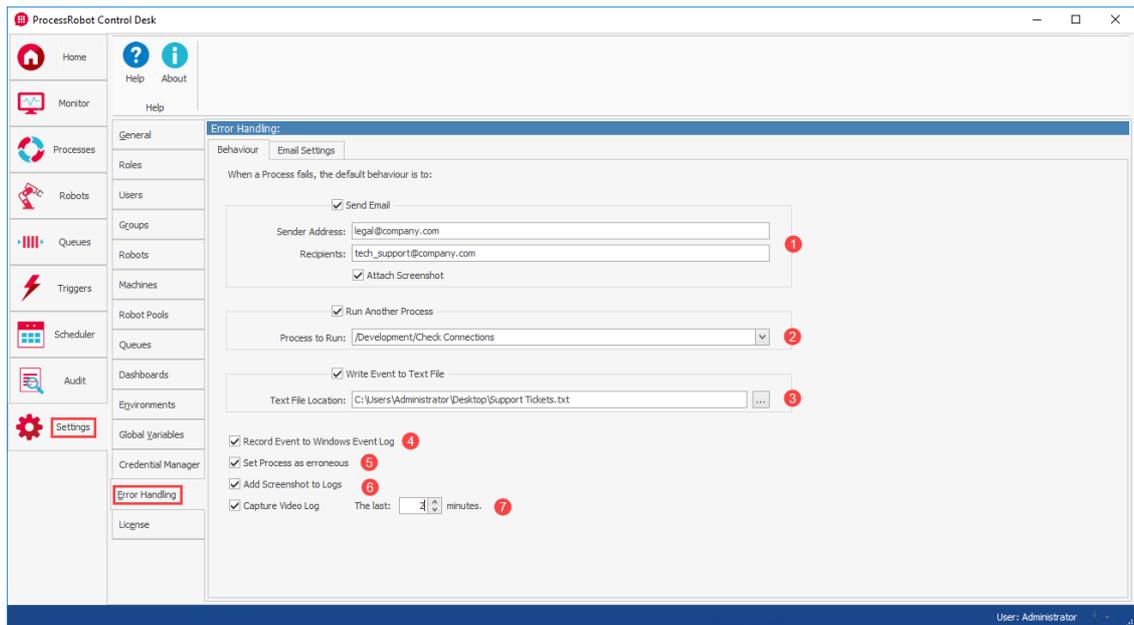
[Process Error Handling](#) ¹³⁹

Process: 3rd Level Of Exception Handling

4th Level

The fourth level refers to All the Processes that run in ProcessRobot. If any Robot fails for any reason, then in the "Control Desk > Settings > Error handling" you can specify the behavior you wish.

[Global Error handling](#) ²⁴⁸



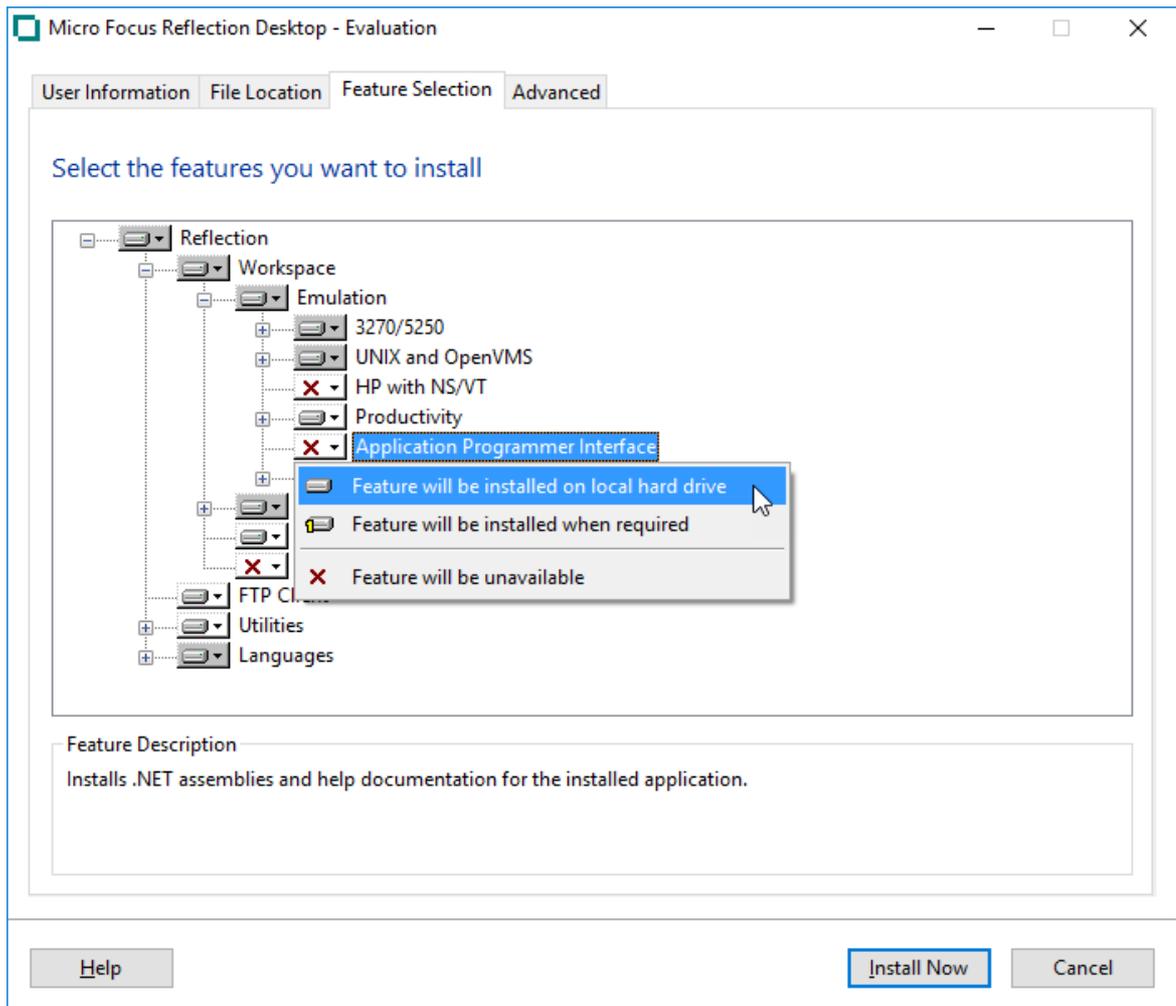
Global: 4th Level Of Exception Handling

3.10.16 Terminal Emulation - Guide

You can choose between two Terminal Emulator providers: Micro Focus Reflection and HLLAPI.

- Micro Focus Reflection

Download Micro Focus Reflection (rdesktop) and run *setup.exe*. During the installation, remember to check the API options that are needed for ProcessRobot actions to work.



Note: Supporting versions of Reflection are 2011, 2014 and 2016

- HLLAPI (theoretically supporting any provider)

Install one or more terminal emulation software (MicroFocus RUMBA, IBM Personal Communications, Cybele zScope, MicroFocus Reflection, RocketSoftware BlueZone).

Open the installed software. Default executable paths:

RUMBA: *C:\Program Files (x86)\Micro Focus\RUMBA\System\RumbaPage.exe*

IBM PC: *C:\Program Files (x86)\IBM\Personal Communications\pcsws.exe*

zScope: *C:\Program Files (x86)\zScope\Classic v6.5\zClassic.exe*

Reflection: *C:\Program Files (x86)\Micro Focus\Reflection\Attachmate.Emulation.Frame.exe.*

BlueZone: *C:\Program Files\BlueZone\7.1\bzsm.exe*

Terminal Emulation Actions can be used ONLY after the Terminal Emulation software has been launched and the connection has been set.

Important Note: Remember to use Close Terminal Session to terminate the connection to the terminal emulation software. For some providers, you won't be able to connect to an already open session if you fail to close the connection and you will need to restart (the software or the connection to the host).

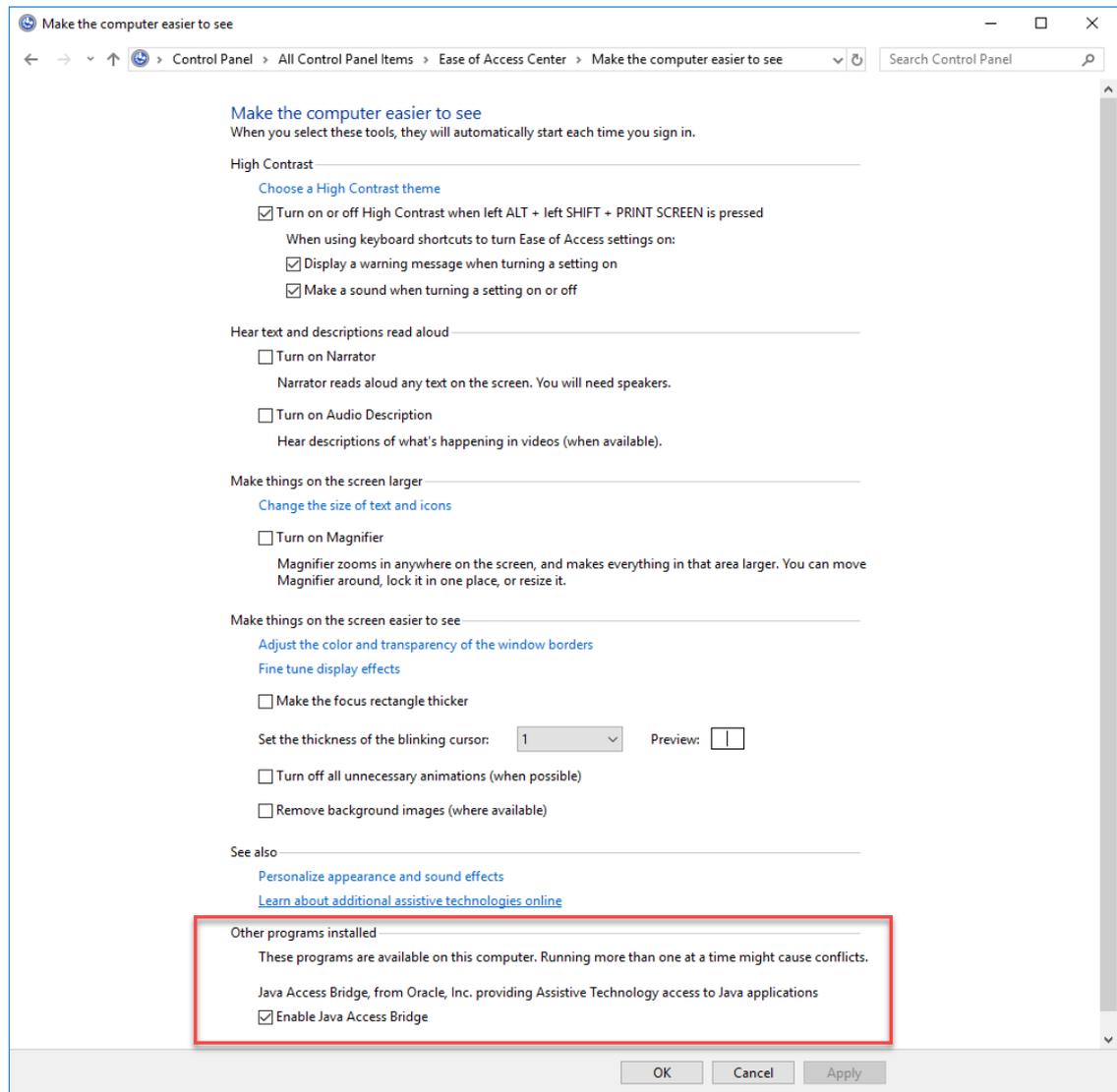
3.10.17 Interaction with JAVA-based applications

In case you are trying to automate the use of a Java - based application, there are some particular settings, that you need to make sure, they are in place.

First, you need to make sure that Java is installed on the machine. Please note, that Process Robot, can interact with Java applications on machines with Java 6 or higher.

Please note, that the bit rate of your Java installation, needs to be the same as the bit rate that Process Robot runs.

Moreover, please navigate to the "Control Panel > Ease of access Center > Make the computer easier to see" and make sure that the option "Enable Java Access Bridge" under "Other programs installed" is ticked as shown in the screen-shot below.

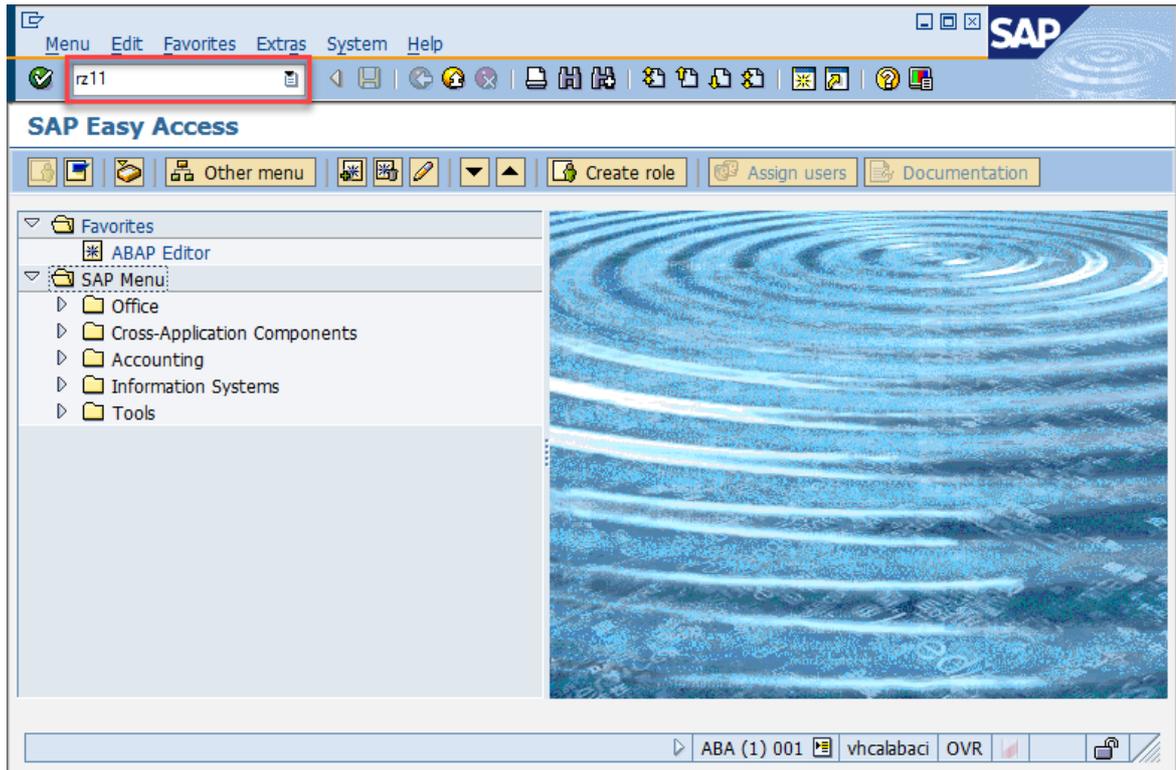


NOTE: Please make sure that you are running both the Java-based application and ProcessRobot under the same user (e.g. if the application runs under a simple user, ProcessRobot should not run under an Administrator).

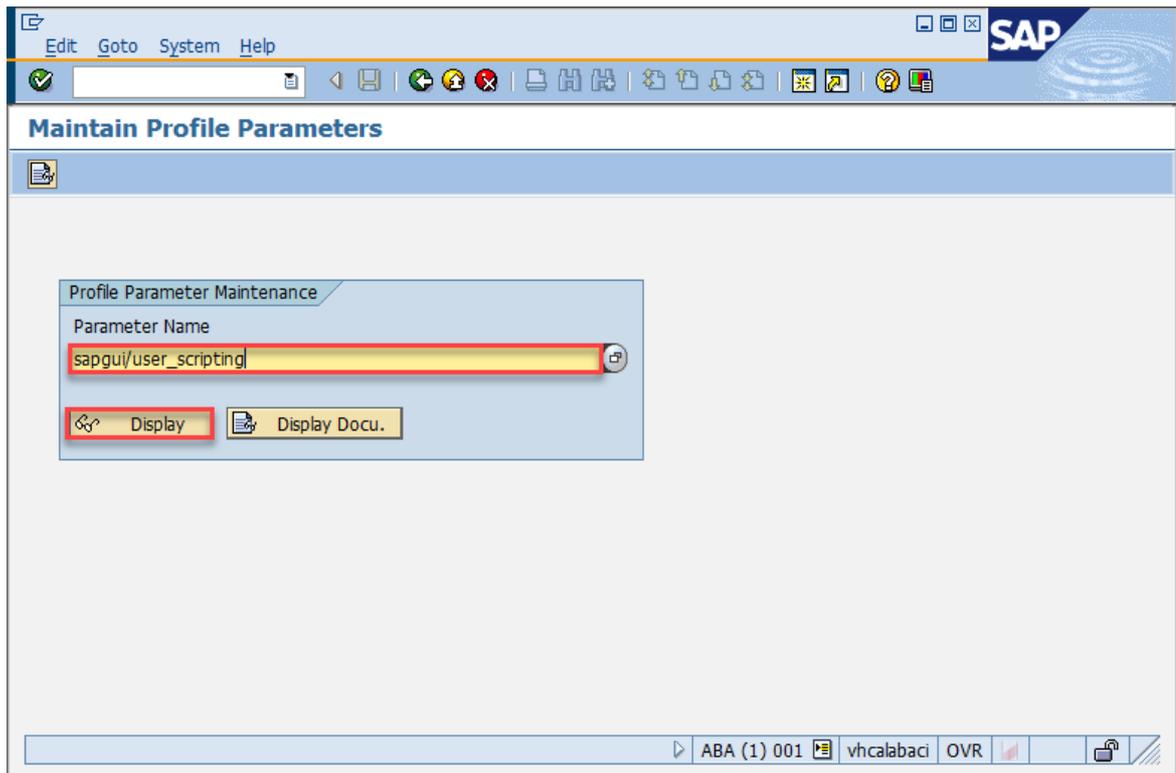
3.10.18 Interaction with SAP

ProcessRobot supports interaction with SAP. For ProcessRobot to be able to select controls within the SAP interface, certain configurations are required within SAP (the following assumes that SAP is fully up to date with the latest support packages installed):

Login to SAP server and enter transaction RZ11.



Specify parameter name as `sapgui/user_scripting` and press "Enter" or click on the "Display" option.



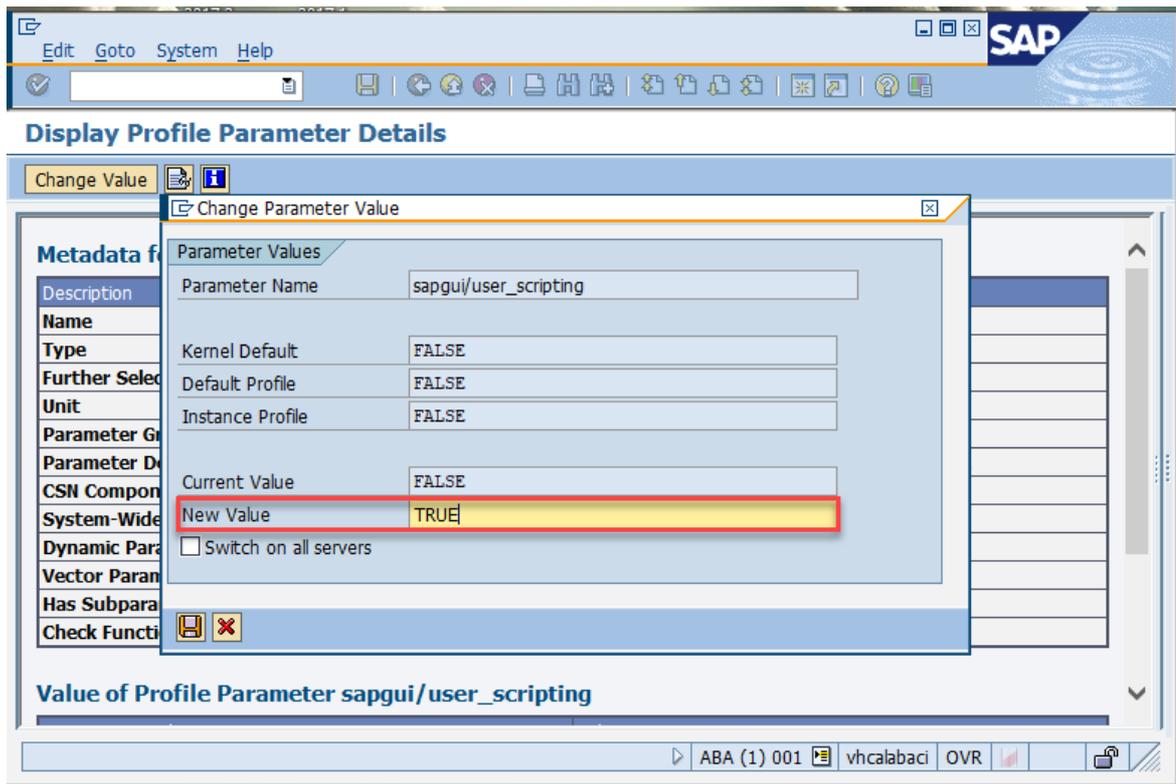
Select the "Change Value" option.

The screenshot shows the SAP 'Display Profile Parameter Details' window. The title bar includes 'Edit Goto System Help' and the SAP logo. Below the title bar is a toolbar with various icons. The main content area is titled 'Display Profile Parameter Details' and contains a 'Change Value' button. Below this is a section titled 'Metadata for Parameter sapgui/user_scripting' which contains a table with the following data:

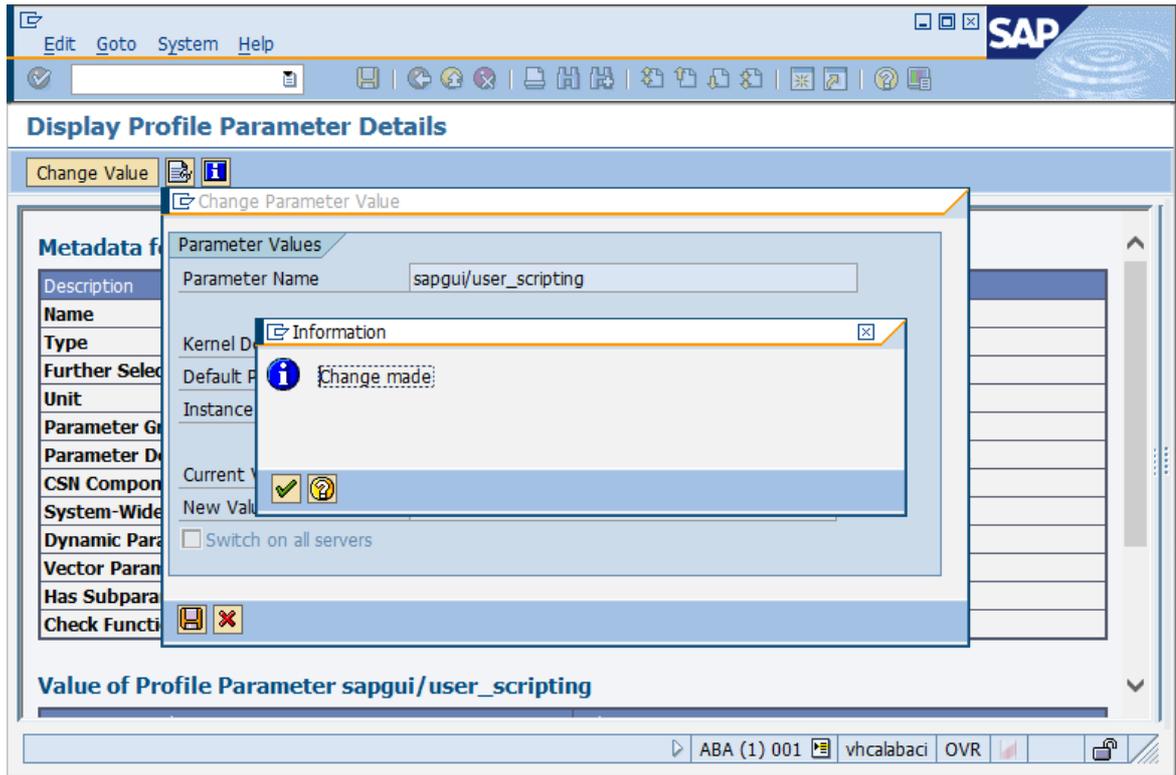
Description	Value
Name	sapgui/user_scripting
Type	Logical Expression
Further Selection Criteria	
Unit	
Parameter Group	Gui
Parameter Description	Enable or disable user scripting on the frontend.
CSN Component	BC-ABA-SC
System-Wide Parameter	No
Dynamic Parameter	Yes
Vector Parameter	No
Has Subparameters	No
Check Function Exists	No

Below the table is a section titled 'Value of Profile Parameter sapgui/user_scripting' which is currently empty. The status bar at the bottom shows 'ABA (1) 001' and 'vhcalabaci OVR'.

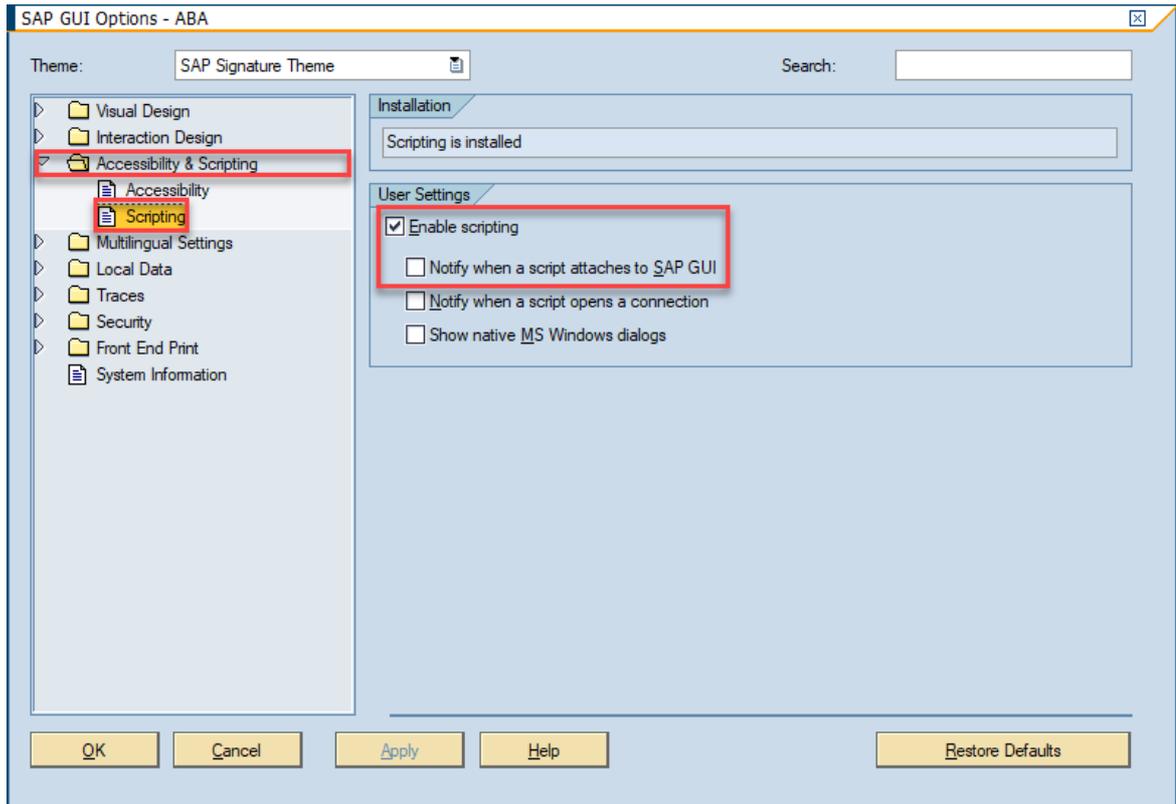
Change the value for sapgui/user_scripting to TRUE and click on Save.



Now the "Server" part alterations have been completed.



On the client side, open SAP GUI Options, navigate under the Accessibility & Scripting tab and Select "Scripting". In this option, make sure the "Enable scripting" option is checked, and that "Notify when a script attaches to SAP GUI" is unchecked.



Confirm with the SAP Administrator that S_SCR (GUI scripting) authorization is granted for all required users.

3.10.19 CaptureFast Actions

CaptureFast is a Cloud-Based Document and Data Capture Application that helps Users extract data from physical or digital documents. Supported files are PDF, TIFF, PNG and JPG.

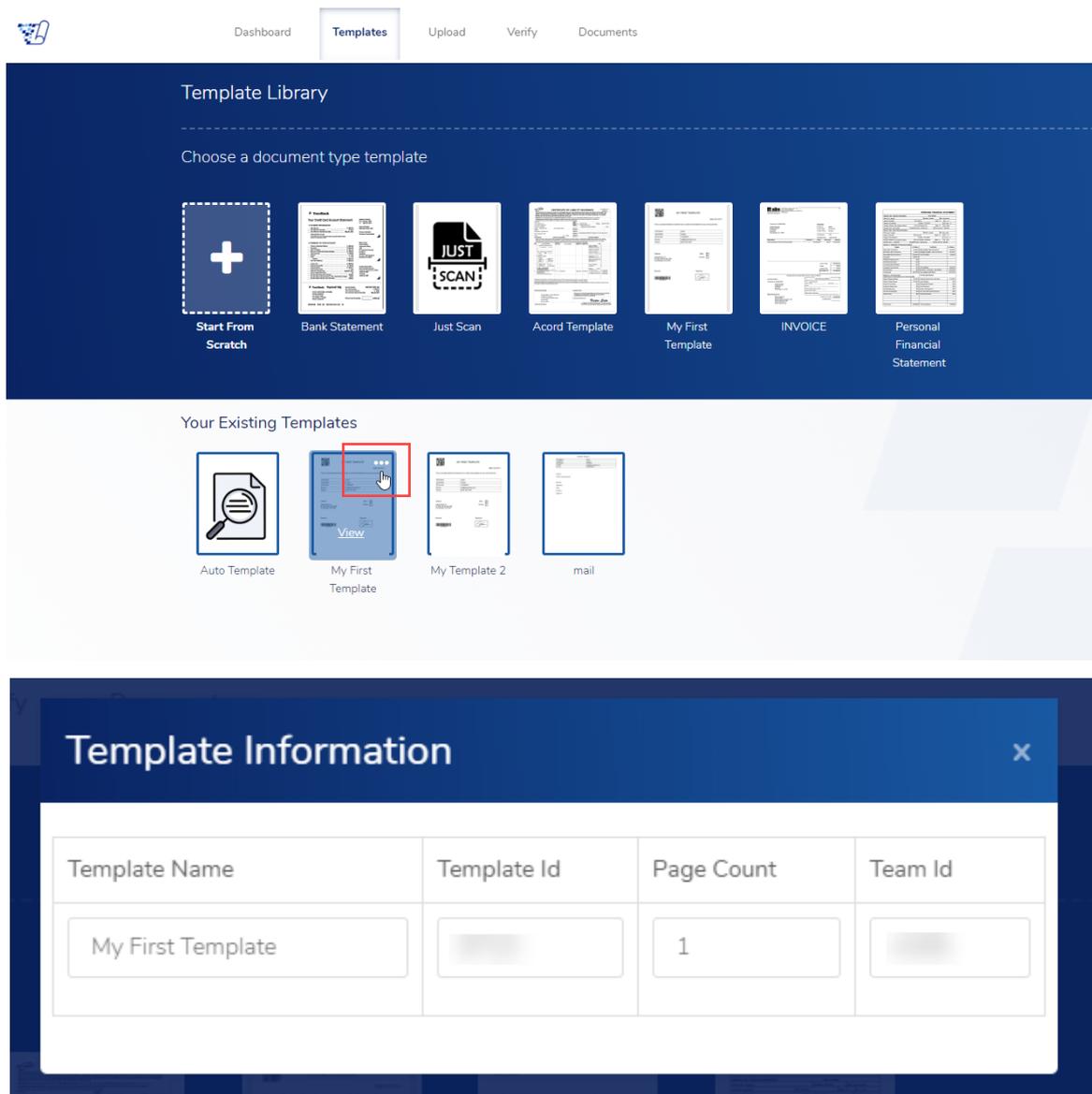
In the CaptureFast web platform, the below steps need to be followed first in order to use the "CaptureFast" group of Actions:

1. Create an account. Note that when creating an account, a Team Id is also created.
2. Create or select an existing document's template. The "no verification" option should be checked, in order to allow Process automation. Otherwise, the User will be asked to verify accuracy of information that was captured.

The screenshot shows the 'Create a New Template' interface in the Process Robot application. At the top, there is a navigation bar with 'Dashboard', 'Templates' (highlighted), 'Upload', 'Verify', and 'Documents'. Below this is a dark blue header with 'Create a New Template' on the left and 'Template Settings' and 'Fields' on the right. The main content area has a section titled 'Name your template' with the text 'My First Template' below it. A red box highlights the 'Advanced options' section, which contains a toggle switch for 'No verification'. Below this is a link that says 'Click here to describe input channels'. At the bottom, there is a preview of a document page labeled 'Page #1'.

3. Specify which fields of the document to capture.

4. Navigate to the "Template Information" window by clicking the ellipsis at the top right corner of an existing template and then selecting "Template Info". Note the Template Id along with the Team Id.



Dashboard Templates Upload Verify Documents

Template Library

Choose a document type template

- Start From Scratch
- Bank Statement
- Just Scan
- Acord Template
- My First Template
- INVOICE
- Personal Financial Statement

Your Existing Templates

- Auto Template
- My First Template
- My Template 2
- mail

Template Information

Template Name	Template Id	Page Count	Team Id
My First Template		1	

Once these steps have been completed, the User can connect to the account and upload a document by using the respective CaptureFast Actions. Note that in the "Upload Document to CaptureFast" Action the User must define the Template and Team Ids mentioned in the fourth step.

Finally, the "Get Results from CaptureFast Document" Action returns all the results in JSON format.

3.10.20 Integration with CyberArk

CyberArk offers an online Privileged Access Management platform in order to provide an Information Security solution to Users.

The CyberArk platform can be accessed through the "Get Password From CyberArk" Action.

Properties of 'Get Password From CyberArk' action

Get Password From CyberArk
This action retrieves a password for a specific application from CyberArk

General | Advanced | Exception Handling

Action Input

Server Address: 1

Application Id: 2

Safe: 3

Folder: 4

Object: 5

Extra data: 6

Certificate Location: Don't use Certificate

Action Output

Cyberark Password: %CyberArkPassword%

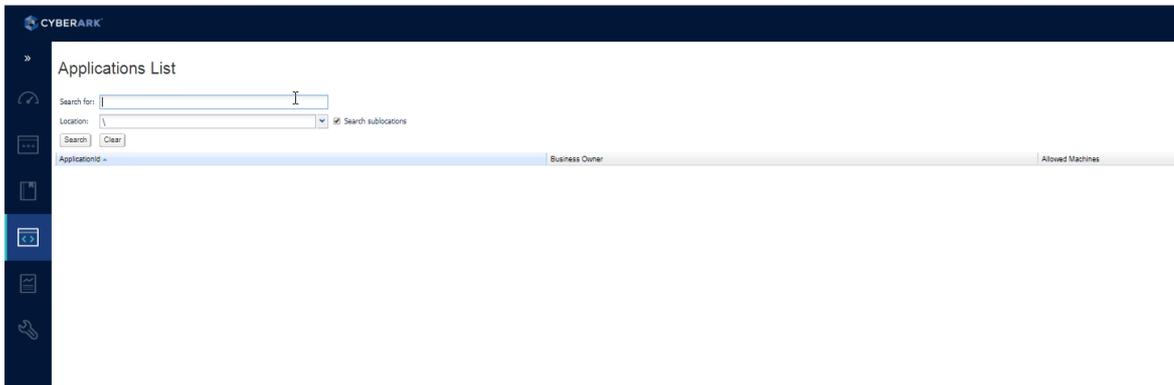
Store Response into: %JSONResponse%

This action is Enabled

More Info OK Cancel

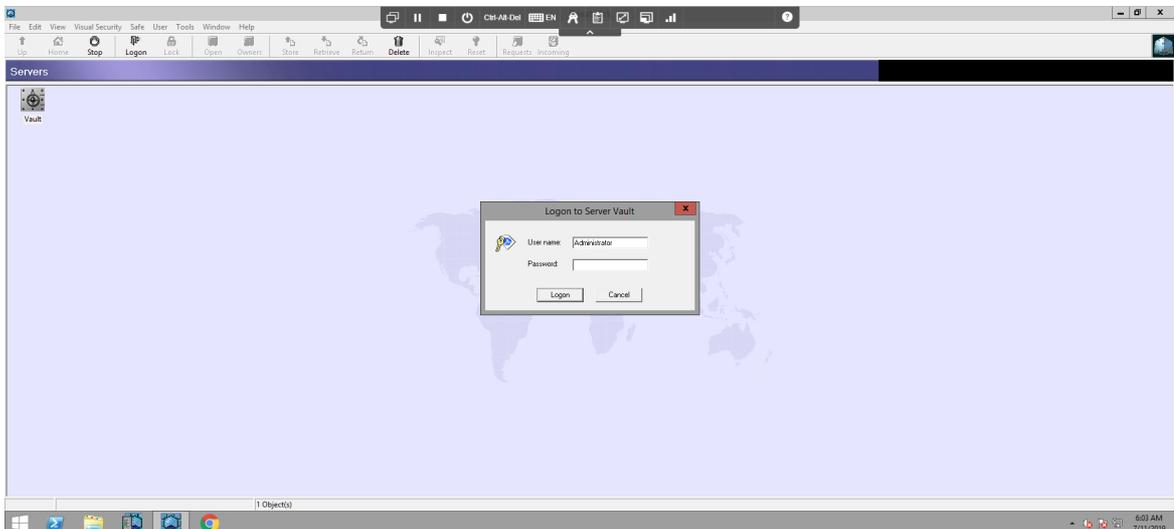
Once the necessary configurations are complete in the CyberArk platform, the "Get Password From CyberArk" Action can be used; information for the required fields (1-6 as labeled above) can be found as follows in the CyberArk platform:

1. Populate the "server Address" field (labeled 1 in the Action Properties screenshot above) with the address of the server where you have installed the CyberArk Central Credential Provider.
2. In case you do not remember the correct Application ID, open the CyberArk Password Vault on a web browser, login, and access the "Applications" tab:



Here, you may search in the Applications List to locate your Application ID, and enter it in the respective field (labeled 2).

3. Open the "PrivateArk" application and login to your Vault.



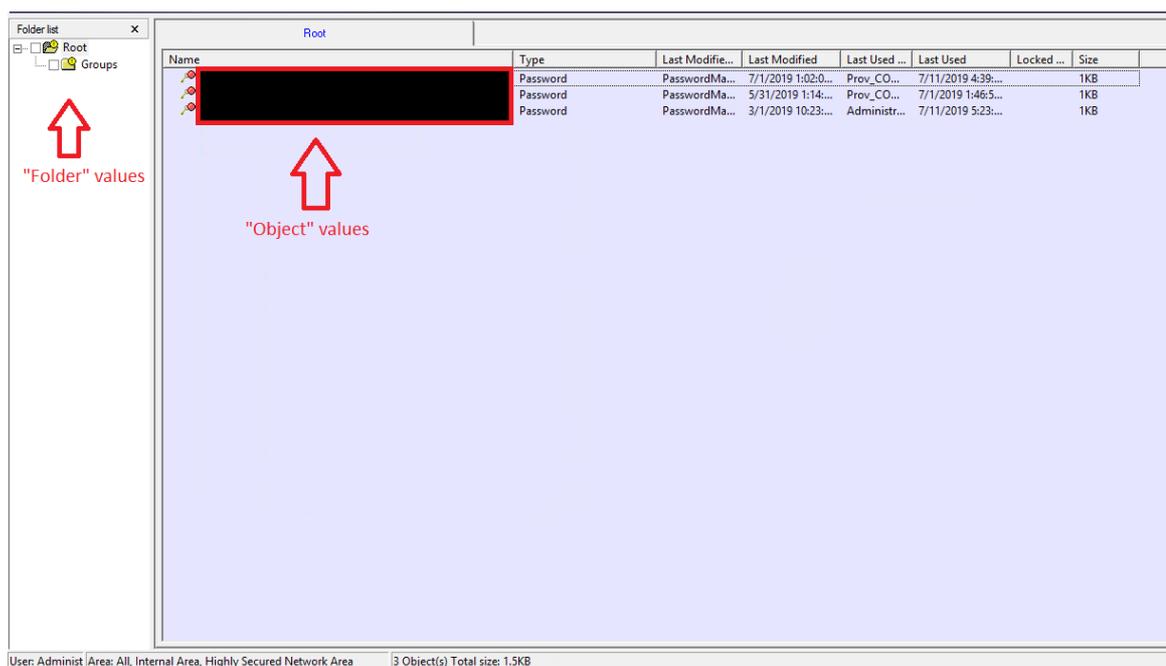
4. The Action's "Safe" field (labeled 3) should be populated with the name of the Safe selected in the following screen:

Vault						
Name	State	Last Used	Authorizations	Expiration Da...	Current S...	
	Closed	7/3/2019 7:08:3...	List, Retrieve,...		11KB	
	Closed	6/30/2019 12:44...	All		6KB	
	Closed	7/3/2019 8:38:1...	All		111KB	
	Closed	7/3/2019 7:11:2...	All		35KB	
	Closed	7/3/2019 8:13:0...	All		4KB	
	Closed	7/3/2019 7:11:2...	All		4KB	
	Closed	7/3/2019 7:11:2...	All		501KB	
	Closed	7/3/2019 8:38:3...	All		7.81MB	
	Closed	6/30/2019 12:44...	All		4KB	
	Closed	6/30/2019 12:44...	All		4KB	
	Closed	6/30/2019 12:44...	All		0KB	
	Closed	7/3/2019 8:29:4...	All		109KB	
	Closed	7/3/2019 6:09:5...	All		0KB	
	Closed	6/30/2019 12:44...	All		0KB	
	Closed	7/3/2019 8:38:5...	All		0KB	
	Closed	6/30/2019 1:53:...	All		3KB	
	Closed	6/30/2019 12:44...	All		0KB	
	Closed	6/30/2019 12:44...	All		0KB	
	Closed	7/3/2019 8:13:1...	All		5KB	
	Closed	6/30/2019 12:44...	All		12KB	
	Closed	6/30/2019 1:45:...	All		0KB	
	Closed	7/3/2019 8:08:5...	All		27KB	
	Closed	6/30/2019 12:44...	All		0KB	
	Closed	7/3/2019 6:09:4...	All		15KB	


 "Safe" values

User: Administ Area: All, Internal Area, Highly Secured Network Area 24 Object(s) Total size: 8.64MB

5. Double-click on the Safe you wish to access. The Action's "Folder" field (labeled 4) should be populated with the name of the folder on the left-hand side of the screen (in this case, "root"), while the "Object" field (labeled 5) should be populated with a name from the main list:



In case you have configured your credentials in a way that requires the use of Extra Data, please enter it in the respective field (labeled 6).

Finally, in case a Certificate is required to retrieve your Credentials, specify it in the "Certificate Location" field.

Using this information, the "Get Password From CyberArk" Action can be configured to retrieve the designated credentials.

3.11 Actions Reference

3.11.1 System

3.11.1.1 Run Application Action

Description:

This action executes an application or opens a document by executing the associated application

Properties of 'Run Application' action

Run Application
This action executes an application or opens a document by executing the associated application

General | Exception Handling

Action Input

Application Path: ⓘ ⚙️ 📄

Command Line Arguments: ⓘ ⚙️

Working Folder: ⓘ ⚙️ 📁

Window Style: ⓘ

After Application Launch: ⓘ

Do not wait for more than seconds ⓘ

Action Output

Store Process Id into: ⓘ

Store Main Window Handle into: ⓘ

This action is Enabled

More Info OK Cancel

Properties:

Application Path:

Insert executable file here, as a complete file path.

Command Line Arguments:

Add extra arguments that would go after the executable file name. For example, you could enter notepad.exe in the Application Path, and a specific text file in the Command Line Arguments.

Working Folder:

Enter the full path of the folder to work out of, if applicable.

Window Style:

Choose the appearance and size of your application Window when it opens.

After Application Launch:

Choose whether the next action execute immediately, or wait until the program loads or completes

Do not wait more than:

Determines whether there is a maximum wait time, and how long before forcing a continue.

Store Process Id into:

Enter a name to be the variable that will store the Process ID output. This will be a numeric value.

Store Main Window Handle into:

Enter a name to be the variable that will store the Window Handle. When opening a new Window, this will catch the value of the Window Handle, and store it in this variable. A Window Handle is useful to specifically identify a Window in a later action.

Store Exit Code into:

Enter a name to be the variable that will store the Application Exit Code as a numeric value.

Other uses:**Display the contents of a folder:**

You can use this action to display a folder inside Windows Explorer. This can be useful if, for example, the Process has just unzipped some files and needs to display them to the user. Simply set "Application path" to "explorer" and "Command Line Arguments" to the folder. (e.g., C:\unzipped).

3.11.1.2 Run DOS Command Action**Description:**

This action executes a DOS command or a console application in invisible mode, waits for the command or application to complete and retrieves its output into a text variable

Properties:

DOS Command or Application:

Enter the name of DOS command or a console application, with arguments if applicable.

Working Folder:

Enter the full path of the folder to work out of, if applicable.

Store Output into:

Enter a name to be the variable that will store the Command Output. The Command Output will be the text output from the DOS Command or Application.

Store Error Output into:

Enter a name to be the variable that will store the Error Output. The Error Output will be the text describing the errors occurred (if any) during the execution of the DOS Command or Application.

Store Exit Code into:

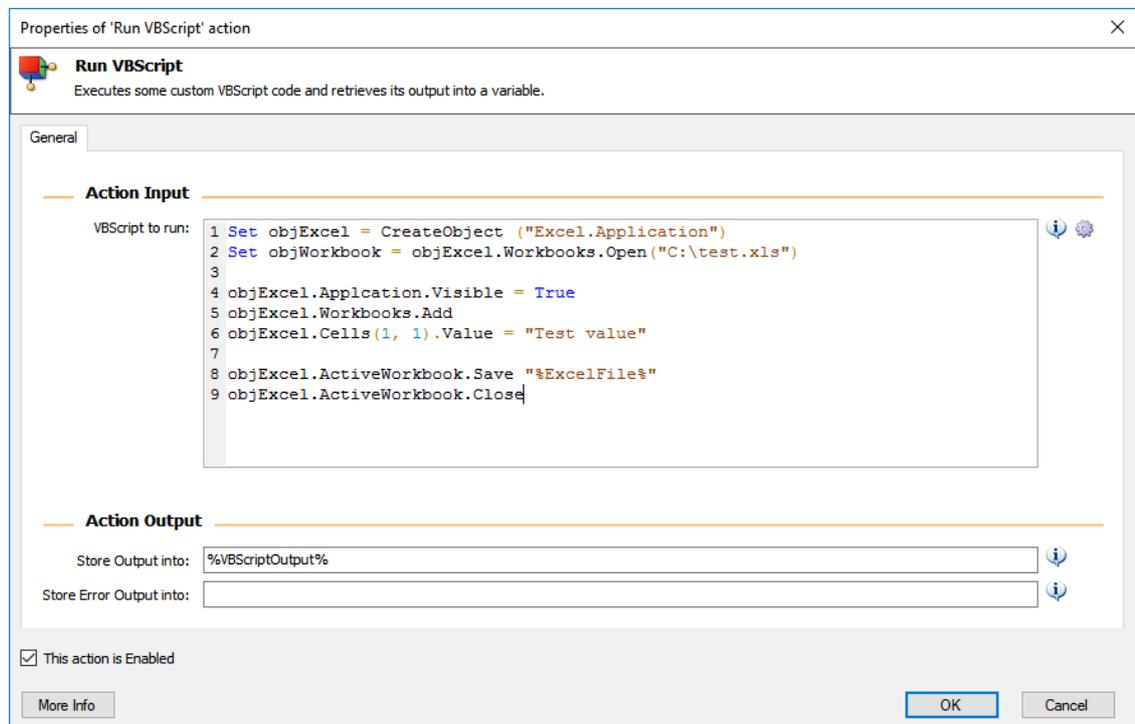
Enter a name to be the variable that will store the Command or Application Exit Code. This will be a numeric value.

3.11.1.3 Run VBScript Action

Description:

Executes some custom VBScript code and retrieves its output into a variable.

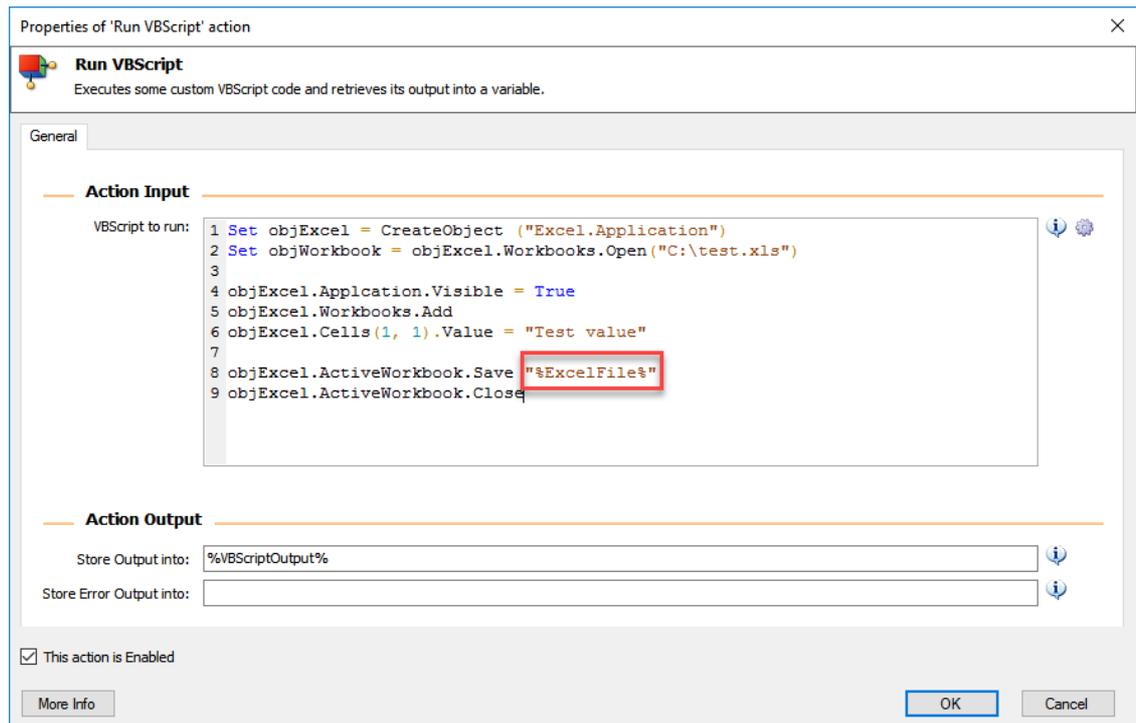
Although ProcessRobot offers a vast number of actions that will cover almost all your automating needs there may be times that you want to inject some VBScript code into your Process. Maybe you already have a script in VBScript that you want to use, maybe you need some really advanced features such as manipulating a COM object. Whatever the need may be this is the action you can use to include VBScript code into your Process.



Properties:

VBScript to run:

Enter here the VBScript code you want to execute. TIP: Within the script, you may include ProcessRobot variables, since they will be evaluated prior to the VBScript code's execution. For example:



If the %ExcelFile% variable holds the value, say, "c:\myexcel.xlsx", the last command in line 8 will equal to

```
objExcel.ActiveWorkBook.Save "%ExcelFile%"
```

Store Output into:

Enter the name for the variable to hold the script's output. A script may write some text to the windows console during its execution. In this case, all the text written by the script is stored into this variable so that it can be processed by later actions.

Store Error Output into:

Enter the name for the variable to hold any errors that may occur during the execution of the VBScript code.

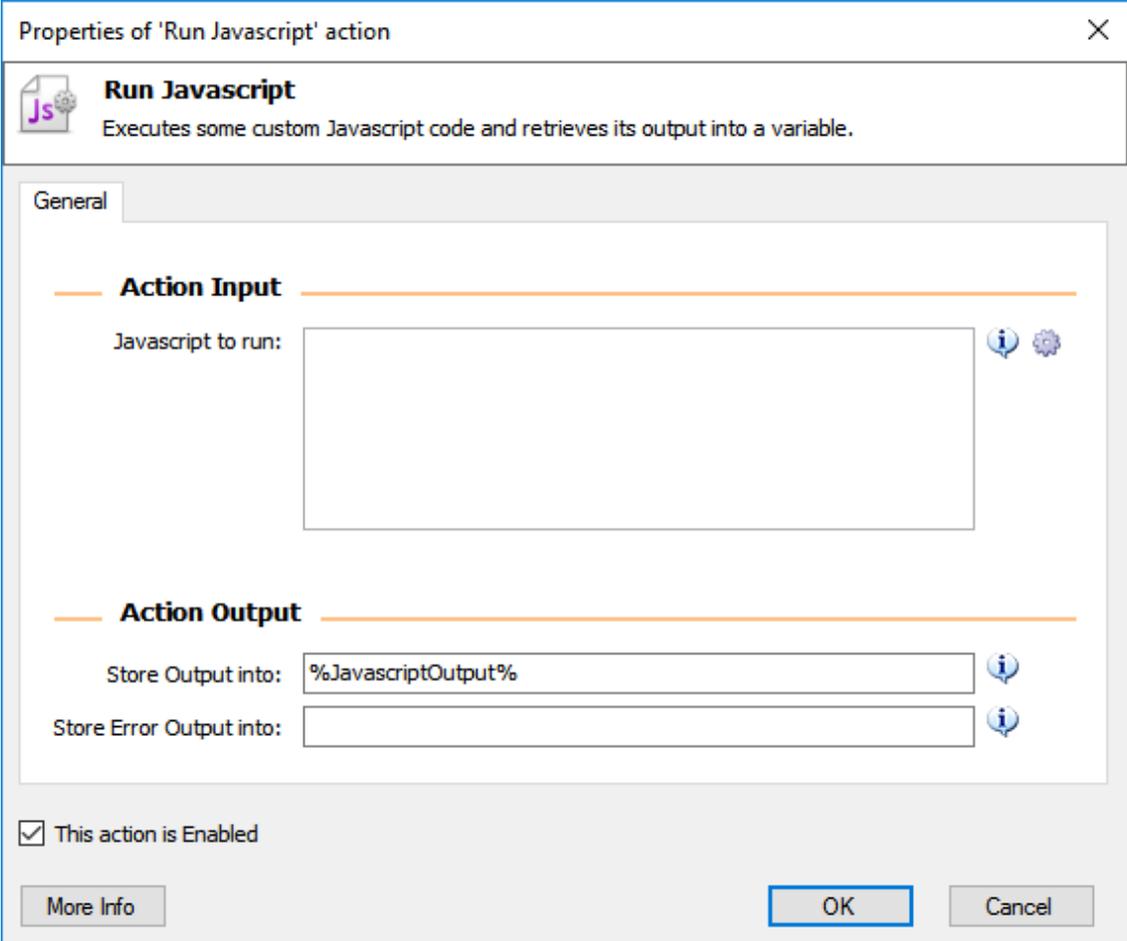
NOTE:

Commands in VbScript will be highlighted accordingly so that you can read your script easily. Also the this action's window can be resized.

3.11.1.4 Run Javascript Action

Description:

Executes some custom Javascript code and retrieves its output into a variable.



The screenshot shows the 'Properties of 'Run Javascript' action' dialog box. The title bar reads 'Properties of 'Run Javascript' action'. The main area has a header with a 'Js' icon and the text 'Run Javascript' and 'Executes some custom Javascript code and retrieves its output into a variable.'. Below this is a 'General' tab. Under 'Action Input', there is a label 'Javascript to run:' followed by a large empty text area. Under 'Action Output', there are two labels: 'Store Output into:' with a text box containing '%JavascriptOutput%' and 'Store Error Output into:' with an empty text box. At the bottom, there is a checked checkbox 'This action is Enabled', a 'More Info' button, and 'OK' and 'Cancel' buttons.

Properties:

Javascript to run:

Enter here the Javascript code you want to execute. TIP: Within the script, you may include ProcessRobot variables, since they will be evaluated prior to the Javascript code's execution.

Store Output into:

Enter the name for the variable to hold the script's output. A script may write some text to the windows console during its execution. In this case, all the text written by the script is stored into this variable so that it can be processed by later actions.

Store Error Output into:

Enter the name for the variable to hold any errors that may occur during the execution of the Javascript code.

NOTE:

Commands in Javascript will be highlighted accordingly so that you can read your script easily. Also this action's window can be resized.

3.11.1.5 Run PowerShell Script

Description:

Executes some custom Powershell Script and retrieves its output into a variable.

The screenshot shows the 'Properties of 'Run PowerShell Script' action' dialog box. The title bar includes a close button (X). The main header area contains a document icon, the title 'Run PowerShell Script', and a description: 'Executes some custom Powershell Script and retrieves its output into a variable.' Below this, there are two tabs: 'General' (selected) and 'Exception Handling'. The 'General' tab is divided into two sections: 'Action Input' and 'Action Output'. The 'Action Input' section has a label 'Powershell script to run:' followed by a large empty text area and an information icon (i) and a settings icon (gear). The 'Action Output' section has two text input fields: 'Store Output into:' with the value '%PowershellOutput%' and an information icon (i), and 'Store Error Output into:' which is empty and also has an information icon (i). At the bottom left, there is a checked checkbox labeled 'This action is Enabled' and a 'More Info' button. At the bottom right, there are 'OK' and 'Cancel' buttons.

Properties:**Powershell Script to run:**

Enter here the Powershell code you want to execute. TIP: Within the script, you may include ProcessRobot variables, since they will be evaluated prior to the Powershell code's execution. In the example presented above, please remember to escape any backslash inside the path variable (%DataDirectory%):

```
C:\\Windows\\SysWOW64
```

Store Output into:

Enter the name for the variable to hold the script's output. A script may write some text to the windows console during its execution. In this case, all the text written by the script is stored into this variable so that it can be processed by later actions.

Store Error Output into:

Enter the name for the variable to hold any errors that may occur during the execution of the Powershell code.

3.11.1.6 Run Python Script Action**Description:**

Executes Python2 script code and retrieves its output into a variable.

Properties of 'Run Python Script' action

Run Python Script
Executes Python2 script code and retrieves its output into a variable.

General | Advanced | Exception Handling

Action Input

Python script to run:

Action Output

Store Output into: %PythonScriptOutput%

Store Error Output into:

This action is Enabled

More Info OK Cancel

Properties:

Python script to run:

Enter here the Python script code you want to execute. TIP: Within the script, you may include ProcessRobot variables, since they will be evaluated prior to the Python script code's execution.

Store Output into:

Enter the name for the variable to hold the script's output.

Store Error Output into:

Enter the name for the variable to hold any errors that may occur during the execution of the Python script code.

NOTE:

Commands in Python will be highlighted accordingly so that you can read your script easily. Also this Action's window can be resized.

3.11.1.7 Terminate Process Action

Description:

This action immediately stops a running process.

Properties:

Specify Process by:

Choose whether you want to specify the process to terminate by its name, or by its ID.

Process Name:

Enter the name of the process that you want to terminate. If more than one process with the same name are running, all of them will be terminated.

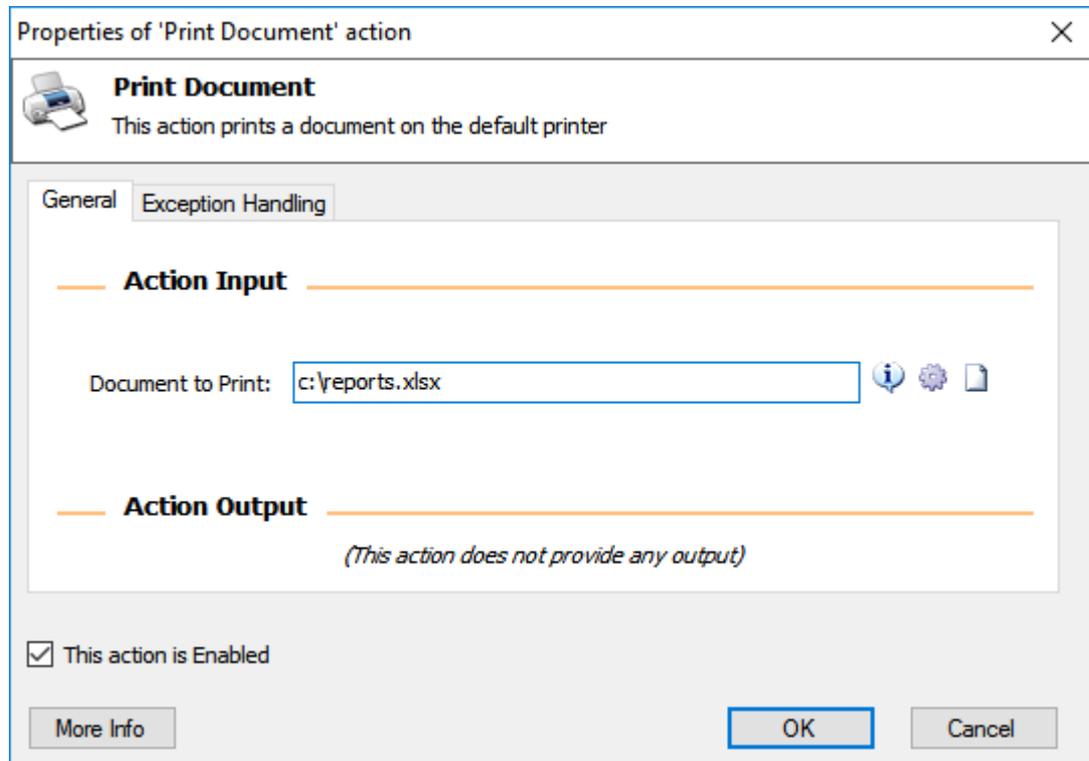
Process ID:

Enter the ID of the process that you want to terminate.

3.11.1.8 Print Document Action

Description:

This action prints a document on the default printer



The screenshot shows a dialog box titled "Properties of 'Print Document' action". It features a printer icon and the text "Print Document" and "This action prints a document on the default printer". There are two tabs: "General" (selected) and "Exception Handling". Under "General", there is an "Action Input" section with a text field labeled "Document to Print:" containing the path "c:\reports.xlsx". To the right of the text field are three icons: an information icon, a gear icon, and a document icon. Below the input field is an "Action Output" section with the text "(This action does not provide any output)". At the bottom, there is a checked checkbox labeled "This action is Enabled", a "More Info" button, and "OK" and "Cancel" buttons.

Properties:

Document to Print:

Insert the file to print here, as [File Variable](#)⁴³⁵ or as a complete file path.

3.11.1.9 Get Default Printer Action

Description:

This action gets the Default Printer name

The screenshot shows a dialog box titled "Properties of 'Get Default Printer' action". It features a printer icon and a checked checkbox. The description reads: "Get Default Printer. This action gets the default printer name." Below this are two tabs: "General" (selected) and "Exception Handling". Under the "General" tab, there are sections for "Action Input" and "Action Output". The "Action Input" section contains the text "(This action does not accept any input)". The "Action Output" section has a label "Store Default Printer's Name into:" followed by a text input field containing "%PrinterName%" and an information icon. At the bottom, there is a checked checkbox "This action is Enabled", a "More Info" button, and "OK" and "Cancel" buttons.

Properties:

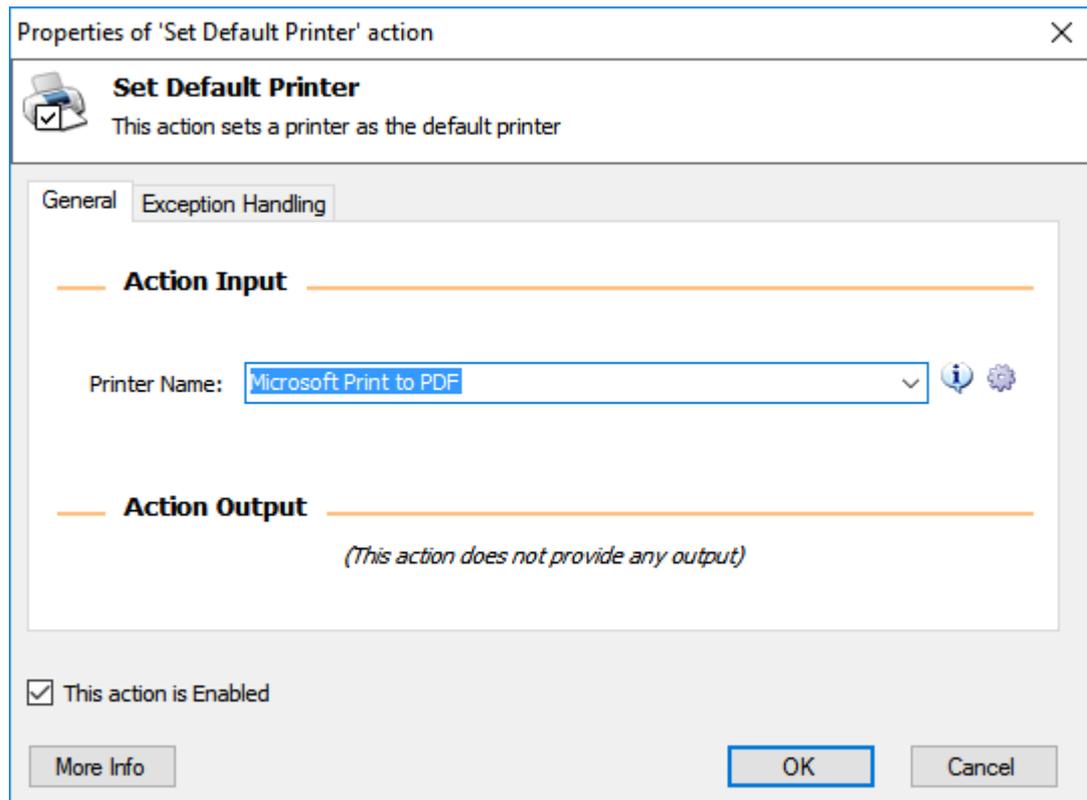
Store Default Printer's Name into:

Choose the variable to hold the acquired default printer's name.

3.11.1.10 Set Default Printer Action

Description:

This action sets a printer as the default printer



Properties:

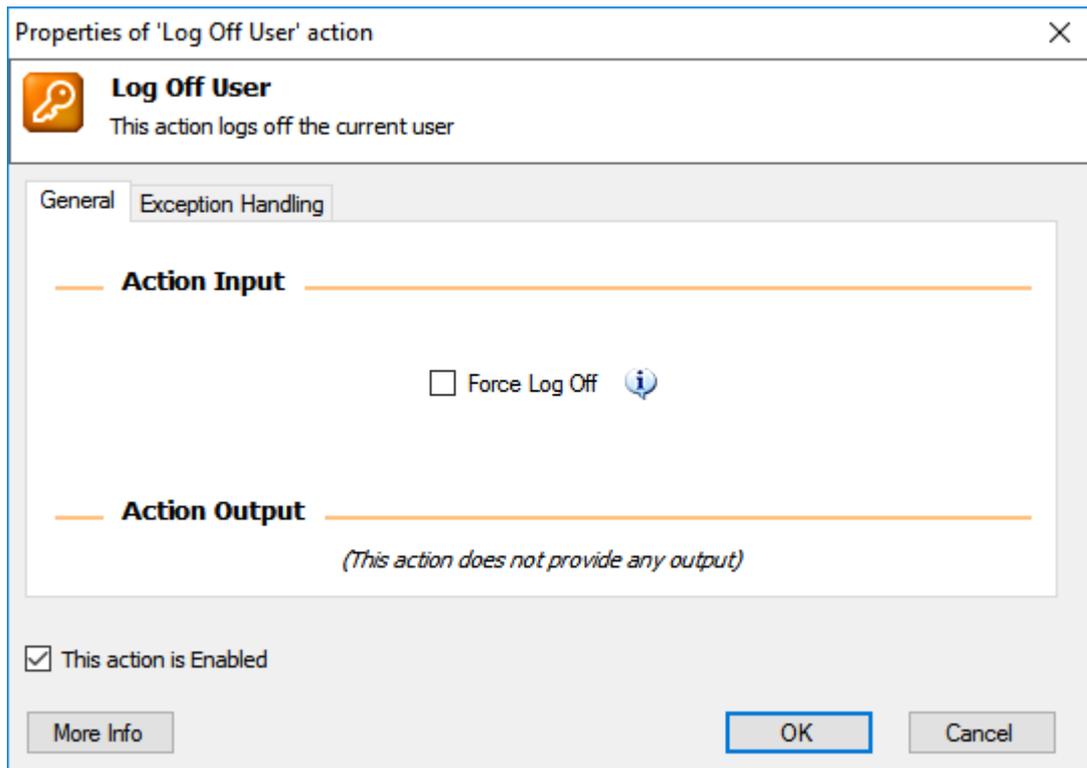
Printer Name:

Choose a printer to be used as default. This will continue to be the default printer until another default printer is chosen.

3.11.1.11 Log Off User Action

Description:

This action logs off the current user



Properties:

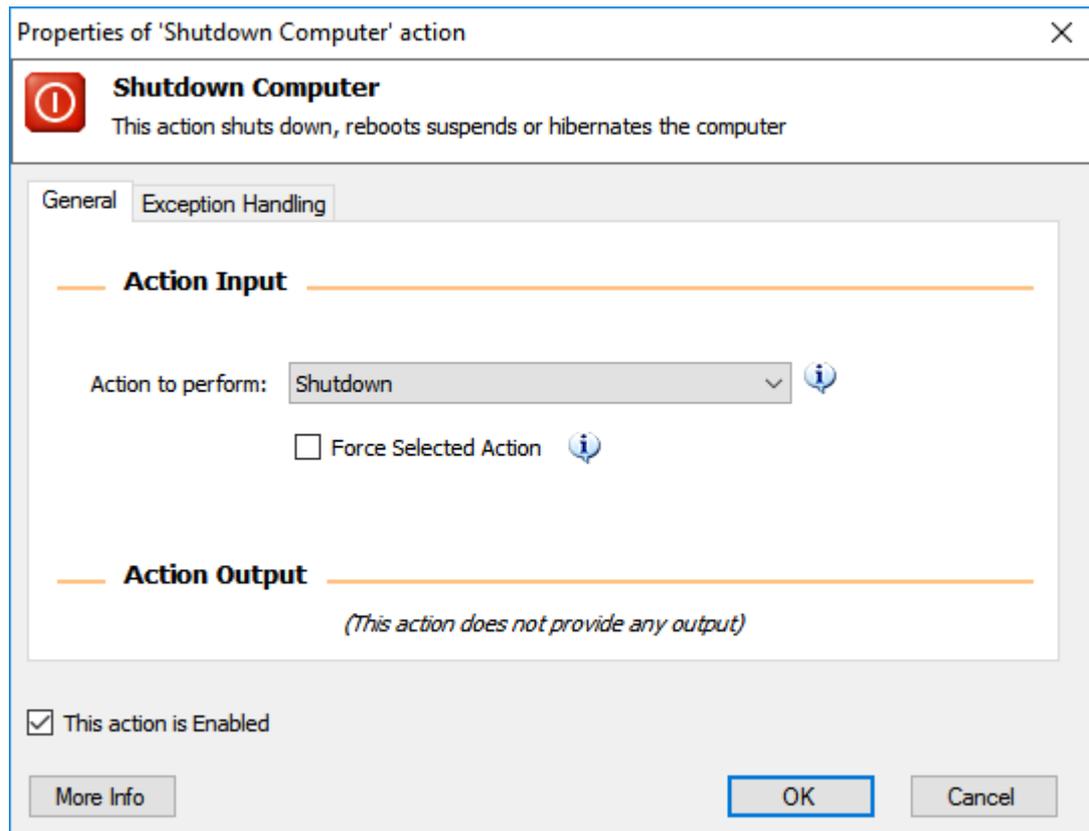
Force Log Off:

Force Log Off, regardless of unsaved files or programs that won't close.

3.11.1.12 Shutdown Computer Action

Description:

This action shuts down, reboots suspends or hibernates the computer



Properties:

Action to Perform:

Choose which Shutdown action to perform.

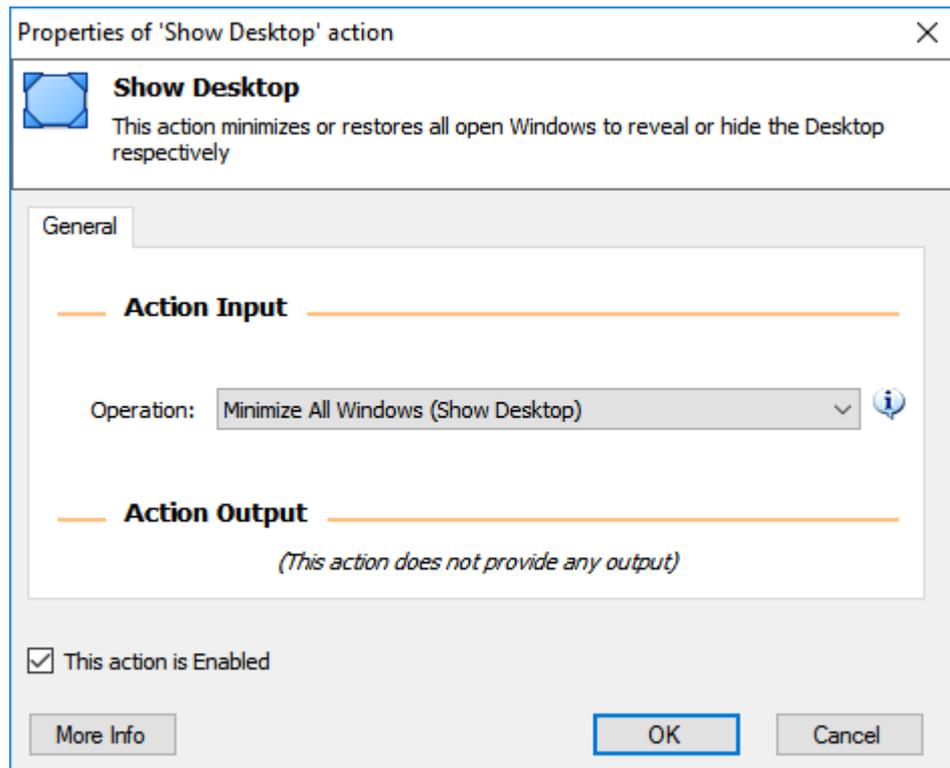
Force Selected Action:

Force selected action, regardless of unsaved files or programs that won't close.

3.11.1.13 Show Desktop Action

Description:

This action minimizes or restores all open Windows to reveal or hide the Desktop respectively.



Properties:

Action to Perform:

Choose whether to minimize all Windows to reveal the Desktop or restore all Windows to their original respective states.

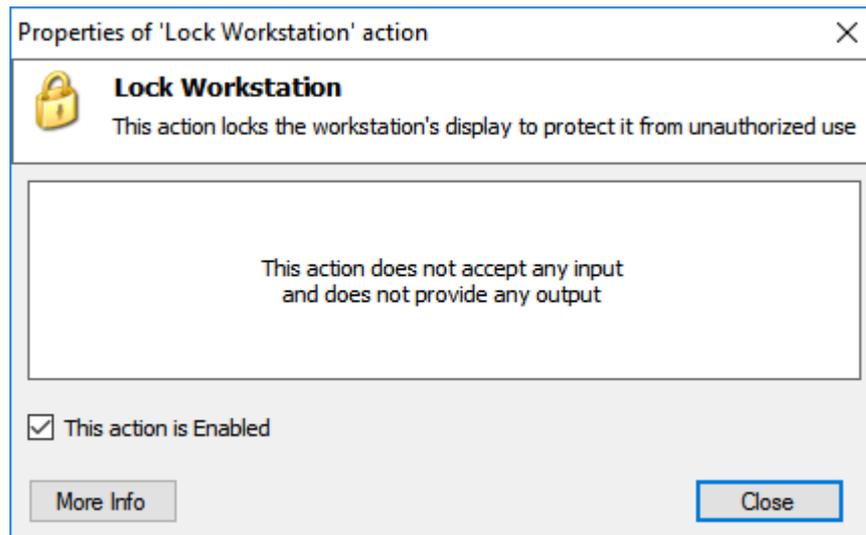
Cautions:

Avoid minimizing any Window you are currently interacting with.

3.11.1.14 Lock Workstation Action

Description:

This action locks the workstation's display to protect it from unauthorized use

**Properties:**

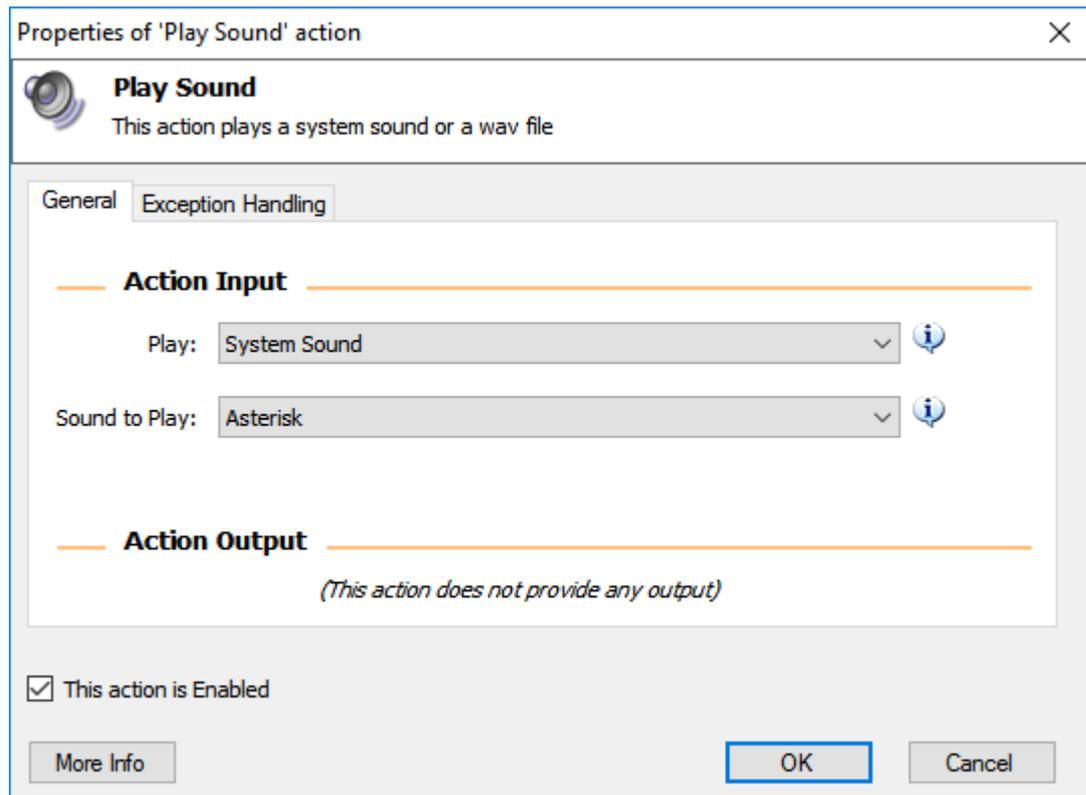
This action does not have any configurable properties.

Cautions:

This action will prevent your Process from interacting with the Desktop. Make sure to use this command only when you're sure the Process won't need to use the mouse, keyboard or request user input after the action has been executed.

3.11.1.15 Play Sound Action**Description:**

This action plays a system sound or a wav file



Properties:

Play:

Choose a type of sound to play.

Sound to Play:

Choose the specific sound to play.

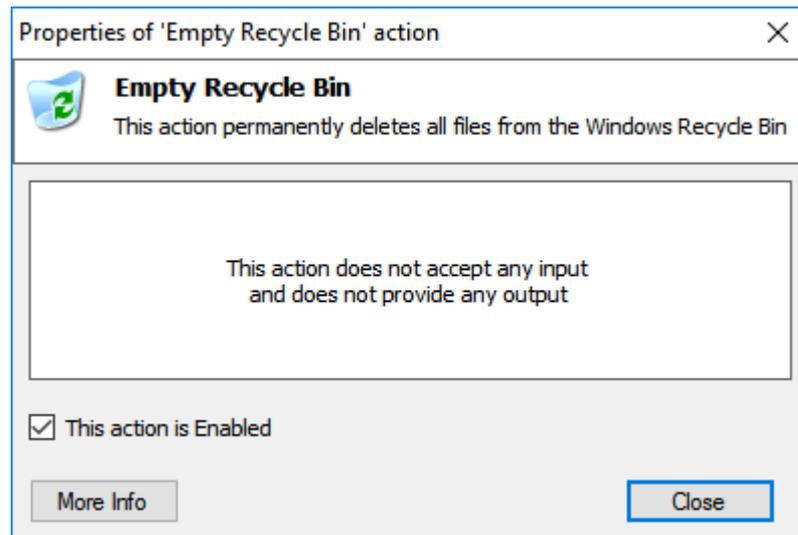
File to Play:

Set the full path of the specific .wav file to play.

3.11.1.16 Empty Recycle Bin Action

Description:

This action permanently deletes all files from the Windows Recycle Bin



Properties:

This action does not have any configurable properties.

3.11.1.17 Take ScreenShot Action

Description:

This action takes a screenshot of the foreground window or the entire screen and saves the image in a file or to clipboard

Properties of 'Take ScreenShot' action

Take ScreenShot
This action takes a screenshot of the foreground window or the entire screen and saves the image in a file or to clipboard

General Exception Handling

Action Input

Capture: Entire Screen 

Save Screen Shot to: File 

Image File: c:\Windows   

Image Format: Bmp 

Action Output

(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties:**Capture:**

Choose an area to capture.

Save Screen Shot to:

Choose the location to save Screen Shot to.

Image File:

Set the full path of the file name where image capture will be saved.

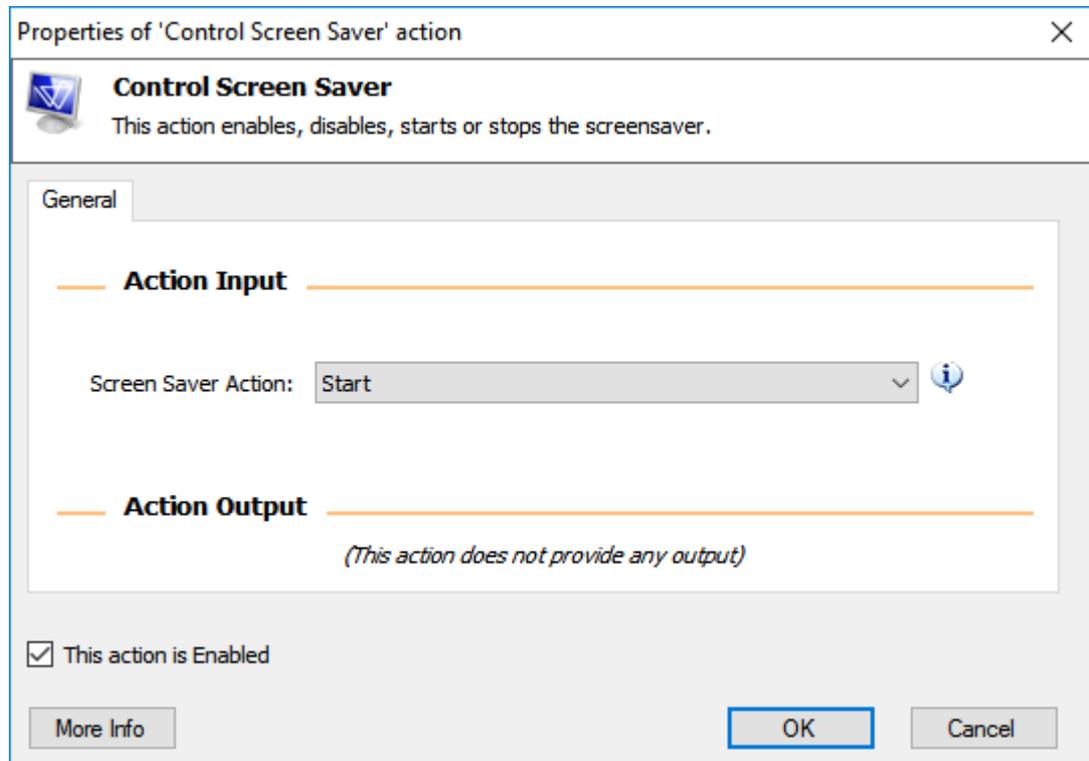
Image Format:

Set the format for the image file to be saved.

3.11.1.18 Control Screen Saver Action

Description:

This action enables, disables, starts or stops the screensaver.



Properties:

Screen Saver Action:

Sets the function of the screensaver.

3.11.1.19 Ping Action

Description:

Sends a message to determine whether a remote computer is accessible over the network.

Properties:

Host Name:

Enter the name of the remote computer or an IP address.

Timeout:

Enter the maximum number of milliseconds (after sending the echo message) to wait for the Ping reply message.

Store Ping Result into:

Enter the name of the variable that will hold the status of the Ping message. The value can be either "Success" or "Failure".

Store Roundtrip Time into:

Enter the name of the variable that will hold the number of milliseconds taken for the Ping to complete.

3.11.1.20 Set Environmental Variable Action

Description:

This action sets an environment variable with a given value.

Properties:

Set Environmental Variable:

Enter the name of the Environmental Variable that you want to set. Variable cannot be greater than or equal to 255 characters

With value:

Enter the value that will be stored in the Environmental Variable

and Type:

Select the type of the Environmental Variable that will be saved in.

3.11.1.21 Get Environmental Variable Action

Description:

This action retrieves the value of an environmental variable

Properties of 'Get Environment Variable' action

Get Environment Variable
This action retrieves the value of an environment variable.

General | Advanced | Exception Handling

Action Input

Environment Variable Name:  

Action Output

Store Environment Variable Value Into: 

This action is Enabled

[More Info](#) [OK](#) [Cancel](#)

Properties:**Environmental Variable Name:**

Enter the name of the Environmental Variable that you wish to retrieve its value

Store Environmental Variable Value Into:

Enter a name to be the variable that will hold the environmental variable's value

3.11.1.22 Delete Environmental Variable Action**Description:**

This action deletes the environment variable from a given scope.

Properties:**Delete Environmental Variable:**

Enter the name of the Environmental Variable that you want to delete. Variable cannot be greater than or equal to 255 characters.

Environmental Variable Type:

Select the type of the Environmental Variable that will be deleted.

3.11.1.23 Get Screen Resolution**Description:**

This actions returns the width, height, bit count and frequency of the selected monitor

Properties of 'Get Screen Resolution' action

Get Screen Resolution
This action return the width, height, bit count and the frequency of the selected monitor

General | Exception Handling

Action Input

Number of Monitor: ⓘ ⚙️

Action Output

Store Width into: ⓘ

Store Height into: ⓘ

Store Bit Count into: ⓘ

Store Frequency into: ⓘ

This action is Enabled

More Info

Properties:

Number of Monitors: Enter the screen identification number of the monitor whose resolution you want to get.

Store Width into: Enter a name to be the variable that will store the width of the monitor. The width will be a numeric output.

Store Height into: Enter a name to be the variable that will store the height of the monitor. The height will be a numeric output.

Store Bit Count into: Enter a name to be the variable that will store the Bit Count of the monitor. The Bit Count will be a numeric output.

Store Frequency into: Enter a name to be the variable that will store the frequency of the monitor. The frequency will be a numeric output.

3.11.1.24 Set Screen Resolution

Description:

This actions sets the width, height, bit count and frequency of the selected monitor

Properties of 'Set Screen Resolution' action ✕

 **Set Screen Resolution**
 This action sets the width, height, bit count and the frequency of a selected monitor

General Exception Handling

Action Input

Number of Monitor: i ⚙

Set Width: i ⚙

Set Height: i ⚙

Set Bit Count: i ⚙

Set Frequency: i ⚙

Available Resolutions

Action Output

(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Available Resolutions ✕

Monitor: ▼

Width	Height ▲	BitCount	Frequency
640	480	32	60 ▲
854	480	32	60
720	480	32	60
800	480	32	60
720	576	32	60
800	600	32	60
1280	720	32	60
1024	768	32	60
1366	768	32	60
1280	768	32	60
1280	800	32	60
1152	864	32	60
1440	900	32	60
1280	960	32	60
1280	1024	32	60
1716	1047	32	60
1400	1050	32	60
1680	1050	32	60
1920	1080	32	60
1600	1200	32	60
1920	1200	32	60
1920	1440	32	60
2560	1440	32	60
2880	1536	32	60

Properties:

Number of Monitors: Enter the index of the monitor whose resolution you want to change

Set Width: Enter the new Width Value that the new screen resolution will have

Set Height: Enter the new Height Value that the new screen resolution will have

Set Bit Count: Enter the new Bit Count that the new screen resolution will have

Set Frequency: Enter the new Frequency that the new screen resolution will have

3.11.2 Conditionals

3.11.2.1 If Action

Description:

This action marks the beginning of a conditional block of actions.

Properties:

First Operand:

Enter a variable output defined by a previous action, or a literal value such as a text or a number to compare with the second Operand.

Operator:

Choose the relationship of first Operand to second Operand.

Ignore Case:

If the comparison is between two text values or variables, this option will specify whether any difference in case between the two operands will be taken into account. This property is available only if the selected operator is any of the following: "Starts with", "Does not start with", "Contains", "Does not contain", "Ends with" or "Does not end with".

Second Operand:

Enter a variable output defined by a previous action, or a literal value such as a text or a number to compare with the first Operand.

Remarks:

The operators "Is Empty" and "Is not Empty" have a special meaning depending on the data type of the first operand. For example, if the first operand is a variable that holds a text value, "Is Empty" is true if the text value is equal to the empty string. If the operand is a variable that holds a list then "Is Empty" is true if the list contains no items.

If the first operand is a value that holds a file (e.g. could be a variable populated by a "Get Files in Folder" action) then "Is Empty" is true only if the file is an empty file (its size equals to 0 bytes). Accordingly, if instead of a file is a folder variable "Is Empty" will be true if the folder does not contain any files or subfolders.

3.11.2.2 If File Exists Action**Description:**

This action marks the beginning of a conditional block of actions depending on whether a file exists or not

Properties of 'If File Exists' action

If File Exists
This action marks the beginning of a conditional block of actions depending on whether a file exists or not

General

Action Input

If File: Exists

File Path: C:\My Files\Example.txt

Action Output
(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties:**If File:**

Choose state of file to be checked.

File Path:

Set the full path where action will look for the file to be checked.

3.11.2.3 If Folder Exists Action**Description:**

This action marks the beginning of a conditional block of actions depending on whether a folder exists or not

Properties of 'If Folder Exists' action

If Folder Exists
This action marks the beginning of a conditional block of actions depending on whether a folder exists or not

General

Action Input

If Folder: Exists

Folder Path: C:\My Files\Folder

Action Output
(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties:**If Folder:**

Choose state of folder to be checked.

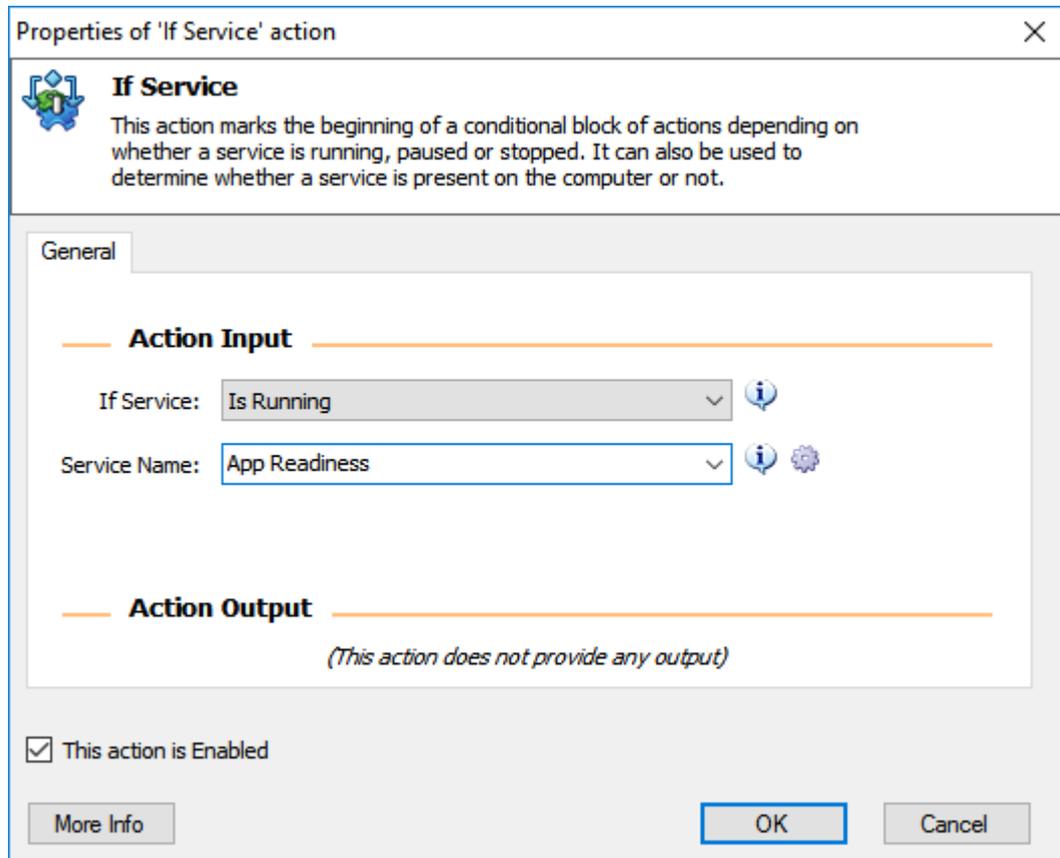
Folder Path:

Set the full path where action will look for the folder to be checked.

3.11.2.4 If Service Action

Description:

This action marks the beginning of a conditional block of actions depending on whether a service is running, paused or stopped. It can also be used to determine whether a service is present on the computer or not.



Properties of 'If Service' action

If Service
This action marks the beginning of a conditional block of actions depending on whether a service is running, paused or stopped. It can also be used to determine whether a service is present on the computer or not.

General

Action Input

If Service: Is Running

Service Name: App Readiness

Action Output
(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties:

If Service:

Choose state of service to be checked. You can check whether the specified service is running, paused or stopped as well as whether it is installed or not.

Service Name:

Choose a specific service installed on your computer to be checked.

3.11.2.5 If Process Action

Description:

This action marks the beginning of a conditional block of actions depending on whether a process is running or not

Properties of 'If Process' action

If Process
This action marks the beginning of a conditional block of actions depending on whether a process is running or not

General

Action Input

If Process: Is Running

Process Name: chrome

Refresh

Action Output
(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties:

If Process:

Choose state of process to be checked.

Process Name:

Choose a specific process to be checked.

3.11.2.6 If Window Action

Description:

This action marks the beginning of a conditional block of actions depending on whether a window is open or not or whether a window is the focused (foreground) window.

Properties of 'If Window' action

If Window
This action marks the beginning of a conditional block of actions depending on whether a window is open or not or whether a window is the focused (foreground) window.

General

Action Input

Get Window: by Window Control

Control:
(no control selected)

Select Control From Repository

Check if Window: Is Open

Action Output
(This action does not provide any output)

This action is Enabled

More Info OK Cancel

by Window Control

Properties of 'If Window' action

If Window
This action marks the beginning of a conditional block of actions depending on whether a window is open or not or whether a window is the focused (foreground) window.

General

Action Input

Get Window: by Window Instance/Handle

Window Instance:

Check if Window: Is Open

Action Output

(This action does not provide any output)

This action is Enabled

More Info OK Cancel

by Window Instance/Handle

Properties of 'If Window' action

If Window
This action marks the beginning of a conditional block of actions depending on whether a window is open or not or whether a window is the focused (foreground) window.

General

Action Input

Get Window: by Title and/or Class 

Window Title:  

Window Class:  

or Click and Drag the target to choose a Window: 

Check if Window: Is Open 

Action Output

(This action does not provide any output)

This action is Enabled

by Title and/or Class

Properties:

Get Window:

Choose whether the action gets the target window from a "Window Instance" variable, or searches for it by Title and/or Class.

Window Instance:

Enter the variable that contains the Window Instance which you want to bring to front. This must be a variable defined by a preceding "[Get Window](#)^[690]" action or a valid Window Handle.

Window Title:

Choose title of window. You can use wildcards, like '?' or '*'. If you can't find the window title in the dropdown list, type it yourself, or open it and press the Refresh button to the right.

Window Class:

Optional: If you have two windows with the same title, Window Class may help differentiate between them. In this case, enter the class of the window you wish to use. Otherwise, leave this blank.

Check if Window:

The state of window to be checked. Choose between: Open, Not Open, Focused, Not Focused.

3.11.2.7 If Image Action

Description:

This action marks the beginning of a conditional block of actions depending on whether a selected image is found on the screen or not.

Properties of 'If Image' action

If Image
This action marks the beginning of a conditional block of actions depending on whether a selected image is found on the screen or not.

General | Exception Handling

Action Input

If Image: exists

Image to wait for: Internet Explorer

Add Image From Images Repository

Delete Image From List

Search for Image on: Entire Screen

Search Mode: Search whole Screen or Foreground Window

Find All Images in List

Tolerance: 10

Action Output

(This action does not provide any output)

This action is Enabled

More Info

OK Cancel

Properties:

If Image:

Choose whether you want to check for the existence or absence of the selected image.

Image to wait for:

Select the Image that the action check if it exists or not. If you need to open a menu or perform steps first, choose 'Capture with Delay'.

Search for Image on:

Choose whether you want to search for the specified Image in the foreground Window only, or the entire visible screen. Neither choice will find the Image if it is not clearly visible on the screen.

Search Mode:

Specify whether you want to scan the entire screen (or window) to find the supplied image or only a narrowed down subregion of it.

Tolerance:

Specify a value for how much the Image searched for can differ from the originally chosen Image.

3.11.2.8 If Text on Screen (OCR)

This action marks the beginning of a conditional block of actions depending on whether a given text [2] is found on the screen or not [4], using an OCR Engine of your choice [1]. The action allows you to search for text on the Entire Screen or the Foreground Window [5] as also to limit your search on a subregion of your working area [6].

Properties of 'If Text on Screen (OCR)' action

If Text on Screen (OCR)
This action marks the beginning of a conditional block of actions depending on whether a given text is found on the screen or not, using OCR.

General | Exception Handling

1 **Action Input**

OCR Engine: %OCREngine%

Text to Find:

Is Regular Expression **3**

If Text: Exists

5 Search for Text on: Entire Screen

Search Mode: Search whole Screen or Foreground Window **6**

2

4

Action Output

(This action does not have any output properties)

This action is Enabled

More Info OK Cancel

1

OCR Engine:

This text field with drop down menu options invites you to enter *or choose* the instance of the OCR Engine you want to work with.

2

Text to Find:

This text field allows you to enter the text you want the OCR Engine to search for as text or as a Variable.

3

Is Regular Expression:

Check this check box if you want to use a Regular Expression to find the text on screen. A Regular Expression creates a range of possibilities and can return a number of results that match your search. The order that the Engine is adding these matches is from top to bottom, left to right as it finds them on the given source to analyze!

4

If Text:

Select if you want the action to check if the text exists or not on the given source to analyze.

5

Search for Text on:

Two options on this drop down menu, *Entire Screen* or *Foreground Window only*.

6

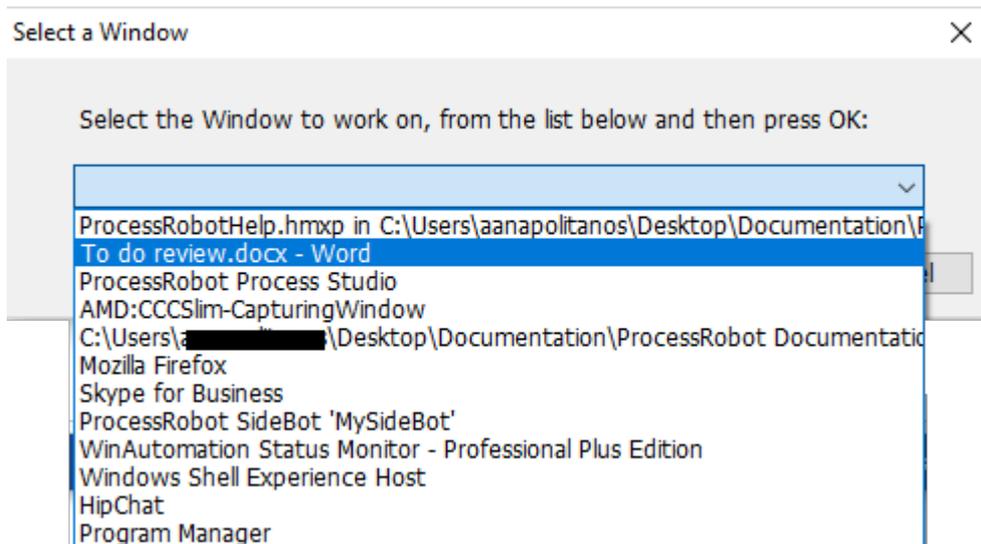
Search Mode:

While the first option of this drop down menu is pretty straightforward (Search Whole Screen or Foreground Window) the other two options that allow you to target a specific subregion of your window or screen produce a whole different set of additional Properties which will see in detail here:

-Search on specified subregion of Screen or Foreground window:

The screenshot shows a configuration panel for searching on a specified subregion. It includes a dropdown for 'Search for Text on:' set to 'Foreground Window only', a dropdown for 'Search Mode:' set to 'Search on specified subregion of Screen or Foreground Window', and four input fields for coordinates: X1 (609), X2 (1003), Y1 (377), and Y2 (554). A 'Select a Different Subregion' button is located to the right of the coordinate fields.

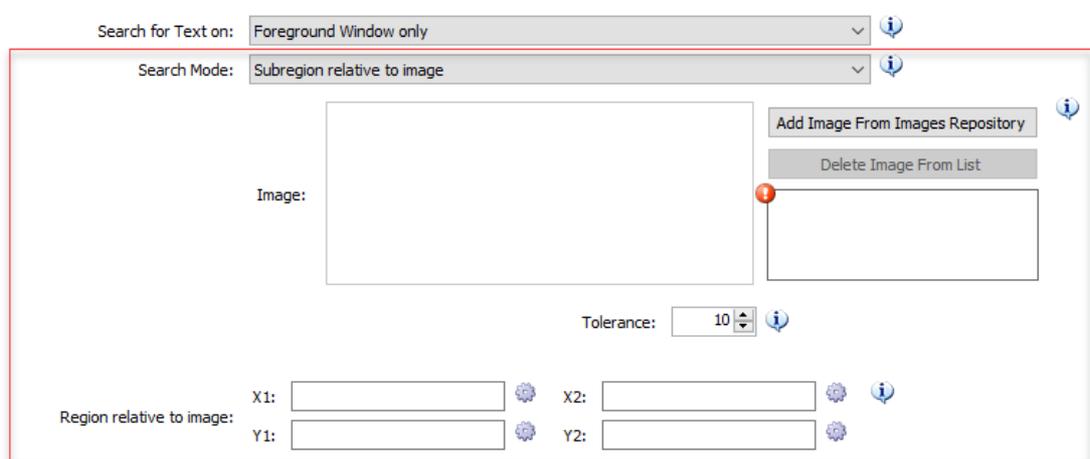
By clicking the "Select a Different Subregion" button on the right of the red box, you will be prompted to select firstly the window you want to work:



and then you will be given the opportunity through an amazing interactive feature to actually select the area you want to narrow down your scan (drag your mouse to select the area). The X1 X2 Y1 Y2 co-ordinates are denoting the start and finish of your dragging manoeuvre.

However, using the "Select a Different Subregion" button is not compulsory. You can enter the values of the co-ordinates by hand or as a Variable (notice the Gear Icon next to the text fields).

-Subregion relative to image:



This search mode enables you to narrow down your scan through a region that is defined relative to the top-left corner of an image from the Image Repository.

3.11.2.9 If Window Contains Action

Description:

Checks whether a specific piece of text or a Window element exists inside a Window.

Option 1: Check for Window Element (Exists/Does not Exist)

Properties of 'If Window Contains' action

If Window Contains
Checks whether a specific piece of text or a Window element exists in a Window.

General

Action Input

Check if Window: Contains Element

Check if Element is: Enabled

Control:
(no control selected)

Select Control From Repository

Action Output
(This action does not provide any output)

This action is Enabled

More Info OK Cancel

You can select the window from your Control Repository or Add it in the Repository.

Option 2: Check for Text (Exists/Does not Exist)

Properties of 'If Window Contains' action

If Window Contains
Checks whether a specific piece of text or a Window element exists in a Window.

General

Action Input

Check if Window: Contains Element
Contains Element
Does Not Contain Element
Contains Text
Does Not Contain Text

Control:
(no control selected)

Select Control From Repository

Action Output
(This action does not provide any output)

This action is Enabled

More Info OK Cancel

This option checks the text inside the Window to determine whether a specified subtext exists or not.

Properties of 'If Window Contains' action

If Window Contains
Checks whether a specific piece of text or a Window element exists in a Window.

General

Action Input

Check if Window: Contains Text

Text to Check for:

Window:
(no control selected)

Select Control From Repository

Action Output
(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties:

Text to Check for:

Enter the text you want check for.

Check if Element is ("Contains Element" should be selected):

Sometimes the target element may be present, but disabled. This property allows you to check whether the target element, if found, is also in a specific state (Enabled or Disabled). The If statement will execute its actions only when the target element is both present and in the required state when this property is checked. This additional "Check if Element is" property becomes available to configure only when the value "Contains Element" is selected in the "Check if Window" field.

Control:

Choose the Control for the element from your Repository or Add it if it is not already there.

3.11.2.10 If Web Page Contains Action

Description:

Checks whether a specific piece of text or a web page element exists in a Web Page.

This action operates on a Web Page that has been previously opened by a "[Launch New Internet Explorer](#)⁷²⁶" action. The latter stores the Web browser instance into a variable. This action accepts this variable and checks whatever page is currently displayed in that Web Browser Instance.

Option 1: Check for Web Element (Exists/Does not Exist)

Properties of 'If Web Page Contains' action

If Web Page Contains
Checks whether a specific piece of text or a web page element exists in a Web Page.

General

Action Input

Web Browser Instance: %InternetExplorer%

Check if Web Page: Contains Element

Control:
(no control selected)

Select Control From Repository

Action Output
(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Choose the Control for the element from your Repository or Add it if it is not already there.

Option 2: Check for Text (Exists/Does not Exist)

This option checks the text of the Web page to determine whether a specified subtext exists or not.

Properties

Web Browser Instance:

Enter the variable that contains the Web Browser Instance you want to work with. This must be a variable defined by a preceding "[Launch New Internet Explorer](#)" action.

Check if Web Page:

Specify whether you want to check for the existence of some text or of an HTML element.

Text to Check for:

Enter the text that you want to check for.

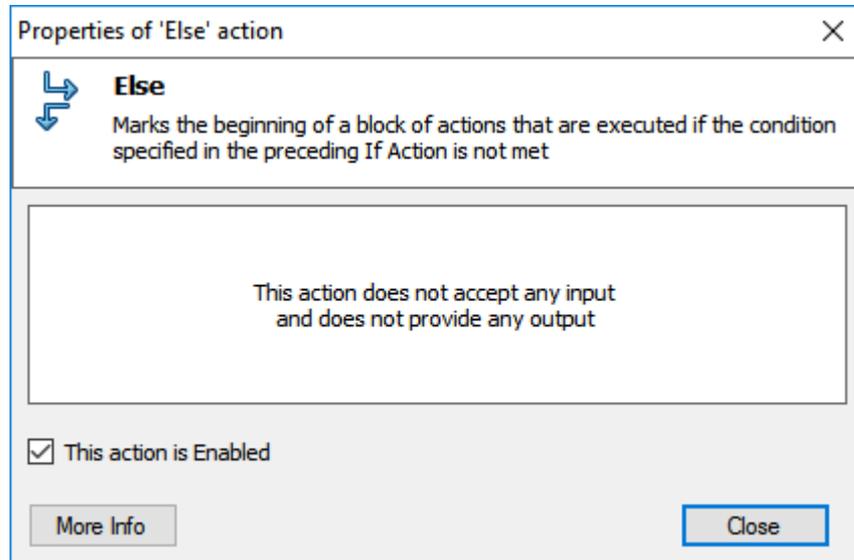
Control:

Choose the Control for the element from your Repository or Add it if it is not already there.

3.11.2.11 Else Action

Description:

Marks the beginning of a block of actions that are executed if the condition specified in the preceding If Action is not met



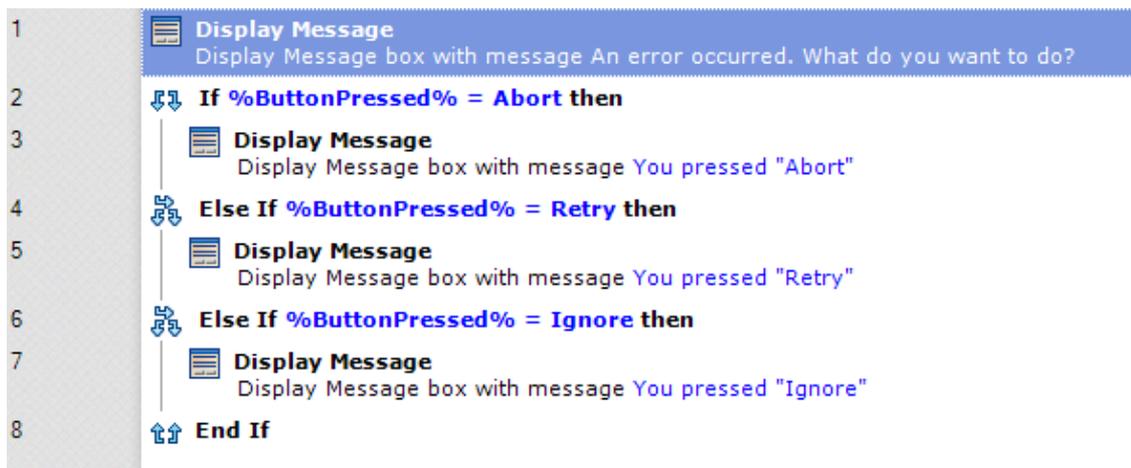
Properties:

This action does not have any configurable properties.

3.11.2.12 Else If Action

Description:

Marks the beginning of a block of actions that are executed if the condition specified in the preceding If Action is not met and the condition specified in this action is met.



This action can only be inserted within an If/End If block and always before the "Else" action, if any. In all other aspects is similar to the ["If" action](#)^[557].

Properties of 'Else If' action

Else If
Marks the beginning of a block of actions that are executed if the condition specified in the preceding If Action is not met and the condition specified in this action is met.

General

Action Input

First Operand: %ButtonPressed%

Operator: =

Second Operand: Retry

Action Output
(This action does not provide any output)

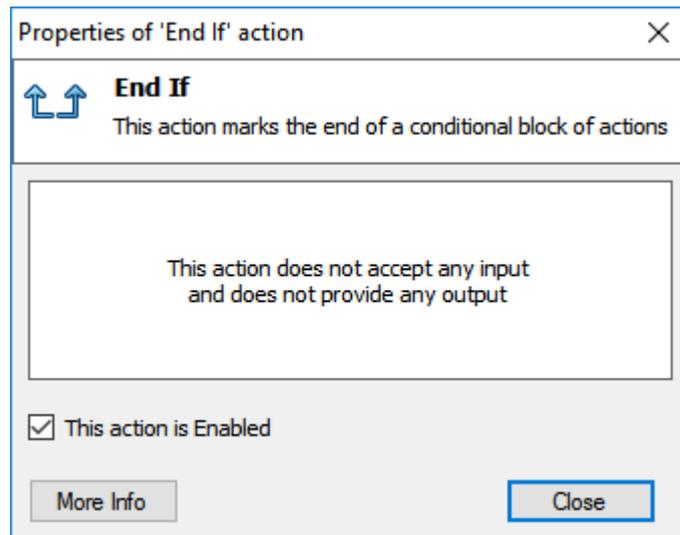
This action is Enabled

More Info OK Cancel

3.11.2.13 End If Action

Description:

This action marks the end of a conditional block of actions

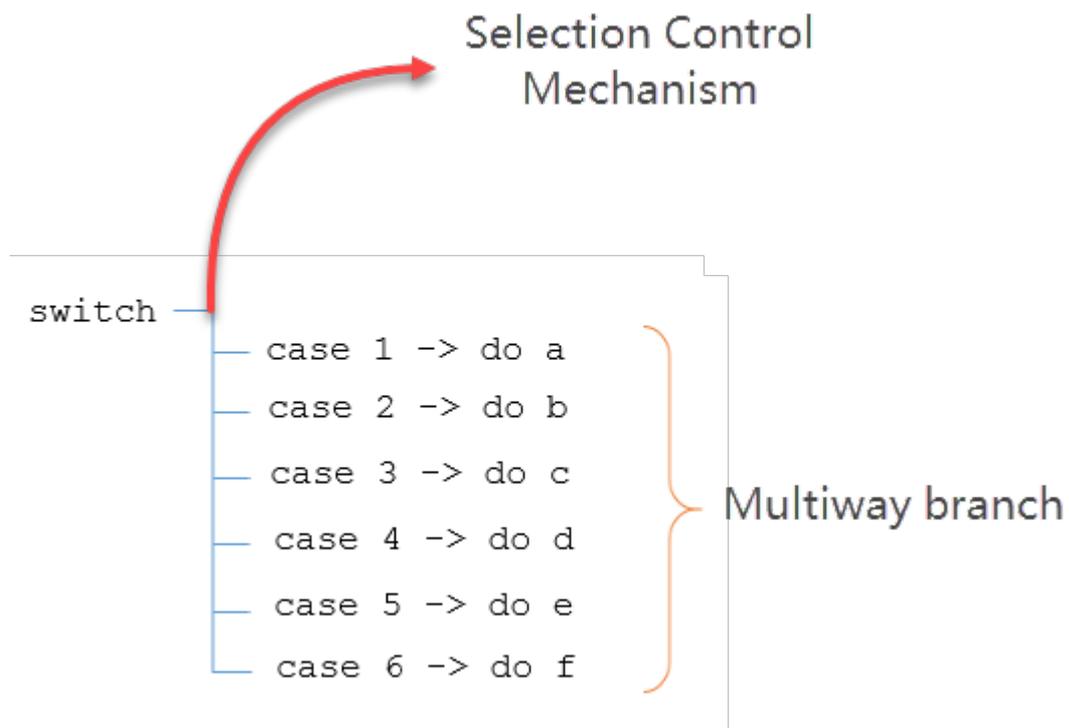


Properties:

This action does not have any configurable properties.

3.11.2.14 Switch

Switch marks the beginning of a switch block. Unlike if-else or else-if conditional statements, a switch block can address a number of possible execution paths. A switch statement is a type of selection control mechanism used to allow the value of a variable or expression to change the control flow of your scripts via a multiway branch:



This action allows you to set a value [1] that will form the basis of any subsequent comparisons with other values using the [Case](#)^[580]/[End Case](#)^[583] actions:

Properties of 'Switch' action

Switch
This action marks the beginning of a switch block.

General

1 **Action Input**

Value to Check: %NewVar%

Action Output
(This action does not provide any output)

This action is Enabled

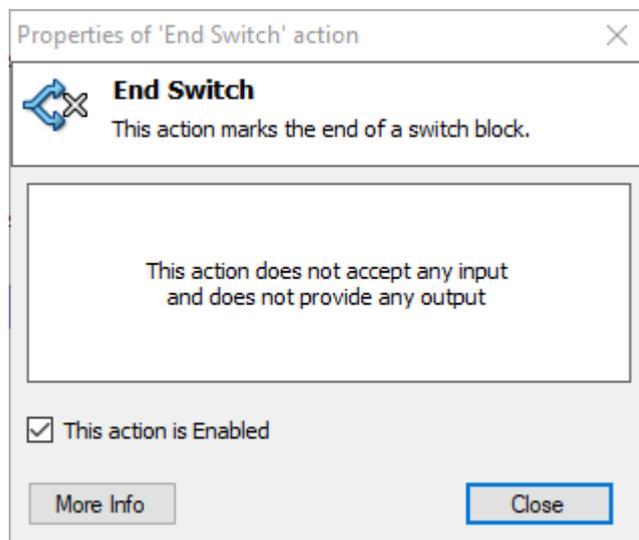
More Info OK Cancel

1 **Value to Check:**

This should be a variable that we expect its value to be derived out of a set of predefined options.

3.11.2.15 End Switch

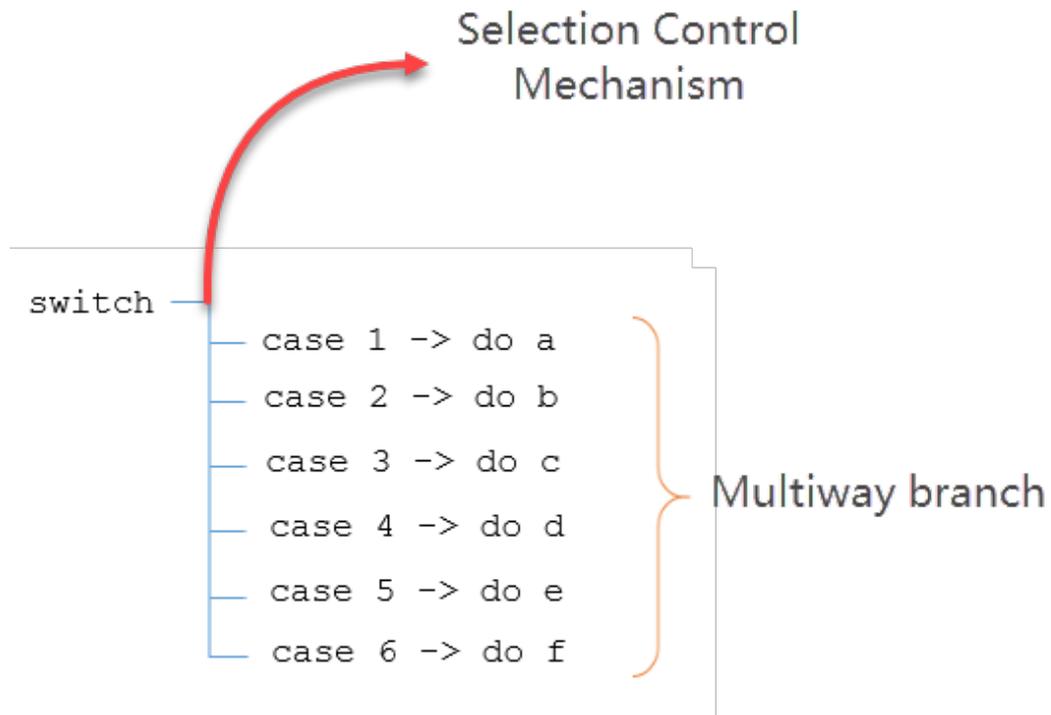
This action simply marks the end of a [Switch](#)^[578] block.



3.11.2.16 Case

This action specifies a condition on an existing [Switch](#)^[578] block and marks the beginning of a sequence of actions that will be executed if the conditional evaluates to true.

The end of this sequence of actions is marked by the complimentary [End Case](#)^[583] action.



The action allows you to choose from a variety of Comparison Types [1] in order to compare the value provided here [2] with that given in the [Switch](#) statement.

Properties of 'Case' action

Case
This action specifies a condition of a switch block.

General

Action Input

Comparison Type: =

Value to Compare: a

Action Output
(This action does not provide any output)

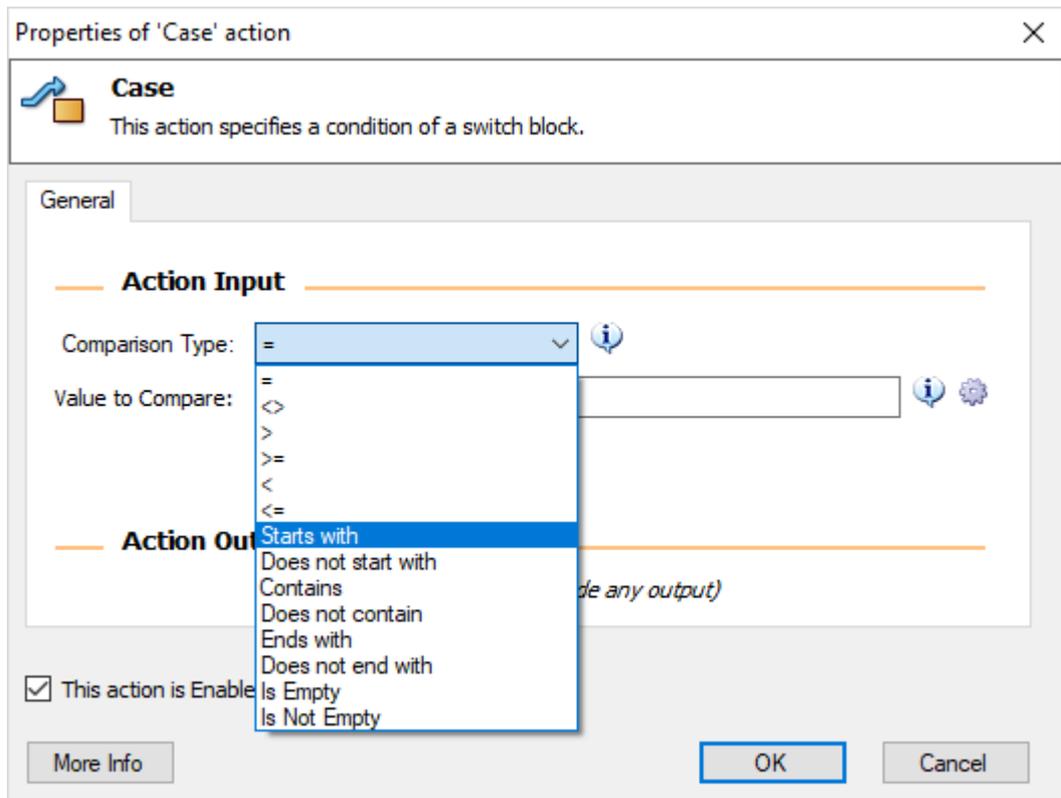
This action is Enabled

More Info OK Cancel

1

Comparison Type:

This property can have a wide variety of predefined values, designed to fulfill your most sophisticated needs:



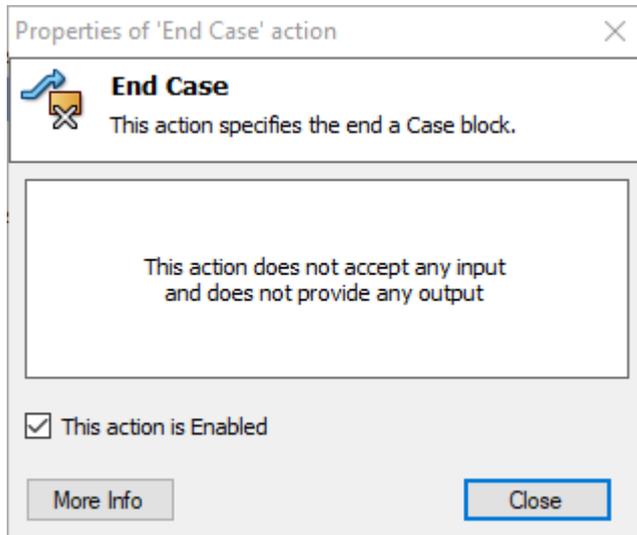
2

Value to Compare:

The Value that you will enter here will be compared after the Comparison Type defined in [1] with the value defined in the [Switch](#)⁵⁷⁸ action. If this criterion is met, then the sequence of actions within the <Case> <sequence of actions> <End Case> block will be executed. If not, the flow will move to the next case condition.

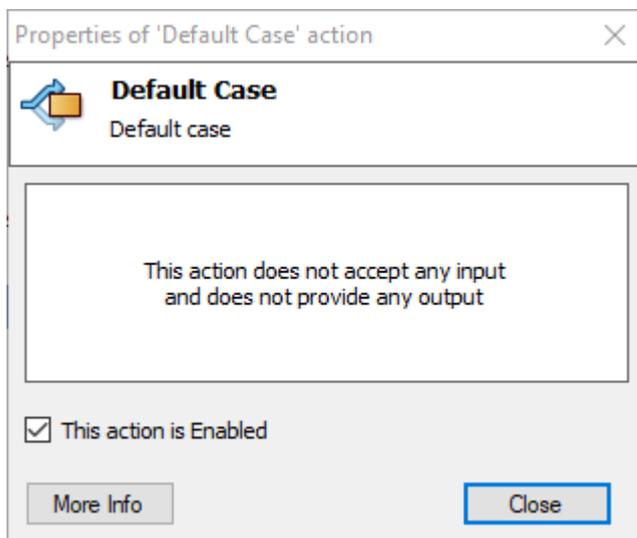
3.11.2.17 End Case

This action simply specifies the end of a [Case](#)⁵⁸⁰ block.



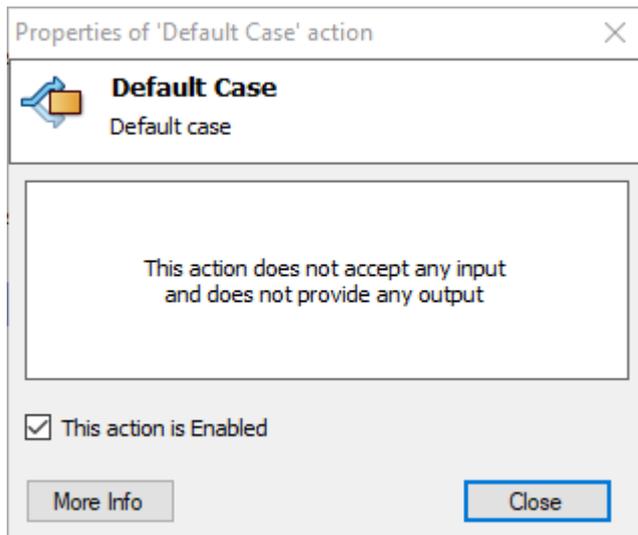
3.11.2.18 Default Case

Use this block in order to execute a sequence of actions if none of your [Case](#)⁵⁸⁰ conditionals are to be met:



3.11.2.19 End Default Case

The complimentary other half of the [Default Case](#)⁵⁸³ action simply marks the end of a Default Case block:



3.11.3 Loops

3.11.3.1 Loop Action

Description:

This action iterates a block of actions for a specific number of times

Properties of 'Loop' action

Loop
This action iterates a block of actions for a specific number of times

General

Action Input

Start From: 1

End To: 10

Increment By: 1

Action Output

Store Current Index into: %LoopIndex%

This action is Enabled

More Info OK Cancel

Properties:

Start From:

Sets the starting point for your Loop counter

End To:

Sets the ending point for your Loop counter

Increment By:

Sets the increments your Loop counter increases by

Store Current Index into:

Enter a name to be the variable that will store the Current Index, starting at the Start From value. The value of this variable will change by the increment with each iteration.

3.11.3.2 Loop Condition Action

Description:

This action iterates a block of actions while a condition is true

Properties:

First Operand:

Enter a variable output defined by a previous action, or a literal value such as a text or a number to compare with the second Operand.

Operator:

Choose the relationship of first Operand to second Operand.

Second Operand:

Enter a variable output defined by a previous action, or a literal value such as a text or a number to compare with the first Operand.

3.11.3.3 For Each Action

Description:

This action loops through the items of a list

The screenshot shows a workflow editor with four steps:

- Read Text from File**: Read content of file C:\Users\admin\Desktop\Testing.txt and store it into %FileContents%
- For Each**: Loop for each item contained in variable %FileContents% and store the current item into %CurrentLine%
- Display Message**: Display Message box with message %CurrentLine%
- End Loop**

The 'Properties of For Each' dialog box is open, showing the following configuration:

- General** tab
- Action Input**: Variable to Iterate: %FileContents%
- Action Output**: Store Current Item into: %CurrentLine%
- This action is Enabled
- Buttons: More Info, OK, Cancel

Properties:

Variable to Iterate:

Enter a previously created variable that contains a list, data table, or data row. The actions in the loop will be executed once for each item in the list, each row of the table, or each column (cell) of the data row.

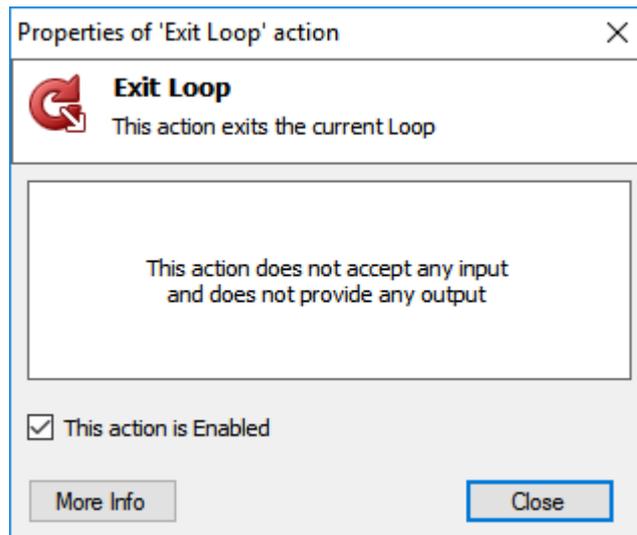
Store Current Item into:

Enter a name to be the variable that will store the Current Item (item, row, or cell), if you wish to access it during later actions in the loop.

3.11.3.4 Exit Loop Action

Description:

This action exits the current Loop



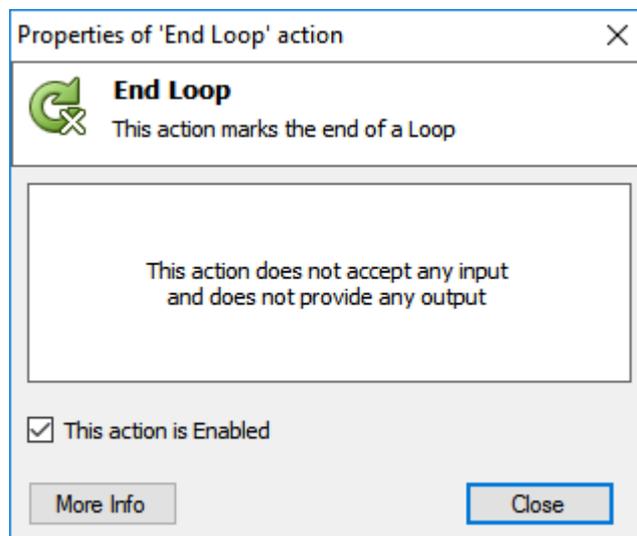
Properties:

This action does not have any configurable properties.

3.11.3.5 End Loop Action

Description:

This action marks the end of a Loop



Properties:

This action does not have any configurable properties.

3.11.4 Wait

3.11.4.1 Wait Action

Description:

This action suspends the execution of the Process for a specified amount of seconds

Properties of 'Wait' action

Wait
This action suspends the execution of the robot for a specified amount of seconds

General

Action Input

Wait for: 10 seconds

Action Output
(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties:

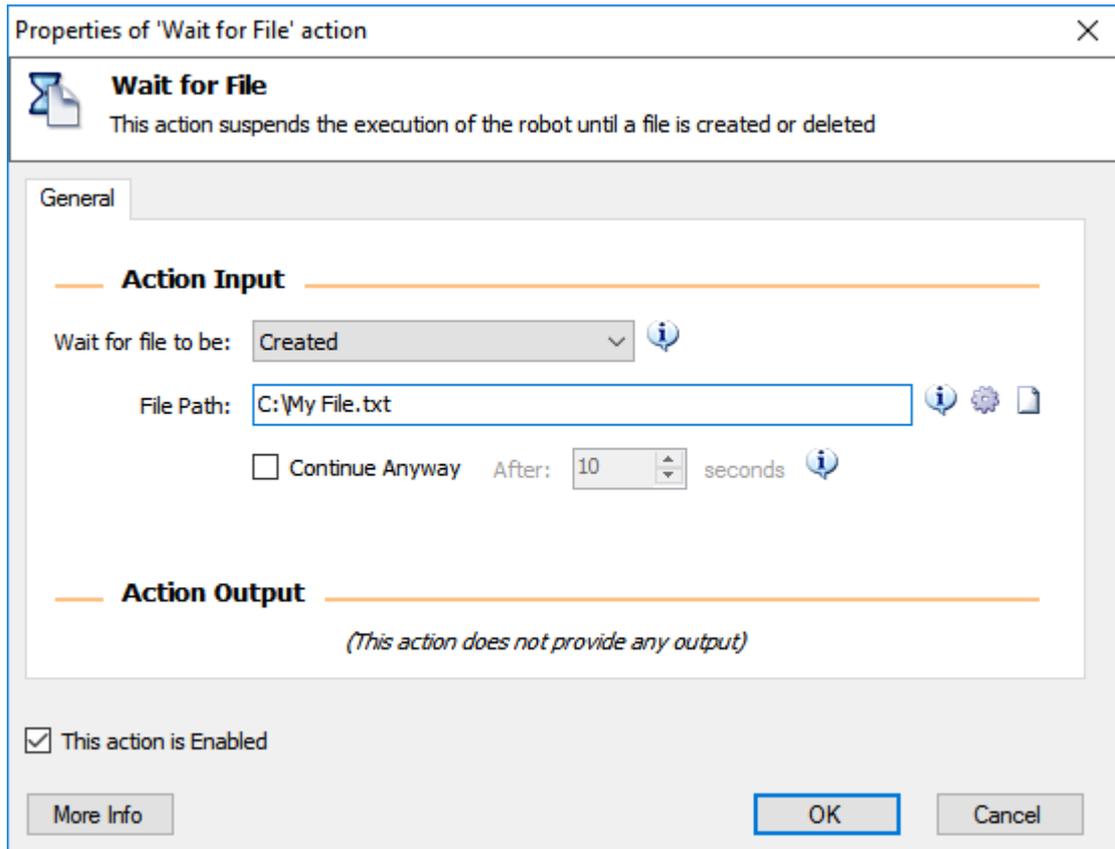
Wait for:

Enter the number of seconds that you want the Process to pause.

3.11.4.2 Wait for File Action

Description:

This action suspends the execution of the Process until a file is created or deleted



Properties:

Wait for file to be:

Choose whether the Process pauses until a certain file is created or deleted.

File Path:

Set the full path of file to be checked.

Continue Anyway:

Choose whether the Process continues after a set number of seconds, regardless of whether the file has been created/deleted.

3.11.4.3 Wait for HotKey Action

Description:

This action suspends the execution of the Process until a specific HotKey is pressed by the user.

There are cases where you want the Process to pause and continue only when the user says so. For example, a Process may open a number of documents for the user to review and needs to know when the user has completely reviewed each document to open the next one.

This is where the "Wait for HotKey" action can be used. You can specify a HotKey and the Process will continue only when this HotKey is pressed.

The screenshot shows a dialog box titled "Properties of 'Wait for HotKey' action". The dialog has a close button (X) in the top right corner. Below the title bar, there is a description of the action: "Wait for HotKey" with a sub-description: "This action suspends the execution of the Robot until a specific HotKey is pressed by the user." The dialog is divided into two tabs: "General" (selected) and "Exception Handling". Under the "General" tab, there are sections for "Action Input" and "Action Output". In the "Action Input" section, there is a field for "HotKey #1" with checkboxes for "Ctrl", "Shift", and "Alt", and a dropdown menu showing "D". Below this is a link: "Specify Additional HotKey to Wait For ...". There is also a checkbox for "Proceed anyway after" followed by a spinner box set to "10" and the word "seconds". The "Action Output" section contains the text: "(This action does not provide any output when only one HotKey is specified)". At the bottom of the dialog, there is a checkbox labeled "This action is Enabled" which is checked. There are three buttons at the bottom: "More Info", "OK", and "Cancel".

To further elaborate our previous example, let's assume that the user, after reviewing each document, has the choice to either archive it or delete it. In this case it would be convenient for the user to be able to press different hotkeys, and the Process to keep track of the hotkey pressed to interpret the user's intention.

Properties of 'Wait for HotKey' action

Wait for HotKey
This action suspends the execution of the Robot until a specific HotKey is pressed by the user.

General | Exception Handling

Action Input

HotKey #1: Ctrl + Shift + Alt + D

HotKey #2: Ctrl + Shift + Alt + 5

[Specify Additional HotKey to Wait For ...](#)

Proceed anyway after 10 seconds

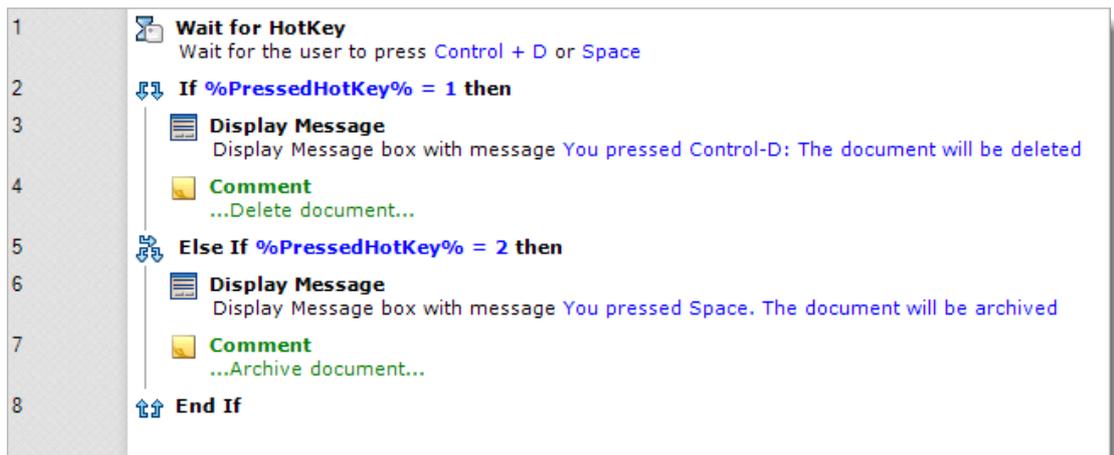
Action Output

Store HotKey Pressed Into: %PressedHotKey%

This action is Enabled

More Info OK Cancel

If you specify more than one hotkeys to wait for, the action populates a variable (%PressedHotKey% in the example above) with the ordinal number of the pressed hotkey. In the example above, if the user pressed Control-D the %PressedHotKey% would hold the value 1 while if the user pressed space the %PressedHotKey% variable would hold the value 2. Then you can check for the hotkey pressed by the user as shown below:



Proceed anyway after:

Choose whether the action will proceed anyway whether the period of time, waiting for a hotkey, expires.

Store HotKey Pressed into:

Enter the name of the variable that will hold the ordinal position of the HotKey pressed by the user. If the user presses the second HotKey the variable will hold the value two.

3.11.4.4 Wait for Process Action

Description:

This action suspends the execution of the Process until a process starts or stops running

Properties of 'Wait for Process' action

Wait for Process
This action suspends the execution of the robot until a process starts or stops running

General | Exception Handling

Action Input

Wait for process to: Start

Process Name: calc Refresh

Continue Anyway After: 10 seconds

Action Output
(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties:

Wait for process to:

Choose whether the Process pauses until a certain process starts or stops.

Process Name:

Choose or enter the process name to be checked.

Continue Anyway:

Choose whether the Process continues after a set number of seconds, regardless of whether the process has started or stopped.

3.11.4.5 Wait for Service Action

Description:

This action suspends the execution of the Process until a service starts, stops or pauses

Properties of 'Wait for Service' action

Wait for Service
This action suspends the execution of the robot until a service starts, stops or pauses

General Exception Handling

Action Input

Wait for service to: Start

Service Name: Windows Audio

Continue Anyway After: 10 seconds

Action Output
(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties:**Wait for service to:**

Choose whether the Process pauses until a certain service starts, stops, or pauses.

Service Name:

Choose or enter the name of a service to be checked.

Continue Anyway:

Choose whether the Process continues after a set number of seconds, regardless of whether the service has started, stopped, or paused.

3.11.4.6 Wait for Window Action

Description:

This action suspends the execution of the Process until a specific window opens, closes, gets or loses the focus.

✕
Properties of 'Wait for Window' action

Wait for Window
 This action suspends the execution of the robot until a specific window opens, closes, gets or loses the focus.

General
Exception Handling

Action Input

Find Window: by Window Control ⓘ

Control:
(no control selected) 📷 ⓘ

Select Control From Repository

Wait For Window To: Open ⓘ

Focus Window after it opens ⓘ

Fail if window does not appear within: 10 seconds ⓘ

Action Output

(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties of 'Wait for Window' action

Wait for Window
This action suspends the execution of the robot until a specific window opens, closes, gets or loses the focus.

General Exception Handling

Action Input

Find Window: by Window Instance/Handle

Window Instance:

Wait For Window To: Open

Focus Window after it opens

Fail if window does not appear within: 10 seconds

Action Output

(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties of 'Wait for Window' action

Wait for Window
This action suspends the execution of the robot until a specific window opens, closes, gets or loses the focus.

General Exception Handling

Action Input

Find Window: by Title and/or Class

Window Title:

Window Class:

or Click and Drag the target to choose a Window:

Wait For Window To: Open

Focus Window after it opens

Fail if window does not appear within: 10 seconds

Action Output

(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties:

Get Window:

Choose whether the action gets the target window from a "Window Instance⁴³⁶" variable, or searches for it by Title and/or Class.

Window Instance:

Enter the variable that contains the Window Instance which you want to bring to front. This must be a variable defined by a preceding "Get Window⁶⁹⁰" action or a valid Window Handle.

Window Title:

Choose title of window. You can use wildcards, like '?' or '*'. If you can't find the window title in the dropdown list, type it yourself, or open it and press the Refresh button to the right.

Window Class:

Optional: If you have two windows with the same title, Window Class may help differentiate between them. In this case, enter the class of the window you wish to use. Otherwise, leave this blank.

Wait for Window to:

Choose the event that this action will wait for: you can choose to wait for a specific window to open, close, become focused (i.e. become the foreground window), or lose focus (stop being the foreground window).

Focus Window after it opens:

Bring the Window to the front after it opens, so later actions are directed at this Window.

Fail if window does not appear/close within:

Choose whether the action will throw an exception after a set number of seconds if the specified Window has not opened/closed/became focused/lost focus. Then go to the Exception Handling tab at top to specify how the Exception will be handled. Otherwise, the Process will stop and issue an error message.

3.11.4.7 Wait for Image Action***Description:***

This action waits until a specific image appears on the screen or on the foreground window. This action uses the [Image Recognition](#)⁵⁰⁶ feature.

Properties of 'Wait for Image' action

Wait for Image
This action waits until a specific image appears on the screen or on the foreground window

General | Exception Handling

Action Input

Wait for Image to: Disappear

Image to wait for: Internet Explorer

Add Image From Images Repository

Delete Image From List

Search for Image on: Foreground Window only

Search Mode: Search on specified subregion of Screen or Foreground Window

X1: 3521 X2: 4036
Y1: 0 Y2: 410

Tolerance: 10

Wait For All Images

Fail if image does not appear within: 10 seconds

Action Output

Location of Image found: X: Y:

This action is Enabled

More Info OK Cancel

Properties:

Image to wait for:

Select the Image that the action will wait to see. If you need to open a menu or perform steps first, choose 'Capture with Delay'.

Search for Image on:

Choose whether you want to search for the specified Image in the foreground Window only, or the entire visible screen. Neither choice will find the Image if it is not clearly visible on the screen.

Search Mode:

Specify whether you want to scan the entire screen (or window) to find the supplied image or only a narrowed down subregion of it.

X1, Y1, X2, Y2:

Specify the subregion of the screen or window to narrow down the scan for the supplied image to. You can visually select a subregion of the screen or window by pressing the "Select a Different Subregion" button.

Tolerance:

Specify a value for how much the Image searched for can differ from the originally chosen Image.

Fail if image does not appear:

Choose whether the action will throw an exception after a set number of seconds if the specified Image has not been found. Then go to the Exception Handling tab at top to specify how the Exception will be handled. Otherwise, the Process will stop and issue an error message.

Location of Image Found:

Enter the names to be the variables that will hold the X and Y coordinates of the point where the image is found on the screen. If the image is being searched for on the foreground window, the coordinates returned are relative to the top left corner of the window.

3.11.4.8 Wait for Text on Screen (OCR)

This action waits (pauses the execution of your script) until a specific text [2] appears/disappears [4] on the entire screen or foreground window [5] allowing you to analyze either the whole of that image or a specific subregion of it [6]. You can also specify whether you want the action to fail [7] if the text does not appear within a custom set of seconds [8]. The action returns the location [9] (coordinates) of the specified text's first occurrence (or match if a regular expression [3] is used) from left to right, top to bottom.

Properties of 'Wait for Text on Screen (OCR)' action

Wait for Text on Screen (OCR)
This action waits until a specific text appears/dissapears on the screen, on the foreground window, or relative to an image on the screen or foreground window using OCR.

General Exception Handling

Action Input

1 OCR Engine: %OCREngine%

2 Text to Find:

3 Is Regular Expression

4 Wait for Text to: Appear

5 Search for Text on: Entire Screen

6 Search Mode: Search whole Screen or Foreground Window

7 Fail if text does not appear within: 10 seconds

8

Action Output

9 Location of Text found: X: %LocationOfTextFoundX% Y: %LocationOfTextFoundY%

This action is Enabled

More Info OK Cancel

1 OCR Engine:

This text field with drop down menu options invites you to enter *or choose* the instance of the OCR Engine you want to work with.

2 Text to Find:

This text field allows you to enter the text you want the OCR Engine to search for as text or as a Variable (please be mindful of the fact that OCR is case sensitive).

3**Is Regular Expression:**

Check this check box if you want to use a Regular Expression to determine the text you are looking for. A Regular Expression creates a range of possibilities and can return a number of results that may match your search. The order that the Engine is adding these matches is from top to bottom, left to right as it scans the given source to analyze.

4**Wait for Text to:**

This drop down menu allows you to specify whether you want to wait for the Text on screen to

5**Search for Text on:**

Two options on this drop down menu, *Entire Screen* or *Foreground Window only*.

6

Search Mode:

While the first option of this drop down menu is pretty straightforward (Search Whole Screen or Foreground Window) the other two options that allow you to target a specific subregion of your window or screen produce a whole different set of additional Properties which will see in detail here:

-Search on specified subregion of Screen or Foreground window:

Search for Text on: ⓘ

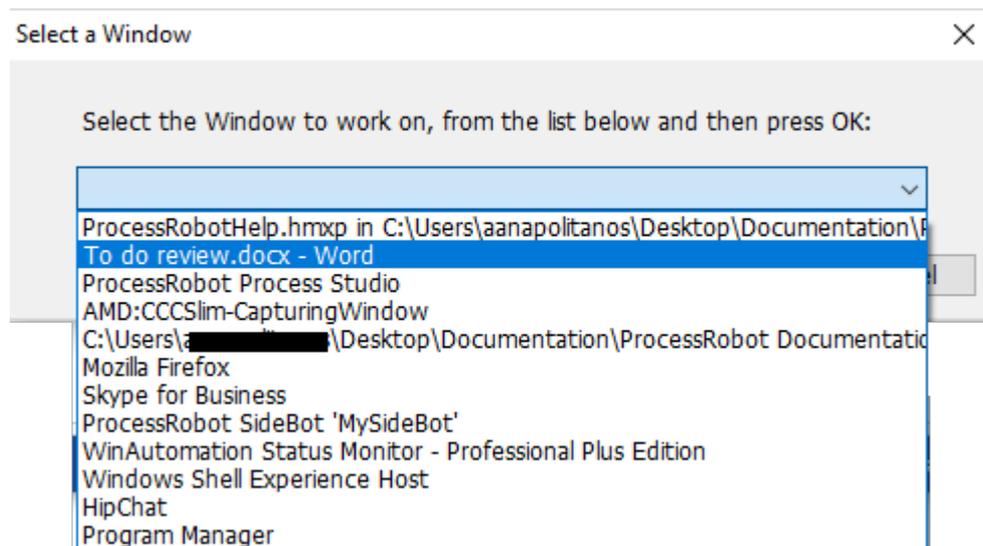
Search Mode: ⓘ

X1: ⓘ X2: ⓘ

Y1: ⓘ Y2: ⓘ

ⓘ

By clicking the "Select a Different Subregion" button on the right of the red box, you will be prompted to select firstly the window you want to work:



and then you will be given the opportunity through an amazing interactive feature to actually select the area you want to narrow down your scan (drag your mouse to select the area). The X1 X2 Y1 Y2 co-ordinates are denoting the start and finish of your dragging manoeuvre.

However, using the "Select a Different Subregion" button is not compulsory. You can enter the values of the co-ordinates by hand or as a Variable (notice the Gear Icon next to the text fields).

-Subregion relative to image:

Search for Text on: ⓘ

Search Mode: ⓘ

Image:

Add Image From Images Repository ⓘ

Delete Image From List ⓘ

Tolerance: ⓘ

Region relative to image: X1: ⓘ X2: ⓘ

Y1: ⓘ Y2: ⓘ

This search mode enables you to narrow down your scan through a region that is defined relative to the top-left corner of an image from the Image Repository.

7

Fail if Text does not appear within:

This checkbox allows you to produce an exception (which you can in turn handle through the state of the art [set](#)^[510] of [exception handling mechanisms](#)^[502] PR is providing you with) in case the specified Text appear or does not appear within a custom set of seconds to be defined on [8].

8

Fail if Text does not appear within specified number of seconds:

The maximum number of seconds you can set in this box is 600.

9

Location of Text found:

This action returns the coordinates of the top-left corner of the text found on screen or a specified window, stored in a couple of variables for later use.

3.11.4.9 Wait for Mouse Action

Description:

This action suspends the execution of the Process until the mouse pointer changes, usually to or from the 'wait cursor' or hourglass.

Properties:

Wait for mouse pointer to:

Choose what action of the mouse cursor you will wait for.

Continue Anyway:

Choose whether the Process continues after a set number of seconds, regardless of whether the mouse pointer has changed or not.

3.11.4.10 Wait for Window Content Action

Description:

Waits until a specific text or Element appears or disappears from a Window.

This action operates on a UI Automation Window Instance that has been previously acquired by a "[Get Window](#)" action. The latter stores the Window instance into a variable. This action accepts this variable and waits until the supplied Window Instance contains or does not contain a specific text or Element.

Option 1: Wait for UI Element (Contained/Not Contained in Window)

The screenshot shows the configuration window for the 'Wait for Window Content' action. The window title is 'Properties of 'Wait for Window Content' action'. The main heading is 'Wait for Window Content' with a sub-description: 'Waits until a specific text or Element appears or disappears from a Window.' Below this, there are three tabs: 'General', 'Advanced', and 'Exception Handling', with 'General' selected. The 'Action Input' section contains a 'Wait Till Window:' dropdown menu set to 'Contains Element', a checkbox for 'Wait for Element to become:' which is unchecked, and a dropdown menu set to 'Enabled'. The 'Control:' section has a text box containing '(no control selected)' and a 'Select Control From Repository' button. The 'Action Output' section is empty and contains the text '(This action does not provide any output)'. At the bottom, there is a checkbox 'This action is Enabled' which is checked, and buttons for 'More Info', 'OK', and 'Cancel'.

Option 2: Wait for Text (Exists/Does not Exist)

This option waits until the specifies text appears or disappears from the Window Contents.

Properties:

Window Instance:

Enter the variable that contains the parent Window Instance of the Element or text you are waiting for. This must be a variable defined by a preceding ["Get Window"](#) action or a valid Window Handle.

Wait Till Window:

Specify whether you want to wait for a UI element or a specific text.

Wait for Element to Become:

This property allows you to wait until the target element is present and in a specific state (Enabled or Disabled). The Wait statement will only stop waiting when the target element is both present and in the required state, if this property is checked.

Text to Wait for:

Enter text you want to wait for.

Description for Element:

Enter a short description for the Window Element you want to wait for. This is optional, meant for documentation purposes and when possible it is auto-populated along with the "UI Selector" property, during the [UI Element Live Selection](#)^[338].

UI Selector of Element:

Enter the UI Selector of the Window Element you want to check for. You do not need to manually enter a value here; this field will be populated automatically when you select an element on any Window, by right-clicking on it while having the action's properties dialog open.

Fail On Timeout:

Choose whether the action will throw an exception after a set timeout of waiting. Then go to the Exception Handling tab to specify how the Exception will be handled, otherwise the Process will stop and issue an error message.

3.11.4.11 Wait for Web Page Content Action

Description:

Waits until a specific text or Element appears or disappears from a Web Page.

This action operates on a Web Page that has been previously opened by a "[Launch New Internet Explorer](#)^[728]" action. The latter stores the Web browser instance into a variable. This action accepts this variable and waits until the page is currently displayed in that Web Browser Instance contains or does not contain a specific element or text.

Option 1: Wait for Web Element (Contained/Not Contained in Page)

Properties of 'Wait for Web Page Content' action

Wait for Web Page Content
Waits until a specific text or Element appears or disappears from a Web Page.

General Exception Handling

Action Input

Web Browser Instance: %InternetExplorer% 

Wait for Web Page to: Contain Element 

Control:
(no control selected)  

Select Control From Repository

Fail On Timeout 

Action Output
(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Option 2: Wait for Text (Exists/Does not Exist)

Properties of 'Wait for Web Page Content' action

Wait for Web Page Content
Waits until a specific text or Element appears or disappears from a Web Page.

General Exception Handling

Action Input

Web Browser Instance: %InternetExplorer%

Wait for Web Page to: Contain Text

Text to Wait for:

Fail On Timeout

Action Output

(This action does not provide any output)

This action is Enabled

More Info OK Cancel

This option waits until the specifies text appears or disappears from the Web page.

Properties:

Web Browser Instance:

Enter the variable that contains the Web Browser Instance you want to work with. This must be a variable defined by a preceding "[Launch New Internet Explorer](#)"⁷²⁶ action.

Wait for Web Page to:

Specify whether you want to wait for an HTML element or a specific text.

Text to Wait for:

Enter text you want to wait for.

Fail On Timeout:

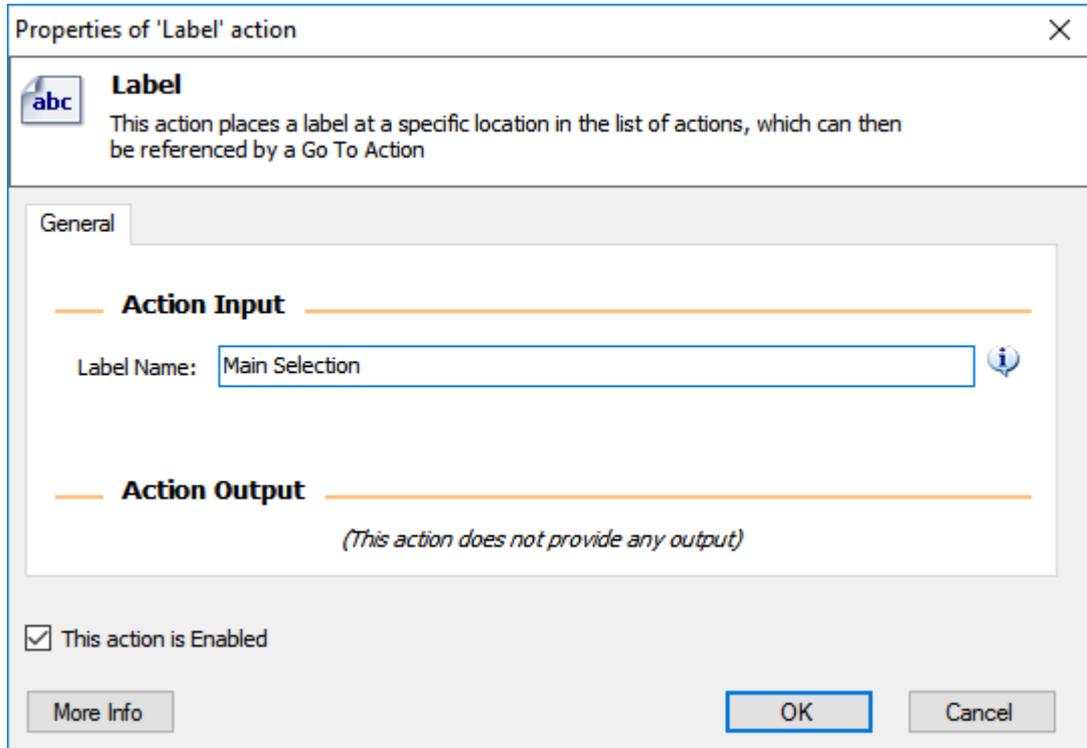
Choose whether the action will throw an exception after a set timeout of waiting. Then go to the Exception Handling tab to specify how the Exception will be handled, otherwise the Process will stop and issue an error message.

3.11.5 Flow Control

3.11.5.1 Label Action

Description:

This action places a label at a specific Process location that can be referenced by a Go To Action



The screenshot shows a dialog box titled "Properties of 'Label' action". At the top, there is a tab labeled "Label" with a description: "This action places a label at a specific location in the list of actions, which can then be referenced by a Go To Action". Below this is a "General" tab. Under "Action Input", there is a text field for "Label Name" containing the text "Main Selection". Under "Action Output", it states "(This action does not provide any output)". At the bottom, there is a checkbox labeled "This action is Enabled" which is checked. There are three buttons at the bottom: "More Info", "OK", and "Cancel".

Properties:

Label Name:

Enter a Label, to be used by a Go To action, to find at a later point in the Process.

3.11.5.2 Go To Action

Description:

This action jumps the execution flow to the specified label

Properties of 'Go To' action

Go To
This action jumps the execution flow to the specified label

General

Action Input

Label Name: ⓘ

Action Output

(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties:

Label Name:

Enter an existing Label Name that this action will find and jump to.

3.11.5.3 Run Function Action

Description:

This action turns the execution flow to the specified function.

Properties of 'Run Function' action

Run Function
This action turns the execution flow to the specified function

General

Action Input

Function Name: Calculate List Sum

Action Output
(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties:

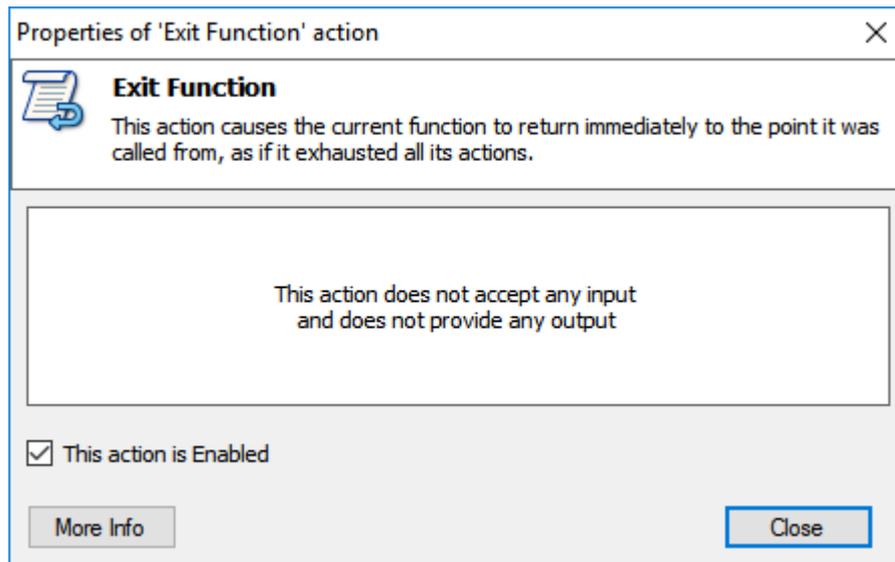
Function Name:

Enter the Name of the function to run.

3.11.5.4 Exit Function Action

Description:

This action causes the current function to return immediately to the point it was called from, as if it exhausted all its actions.



Properties:

This action does not have any configurable properties.

3.11.5.5 Stop Process Action

Description:

This action stops the execution of the Process

Properties:

End Process:

Choose whether this action ends the Process with an error message or not.

Error Message:

Enter text to be a descriptive error message explaining why the Process was ended.

Exit Code:

Enter a number to be the exit code of the Process. This applies only to Processes that are compiled to exe; for Processes that run within ProcessRobot this setting will have no effect.

3.11.5.6 Begin Exception Block Action

Description:

This action sets the beginning of an exception handling region

Properties of 'Begin Exception Block' action

Begin Exception Block
This action sets the beginning of an exception handling region.

General

Action Input

Name:

Set Variable:

Set Value:

into Variable:

Run Function:

Continue Execution:

Action Output

(This action does not provide any output)

This action is Enabled

Properties:**Name:**

Enter the name of the Exception Block for virtual purposes only.

Set Variable - Set Value:

Enter the value to be set into the variable when exception in any action occurs.

Set Variable - into Variable:

Enter the variable into which the value should be set when exception in any action occurs.

Run Function:

Set a function to run when exception in any action occurs.

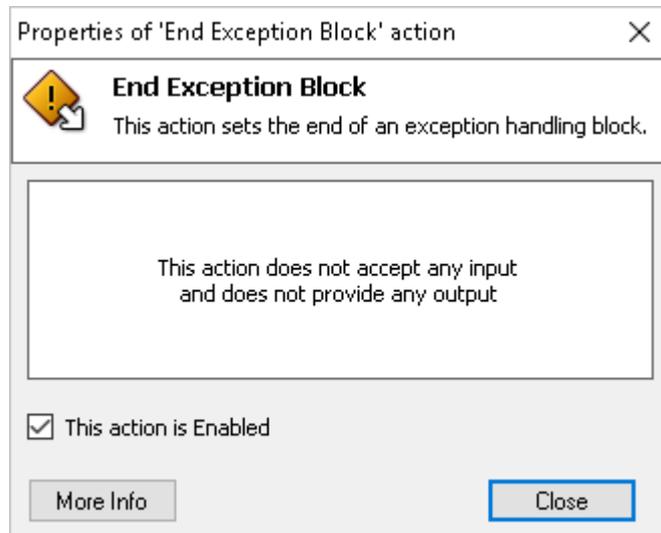
Continue Execution:

Set to continue the robot execution with a selected command when exception in any action occurs.

3.11.5.7 End Exception Block Action

Description:

This action sets the end of an exception handling block



Properties:

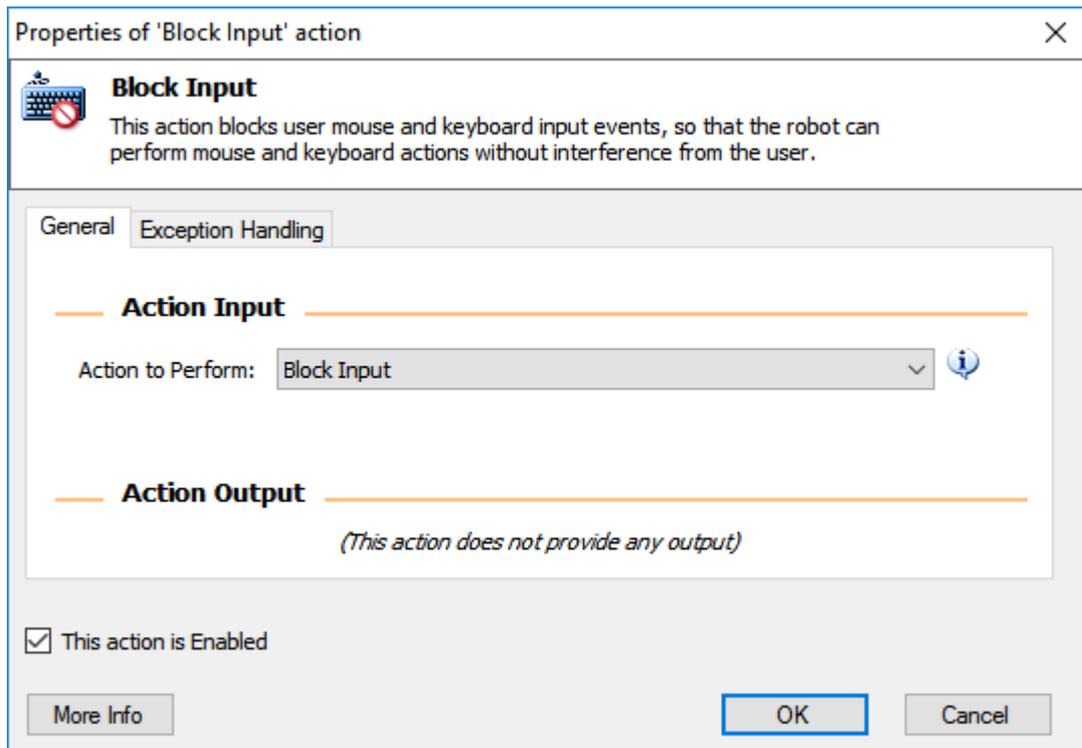
This action does not have any configurable properties.

3.11.6 Mouse and Keyboard

3.11.6.1 Block Input Action

Description:

This action blocks user mouse and keyboard input events, so the Process can perform mouse and keyboard actions without interference from the user.



Properties:

Action to Perform:

Choose whether this action will block or unblock Mouse and Keyboard input.

Cautions:

- If your Process fails while your mouse and keyboard input is locked, your control will not be unlocked. To regain control, press ctrl + alt + delete on your keyboard. This works even if the Process has not failed.
- For this Action to function correctly, the Process must be run with Administrator privileges.

3.11.6.2 Get Mouse Position Action

Description:

Retrieves the current position of the mouse cursor on the screen in pixel coordinates.

Properties:

Get Position Relative to:

Specify whether to retrieve the mouse position in screen coordinates or you want the mouse position relative to the top left corner of the active window.

Store Mouse Position X value into:

Enter a name to be the variable that will store the horizontal (X) value of the mouse position.

Store Mouse Position Y value into:

Enter a name to be the variable that will store the vertical (Y) value of the mouse position.

3.11.6.3 Move Mouse Action

Description:

This action moves the mouse to a specific position

Properties of 'Move Mouse' action

Move Mouse
This action moves the mouse to a specific position

General | Exception Handling

Action Input

Move Mouse to: X:  

Y: 

Relative to: 

Move Mouse From Previous Position: 

Current cursor position (Relative to screen): X=3401, Y=581

Press CONTROL KEY + SHIFT KEY to grab current cursor position
or hold down CONTROL KEY + SHIFT KEY to record mouse move

Action Output

(This action does not provide any output)

This action is Enabled

Properties:

Move Mouse to:

Set the position of the mouse in X and Y pixel coordinates. Press CONTROL + SHIFT keys if you wish to grab the current cursor position, or hold them both down to record a mouse movement.

Relative to:

Choose whether the new Mouse position will be relative to the top left corner of the screen, or of the foremost Window, or relative to the current mouse position.

Move Mouse From Previous Position:

Choose the style of movement in which the mouse will move from it's previous position to the beginning of the recorded route (or to it's final position). The mouse can move instantly (jump) to the new position or navigate smoothly at three different speeds (slow/normal/fast).

3.11.6.4 Move Mouse to Image Action

Description:

This action moves the mouse over an image found on the screen or on the foreground window. This action uses the [Image Recognition](#) ⁵⁰⁶ feature.

Properties of 'Move Mouse to Image' action

Move Mouse to Image
This action moves the mouse over an image found on the screen or on the foreground window

General | Advanced | Exception Handling

Action Input

Image to move mouse to: Image

Add Image From Images Repository

Delete Image From List

Search for Image on: Entire Screen

Search Mode: Search on specified subregion of Screen or Foreground Window

X1: 1665 X2: 2114

Y1: 0 Y2: 421

Mouse position relative to image:

Offset X: 0

Offset Y: 0

Tolerance: 10

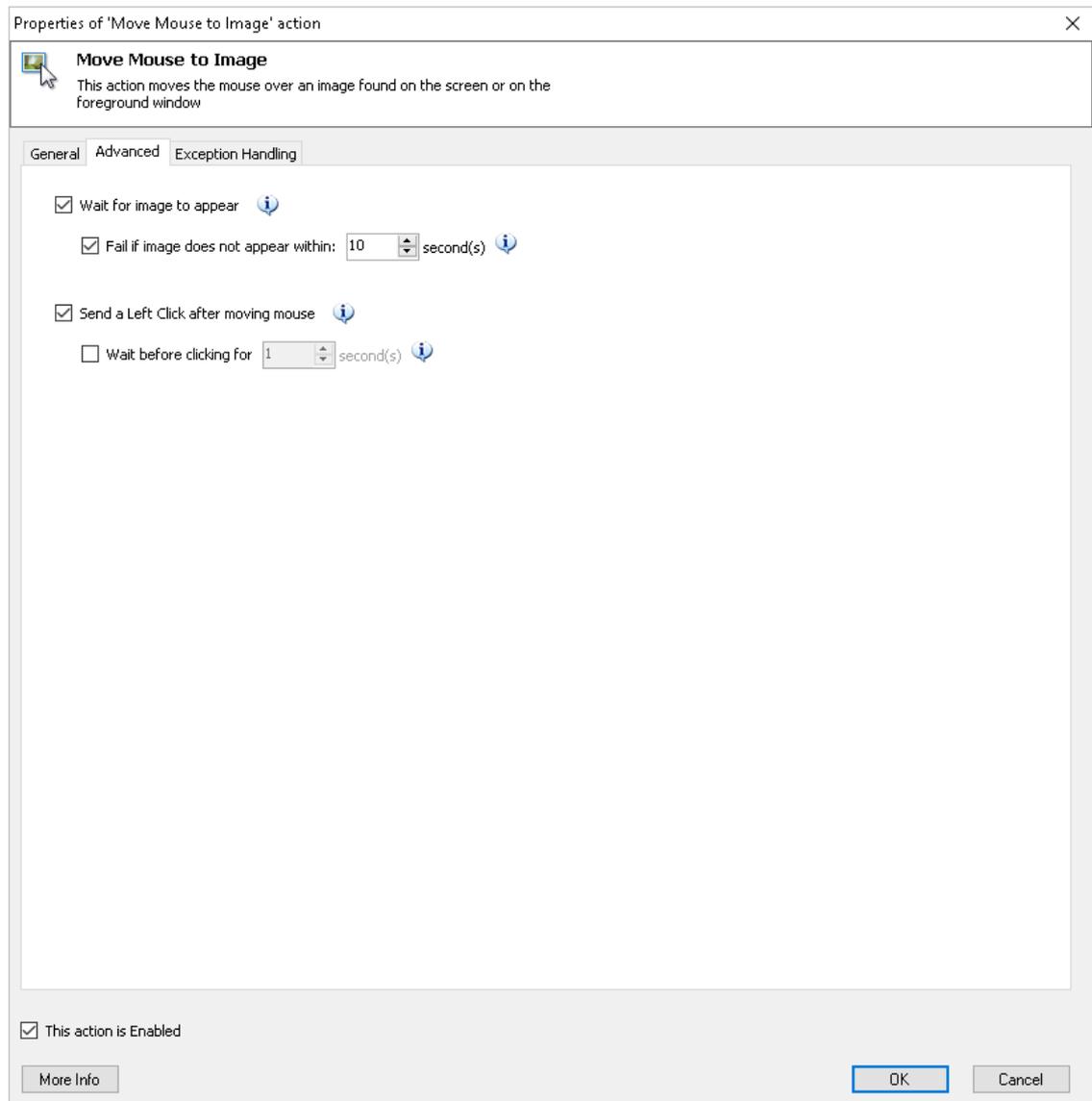
Move Mouse From Previous Position: Instantly

Action Output

Location of Image found: X: %ImageFoundAtX% Y: %ImageFoundAtY%

This action is Enabled

More Info OK Cancel



Properties:

Image to move mouse on:

This is the Image that the action will move the mouse to. Press 'Capture Image' to select the image. If you need to open a menu or perform steps first, press 'Capture with Delay'.

Search for Image on:

Choose whether you want to search for the specified Image in the foremost Window only, or the entire visible screen. Neither choice will find the Image if it is not clearly visible on the screen.

Search Mode:

Specify whether you want to scan the entire screen (or window) to find the supplied image or only a narrowed down subregion of it.

X1, Y1, X2, Y2:

Specify the subregion of the screen or window to narrow down the scan for the supplied image to. You can visually select a subregion of the screen or window by pressing the "Select a Different Subregion" button.

Mouse position relative to image:

Choose which section of the image the mouse will be moved to.

Offset X:

Offset the mouse from the Position by this many pixels to the right.

Offset Y:

Offset the mouse from the Position by this many pixels down.

Tolerance:

Specify a value for how much the Image searched for can differ from the originally chosen Image.

Move Mouse From Previous Position:

Choose the style of movement in which the mouse will move from it's current position to the position of the image found. The mouse can move instantly (jump) to the new position or navigate smoothly at three different speeds (slow/normal/fast).

Location of Image Found:

Enter the names to be the variables that will hold the X and Y coordinate of the point where the image is found on the screen. If the image is being searched for on the foreground window, the coordinates returned are relative to the top left corner of the window.

Wait for image to appear:

Choose whether you want the action to wait if the image is not found on the screen or foreground window. If this property is not checked and the image is not found the action will through an exception, otherwise the action will wait until the image appears. In the next property you can specify the maximum number of seconds (timeout) to wait.

Fail if image does not appear within x seconds:

Choose whether the action wait indefinitely for the image to appear or it will throw an exception after a set number of seconds if the specified Image has not been found. You can then go to the Exception Handling tab at top to specify how the Exception will be handled, otherwise, if the image is not found within the timeout, the Process will stop and issue an error message.

Send a Left Click after moving mouse:

Choose whether you want the action to send a left click after the mouse is positioned over the image.

3.11.6.5 Move Mouse to Text on Screen (OCR)

This action allows you to move the mouse over a specific image some specific text [2] found on screen or a foreground window [5] using an OCR Engine of your choice [1]. If this text is found more than once on that surface, you can target a specific occurrence [4] (2nd, 3rd etc.) given the fact that the OCR is adding these matches first to last, top to bottom, left to right. Just like in other actions that use OCR, you can work with a variety of options that allow you to search on very specific subregions of your screen [6], but the action is also giving you option to select the speed with which the mouse [7] will be moved on the spot.

The action returns co-ordinates [8] of where the text is found relative to the top left of your screen or the top left corner of the window you are working with. Along the position of the text it returns the width and height of that image in two [9] corresponding Variables.

Properties of 'Move Mouse to Text on Screen (OCR)' action

Move Mouse to Text on Screen (OCR)
This action moves the mouse over a text found on the screen or on the foreground window using OCR.

General | Advanced | Exception Handling

Action Input

1 OCR Engine: %OCREngine%

2 Text to Find: Control Desk

3 Is Regular Expression

4 Occurrence: 2

5 Search for Text on: Entire Screen

6 Search Mode: Search whole Screen or Foreground Window

7 Move Mouse From Previous Position: With Animation (Natural Speed)

Action Output

8 Location of Image found:

X: %LocationOfTextFoundX%

Y: %LocationOfTextFoundY%

9 Width: %WidthOfTextFound%

Height: %HeightOfTextFound%

This action is Enabled

More Info OK Cancel

General Tab

1

OCR Engine:

This text field with drop down menu options invites you to enter *or choose* the instance of the OCR Engine you want to work with.

2

Text to Find:

This text field allows you to enter the text you want the OCR Engine to search for as text or as a Variable.

3

Is Regular Expression:

Check this check box if you want to use a Regular Expression to find the text on screen. A Regular Expression creates a range of possibilities and can return a number of results that match your search. The order that the Engine is adding these matches is from top to bottom, left to right as it finds them on target!

4

Occurence:

This positive integer should point to the match that interests you, if your search returns more than one occurrences or a collection of values (through a regular expression for example).

5

Search for Text on:

Two options on this drop down menu, *Entire Screen* or *Foreground Window only*.

6

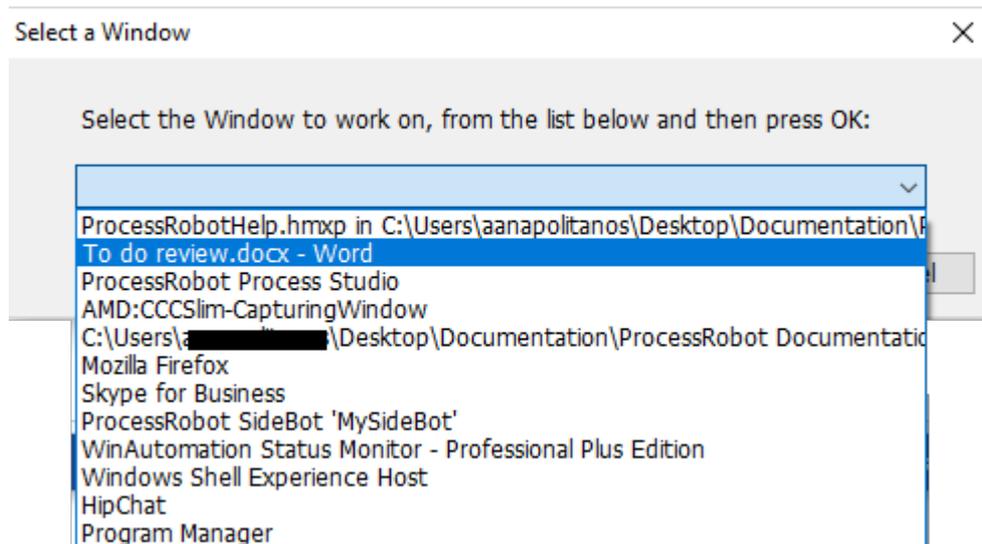
Search Mode:

While the first option of this drop down menu is pretty straightforward (Search Whole Screen or Foreground Window) the other two options that allow you to target a specific subregion of your window or screen produce a whole different set of additional Properties which will see in detail here:

-Search on specified subregion of Screen or Foreground window:

The screenshot shows a search configuration interface. At the top, there is a dropdown menu labeled "Search for Text on:" with the value "Foreground Window only" and an information icon. Below it, another dropdown menu labeled "Search Mode:" has the value "Search on specified subregion of Screen or Foreground Window" and an information icon. Underneath, there are four input fields for coordinates: "X1:" with the value "609", "X2:" with "1003", "Y1:" with "377", and "Y2:" with "554". Each coordinate field has a gear icon to its right. To the right of these fields is a button labeled "Select a Different Subregion" with an information icon.

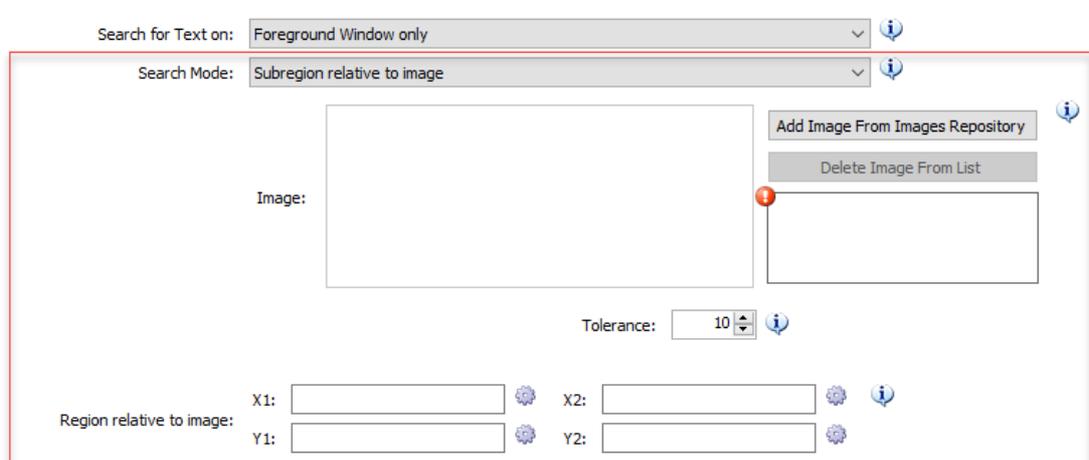
By clicking the "Select a Different Subregion" button on the right of the red box, you will be prompted to select firstly the window you want to work:



and then you will be given the opportunity through an amazing interactive feature to actually select the area you want to narrow down your scan (drag your mouse to select the area). The X1 X2 Y1 Y2 co-ordinates are denoting the start and finish of your dragging manoeuvre.

However, using the "Select a Different Subregion" button is not compulsory. You can enter the values of the co-ordinates by hand or as a Variable (notice the Gear Icon next to the text fields).

-Subregion relative to image:



This search mode enables you to narrow down your scan through a region that is defined relative to the top-left corner of an image from the Image Repository.

7

Move Mouse From Previous Position:

This drop down menu gives you a number of options in case the speed with which the mouse will move is critical to your process.

8

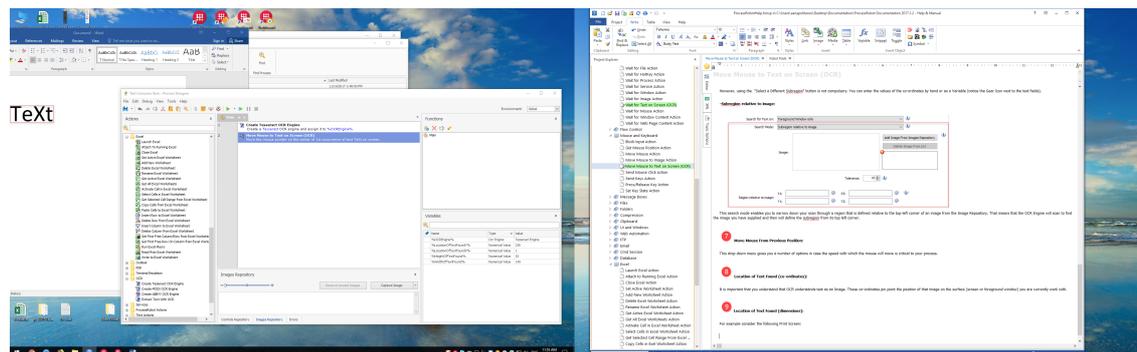
Location of Text Found (co-ordinates):

It is important that you understand that OCR understands text as an Image. These co-ordinates pin point the position of that image on the surface (screen or foreground window) you are currently work with.

9

Location of Text Found (dimensions):

For example consider the following Print Screen:



If we run the process shown on Robot Designer, which is designed to move the mouse on the center of the image text "TeXt", the co-ordinates we will get for that image are (1, 336). This makes sense since the "TeXt" text on Word is positioned at the very left ($\%LocationOfTextFoundX\% = 1$) of our screen!

The width (145) and height (52) values also make sense, accurately describing the image the OCR Engine has found.

3.11.6.6 Send Mouse Click Action

Description:

This action sends a mouse click event

Properties of 'Send Mouse Click' action

Send Mouse Click
This action sends a mouse click event

General Exception Handling

Action Input

Mouse Event to Send: Left Click

Wait 0 milliseconds before sending mouse event

Move mouse before sending mouse event

Move Mouse to: X: 3330
Y: 491

Relative to: Screen

Move Mouse From Previous Position: Instantly

Current cursor position (Relative to screen): X=3330, Y=491
Press CONTROL KEY + SHIFT KEY to grab the current cursor position

Action Output

(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties:

Move Event to Send:

Choose what form of mouse event to send.

Wait:

Set a time for the action to delay before sending the mouse event, in 1/1000 of a second. In other words, 1000 means one second, 1500 means one and a half seconds, 2250 means two and a quarter seconds and so on.

Move Mouse to:

Set the position of the mouse in X and Y pixel coordinates. Press CONTROL + SHIFT keys if you wish to grab the current cursor position.

Relative to:

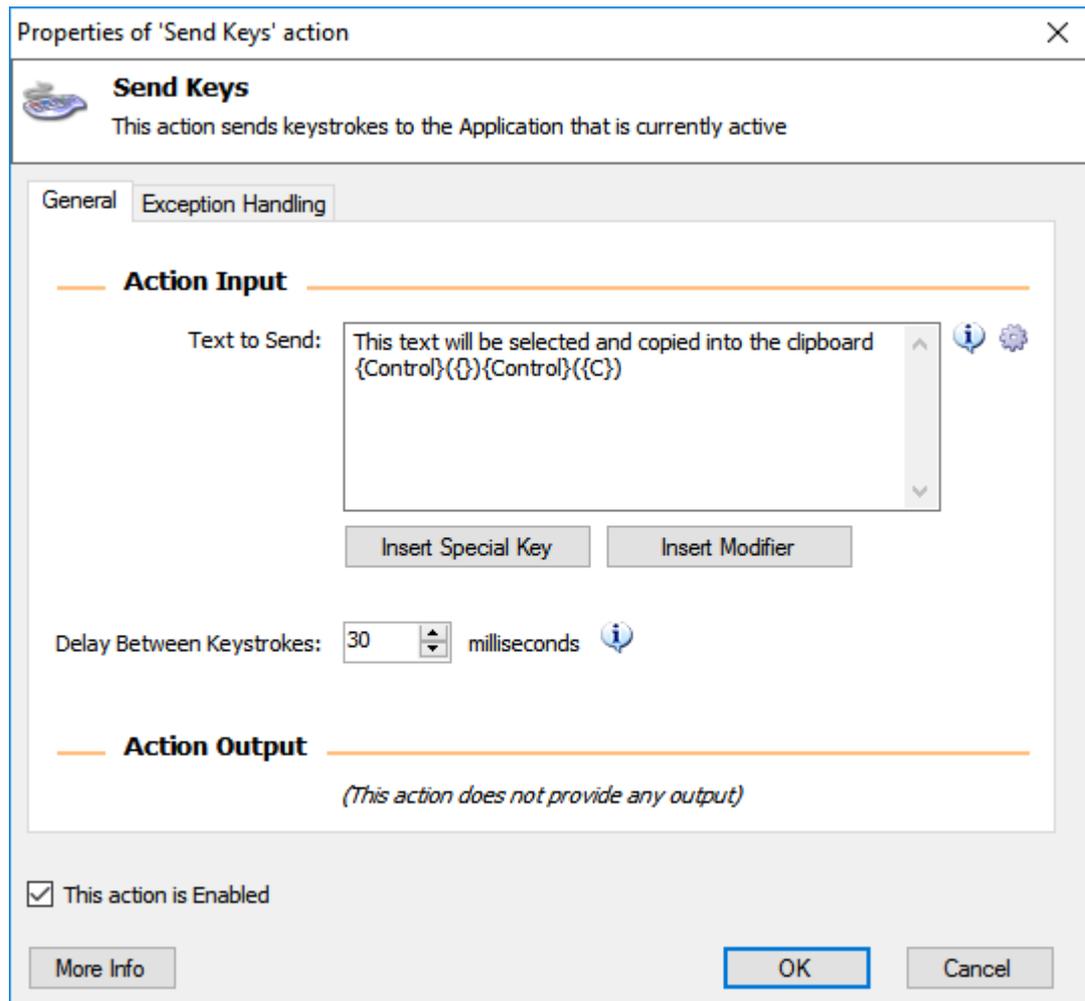
Choose whether the new Mouse position will be relative to the top left corner of the screen, or to the foremost Window, or to the current mouse position.

Move mouse from previous position:

Choose the style of movement in which the mouse will move from its previous position to the beginning of the recorded route (or to its final position)

3.11.6.7 Send Keys Action***Description:***

This action sends keystrokes to the Application that is currently active



Properties:

Text to Send:

Insert keystrokes to send as text. Special key and modifiers can be accessed below the box. Control is under Insert Modifier, and Control-a, for example, needs to appear in the form {Control}{A}. Any Key, as compared to a character, must be written here in braces and capitalized, such as {A}. Within Modifiers, Keys must be used, not characters like 'a' or '\$'. For a complete list of the Key Codes corresponding to the keyboard keys, please consult the topic "[Valid Key Codes for the Send Keys Action](#)".

Delay Between Keystrokes:

Choose a time to delay between keystrokes to avoid input errors.

3.11.6.8 Press/Release Key Action

Description:

This action presses (and holds) or releases one or more modifier keys (Alt, Control, or Shift). This is useful mainly taking steps that call for a key to be pressed while operating the mouse, and requires another use of this action to release the key(s) being held.

Properties of 'Press/Release Key' action

Press/Release Key
This action presses (and holds) or releases one or more modifier keys (Alt, Control, or Shift). This is useful mainly taking steps that call for a key to be pressed while operating the mouse, and requires another use of this action to release the key(s) being held.

General | Exception Handling

Action Input

Action to perform: Press the following key(s) (dropdown menu)
 Control
 Alt
 Shift
 Win

Action Output
(This action does not provide any output)

This action is Enabled

More Info | OK | Cancel

Properties:

Action to Perform:

Choose whether to Press or Release keys with this action.

Control:

Choose whether the CTRL key is pressed/released or not.

Alt:

Choose whether the Alt key is pressed/released or not.

Shift:

Choose whether the Shift key is pressed/released or not.

Win:

Choose whether the Windows key is pressed/released or not.

3.11.6.9 Set Key State Action

Description:

This action sets the state (on or off) for the keys CAPS LOCK, NUM LOCK or SCROLL LOCK.

Properties of 'Set Key State' action

Set Key State
This action sets the state (on or off) for the keys CAPS LOCK, NUM LOCK, or SCROLL LOCK.

General | Exception Handling

Action Input

Set Key: CAPS LOCK to: Off

Action Output
(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties:**Set Key to:**

Choose the key you wish to set, and select 'On' or 'Off'.

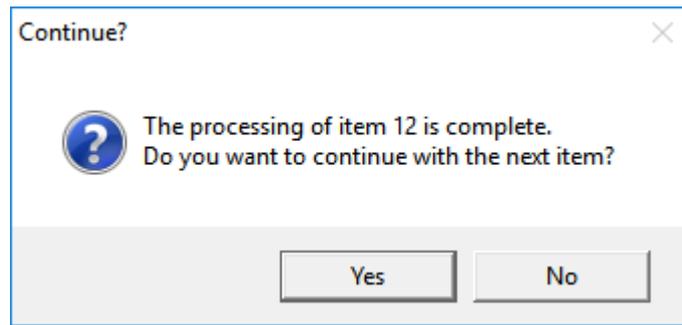
3.11.7 Message Boxes

3.11.7.1 Display Message Action

Description:

This action displays a message box

The screenshot shows the configuration window for the 'Display Message' action. The window title is 'Properties of Display Message' and it has a close button (X) in the top right corner. The main title is 'Display Message' with a sub-description: 'This action displays a message box'. There are two tabs: 'General' (selected) and 'Exception Handling'. The 'General' tab is divided into two sections: 'Action Input' and 'Action Output'.
Action Input:
- Message Box Title: 'Continue?'
- Message to Display: 'The processing of item %ItemNumber% is complete. Do you want to continue with the next item?'
- Message Box Icon: 'Question'
- Message Box Buttons: 'Yes - No'
- Default Button: 'First Button'
- Checkboxes: 'Keep Message Box Always on Top' and 'Close Message Box Automatically' are both checked.
- After: '10' seconds.
Action Output:
- Store Button Pressed into: '%ButtonPressed%'.
At the bottom, there is a checkbox 'This action is Enabled' which is checked. There are three buttons: 'More Info', 'OK', and 'Cancel'.



The Message Dialog in action

Properties:

Message Box Title:

Enter the text, or a previously defined variable, to be used as the Message Box title.

Message To Display:

Enter the text, or a previously defined variable, to be displayed as the actual message.

Message Box Icon:

Choose which Icon to display with the Message Box.

Message Box Buttons:

Choose which buttons to display on the Message Box.

Default Button:

Choose which button is highlighted by default. If the user presses {Enter}, this button will be pressed.

Keep Message Box Always on Top:

Choose whether the Message Box should always remain on top of all other windows.

Close Message Box Automatically:

Choose whether Message Box closes automatically after a preset time, as if the default button was pressed. Otherwise, the Process will wait until a button is pressed by the user.

Timeout:

Choose how many seconds to pause the Process while waiting for input, until continuing automatically.

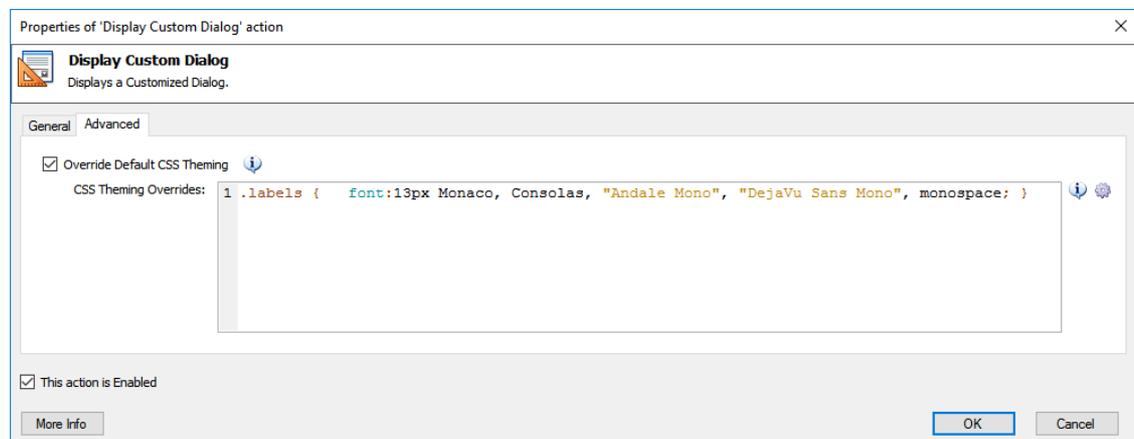
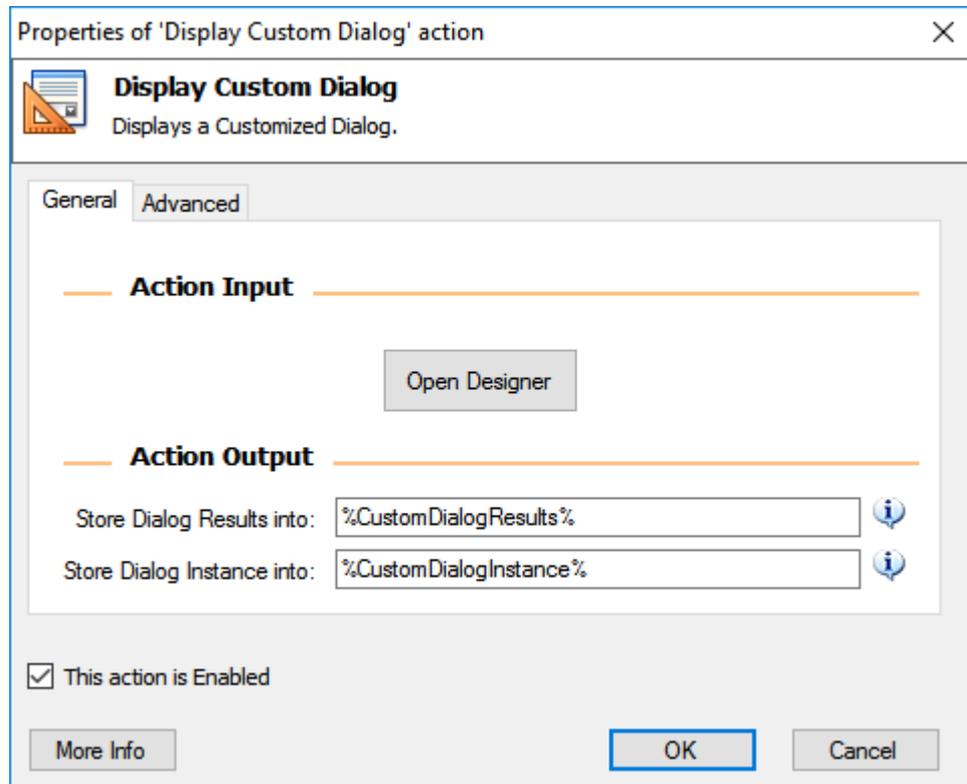
Store Button Pressed into:

Enter a name to be the variable that will store the text of the button pressed.

3.11.7.2 Display Custom Dialog Action

Description:

This action displays a Customized Dialog.



Properties:

"Open Designer" Button:

Pressing this button will open up the Custom Dialog Designer window. You will find more details regarding the Custom Dialog Designer in the section "[Designing a Custom Dialog](#)".

Store Dialog Results into:

Enter a name to be the variable that will store the values populated in the Custom Dialog.

Store Dialog Instance into:

Enter a name to be the variable that will store the specific Custom Dialog instance for use with later UI Automation actions.

Advanced Properties Tab - Override Default CSS Theming:

Specify whether you would like to override the default theming of this Custom Dialog with your own CSS rules.

Advanced Properties Tab - CSS Theming Overrides:

Enter the CSS rules you desire to be applied over the default theming of this Custom Dialog.

Control Specific Tips:

Date Control:

Be forewarned that the formatting flags used by the Date Control inside the Custom Dialog are different from the flags used for converting text-to-datetime by means of the 'Convert Text to DateTime' action. Having said this, the date-format field in the properties of the Date Control, supports the following flags:

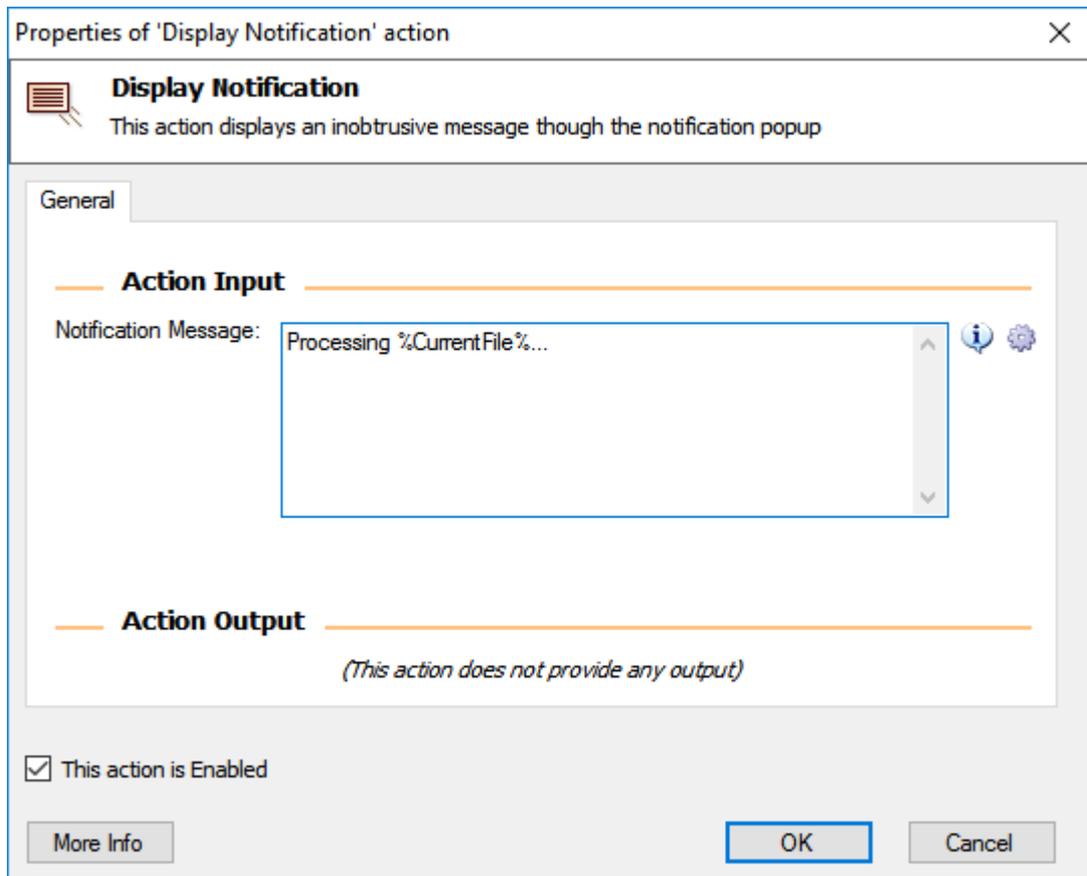
- d - day of month (no leading zero)
- dd - day of month (two digit)
- o - day of year (no leading zeros)
- oo - day of year (three digit)
- D - day name short
- DD - day name long
- m - month of year (no leading zero)
- mm - month of year (two digit)
- M - month name short
- MM - month name long
- y - year (two digit)
- yy - year (four digit)
- @ - Unix timestamp (ms since 01/01/1970)
- ! - Windows ticks (100ns since 01/01/0001)
- '...' - literal text
- " - single quote
- anything else - literal text

For example in order to parse a date in ISO format like "2007-01-26" one would have to set the date-format field of the Date Control to the pattern "yy-mm-dd".

3.11.7.3 Display Notification Action

Description:

This action displays an inobtrusive message through the [notification pop-up](#)^[284]



Properties:

Notification Message:

Enter the text, or a previously defined text variable, to be displayed in the Notification Popup while the Process is being run from the ProcessRobot Console Window (not Process Designer) or as a Console Application if the Process has been converted to .exe.

3.11.7.4 Display Input Dialog Action

Description:

This action displays a dialog box that prompts the user to enter text

Properties of 'Display Input Dialog' action

Display Input Dialog
This action displays a dialog box that prompts the user to enter text

General Exception Handling

Action Input

Input Dialog Title: ⓘ ⚙️

Input Dialog Message: ⓘ ⚙️

Default Value: ⓘ ⚙️

Input Type: ⓘ

Keep Input Dialog Always on Top ⓘ

Action Output

Store User Input into: ⓘ

Store Button Pressed into: ⓘ

This action is Enabled

What is your age?

Hello User, please, enter your age:

The Input Dialog in action

Properties:

Input Dialog Title:

Enter the text, or a previously defined text variable, to be used as the Dialog title.

Input Dialog Message:

Enter the text, or a previously defined variable, to be displayed as the actual message.

Default Value:

Enter the text to be displayed by default. If the user wishes to change this, they can type over it. Otherwise, the default text will be used.

Input Type:

Choose the format for the input text. Choose Single Line - Password if you wish to hide the text, or Multi Line if you want a larger display box to make visible more than one line of text. Otherwise choose Single Line.

Keep Input Dialog Always on Top:

Choose whether the Input Dialog should always remain on top of all other windows.

Store User Input into:

Enter a name to be the variable that will store the text entered by the user, or the default text.

Store Button Pressed into:

The user will automatically be given the choice of OK or Cancel. Enter a name to be the variable that will store the text of the button pressed.

3.11.7.5 Display Select Date Dialog***Description:***

This action displays a dialog box that prompts the user to enter a date or a date range

Properties of 'Display Select Date Dialog' action

Display Select Date Dialog
This action displays a dialog box that prompts the user to enter a date or a date range

General Exception Handling

Action Input

Dialog Title: Enter Date Range

Dialog Message: Please, choose the date range for the report you want to generate

Dialog Type: Date Range (2 Dates)

Prompt for: Date only

Default Value: 1/1/2017

Default Value for Second Date: %CurrentDateTime%

Keep Date Selection Dialog Always on Top

Action Output

Store Selected Date into: %SelectedDate%

Store Second Selected Date into: %SecondSelectedDate%

Store Button Pressed into: %ButtonPressed3%

This action is Enabled

More Info OK Cancel

Enter Date Range

Please, choose the date range for the report you want to generate

Sunday . January 1, 2017

Tuesday . March 21, 2017

OK Cancel

The Select Date Dialog in action

Properties:

Dialog Title:

Enter the text, or a previously defined text variable, to be used as the Dialog title.

Dialog Message:

Enter the text, or a previously defined variable, to be displayed as the actual message.

Dialog Type:

Choose whether the user will enter a single date or two dates to be the endpoints of a range of dates.

Prompt for:

Choose whether the user will enter the date only or the date and time.

Default Value:

Enter a DateTime value, or a previously defined DateTime variable, to be used as the default value.

Keep Date Selection Dialog Always on Top:

Choose whether the Date Selection Dialog should always remain on top of all other windows.

Default Value for Second Date:

Enter a DateTime value, or a previously defined DateTime variable, to be used as the default value for the end date in a range.

Store Selected Date into:

Enter a name to be the variable that will store the date entered by the user or the default date.

Store Second Selected Date into:

Enter a name to be the variable that will store the second date entered by the user or that default date.

Store Button Pressed into:

The user will automatically be given the choice of OK or Cancel. Enter a name to be the variable that will store the text of the button pressed by the user.

3.11.7.6 Display Select From List Dialog Action

Description:

This action displays a dialog box with options that lets the user select from a list

Properties of 'Display Select from List Dialog' action ✕

 **Display Select from List Dialog**
This action displays a dialog box with options that lets the user select from a list

General Exception Handling

Action Input

Dialog Title:  

Dialog Message:  

List to choose from:  

Keep Select Dialog Always on Top 

Limit to List 

Allow Empty Selection 

Allow Multiple Selection 

Preselect Items Starting With a + Sign 

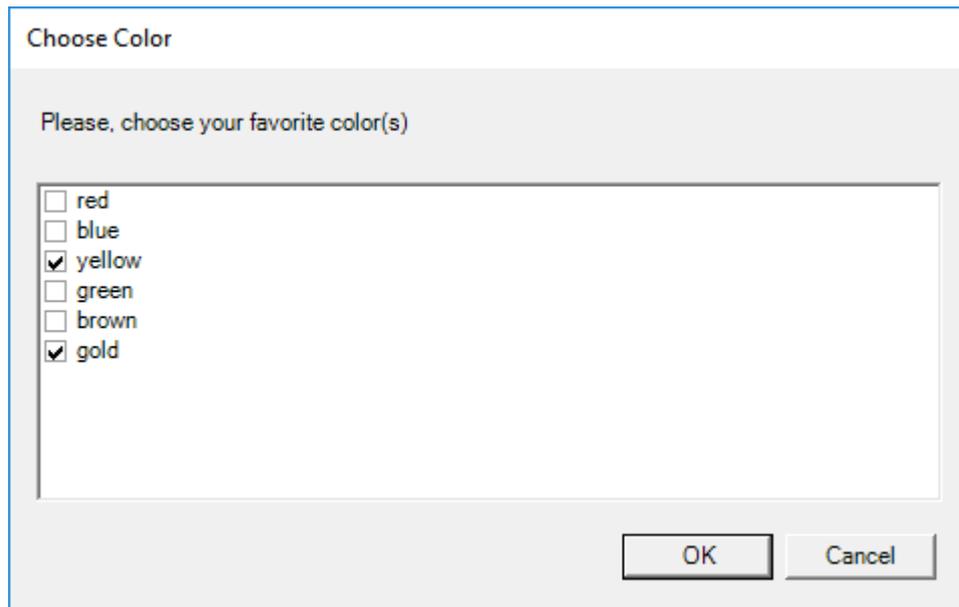
Action Output

Store Selected Item into: 

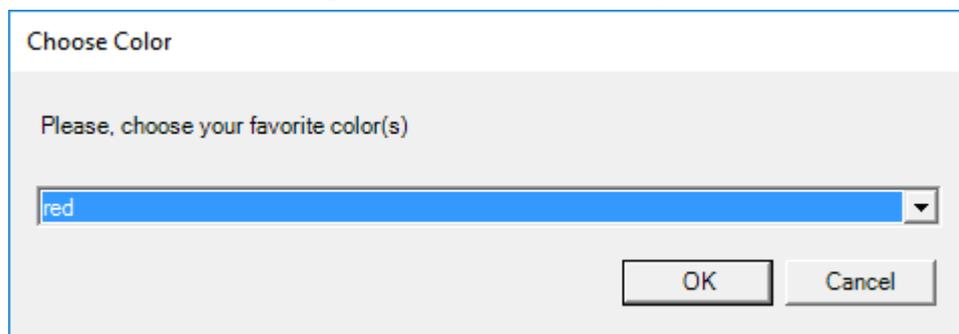
Store Selected Index into: 

Store Button Pressed into: 

This action is Enabled



The Select From List Dialog in action with "Allow Multiple Selection" unchecked



The Select From List Dialog with "Allow Multiple Selection" checked

Properties:

Dialog Title:

Enter the text, or a previously defined text variable, to be used as the Dialog title.

Dialog Message:

Enter the text, or a previously defined text variable, to be displayed as a Prompt Message for the user.

List to choose from:

Enter a list of choices - one per line, or a previously defined variable that contains a list. This will be displayed as a drop-down menu for the user to choose from.

Keep Select Dialog Always on Top:

Choose whether the Selection Dialog should always remain on top of all other windows.

Limit to List:

Deselect this box if you wish to allow the user to enter their own answer outside of the list being displayed. This means the action will act more like the "Display Input Dialog" action.

Allow Empty Selection:

Allow the user to not select anything, creating an empty Selected Item output.

Allow Multiple Selection:

Allow the user to select more than one choice. This means that the Selected Item and Selected Index variables will hold a list of items.

Preselect Items Starting with a + Sign:

This option is available only when the previous one (Allow Multiple Selection) is checked. By selecting this option all items with a '+' sign prepended on their name will appear automatically preselected (and the '+' sign will be removed from the item's text).

Store Selected Item into:

Enter a name to be the variable that will store the item selected from the list as text (or a list of text values if 'Allow Multiple Selection' is checked).

Store Selected Index into:

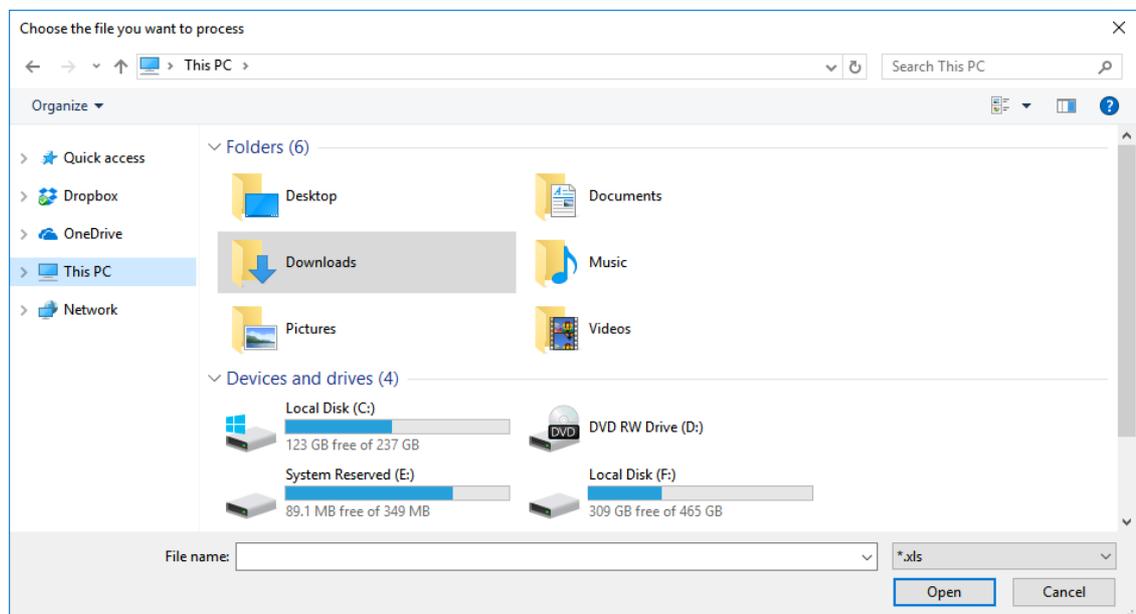
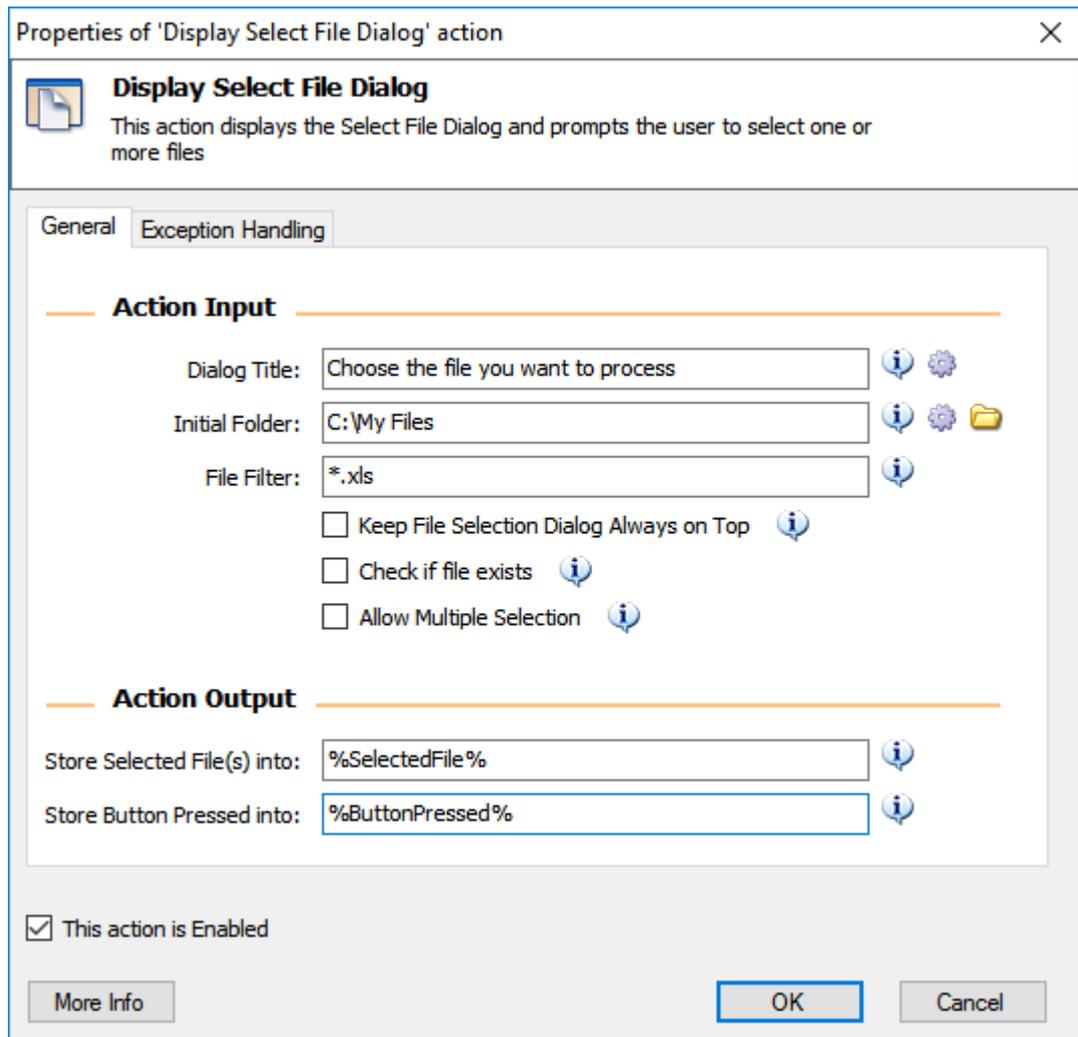
Enter a name to be the variable that will store the index number of the item(s) selected from the list. This allows you to use the item number instead of the full text of the user's choice. The variable that will store the index of the selected item as number (or a list of numeric values if 'Allow Multiple Selection' is checked). If the 'Limit to List' box is not checked and the user enters a new value, the index value will be '-1'.

Store Button Pressed into:

Enter a name to be the variable that will store the name of the button pressed by the user (OK or Cancel).

3.11.7.7 Display Select File Dialog Action**Description:**

This action displays the Select File Dialog and prompts the user to select one or more files



The Select File Dialog in action (as displayed in Windows Vista)

Properties:**Dialog Title:**

Enter the text, or a previously stored text variable, to be used as the Dialog title.

Initial Folder:

Enter or choose the initial folder to be opened when browsing for a file. This is where the Select File Dialog action will start the user looking for the file(s).

File Filter:

Choose a filter to limit the files retrieved. This allows wild cards, for example "*.txt" or "document?.doc" (without the quotes). If you want to allow for multiple file filters the user can choose from, separate your choices with a semi-colon, for example, "*.txt;.exe".

Keep File Selection Dialog Always on Top:

Choose whether the File Selection Dialog should always remain on top of all other windows.

Check if File Exists:

If checked, this action will then only accept a file (or multiple files) that already exists.

Allow Multiple Selection:

Choose whether this action will allow the user to select more than one file or not.

Store Selected File(s) into:

Enter a name to be the variable that will store the file(s). If Allow Multiple Selection is checked, this variable will contain a list of files, even if the list is only one entry. If Allow Multiple Selection is not checked, this variable will contain a single file, not a list.

Store Button Pressed into:

The user will automatically be given the choice of Open or Cancel. Enter a name to be the variable that will store the text of the button pressed.

3.11.7.8 Display Select Folder Dialog Action**Description:**

This action displays the Select Folder Dialog and prompts the user to select a folder

Properties of 'Display Select Folder Dialog' action

Display Select Folder Dialog
This action displays the Select Folder Dialog and prompts the user to select a folder

General | Exception Handling

Action Input

Dialog Description: Please, select a folder

Initial Folder: C:\

Keep Folder Selection Dialog Always on Top

Action Output

Store Selected Folder into: %SelectedFolder%

Store Button Pressed into: %ButtonPressed%

This action is Enabled

More Info OK Cancel

Properties:

Dialog Description:

Enter text to explain why you want the user to select a folder. For example, "Please select the folder into which you wish to copy the files."

Initial Folder:

Enter or choose the initial folder to be opened. This will be the default folder unless the user picks a new one.

Keep Folder Selection Always on Top:

Choose whether the Folder Selection Dialog should always remain on top of all other windows.

Store Selected Folder into:

Enter a name to be the variable that will store the selected folder.

Store Button Pressed into:

The user will automatically be given the choice of OK or Cancel. Enter a name to be the variable that will store the text of the button pressed.

3.11.8 Files

3.11.8.1 Get Files in Folder Action

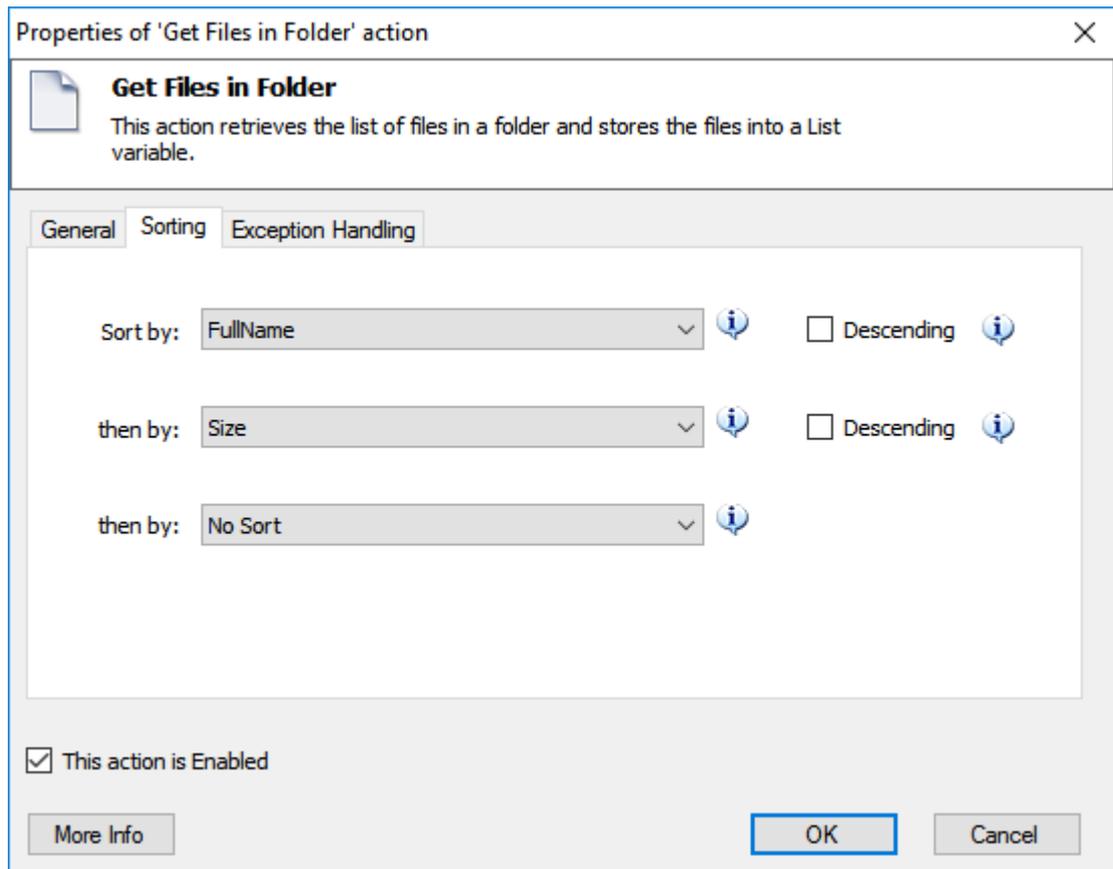
Description:

This action retrieves the list of files in a folder and stores the files info into a "List of Files" variable

The screenshot shows a dialog box titled "Properties of 'Get Files in Folder' action". It features a description: "Get Files in Folder. This action retrieves the list of files in a folder and stores the files into a List variable." Below this are three tabs: "General", "Sorting", and "Exception Handling". The "General" tab is active and contains the following fields:

- Action Input:**
 - Folder: C:\MyFiles (with folder icon and help/info icons)
 - File Filter: *.docx (with help/info icons)
 - Include Subfolders (with help/info icon)
- Action Output:**
 - Store Retrieved Files into: %Files% (with help/info icon)

At the bottom, there is a checked checkbox "This action is Enabled", a "More Info" button, and "OK" and "Cancel" buttons.



Properties:

Folder:

Enter or choose the full path of the folder, or a variable containing the folder, you wish to get files from.

File Filter:

Choose a filter to limit the files retrieved. This allows wild cards, for example "*.txt" or "document?.doc" (without the quotes). If you want to allow for multiple file filters, separate your choices with a semi-colon, for example, "*.txt;*.exe".

Include Subfolders:

Check this if you wish to look into Subfolders as well.

Store Retrieved Files into:

Enter a name to be the variable that will store the files selected by this action as a list of file objects.

Sort by:

Choose if this action will sort the results, and by what criteria.

Descending:

Does this action sort into ascending or descending order?

3.11.8.2 Copy File(s) Action

Description:

This action copies one or more files into a destination folder.

Properties of 'Copy File(s)' action

Copy File(s)
This action copies one or more files into a destination folder.

General Exception Handling

Action Input

File(s) to Copy: %MyImages%

Destination Folder: C:\Images

If File(s) Exists: Do not Copy File

Action Output

Store Copied Files into: %CopiedFiles%

This action is Enabled

More Info OK Cancel

Properties:

File to Copy:

Enter or choose the file(s) that will be copied. This can be a file path, or a variable containing a file, a list of files, a text path, or a list of text paths. You can use the 'Get Files in Folder' action to populate a variable with a list of files.

Destination Folder:

Enter or choose the folder, or a variable containing a folder, that this action will use as the destination for the copied files.

If File Exists:

Choose what will happen if a file with the same name already exists in the destination folder.

Store Copied Files into:

Enter a name to be the variable that will store the copied file(s) as a list of files.

3.11.8.3 Move File(s) Action**Description:**

This action moves one or more files into a destination folder

Properties of 'Move File(s)' action

Move File(s)
This action moves one or more files into a destination folder

General | Exception Handling

Action Input

File(s) to Move: %Images%

Destination Folder: C:\Images

If File Exists: Do not Move File

Action Output

Store Moved Files into: %MovedFiles%

This action is Enabled

More Info OK Cancel

Properties:**File to Move:**

Enter or choose the the file(s) that will be moved. This can be a file path, or a variable containing a file, a list of files, a textual path, or a list of text paths. You can use the 'Get Files in Folder' action to populate a variable with a list of files.

Destination Folder:

Enter or choose the folder, or a variable containing a folder, that this action will use as the destination for the moved files.

If File Exists:

Choose what will happen if a file with the same name already exists in the destination folder.

Store Moved Files into:

Enter a name to be the variable that will store the moved file(s) as a list of files.

3.11.8.4 Delete File(s) Action**Description:**

This action deletes one or more files

Properties:**File(s) to Delete:**

Enter or choose the file(s) that will be deleted. This can be a file path, or a variable containing a file, a list of files, a text path, or a list of text paths. You can use the 'Get Files in Folder' action to populate a variable with a list of files.

3.11.8.5 Rename File(s) Action

Description:

This action changes the name of one or more files

Properties of 'Rename File(s)' action

Rename File(s)
This action changes the name of one or more files

General | Exception Handling

Action Input

File(s) to Rename: %LogFiles%

Rename Scheme: Change Extension

New Extension: txt

If File Exists: Do not Rename File

Action Output

Store Renamed Files into: %RenamedFiles%

This action is Enabled

More Info OK Cancel

Properties:

File to Rename:

Enter or choose the the file(s) that will be renamed. This can be a file path, or a variable containing a file, a list of files, a text path, or a list of text paths. You can use the 'Get Files in Folder' action to populate a variable with a list of files.

Rename Scheme:

Choose how you wish to rename the file(s).

Rename Scheme: Set New Name**New File Name:**

Enter the text, or a previously stored text variable, to be the new name of the file(s).

Keep Extension:

Check this box if you wish to include the previous extension with the file name(s). If this box is not checked, you must include the extension you wish the file(s) to have, or it will have none.

Rename Scheme: Add Text**Text to Add:**

Enter the text, or a previously stored text variable, to add to the original file name(s).

Add Text:

Choose whether to add the text before or after the original name(s).

Rename Scheme: Remove Text**Text to Remove:**

Enter the text, or a previously stored text variable, to remove from the original file name(s). This will search each file name, and remove the entered text from anywhere in the name. If this text exists more than once in the original name(s), it will be removed every time.

Rename Scheme: Replace Text**Text to Replace:**

Enter the text, or a previously stored text variable, to be replaced in the original file name(s). This will search each file name, and replace the entered text anywhere in the name, each time it occurs.

Replace with:

Enter the text, or a previously stored text variable, to replace the original text.

Rename Scheme: Change Extension

New Extension:

Enter the text, or a previously stored text variable, to be the new extension for the file(s).

Rename Scheme: Add Date or Time

DateTime to Add:

Choose what DateTime value to add to the file name(s).

Custom DateTime:

Enter a previously stored DateTime variable to be added to the file name(s).

Add DateTime:

Choose whether to add the DateTime before or after the original name(s).

Separator:

Choose what to use to separate the original file name and the DateTime value added, including 'no separator'.

DateTime Format:

Enter or choose a Format this action will use to express the DateTime value that will be added to the file name. You can custom format a DateTime as, for example, MM/dd/yyyy for date, and hh:mm:ssstt for time.

Example:

Here you can see an example of the final output.

Rename Scheme: Make Sequential

Add Number to:

Choose whether to add the number to the existing file name(s) or a new name.

New Name:

Enter the text, or a previously stored text variable, to be the new base name of the file(s).

Add Number:

Choose whether to add the number before or after the original name(s) or a new base name.

Start Numbering at:

Enter a number, or a previously stored number variable, to be the starting number value.

Increment by:

Enter a number, or a previously stored number variable, to be the increment this action will count by.

Separator:

Choose what to use to separate the original file name and the number added, including 'no separator'.

Make each number at least:

Choose to have a set minimum length for each number added.

Example:

Here you can see an example of the final output.

If File Exists:

Choose what will happen if a file with the same name already exists in the folder.

Stored Renamed Files into:

Enter a name to be the variable that will store the renamed file(s) as a list of files.

3.11.8.6 Read Text from File Action

Description:

This action reads the content of a text file and stores it into a variable

Properties of 'Read Text from File' action

Read Text from File
This action reads the content of a text file and stores it into a variable

General Exception Handling

Action Input

File Path: C:\MyFile.txt

Store file content as: Single Text Value

Encoding: Default

Action Output

Store content into: %FileContents%

This action is Enabled

More Info OK Cancel

Properties:

File Path:

Enter or choose the the file to be read. This can be a file path, or a variable containing a file or a textual path.

Store Content As:

Choose how to store the text. If you choose Single Text Value, the entire text will be stored as a single text value. If you choose to store as a List, each line of the original text will be a text item in the list.

Encoding:

Select the encoding of the input text File.

Store Content Into:

Enter a name to be the variable that will store the contents as a text or list of texts, depending on your choice in 'Store File Content As'.

3.11.8.7 Write Text to File Action

Description:

This action writes or appends text to a file

Properties:

File Path:

Enter or choose the file to be written to. This can be a file path, or a variable containing a file or a textual path.

Text to Write:

Enter the text, or a previously stored variable, to be the content to write.

Append New Line:

Defines whether a new line character should be appended at the end of the overall text to be written to the file.

If File Exists:

Choose whether to overwrite the existing content, or to append to the end of the existing content. If the file does not exist, this action will automatically create it.

Encoding:

Select the encoding used for the specified text to be written into the text file.

3.11.8.8 Read From CSV File Action

Description:

Reads a CSV file into a Data Table Variant.

The screenshot shows a dialog box titled "Properties of 'Read from CSV File' action". The dialog has a close button (X) in the top right corner. Below the title bar, there is a document icon and the text "Read from CSV File" followed by "Reads a CSV file into a Data Table Variant." Below this, there are three tabs: "General", "Advanced", and "Exception Handling". The "General" tab is selected. The dialog is divided into two sections: "Action Input" and "Action Output". In the "Action Input" section, there is a "File Path:" text box with a file icon, an "Encoding:" dropdown menu set to "UTF8" with an information icon, and a "Store table into:" text box with the value "%CSVTable%" and an information icon. At the bottom of the dialog, there is a checkbox labeled "This action is Enabled" which is checked. There are three buttons at the bottom: "More Info", "OK", and "Cancel".

Properties:**File Path:**

Enter or choose the CSV file to be read. This can be a file path, or a variable containing a file or a textual path.

Encoding:

Select the encoding to be used for reading the CSV file.

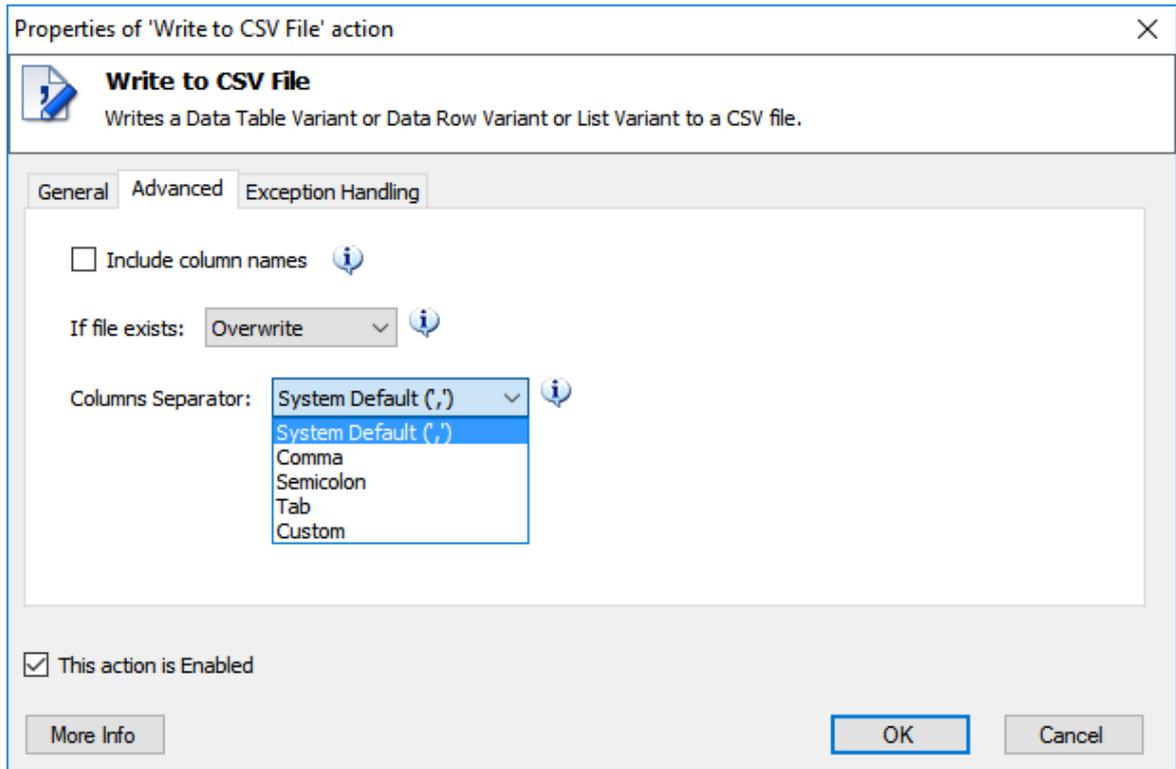
Store Table into:

Enter the name of the variable that will store the contents of the CSV file as a Data Table.

3.11.8.9 Write to CSV File Action**Description:**

Writes a Data Table Variant or Data Row Variant or List Variant to a CSV file.

The screenshot shows a dialog box titled "Properties of 'Write to CSV File' action". The dialog has a close button (X) in the top right corner. Below the title bar, there is a header section with a document icon and the text "Write to CSV File" and "Writes a Data Table Variant or Data Row Variant or List Variant to a CSV file." Below this, there are three tabs: "General", "Advanced", and "Exception Handling". The "General" tab is selected. The dialog is divided into two main sections: "Action Input" and "Action Output". In the "Action Input" section, there are three fields: "Variable to Write:" with an empty text box and an information icon; "File Path:" with an empty text box, an information icon, a gear icon, and a file icon; and "Encoding:" with a dropdown menu showing "UTF8" and an information icon. The "Action Output" section contains the text "(This action does not provide any output)". At the bottom of the dialog, there is a checkbox labeled "This action is Enabled" which is checked. There are three buttons at the bottom: "More Info", "OK", and "Cancel".



Properties:

Variable to Export:

Enter the name of the Data Table Variable or Data Row Variable or List Variable which must be written into the target CSV file.

File Path:

Enter or choose the CSV file to export the variable to. This can be a file path, or a variable containing a file or a textual path.

Encoding:

Select the encoding to be used for writing to the CSV file specified.

Include column names:

Choose whether the column names of the variant specified should become the very first row of the CSV file. This option takes effect if and only if the target CSV file either does not initially exist or exists but is otherwise empty of text.

If file exists:

Specify the desired behavior when the targeted CSV file is found to be already present in the filesystem.

Columns Separator:

Specify the column separator to use in the CSV file to be created.

3.11.8.10 Get FilePath Part Action**Description:**

This action retrieves one or more parts (directory, filename, extension etc) from a text that represents a FilePath

Properties of 'Get FilePath Part' action

Get FilePath Part
This action retrieves one or more parts (directory, filename, extension etc) from a text that represents a FilePath

General Exception Handling

Action Input

File Path: %CurrentFile%

Action Output

Root Path: %RootPath%

Directory: %Directory%

File Name: %FileName%

File Name without Extension: %FileNameNoExtension%

Extension: %FileExtension%

This action is Enabled

More Info OK Cancel

Properties:

File Path:

Enter a previously defined text variable containing a file path to be used as the source.

Root Path:

Enter a name to be the variable that will contain the Root Path of the source file.

Directory:

Enter a name to be the variable that will contain the Directory of the source file.

File Name:

Enter a name to be the variable that will contain the name of the source file.

File Name without Extension:

Enter a name to be the variable that will contain the file name (without the extension) of the source file.

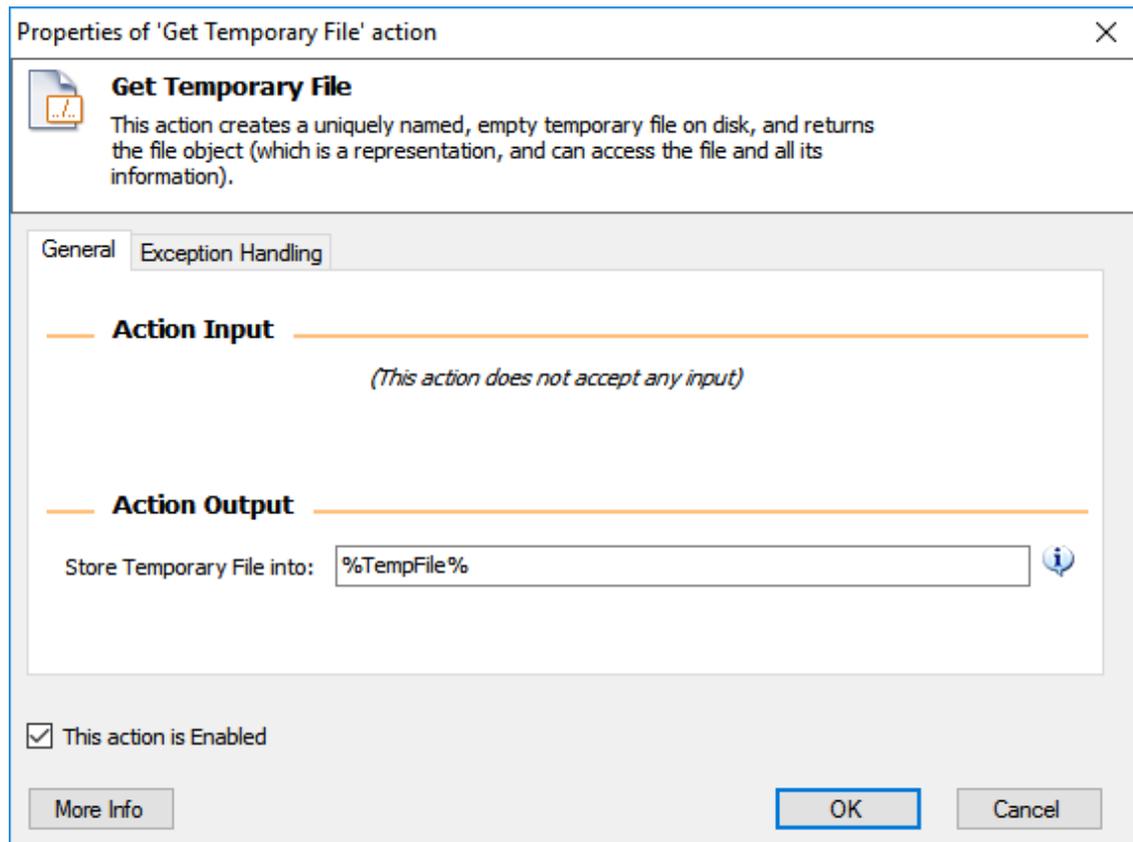
Extension:

Enter a name to be the variable that will contain the Extension (for example, .doc) of the source file.

3.11.8.11 Get Temporary File Action

Description:

This action creates a uniquely named, empty temporary file on disk, and returns the file object (which is a representation, and can access the file and all its information).



Properties:

Store Temporary File into:

Enter a name to be the variable that will contain the temporary file object.

3.11.9 Folders

3.11.9.1 Get Subfolders in Folder Action

Description:

This action retrieves the list of subfolders in a folder and stores the subfolders info into a DataTable variable

Properties of 'Get Subfolders in Folder' action

Get Subfolders in Folder
This action retrieves the list of subfolders in a folder, and stores the information into a list.

General | **Sorting** | Exception Handling

Action Input

Folder: ⓘ ⚙️ 📁

Include Subfolders ⓘ

Action Output

Store Retrieved Subfolders into: ⓘ

This action is Enabled

Properties of 'Get Subfolders in Folder' action

Get Subfolders in Folder
This action retrieves the list of subfolders in a folder, and stores the information into a list.

General | **Sorting** | Exception Handling

Sort by: ⓘ Descending ⓘ

then by: ⓘ Descending ⓘ

then by: ⓘ

This action is Enabled

Properties:

Folder:

Enter or choose the full path of the folder, or a variable containing the folder, you wish to retrieve the list of Subfolders from.

Include Subfolders:

Check this if you wish to look into the Subfolders, and retrieve their Subfolders (and so on) as well.

Store Retrieved Subfolders into:

Enter a name to be the variable that will store all of the Subfolders selected by this action as a list of folders.

Sort By:

Choose if this action will sort the results, and by what criteria.

Descending:

Does this action sort into ascending or descending order?

3.11.9.2 Create Folder Action***Description:***

This action creates a new folder

Properties of 'Create Folder' action

Create Folder
This action creates a new folder

General | Exception Handling

Action Input

Create New Folder into: C:\MyData

New Folder Name: My Images

Action Output

Store Newly Created Folder into: %NewFolder%

This action is Enabled

More Info OK Cancel

Properties:

Create New Folder into:

Enter or choose the full path of the folder, or a variable containing the folder, you wish to create a new Folder in.

New Folder Name:

Enter the text, or a text variable, to be the name of the new Folder.

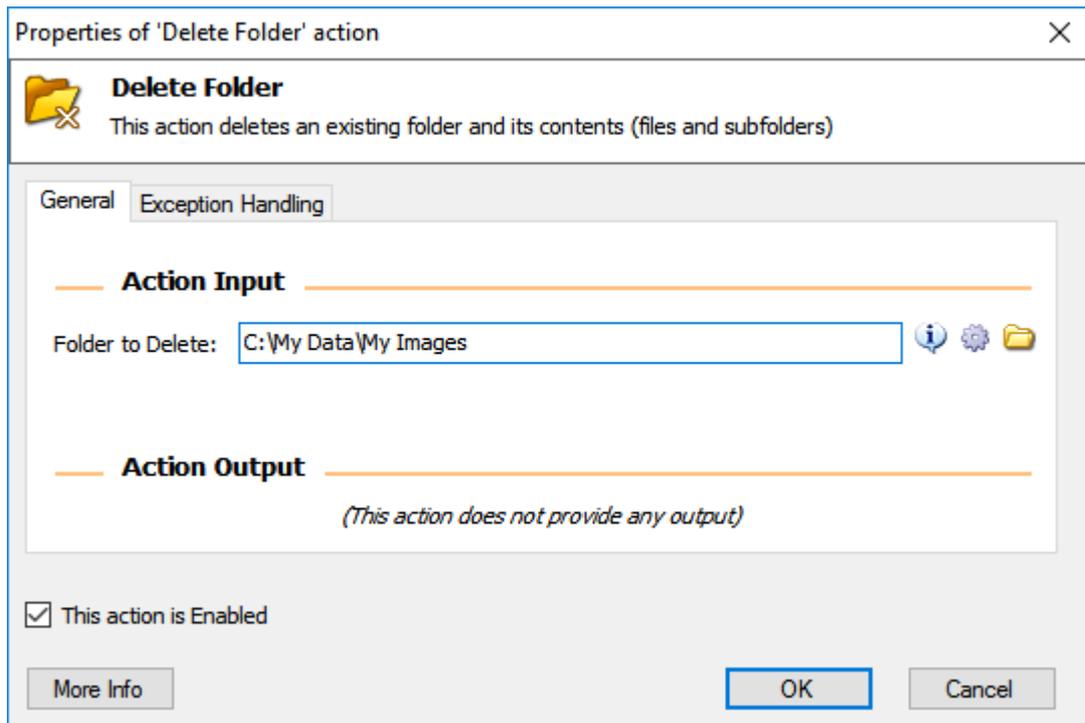
Store Newly Created Folder into:

Enter a name to be the variable that will contain the created Folder object (which is a representation, and can access the folder and all its information).

3.11.9.3 Delete Folder Action

Description:

This action deletes an existing folder and its contents (files and subfolders)



Properties:

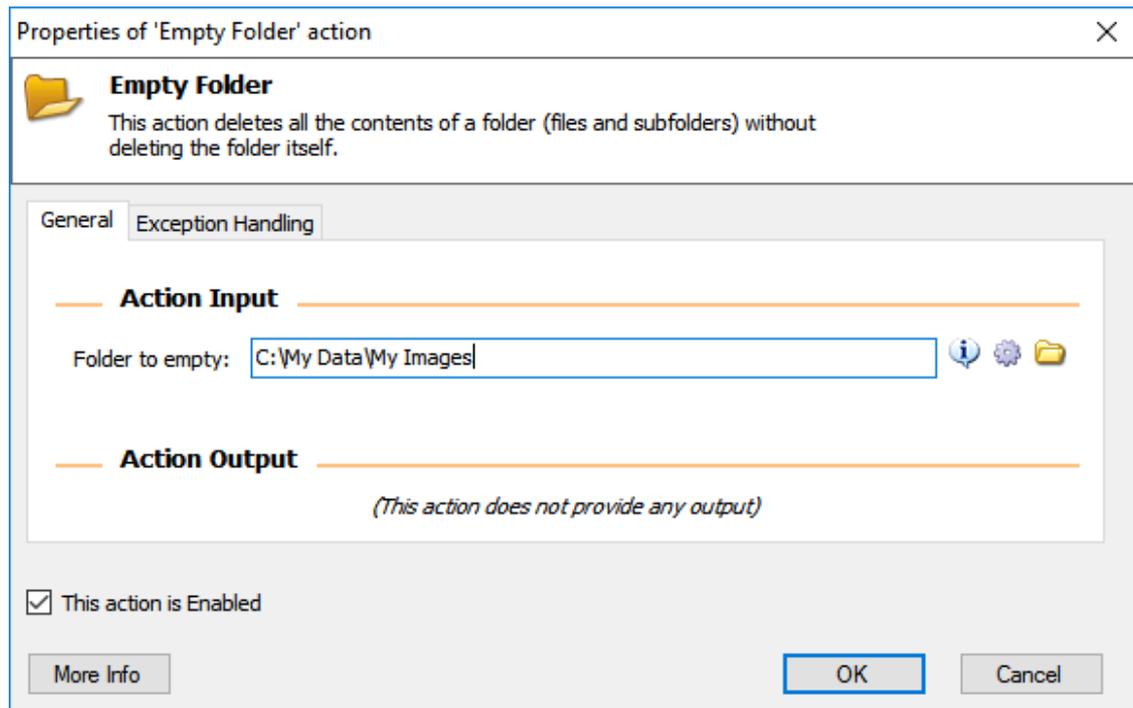
Folder to Delete:

Enter or choose the full path of the folder, or a variable containing the folder, you wish to delete. Remember that all contents of that folder, and its Subfolders, will be deleted too.

3.11.9.4 Empty Folder Action

Description:

This action deletes all the contents of a folder (files and subfolders) without deleting the folder itself.



Properties:

Folder to empty:

Enter or choose the full path of the folder, or a variable containing the folder whose contents you want to delete.

3.11.9.5 Copy Folder Action

Description:

This action copies a folder into a destination folder

Properties of 'Copy Folder' action

Copy Folder
This action copies a folder into a destination folder

General | Exception Handling

Action Input

Folder to Copy: C:\My Data\My Images

Destination Folder: D:\Backup

If Folder Exists: Do not Copy

Action Output

Store Copied Folder into: %CopiedFolder%

This action is Enabled

More Info OK Cancel

Properties:

Folder to Copy:

Enter or choose the full path of the folder, or a variable containing the folder, you wish to copy. If the folder path ends with a \ only the contents of the folder (files + subfolders) will be copied. Otherwise, the folder itself (along with its contents) will be copied as a subfolder into the destination folder

Destination Folder:

Enter or choose the full path of the folder, or a variable containing the folder, to be the destination Folder.

If Folder Exists:

Choose whether to overwrite files or to not copy if the destination folder already exists. If the folder exists, but the files have different names, the old files will still be in the folder.

Store Copied Folder into:

Enter a name to be the variable that will contain the copied folder object (which is a representation, and can access the folder and all its information).

3.11.9.6 Move Folder Action

Description:

This action moves an existing folder into a destination folder

The screenshot shows a dialog box titled "Properties of 'Move Folder' action". It features a yellow folder icon and the text "Move Folder" and "This action moves an existing folder into a destination folder". The dialog has two tabs: "General" (selected) and "Exception Handling". Under "General", there are two sections: "Action Input" and "Action Output".

Action Input:

- Folder to Move: C:\My Data\My Images
- Destination Folder: D:\All Media

Action Output:

- Store Moved Folder into: %MovedFolder%

At the bottom, there is a checkbox labeled "This action is Enabled" which is checked. There are also buttons for "More Info", "OK", and "Cancel".

Properties:

Folder to Move:

Enter or choose the full path of the folder, or a variable containing the folder, you wish to move.

Destination Folder:

Enter or choose the full path of the folder, or a variable containing the folder, to be the destination Folder.

Store Moved Folder into:

Enter a name to be the variable that will contain the moved folder object (which is a representation, and can access the folder and all its information).

3.11.9.7 Rename Folder Action

Description:

This action changes the name of a folder

Properties of 'Rename Folder' action

Rename Folder
This action changes the name of a folder

General Exception Handling

Action Input

Folder to Rename: C:\My Data\My Images

New Folder Name: My Photos

Action Output

Store Renamed Folder into: %RenamedFolder%

This action is Enabled

More Info OK Cancel

Properties:

Folder to Rename:

Enter or choose the full path of the folder, or a variable containing the folder, to be the Folder that will be renamed.

New Folder Name:

Enter or choose the full path, or a variable containing the folder, to be the new Folder name.

Store Renamed Folder into:

Enter a name to be the variable that will contain the renamed folder object (which is a representation, and can access the folder and all its information).

3.11.9.8 Get Special Folder Action

Description:

This action retrieves the path of a Windows' special folder (such as Desktop, My Pictures, Internet Cache etc)

Properties of 'Get Special Folder' action

Get Special Folder
This action retrieves the path of a Windows' special folder (such as Desktop, My Pictures, Internet Cache etc)

General

Action Input

Special Folder Name: CommonApplicationData

Special Folder Path: C:\ProgramData

Action Output

Store Special Folder Path into: %SpecialFolder%

This action is Enabled

More Info OK Cancel

Properties:

Special Folder Name:

Choose the name of the Special Folder (like My Documents or Desktop). This will be independent of path, so the Special Folder can be found on any computer regardless of path specifics. The Special Folder Path for the current computer will be displayed in the field below.

Store Special Folder Path into:

Enter a name to be the variable that will contain the Special Folder object (which is a [Folder object](#)^[440]), and can access the folder and all its information).

More Information:

This command is extremely useful if you plan to send a compiled Process to many people. Let's say you're unzipping a file to the end user's desktops. The path would look something like this: "C:

\Users\Jane Doe\Desktop". Obviously, this will only work if "Jane Doe" is the current user. If "John Doe" tried to run the Process it would fail. This action could be used to solve such problems by replacing the names of folders when needed automatically.

3.11.10 Compression

3.11.10.1 Zip Files Action

Description:

This action compresses one or more files or folders into a Zip archive

Properties:

Archive Path:

Enter the full path of zip file you wish to create. If this file already exists, the new zipped files and/or folders will be added to the zip file. If the zip file already contains a file or folder with the same name, the existing one will be overwritten by the one brought in by this action.

File(s) to Zip:

Enter or choose the full path of a file or folder, or the name of a variable containing a file, a folder, or a list of files or folders, that you wish to include in the zip file. A list can be built with the 'Get Files in Folder' or 'Get Subfolders in Folder' actions. If you create a list of files, all subfolder information will be lost, and these will be extracted to the base folder. If you create a list of folders, subfolders will be kept, and the files will be extracted to their same directory structure.

Compression Level:

Choose the level of compression you want to use. The higher the compression, the smaller the file, though it takes longer to create or access.

Password:

Enter a password if you want the archive to be password protected. Otherwise leave this field blank and it will not have a password.

Archive Comment:

Enter any comment you wish to be included in the zip file as a file property.

Store Zip File into:

Enter a name to be the variable that will store the zip file created by this action.

3.11.10.2 Unzip Files Action**Description:**

This action uncompresses one or more files or folders contained in a Zip archive

Properties:

Archive Path:

Enter or choose the full path of a zip file, or the name of a variable containing a zip file, to be extracted.

Destination Folder:

Enter or choose the full path of a folder, or the name of a variable containing a folder, to extract the archive to. This will overwrite files in the folder with the same name as a file in the archive.

Password:

Include the password, if any, that is used for this archive. If the zip file is not password-protected, leave this blank.

Include Mask:

Choose a filter to limit the files extracted to those entered here. This allows wild cards, for example "*.txt" or "document?.doc" (without the quotes). If you want to allow for multiple file filters, separate your choices with a semi-colon, for example, "*.txt;*.exe". Other files will not be extracted.

Exclude Mask:

Choose a filter to limit the files extracted by excluding those entered here. This allows wild cards, for example "*.txt" or "document?.doc" (without the quotes). If you want to allow for multiple file filters, separate your choices with a semi-colon, for example, "*.txt;.exe". These files will not be extracted.

3.11.11 Clipboard

3.11.11.1 Get Clipboard Text Action

Description:

This action copies the text from the clipboard into a variable

Properties:

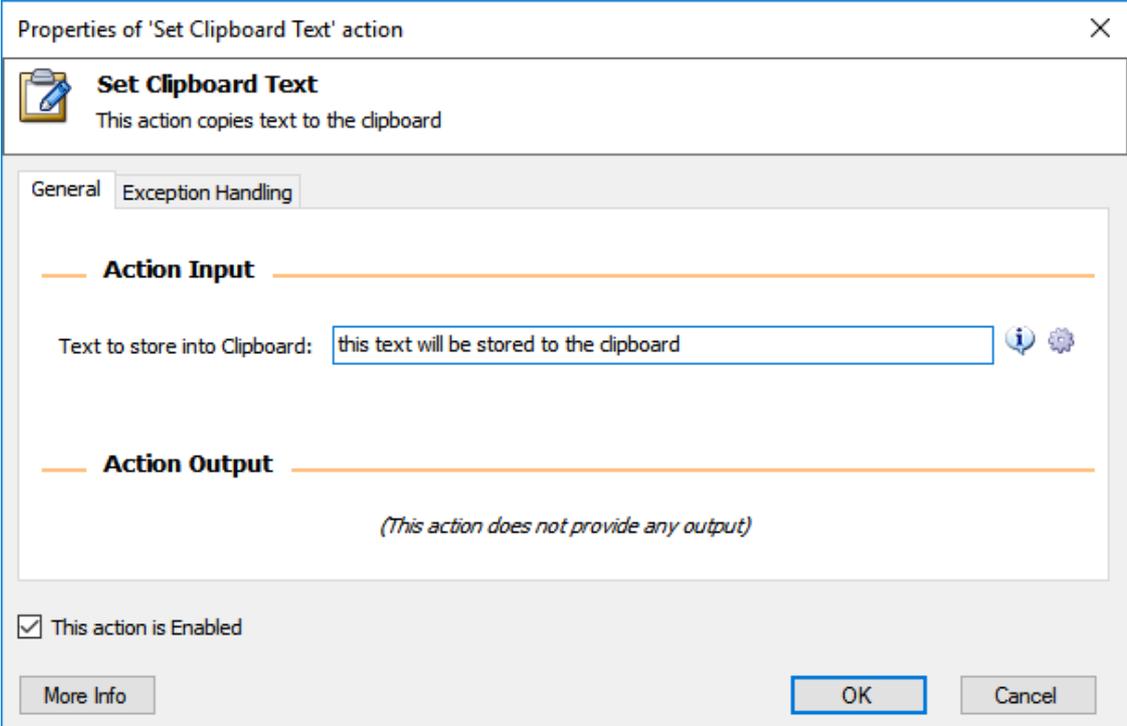
Store Clipboard Text Into:

Enter a name to be the variable that will store the text currently in the clipboard.

3.11.11.2 Set Clipboard Text Action

Description:

This action copies text to the clipboard



The screenshot shows a dialog box titled "Properties of 'Set Clipboard Text' action". It features a header with a clipboard icon and the text "Set Clipboard Text" and "This action copies text to the clipboard". Below this are two tabs: "General" (selected) and "Exception Handling". Under the "General" tab, there are two sections: "Action Input" and "Action Output". The "Action Input" section contains a text field labeled "Text to store into Clipboard:" with the value "this text will be stored to the clipboard" and an information icon. The "Action Output" section contains the text "(This action does not provide any output)". At the bottom, there is a checkbox labeled "This action is Enabled" which is checked, and three buttons: "More Info", "OK", and "Cancel".

Properties:

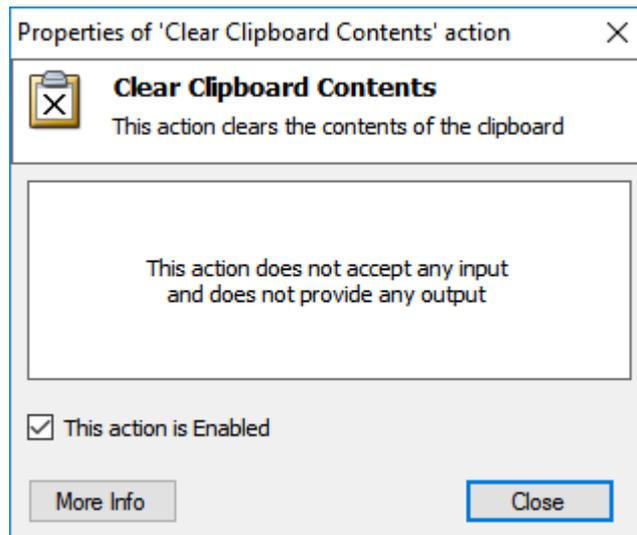
Text to store into Clipboard:

Enter the text, or a previously stored text variable, to be copied into the clipboard.

3.11.11.3 Clear Clipboard Contents Action

Description:

This action clears the contents of the clipboard



Properties:

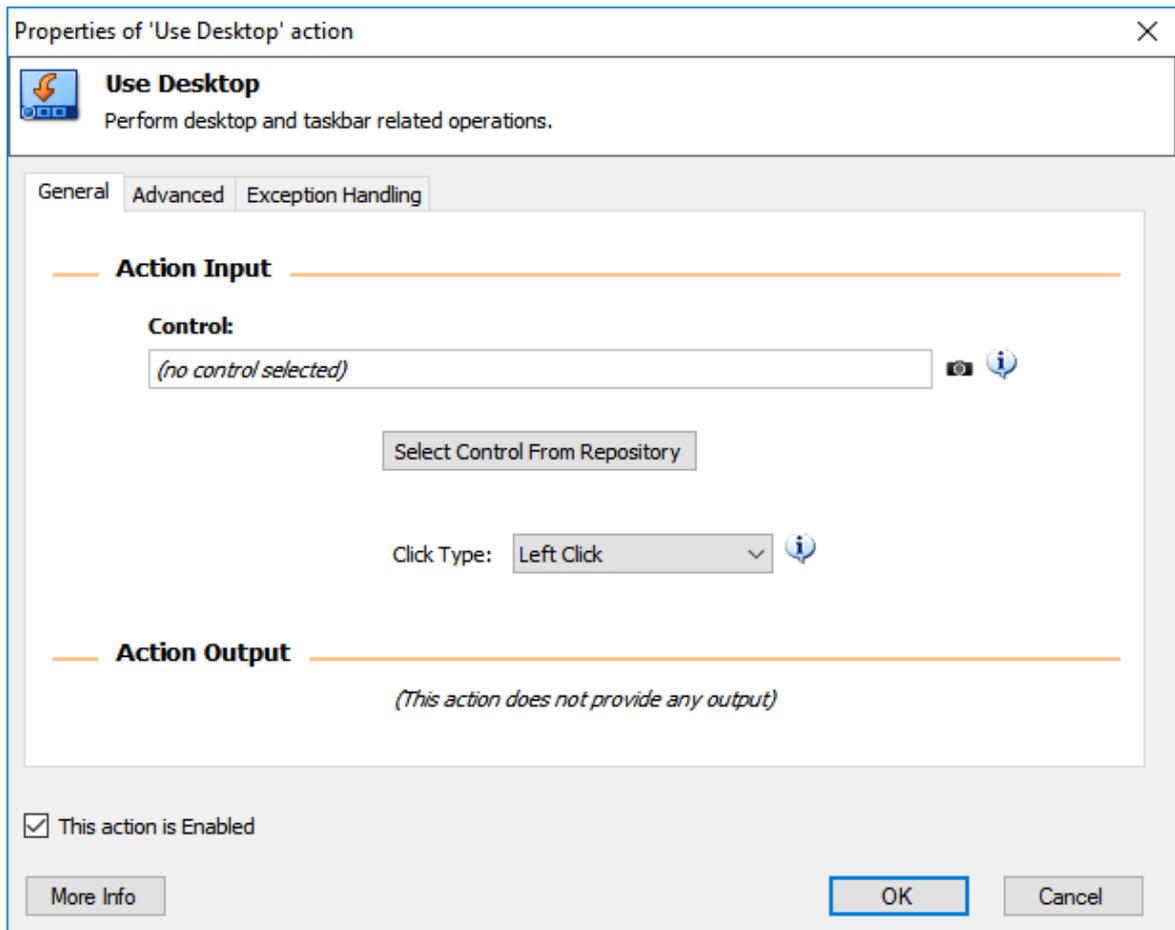
This action does not have any configurable properties.

3.11.12 UI and Windows

3.11.12.1 Use Desktop Action

Description:

Perform desktop and taskbar related operations



Properties:

Control:

Click on Select Control from Repository button and a box with all controls will appear. Hover over camera icon to preview the control's image.

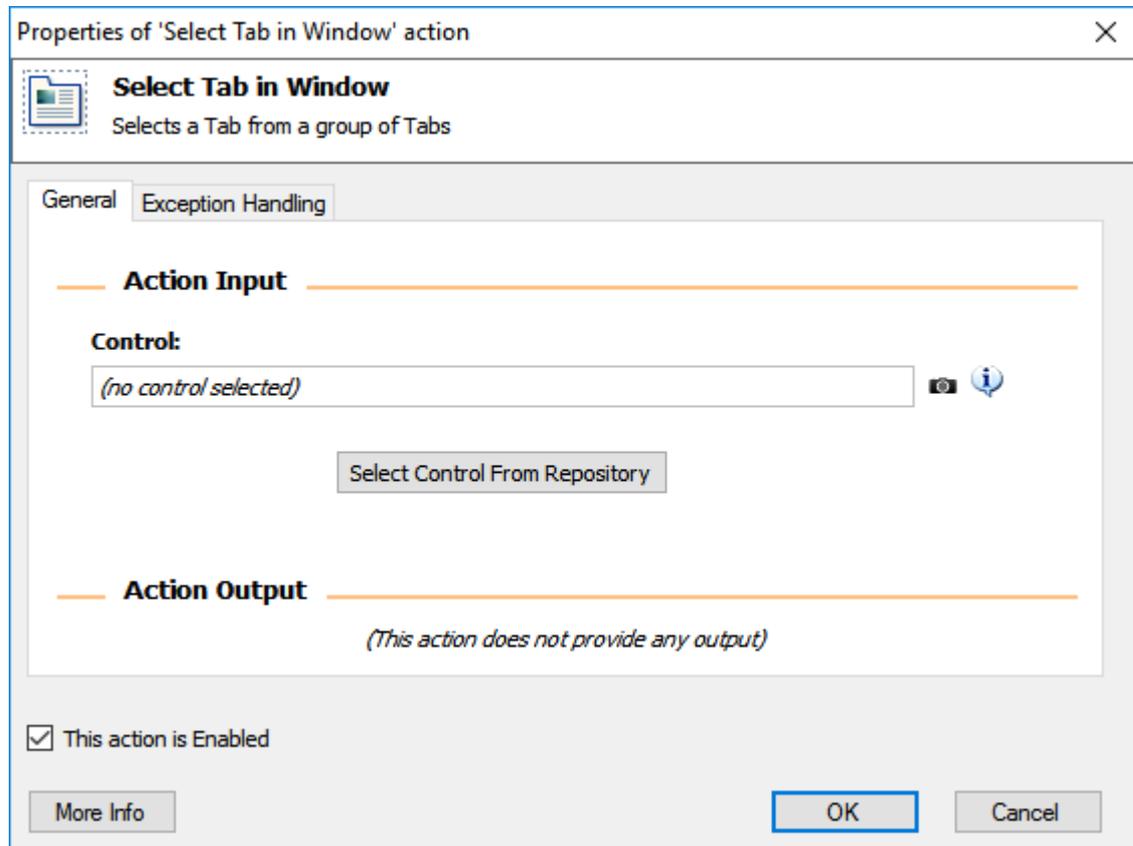
Click-Type:

Set the click action you want to perform on the specified element by selecting the click-type (left/right/double click). Quick-launch items will attempt to spawn a new application window, whenever this is possible.

3.11.12.2 Select Tab in Window Action

Description:

Select a Tab from a group of Tabs.



Properties:

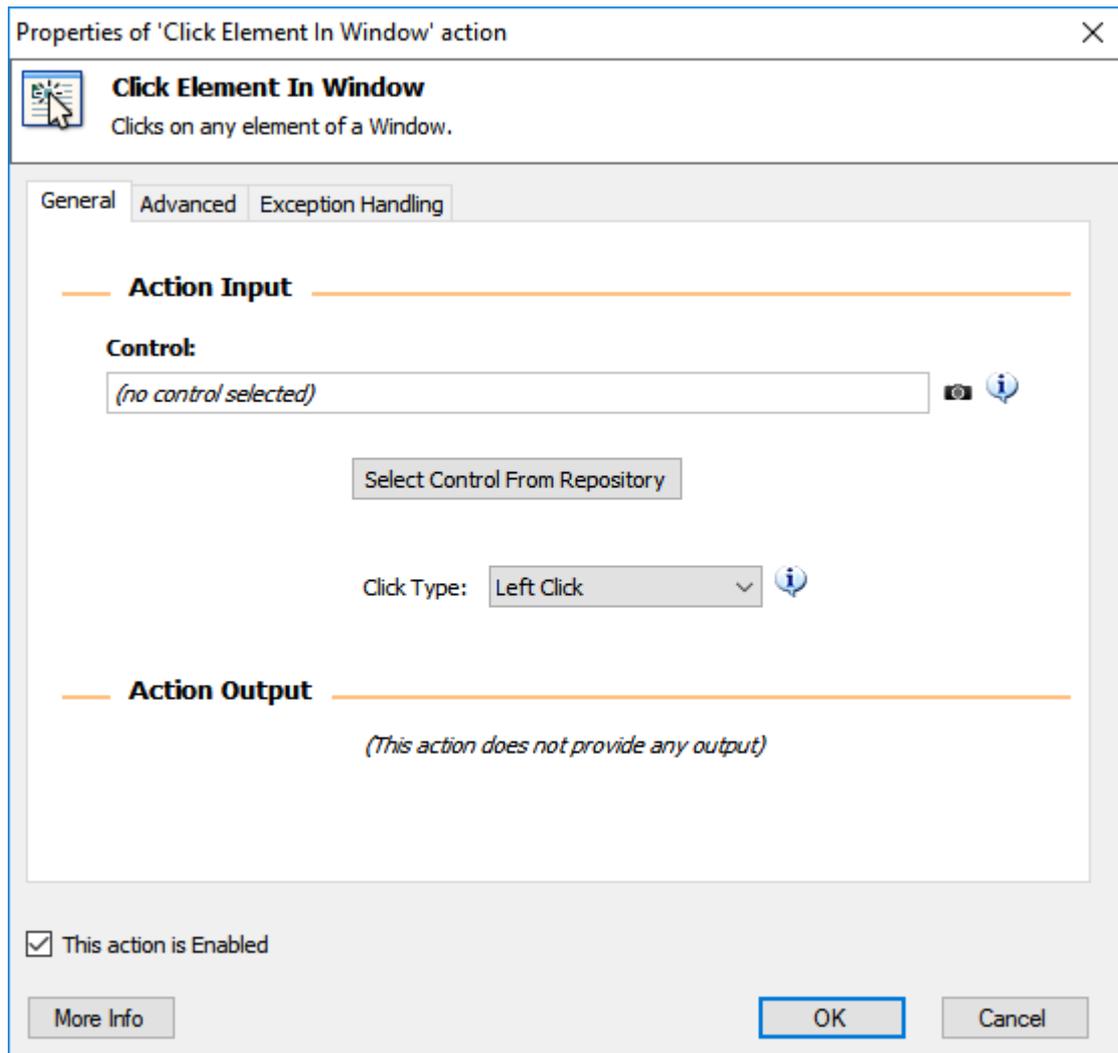
Control:

Click on Select Control from Repository button and a box with all controls will appear. Hover over camera icon to preview the control's image.

3.11.12.3 Click Element in Window Action

Description:

Clicks on any element of a Window.



Properties

Control:

Click on Select Control from Repository button and a box with all controls will appear. Hover over camera icon to preview the control's image.

3.11.12.4 Select Menu Option in Window Action

Description:

Selects an Option in a Menu of a Window

Properties of 'Select Menu Option in Window' action ✕

 **Select Menu Option in Window**
Selects an Option in a Menu of a Window

General Exception Handling

Action Input

Control:
 

Select Control From Repository

Action Output

(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties

Control:

Click on Select Control from Repository button and a box with all controls will appear. Hover over camera icon to preview the control's image.

i.e. To select the "Save As" menu option on a text file, once you select the control the properties of the action will look like this:

Properties of 'Select Menu Option in Window' action

Select Menu Option in Window
Selects an Option in a Menu of a Window

General Exception Handling

Action Input

Control:
MenuItem: File → Save As...  

On
Window 'Hello.txt - Notepad'

Select Control From Repository

Option:
File → Save As...

Action Output
(This action does not provide any output)

This action is Enabled

More Info OK Cancel

3.11.12.5 Drag and Drop Element in Window Action

Description:

Drags and Drops an Element of a Window

Properties of 'Drag and Drop Element in Window' action

Drag and Drop Element in Window
Drags and Drops an Element of a Window.

General | Advanced | Exception Handling

Action Input

Drag ...

Control:
(no control selected) [Camera] [Info]

Select Control From Repository

... and Drop it Over:

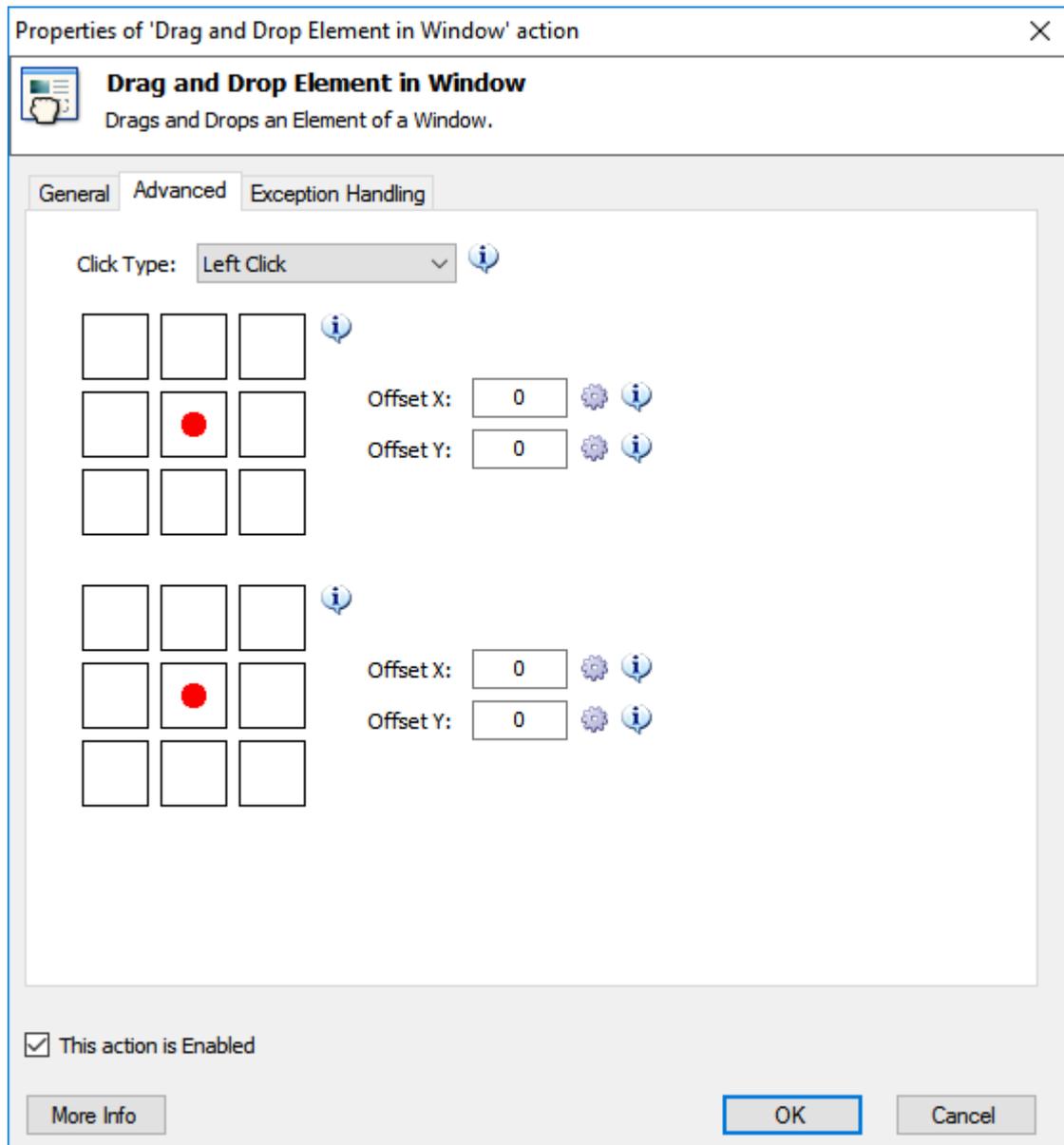
Control:
(no control selected) [Camera] [Info]

Select Control From Repository

Action Output
(This action does not provide any output)

This action is Enabled

More Info | OK | Cancel



Properties:

Control:

Click on Select Control from Repository button and a box with all controls will appear. Hover over camera icon to preview the control's image.

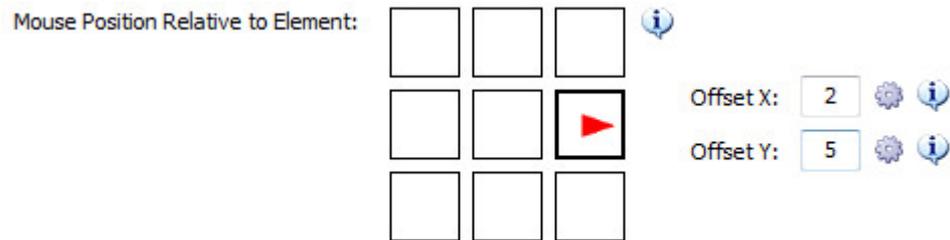
Advanced Properties Tab - Click-Type:

Specify which mouse button to use for clicking and holding down, while dragging the element over to its destination.

Advanced Properties Tab - Drag and Drop Targets Mouse Position Relative to Element:

For occasions where the target element is part of a parent element and cannot be identified as a separate control (when selected from the General Properties tab), you have to select the overall element and in the Advanced Properties Tab set the specific area of the element you want to place your cursor and click.

To do so you may initially select one of the nine parts in which the element is divided to place your the cursor, while for more fine tuning of cursor's position you may also define the Offset (in pixels) for X and Y axes relative to the control element's border or corner (as defined by the red arrow in each part).



e.g. for the setting above the cursor will be placed in the middle right section of the control element 2 px left from the right border and 5px lower from the middle.

3.11.12.6 Expand / Collapse Tree Node in Window Action

Description:

Expands or Collapses a Node of a Tree view residing in a Window

Properties of 'Expand/Collapse Tree Node in Window' action

Expand/Collapse Tree Node in Window
Expands or Collapses a Node of a Tree View residing in a Window

General Exception Handling

Action Input

Control:

Select Control From Repository

Folders Path:

Use Regular Expressions

Operation:

Action Output

(This action does not provide any output)

This action is Enabled

More Info

Properties:

Control:

Click on Select Control from Repository button and a box with all controls will appear. Hover over camera icon to preview the control's image.

Folders Path:

Enter a forward slash separated path made out of folders names leading to the Tree Node you want to expand or collapse.

Note that you may also use a value of a variable (generated in an earlier step of the Process) to specify the path to the tree node you wish to expand or collapse.

Use Regular Expression:

In the case of using a value of a variable for Folder Path input and depending on its notation, you may also need to check the 'Use Regular Expression' option so that action will interpret the name of the folder as a regular expression.

Operation:

Select whether you want to expand or collapse the tree node.

3.11.12.7 Windows

3.11.12.7.1 Get Window Action

Description:

Gets a running window, for automating desktop applications.

The screenshot shows the 'Properties of Get Window' action dialog box. The title bar reads 'Properties of Get Window' action. The main title is 'Get Window' with a subtitle 'Gets a running window, for automating desktop applications'. The dialog has three tabs: 'General', 'Advanced', and 'Exception Handling', with 'General' selected. Under the 'Action Input' section, there is a 'Get:' dropdown menu set to 'Specific Window'. Below this is a 'Window:' label with the instruction '(Move the mouse over a Window to select it)' and a 'Select Window' button. The 'Associate window with control:' dropdown is set to 'Existing Control'. Below this is a 'Window:' text box containing '(no control selected)' and a 'Select Control From Repository' button. There is a checkbox for 'Bring Window to Front' which is unchecked. Under the 'Action Output' section, the 'Store Window Instance Into:' text box contains '%AutomationWindow%'. At the bottom, there is a checked checkbox for 'This action is Enabled', a 'More Info' button, and 'OK' and 'Cancel' buttons.

Get Specific Window

Properties of 'Get Window' action

Get Window
Gets a running window, for automating desktop applications

General | Advanced | Exception Handling

Action Input

Get: ⓘ

Associate window with control: ⓘ

Window:
 ⓘ

Action Output

Store Window Title Into: ⓘ

Store Window Instance Into: ⓘ

This action is Enabled

Get Foreground Window

Properties:

Get:

Choose whether the action looks for the Window using a UIAutomation Instance Variable or a combination of Window Title/Class.

Depending on the Get option you will have some of the following inputs

Associate window with control:

Choose whether the window that this action trying to find will be associated with an existing control into controls repository or a new one that will be created.

Window:

Click on Select Control from Repository button and a box with all controls will appear. Hover over camera icon to preview the control's image.

Store Window Title into (when Get is "Foreground Window"):

Enter a name to be the variable that will store the title of the Foreground Window

Store Window Instance into:

Enter a name to be the variable that will store the title of the specific Window Instance for use with later UI Windows actions

3.11.12.7, Focus Window Action

Description:

This action activates and brings a specific window to the foreground

The screenshot shows the 'Properties of 'Focus Window' action' dialog box. The title bar reads 'Properties of 'Focus Window' action'. The main content area is titled 'Focus Window' and includes the description: 'This action activates and brings to foreground a specific window'. There are two tabs: 'General' (selected) and 'Exception Handling'. Under the 'General' tab, there is an 'Action Input' section with a 'Get Window:' dropdown menu set to 'by Window Control' and an information icon. Below it is a 'Control:' field with the text '(no control selected)' and a camera icon. A 'Select Control From Repository' button is positioned below the control field. The 'Action Output' section is empty, with the text '(This action does not provide any output)'. At the bottom left, there is a checked checkbox labeled 'This action is Enabled' and a 'More Info' button. At the bottom right, there are 'OK' and 'Cancel' buttons.

by Window Control

Properties of 'Focus Window' action

Focus Window
This action activates and brings to foreground a specific window

General | Exception Handling

Action Input

Get Window: by Window Instance/Handle

Window Instance: %AutomationWindow%

Action Output
(This action does not provide any output)

This action is Enabled

More Info OK Cancel

by Window Instance/Handle

Properties of 'Focus Window' action

Focus Window
This action activates and brings to foreground a specific window

General | Exception Handling

Action Input

Get Window: by Title and/or Class

Window Title:

Window Class:

or Click and Drag the target to choose a Window:

Action Output
(This action does not provide any output)

This action is Enabled

More Info OK Cancel

by Title and/or Class

Properties:**Get Window:**

Choose whether the action looks for the Window using a UIAutomation Instance Variable or a combination of Window Title/Class.

Depending on the Get Window option you will have some of the following inputs

Control:

Click on Select Control from Repository button and a box with all controls will appear. Hover over camera icon to preview the control's image.

Window Instance:

Enter the variable of a variable that contains a UI Automation Instance, acquired by a Get Window Action.

Window Title:

Choose title of window. You can use wildcards, like '?' or '*'. If you can't find the window title in the dropdown list, type it yourself, or open it and press the Refresh button to the right.

Window Class:

Optional: If you have two windows with the same title, Window Class may help differentiate between them. In this case, enter the class of the window you wish to use. Otherwise, leave this blank.

3.11.12.7,Set Window State Action**Description:**

This action restores, maximizes or minimizes a specific window

Properties of 'Set Window State' action

Set Window State
This action restores, maximizes or minimizes a specific window

General | Exception Handling

Action Input

Get Window: by Window Control

Control:
(no control selected)

Select Control From Repository

Set Window State to: Restored

Action Output
(This action does not provide any output)

This action is Enabled

More Info

OK Cancel

by Window Control

Properties of 'Set Window State' action

Set Window State
This action restores, maximizes or minimizes a specific window

General | Exception Handling

Action Input

Get Window: by Window Instance/Handle

Window Instance: %AutomationWindow%

Set Window State to: Restored

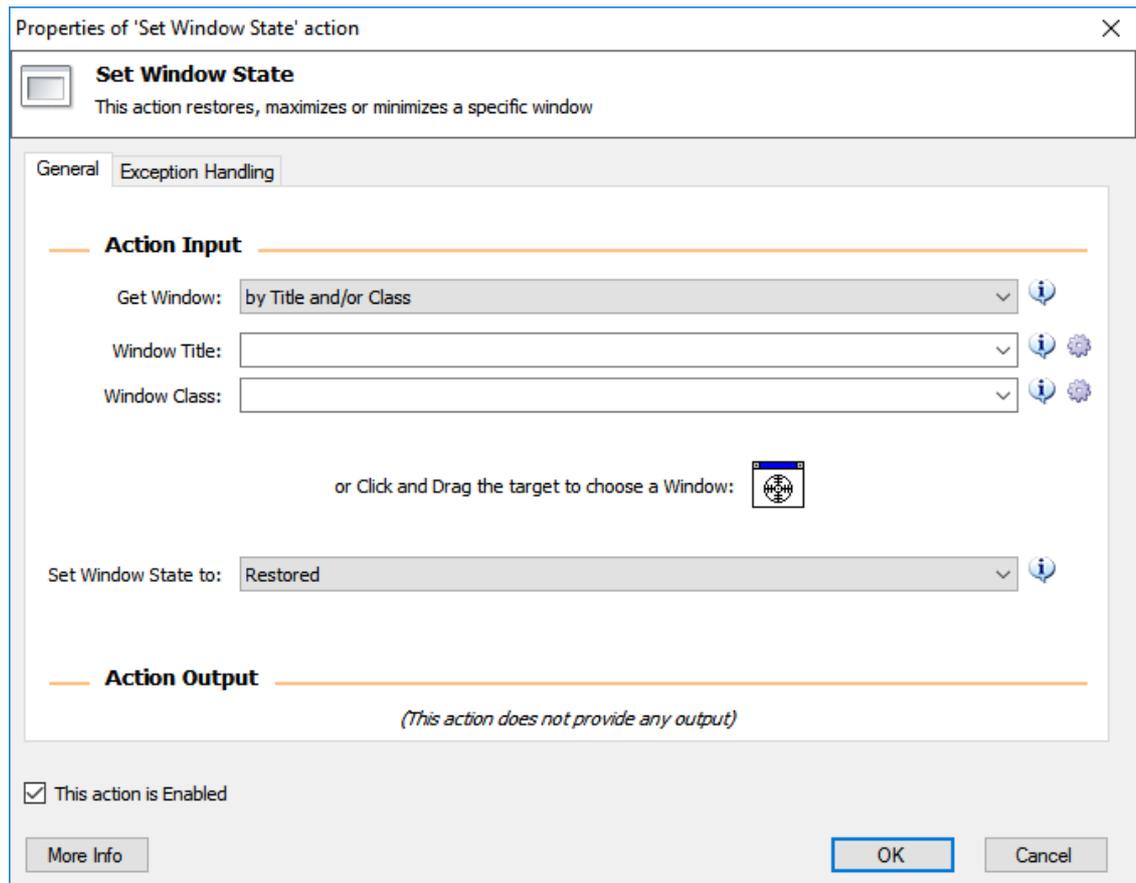
Action Output

(This action does not provide any output)

This action is Enabled

More Info OK Cancel

by Window Instance/Handle



by Title and/or Class

Properties:

Get Window:

Choose whether the action looks for the Window using a UIAutomation Instance Variable or a combination of Window Title/Class.

Depending on the Get Window option you will have some of the following inputs

Control:

Click on Select Control from Repository button and a box with all controls will appear. Hover over camera icon to preview the control's image.

Window Instance:

Enter the variable of a variable that contains a UI Automation Instance, acquired by a Get Window Action.

Window Title:

Choose title of window. You can use wildcards, like '?' or '*'. If you can't find the window title in the dropdown list, type it yourself, or open it and press the Refresh button to the right.

Window Class:

Optional: If you have two windows with the same title, Window Class may help differentiate between them. In this case, enter the class of the window you wish to use. Otherwise, leave this blank.

Set Window State to:

Choose in which state you would like to display the window.

3.11.12.7, Set Window Visibility Action

Description:

This action shows a hidden window or hides a visible window

Properties of 'Set Window Visibility' action

Set Window Visibility
This action shows a hidden window or hides a visible window

General Exception Handling

Action Input

Get Window: by Window Control

Control:
(no control selected)

Select Control From Repository

Set Window Visibility to: Hidden

Action Output
(This action does not provide any output)

This action is Enabled

More Info OK Cancel

by Window Control

Properties of 'Set Window Visibility' action

Set Window Visibility
This action shows a hidden window or hides a visible window

General | Exception Handling

Action Input

Get Window: by Window Instance/Handle

Window Instance: %AutomationWindow%

Set Window Visibility to: Hidden

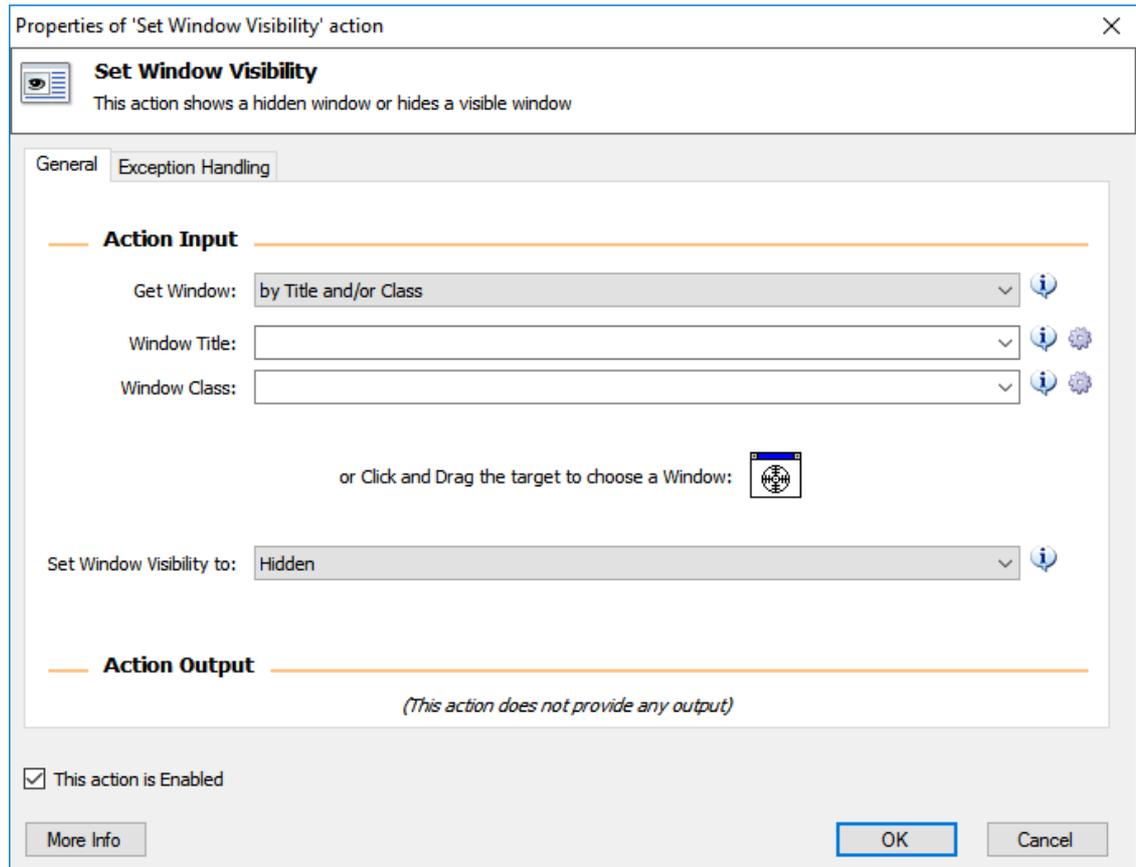
Action Output

(This action does not provide any output)

This action is Enabled

More Info OK Cancel

by Window Instance/Handle



by Title and/or Class

Properties:

Get Window:

Choose whether the action looks for the Window using a UIAutomation Instance Variable or a combination of Window Title/Class.

Depending on the Get Window option you will have some of the following inputs

Control:

Click on Select Control from Repository button and a box with all controls will appear. Hover over camera icon to preview the control's image.

Window Instance:

Enter the variable of a variable that contains a UI Automation Instance, acquired by a Get Window Action.

Window Title:

Choose title of window. You can use wildcards, like '?' or '*'. If you can't find the window title in the dropdown list, type it yourself, or open it and press the Refresh button to the right.

Window Class:

Optional: If you have two windows with the same title, Window Class may help differentiate between them. In this case, enter the class of the window you wish to use. Otherwise, leave this blank.

Set Window Visibility to:

Choose in which state you would like to display the window.

3.11.12.7. Move Window Action

Description:

This action sets the position of a specific window

by Window Control

Properties of 'Move Window' action

Move Window
This action sets the position of a specific window

General Exception Handling

Action Input

Get Window: by Window Instance/Handle

Window Instance: %AutomationWindow%

New Window Position: X: 400 Y: 400

Action Output
(This action does not provide any output)

This action is Enabled

More Info OK Cancel

by Window Instance/Handle

Properties of 'Move Window' action

Move Window
This action sets the position of a specific window

General | Exception Handling

Action Input

Get Window: by Title and/or Class 

Window Title:  

Window Class:  

or Click and Drag the target to choose a Window: 

New Window Position: X:  Y:  

Action Output

(This action does not provide any output)

This action is Enabled

by Title and/or Class

Properties:

Get Window:

Choose whether the action looks for the Window using a UIAutomation Instance Variable or a combination of Window Title/Class.

Depending on the Get Window option you will have some of the following inputs

Control:

Click on Select Control from Repository button and a box with all controls will appear. Hover over camera icon to preview the control's image.

Window Instance:

Enter the variable of a variable that contains a UI Automation Instance, acquired by a Get Window Action.

Window Title:

Choose title of window. You can use wildcards, like '?' or '*'. If you can't find the window title in the dropdown list, type it yourself, or open it and press the Refresh button to the right.

Window Class:

Optional: If you have two windows with the same title, Window Class may help differentiate between them. In this case, enter the class of the window you wish to use. Otherwise, leave this blank.

New Window Position:

Set the position of the top left corner of the Window on the X and Y coordinates, based on the pixels of the screen.

3.11.12.7, Resize Window Action***Description:***

This action sets the size of a specific window

Properties of 'Resize Window' action

Resize Window
This action sets the size of a specific window

General | Exception Handling

Action Input

Get Window: by Window Control

Control:
(no control selected)

Select Control From Repository

New Window Size: Width: 400 Height: 400

Action Output
(This action does not provide any output)

This action is Enabled

More Info OK Cancel

by Window Control

Properties of 'Resize Window' action

Resize Window
This action sets the size of a specific window

General | Exception Handling

Action Input

Get Window: by Window Instance/Handle ⓘ

Window Instance: %AutomationWindow% ⓘ ⚙

New Window Size: Width: 400 ⚙ Height: 400 ⓘ

Action Output

(This action does not provide any output)

This action is Enabled

More Info OK Cancel

by Window Instance/Handle

Properties of 'Resize Window' action

Resize Window
This action sets the size of a specific window

General Exception Handling

Action Input

Get Window: by Title and/or Class

Window Title:

Window Class:

or Click and Drag the target to choose a Window:

New Window Size: Width: 400 Height: 400

Action Output

(This action does not provide any output)

This action is Enabled

More Info OK Cancel

by Title and/or Class

Properties:

Get Window:

Choose whether the action looks for the Window using a UIAutomation Instance Variable or a combination of Window Title/Class.

Depending on the Get Window option you will have some of the following inputs

Control:

Click on Select Control from Repository button and a box with all controls will appear. Hover over camera icon to preview the control's image.

Window Instance:

Enter the variable of a variable that contains a UI Automation Instance, acquired by a Get Window Action.

Window Title:

Choose title of window. You can use wildcards, like '?' or '*'. If you can't find the window title in the dropdown list, type it yourself, or open it and press the Refresh button to the right.

Window Class:

Optional: If you have two windows with the same title, Window Class may help differentiate between them. In this case, enter the class of the window you wish to use. Otherwise, leave this blank.

New Window Size:

Set the new size of the Window, in pixels.

3.11.12.7, Close Window Action

Description:

This action closes a specific window

Properties of 'Close Window' action

Close Window
This action closes a specific window

General | Exception Handling

Action Input

Get Window: by Window Control

Control:
(no control selected)

Select Control From Repository

Action Output
(This action does not provide any output)

This action is Enabled

More Info | OK | Cancel

by Window Control

Properties of 'Close Window' action

Close Window
This action closes a specific window

General Exception Handling

Action Input

Get Window: by Window Instance/Handle

Window Instance: %AutomationWindow%

Action Output

(This action does not provide any output)

This action is Enabled

More Info OK Cancel

by Window Instance/Handle

Properties of 'Close Window' action

Close Window
This action closes a specific window

General Exception Handling

Action Input

Get Window: by Title and/or Class

Window Title:

Window Class:

or Click and Drag the target to choose a Window:

Action Output

(This action does not provide any output)

This action is Enabled

More Info OK Cancel

by Title and/or Class

Properties :**Get Window:**

Choose whether the action looks for the Window using a UIAutomation Instance Variable or a combination of Window Title/Class.

Depending on the Get Window option you will have some of the following inputs

Control:

Click on Select Control from Repository button and a box with all controls will appear. Hover over camera icon to preview the control's image.

Window Instance:

Enter the variable of a variable that contains a UI Automation Instance, acquired by a Get Window Action.

Window Title:

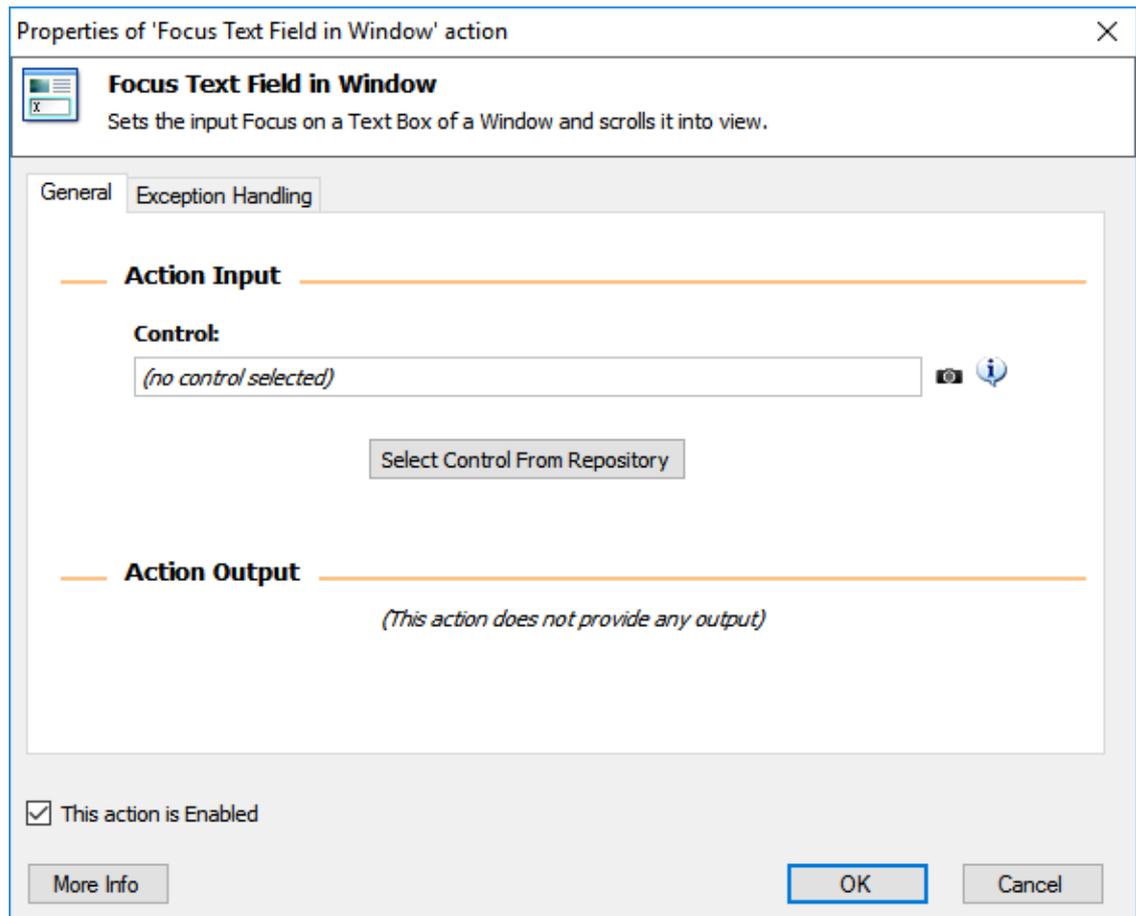
Choose title of window. You can use wildcards, like '?' or '*'. If you can't find the window title in the dropdown list, type it yourself, or open it and press the Refresh button to the right.

Window Class:

Optional: If you have two windows with the same title, Window Class may help differentiate between them. In this case, enter the class of the window you wish to use. Otherwise, leave this blank. this blank.

3.11.12.8 Form Filling**3.11.12.8,Focus Text Field in Window Action****Description:**

Sets the input Focus on a Text Box of a Window and scrolls it into view.



Properties:

Control:

Click on Select Control from Repository button and a box with all controls will appear. Hover over camera icon to preview the control's image.

3.11.12.8,Populate Text Field in Window Action

Description:

Fills a TextBox in a Window with the specified text.

Properties:

Control:

Click on Select Control from Repository button and a box with all controls will appear. Hover over camera icon to preview the control's image.

Text to Fill In:

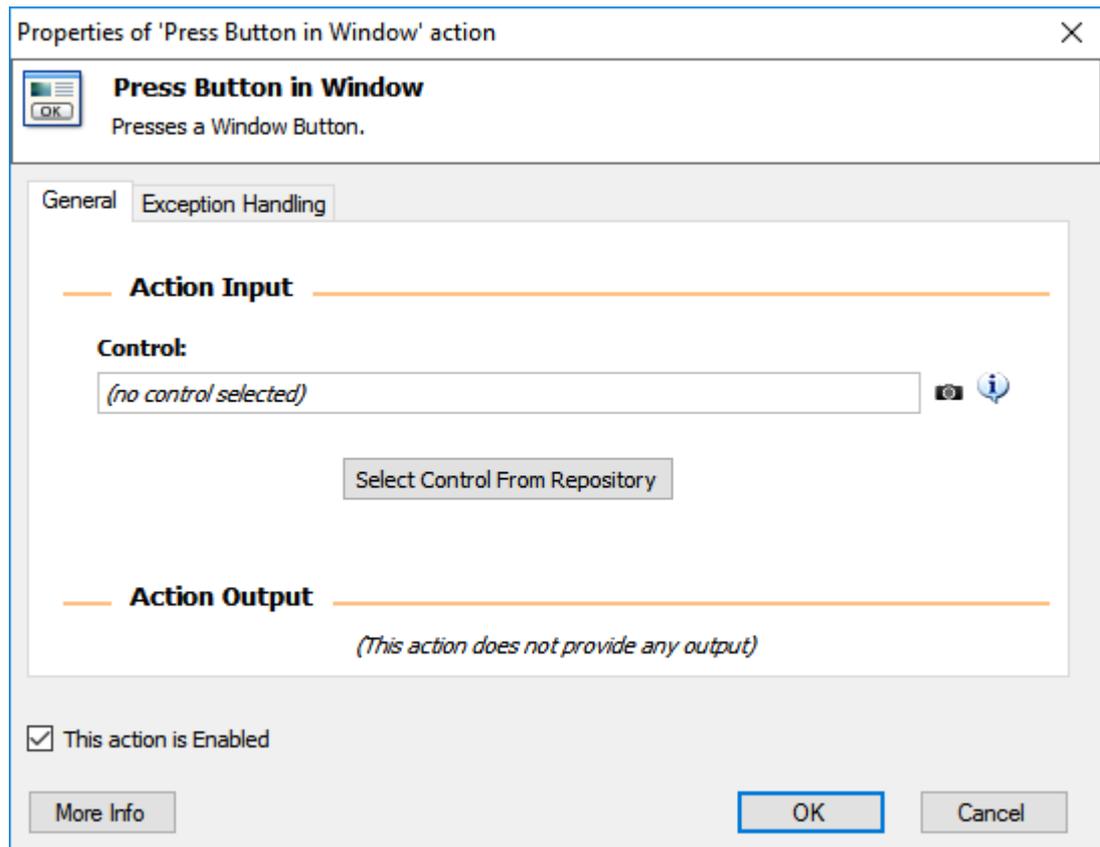
Enter here the text you want to fill in the text field with.

You may also use as input the value of a variable defined in a preceding action of your Process.

3.11.12.8,Press Button in Window Action

Description:

Presses a Window Button



Properties:

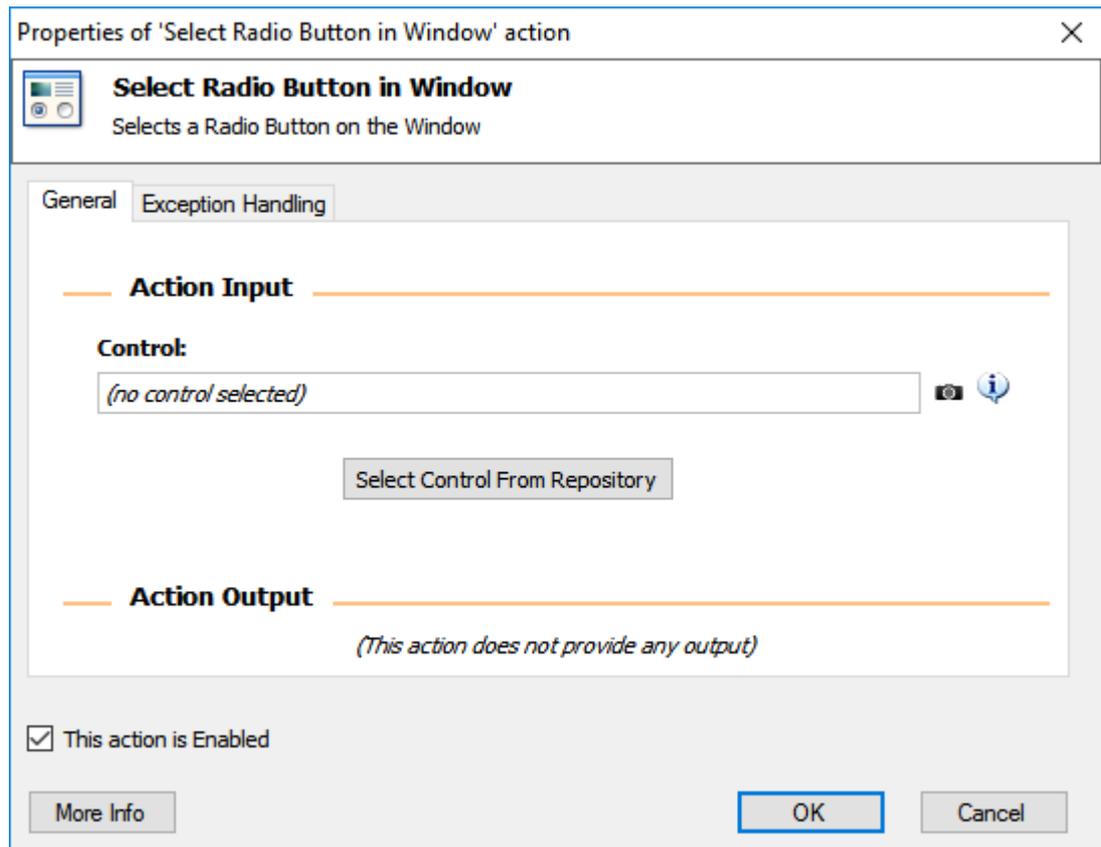
Control:

Click on Select Control from Repository button and a box with all controls will appear. Hover over camera icon to preview the control's image.

3.11.12.8 Select Radio Button in Window Action

Description:

Selects a Radio Button in the Window



Properties:

Control:

Click on Select Control from Repository button and a box with all controls will appear. Hover over camera icon to preview the control's image.

3.11.12.8,Set Checkbox State in Window Action

Description:

Checks or unchecks a Checkbox in a Window Form

Properties of 'Set Checkbox State in Window' action

Set Checkbox State in Window
Checks or unchecks a Checkbox in a Window Form

General Exception Handling

Action Input

Control:
(no control selected)  

Select Control From Repository

Set Checkbox State to: Checked 

Action Output
(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties:**Control:**

Click on Select Control from Repository button and a box with all controls will appear. Hover over camera icon to preview the control's image.

Set Checkbox State to:

Select whether you want the checkbox to be checked or unchecked.

3.11.12.8,Set Dropdown List Value in Window Action**Description:**

Sets or clears the Selected Option from a Drop Down List in a Window Form

Properties of 'Set DropDown List Value in Window' action

Set DropDown List Value in Window
Sets or clears the Selected Option for a Drop Down List in a Window Form

General Exception Handling

Action Input

Control:
(no control selected)  

Select Control From Repository

Operation: Clear Selected Options 

Action Output
(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties of 'Set DropDown List Value in Window' action

Set DropDown List Value in Window
Sets or clears the Selected Option for a Drop Down List in a Window Form

General Exception Handling

Action Input

Control:
(no control selected)  

Select Control From Repository

Operation: Select Option(s) 

Select Option(s) by: Name 

Option Name(s):  

Action Output
(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Operation: Select Option(s)

Properties:**Control:**

Click on Select Control from Repository button and a box with all controls will appear. Hover over camera icon to preview the control's image.

Operation:

Set whether you want to select a value or clear the selected value from a dropdown list.

Select Option(s) by:

Set the selection method for the value(s) of the Dropdown list.

You may choose to do so either by name or by ordinal position.

Depending on the dropdown list control element design you may be able to select a single or multiple items.

For selection by name, multiple selection is available by entering more than one names (one per line). For selection by index, multiple selection is available by entering more than one items' index separated by a space or a - e.g. 1-3-4.

Note that if the Dropdown list only supports single selection and you have specified more than one items then only the first specified will be used.

3.11.12.9 Data Extraction**3.11.12.9 Get Details of Window Action****Description:**

Get a property of a Window, such as its title or its source text.

Properties of 'Get Details of Window' action

Get Details of Window
Get a property of a Window, such as its title or its source text.

General | Exception Handling

Action Input

Control:
(no control selected)  

Select Control From Repository

Operation: Get Window Title 
Get Window Title
Get Window Text
Get Window Location and Size
Get Process Name

Action Output

Store Property Into: %WindowProperty% 

This action is Enabled

More Info OK Cancel

Properties:

Control:

Click on Select Control from Repository button and a box with all controls will appear. Hover over camera icon to preview the control's image.

Operation:

Specify the information of the running Window that you want to retrieve. You have the option to get details such as Window title, Window text, Window location and size.

Store Property Into:

Set the name of the variable that will hold the retrieved information of the Window. The value of this variable can be used in a following action (e.g. displayed through a "Display Message" action).

3.11.12.9,Get Details of Element in Window Action

Description:

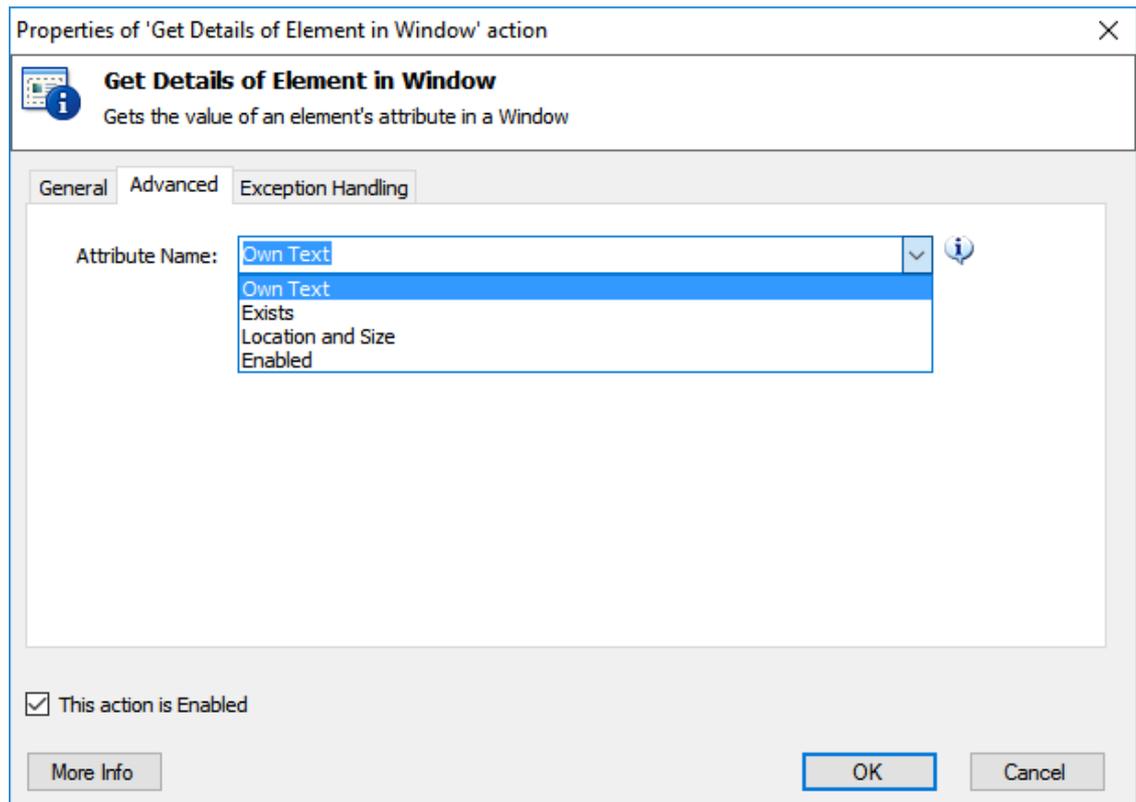
Gets the value of an element's attribute in a Window

The screenshot shows a dialog box titled "Properties of 'Get Details of Element in Window' action". The dialog has a close button (X) in the top right corner. Below the title bar, there is a header section with an information icon and the text "Get Details of Element in Window" and "Gets the value of an element's attribute in a Window".

The main area of the dialog is divided into three tabs: "General", "Advanced", and "Exception Handling". The "General" tab is selected. Under the "General" tab, there are two sections:

- Action Input:** A section with a header "Action Input" and a sub-header "Control:". Below this is a text input field containing "(no control selected)". To the right of the field are a camera icon and an information icon. Below the field is a button labeled "Select Control From Repository".
- Action Output:** A section with a header "Action Output" and a sub-header "Store Element Text Into:". Below this is a text input field containing "%AttributeValue%". To the right of the field is an information icon.

At the bottom of the dialog, there is a checkbox labeled "This action is Enabled" which is checked. Below the checkbox are three buttons: "More Info", "OK", and "Cancel".



Properties:

Control:

Click on Select Control from Repository button and a box with all controls will appear. Hover over camera icon to preview the control's image.

Attribute Name:

You have the option to get details for element attributes such as element's own text, whether it exists or not, its location and size, and whether it is enabled or not. To select the attribute for which the details you want to retrieve use the 'Attribute Name' property menu options in the Advanced Tab.

Store Element Text Into:

Set the name of the variable that will hold the retrieved information of the Element's attribute. The value of this variable can be used in a following action (e.g. displayed through a "Display Message" action).

3.11.12.9,Get Selected Checkboxes in Window Action

Description:

Retrieves the names of the Selected Checkboxes in a Checkbox Group or the State of a specific Checkbox.

Properties of 'Get Selected Checkboxes in Window' action

Get Selected Checkboxes in Window
Retrieves the names of the Selected Checkboxes in a Checkbox Group or the State of a specific Checkbox.

General | Exception Handling

Action Input

Control: (no control selected) [camera icon] [info icon]
[Select Control From Repository]

Operation: Get Names of Selected Checkboxes in Group (selected)
Get Names of Selected Checkboxes in Group
Get State of Checkbox

Action Output

Store Selected Checkboxes into: %SelectedCheckboxes% [info icon]

This action is Enabled

[More Info] [OK] [Cancel]

Properties:

Control:

Click on Select Control from Repository button and a box with all controls will appear. Hover over camera icon to preview the control's image.

Operation:

Specify whether you want to retrieve the names of multiple Checkboxes in the running window or just the State of a Specific Checkbox

Store Selected Checkboxes/Checkbox Into:

Set the name of the variable that will hold the names of the Selected Checkboxes/state of the specific Checkbox. The value of this variable can be used in a following action (e.g. displayed through a "Display Message" action).

3.11.12.9, Get Selected Radiobutton in Window Action

Description:

Retrieves the names of the Selected Radiobutton in a Radiobutton Group or the State of a specific Radiobutton.

The screenshot shows the configuration window for the 'Get Selected Radiobutton in Window' action. The window title is 'Properties of 'Get Selected Radiobutton in Window' action'. The main title is 'Get Selected Radiobutton in Window' with a description: 'Retrieves the name of the Selected Radiobutton in a Radiobutton Group or the State of a specific Radiobutton.' The 'General' tab is selected. Under 'Action Input', the 'Control' field is empty with the text '(no control selected)'. Below it is a 'Select Control From Repository' button. The 'Operation' dropdown menu is open, showing three options: 'Get Selected Radiobutton Name in Group' (selected), 'Get Selected Radiobutton Name in Group', and 'Get State of Radiobutton'. Under 'Action Output', the 'Store Selected Radiobutton into:' field contains the text '%SelectedRadiobutton%'. At the bottom, there is a checkbox 'This action is Enabled' which is checked, a 'More Info' button, and 'OK' and 'Cancel' buttons.

Properties:**Control:**

Click on Select Control from Repository button and a box with all controls will appear. Hover over camera icon to preview the control's image.

Operation:

Specify whether you want to retrieve the name of the Radiobutton that is selected within a Radiobutton Group in the running window or just the State of a Single Radiobutton.

Store Selected Checkboxes/Checkbox Into:

Set the name of the variable that will hold the name of the Selected Radiobutton/state of the specific Radiobutton. The value of this variable can be used in a following action (e.g. displayed through a "Display Message" action).

3.11.12.9 Extract Data from Window Action**Description:**

Extracts Data from specific parts of a Window in the form of single values, lists or tables.

Properties of 'Extract Data from Window' action ✕

 **Extract Data from Window**
Extracts Data from specific parts of a Window in the form of single values, lists, or tables.

General Exception Handling

Action Input

Control:
  

Store extracted data in 

an Excel Spreadsheet
an Excel Spreadsheet
a Variable

Action Output

Store New Excel Instance Into: 

This action is Enabled

Properties of 'Extract Data from Window' action

Extract Data from Window
Extracts Data from specific parts of a Window in the form of single values, lists, or tables.

General | Exception Handling

Action Input

Control:
(no control selected)  

Select Control From Repository

Store extracted data in a Variable 

Action Output

Store Extracted Data Into: %DataFromWindow% 

This action is Enabled

More Info OK Cancel

Properties:

Control:

Click on Select Control from Repository button and a box with all controls will appear. Hover over camera icon to preview the control's image.

Store Extracted data in:

Specify whether you want to store the Extracted Data in an Excel Spreadsheet or as the value of a variable.

Store New Excel Instance Into:

In the case that you have selected to store the Extracted Data in an Excel Spreadsheet, enter the name to be the variable that will hold the Excel Instance with the Extracted Data. This instance can be used in consequent dedicated Excel Actions to manipulate (or save and close) the spreadsheet.

Store Extracted Data Into:

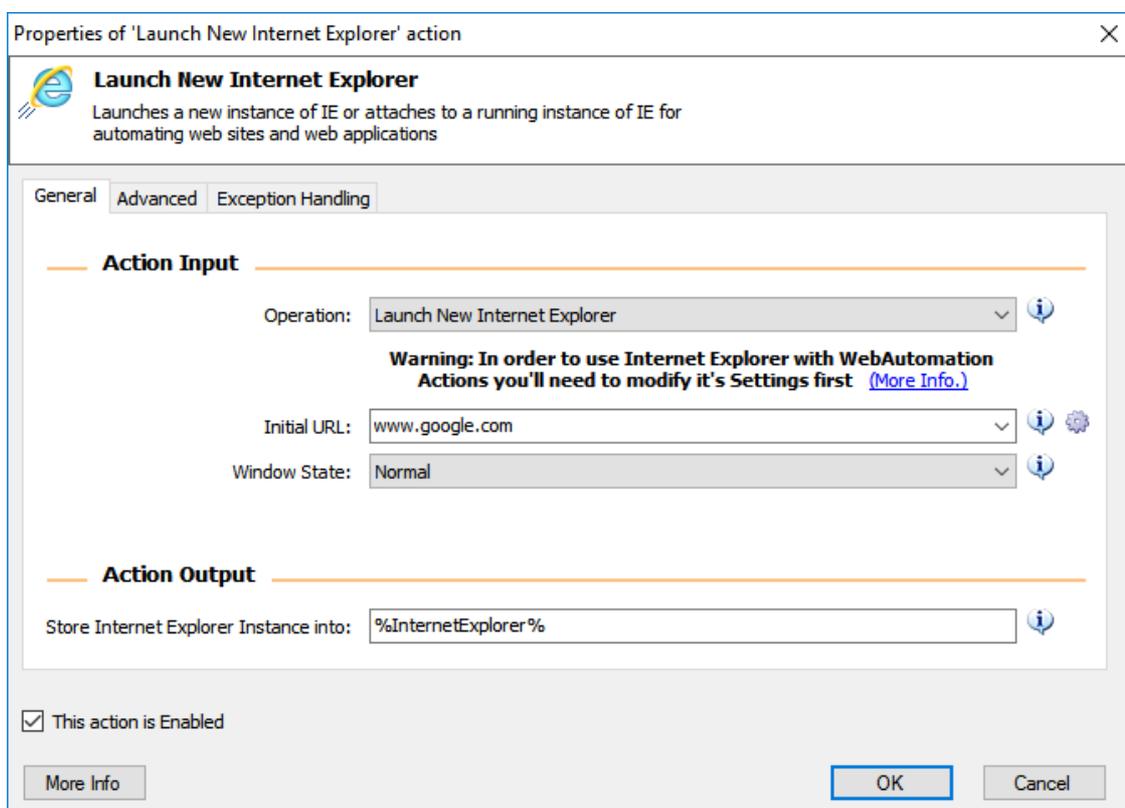
In the case that you have selected to store the Extracted Data as the value of a variable, set the name of the variable that will hold these Data. The value of this variable can be used in a following action (e.g. displayed through a "Display Message" action).

3.11.13 Web Automation

3.11.13.1 Launch New Internet Explorer Action

Description:

Launches a new instance of IE or attaches to a running instance of IE for automating web sites and web applications.



The screenshot shows the 'Properties of 'Launch New Internet Explorer' action' dialog box. The title bar reads 'Properties of 'Launch New Internet Explorer' action'. The main title is 'Launch New Internet Explorer' with a description: 'Launches a new instance of IE or attaches to a running instance of IE for automating web sites and web applications'. There are three tabs: 'General', 'Advanced', and 'Exception Handling'. The 'General' tab is active. Under 'Action Input', there are three fields: 'Operation' (set to 'Launch New Internet Explorer'), 'Initial URL' (set to 'www.google.com'), and 'Window State' (set to 'Normal'). A warning message is displayed: 'Warning: In order to use Internet Explorer with WebAutomation Actions you'll need to modify it's Settings first (More Info.)'. Under 'Action Output', there is a field 'Store Internet Explorer Instance into:' set to '%InternetExplorer%'. At the bottom, there is a checkbox 'This action is Enabled' which is checked, and buttons for 'More Info', 'OK', and 'Cancel'.

Launch New Internet Explorer

Properties of 'Launch New Internet Explorer' action

Launch New Internet Explorer
Launches a new instance of IE or attaches to a running instance of IE for automating web sites and web applications

General Advanced Exception Handling

Action Input

Operation: Attach to Running Internet Explorer ⓘ

Warning: In order to use Internet Explorer with WebAutomation Actions you'll need to modify it's Settings first [\(More Info.\)](#)

Select Internet Explorer: by Title ⓘ
by Title ⓘ
by URL ⓘ
use Foreground Internet Explorer ⓘ

Action Output

Store Internet Explorer Instance into: %InternetExplorer% ⓘ

This action is Enabled

More Info OK Cancel

Attach to running IE

Properties of 'Launch New Internet Explorer' action

Launch New Internet Explorer
Launches a new instance of IE or attaches to a running instance of IE for automating web sites and web applications

General Advanced Exception Handling

Clear Cache ⓘ
 Clear Cookies ⓘ
 Wait for Page to Load ⓘ

If a Pop-Up Dialog Appears: Do Nothing ⓘ
Close it
Press a Button
Do Nothing

This action is Enabled

More Info OK Cancel

Advanced tab of "Launch New Internet Explorer"

Properties:**Operation:**

Select the type of Internet Explorer you wish to work with. Note that if you want to use Internet Explorer to automate your web related tasks you will need to [configure its security settings](#)^[383] first.

Initial URL:

Enter the URL of the web site you want to visit as soon as the web browser is launched.

Window State:

Specify whether the Web Browser window will be launched in normal, minimized or maximized state.

Attach to Internet Explorer Tab:

Choose whether the action will attach to an Internet Explorer Tab by its Title, by its URL, or attach to the Active Tab of the Internet Explorer running as the Foreground Window.

Tab Title:

Specify the title (or part of it) of the Internet Explorer Tab you want to attach to. In the dropdown list you can choose between the titles of all the IE tabs that are currently open.

Tab URL:

Enter the URL of the Internet Explorer Tab you want to attach to.

Store Internet Explorer Instance into:

Enter a name to be the variable that will store the specific Internet Explorer Instance for use with later web automation actions.

Clear Cache:

Specify whether to clear the entire cache of the web browser right after launching it.

Clear Cookies:

Specify whether to clear all stored cookies in the web browser right after launching it.

Wait for Page to Load:

Choose whether you want the Process to wait for the new web page to load completely before proceeding to the next action.

If a pop-up dialog appears:

Specify what you want the Process to do if a pop-up dialog appears while loading the initial web page.

3.11.13.2 Launch New Firefox

Description:

Launches a new instance of Mozilla Firefox or attaches to a running instance of Firefox for automating web sites and web applications.

Properties of 'Launch New Firefox' action

Launch New Firefox
Launches a new instance of Firefox or attaches to a running instance of Firefox for automating web sites and web applications

General | Advanced | Exception Handling

Action Input

Operation: Launch New Firefox

Warning: In order to use Firefox with WebAutomation Actions you'll need to modify it's Settings first (More Info.)

Initial URL: https://en.wikipedia.org/wiki/Domain_controller

Window State: Normal

Action Output

Store Firefox Instance into: %Browser%

This action is Enabled

More Info OK Cancel

Launch New Firefox

Properties of 'Launch New Firefox' action

Launch New Firefox
Launches a new instance of Firefox or attaches to a running instance of Firefox for automating web sites and web applications

General | Advanced | Exception Handling

Action Input

Operation: Attach to Running Firefox ⓘ

Warning: In order to use Firefox with WebAutomation Actions you'll need to modify it's Settings first (More Info.)

Select Firefox: by URL ⓘ

Tab URL: ⓘ ⚙️

Action Output

Store Firefox Instance into: %Browser% ⓘ

This action is Enabled

More Info OK Cancel

Attach to running Firefox Instance

Properties:

Operation:

Select whether you wish to work with an existing running instance of your browser or if you want to launch a new one. You can select either the 'Launch New Firefox' option or the 'Attach to Running Firefox' option.

Action Input

Operation: Attach to Running Firefox ⓘ
Launch New Firefox
Attach to Running Firefox
Warning: In order to use Firefox with WebAutomation Actions you'll need to modify it's Settings first (More Info.)

Select Firefox: by Title ⓘ

Tab Title: ⓘ ⚙️

Action Output

Store Firefox Instance into: %Browser% ⓘ

The 'Attach to Running Firefox' option should be selected when we have (or we are about to have through a running script) an open Instance of Firefox and we want to reference that Instance (or Tab) somewhere within our scripts via a Variable.

Initial URL:

If you have selected to work with a new Firefox Instance (Launch New Firefox) in the Operation drop down menu, the Initial URL text box will invite you to enter the URL of the web site you want to visit as soon as the web browser is launched.

Window State:

If you have selected to work with a new Firefox Instance (Launch New Firefox) in the Operation drop down menu, the Window State text box will invite you to specify whether the new web browser window will be launched in normal, minimized or maximized state.

Select Firefox:

If you have selected to work with an already running Firefox Instance (Attach to Running Firefox) the Select Firefox drop down menu will invite you to specify whether you want to identify that instance via its Title, its URL or simply by the fact that the Instance is running currently in the Foreground:

Action Input

Operation: ⓘ

**Warning: In order to use Firefox with WebAutomation
Actions you'll need to modify it's Settings first [\(More Info.\)](#)**

Select Firefox: ⓘ

Tab Title: ⓘ ⚙️

Action Output

Store Firefox Instance into: ⓘ

Once more, according to our option we will have the opportunity to work with two different text boxes, the Tab URL and Tab Title respectively, that will allow us to enter the info necessary in order to locate the running Instance that is interesting us.

Tab Title:

If you have selected to work with a running Firefox Instance (Attach to Running Firefox) this drop down menu allows you to specify the title (or part of it) of the Firefox Window or Tab you want to use in your scripts. From this drop down list you can choose between the titles of all the Firefox windows or tabs that are currently open.

Tab URL:

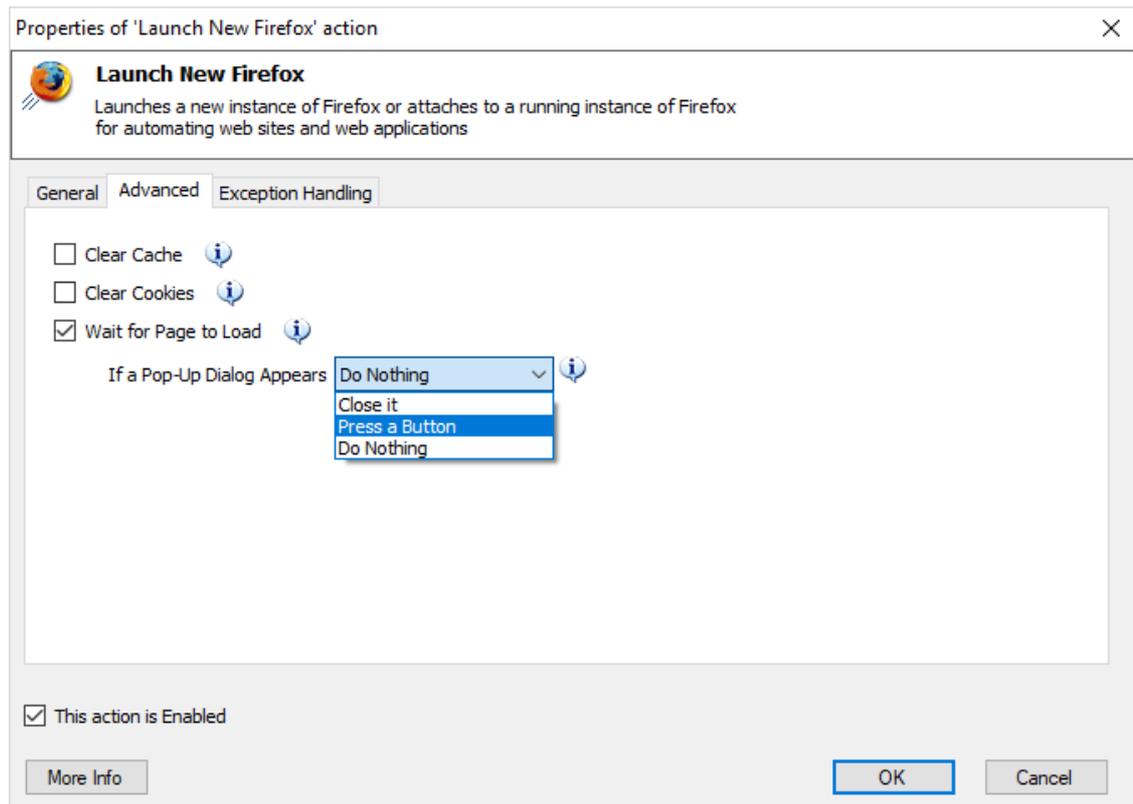
Enter the URL of the Firefox Tab you want to use in your scripts.

Store Firefox Instance into:

Enter a name to be the variable that will store the specific Firefox Instance for use in later web automation actions.

The Advanced tab:

The Advanced tab options are available to you only if you have previously selected to work with a brand new Firefox Instance.



Advanced tab of "Launch New Firefox"

Clear Cache:

Specify whether to clear the entire cache of the web browser right after launching it.

Clear Cookies:

Specify whether to clear all stored cookies in the web browser right after launching it.

Wait for Page to Load:

Choose whether you want the Process to wait for the new web page to load completely before proceeding to the next action.

If a pop-up dialog appears:

Specify what you want the Process to do if a pop-up dialog appears while loading the initial web page.

3.11.13.3 Launch New Chrome

Description:

Launches a new instance of Google Chrome or attaches to a running instance of Chrome for automating web sites and web applications.

Properties of 'Launch New Chrome' action

Launch New Chrome
Launches a new instance of Chrome or attaches to a running instance of Chrome for automating web sites and web applications

General Advanced Exception Handling

Action Input

Operation: Launch New Chrome

Warning: In order to use Chrome with WebAutomation Actions you'll need to modify it's Settings first (More Info.)

Initial URL: https://en.wikipedia.org/wiki/Domain_controller

Window State: Normal

Action Output

Store Chrome Instance into: %Browser%

This action is Enabled

More Info OK Cancel

Launch New Chrome

Properties of 'Launch New Firefox' action

Launch New Firefox
Launches a new instance of Firefox or attaches to a running instance of Firefox for automating web sites and web applications

General | Advanced | Exception Handling

Action Input

Operation: Attach to Running Firefox ⓘ

Warning: In order to use Firefox with WebAutomation Actions you'll need to modify it's Settings first (More Info.)

Select Firefox: by URL ⓘ

Tab URL: ⓘ ⚙️

Action Output

Store Firefox Instance into: %Browser% ⓘ

This action is Enabled

More Info OK Cancel

Attach to running Chrome Instance

Properties:

Operation:

Select whether you wish to work with an existing running instance of your browser or if you want to launch a new one. You can select either the 'Launch New Chrome' option or the 'Attach to Running Chrome' option.

Action Input

Operation: Attach to Running Chrome
Launch New Chrome
Attach to Running Chrome ⓘ

Select Chrome: by Title ⓘ

Tab Title: ⓘ ⚙️

Action Output

Store Chrome Instance into: %Browser% ⓘ

The 'Attach to Running Chrome' option should be selected when we have (or we are about to have through a running script) an open Instance of Chrome and we want to reference that Instance (or Tab) somewhere within our scripts via a Variable.

Initial URL:

If you have selected to work with a brand new Google Chrome Instance (Launch New Chrome) in the Operation drop down menu, the Initial URL text box will invite you to enter the URL of the web site you want to visit as soon as the web browser is launched.

Window State:

If you have selected to work with a brand new Google Chrome Instance (Launch New Chrome) in the Operation drop down menu, the Window State text box will invite you to specify whether the new web browser window will be launched in normal, minimized or maximized state.

Select Chrome:

If you have selected to work with an already running Google Chrome Instance (Attach to Running Chrome) the Select Chrome drop down menu will invite you to specify whether you want to identify that instance via its Title, its URL or simply by the fact that the Instance is running currently in the Foreground:

Action Input

Operation: ⓘ

Warning: In order to use Chrome with WebAutomation Actions you'll need to modify it's Settings first [\(More Info.\)](#)

Select Chrome: ⓘ

Tab Title: ⓘ ⚙️

Action Output

Store Chrome Instance into: ⓘ

Once more, according to our option we will have the opportunity to work with two different text boxes, the Tab URL and Tab Title respectively, that will allow us to enter the info necessary in order to locate the running Instance that is interesting us.

Tab Title:

If you have selected to work with a running Chrome Instance (Attach to Running Chrome) this drop down menu allows you to specify the title (or part of it) of the Chrome Tab you want to work with in your scripts. From this drop down list you can choose between the titles of all the Chrome tabs that are currently open.

Tab URL:

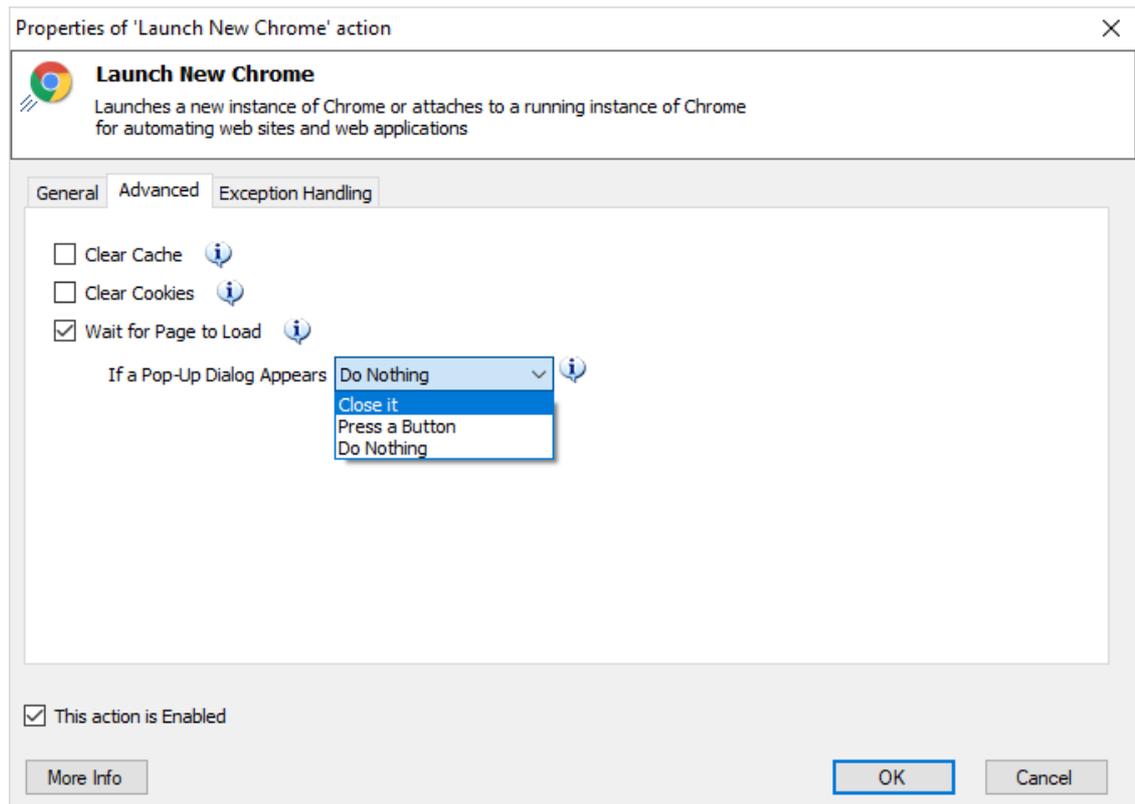
Enter the URL of the Chrome Tab you want to use in your scripts.

Store Chrome Instance into:

Enter a name to be the variable that will store the specific Chrome Instance for use in later web automation actions.

The Advanced tab:

The Advanced tab options are available to you only if you have previously selected to work with a brand new Chrome Instance.



Advanced tab of "Launch New Chrome"

Clear Cache:

Specify whether to clear the entire cache of the web browser right after launching it.

Clear Cookies:

Specify whether to clear all stored cookies in the web browser right after launching it.

Wait for Page to Load:

Choose whether you want the Process to wait for the new web page to load completely before proceeding to the next action.

If a pop-up dialog appears:

Specify what you want the Process to do if a pop-up dialog appears while loading the initial web page.

3.11.13.4 Create New Tab

The Create new Tab action allows you to create a new tab on a running web browser instance. The Web Browser Instance in which the tab will be added is provided to the action via a Variable [1] and the URL you want the tab to navigate to can be provided as either a string or a Variable [2]. The output of the Action is another Variable [3] of type Web Browser Instance.

Properties of 'Create New Tab' action

Create New Tab
Creates a new tab and navigates to the given url.

General | Advanced | Exception Handling

Action Input

Web Browser Instance: %Browser%

URL to navigate to: https://en.wikipedia.org/wiki/Domain_controller

Action Output

Store New Web Browser Instance into: %NewBrowser%

This action is Enabled

More Info OK Cancel

1

The Web Browser Instance:

This drop down menu will present to you a list with all the available variables of web browser instance type. You have not but to select the one that interests you:

Action Input

Web Browser Instance: ⓘ

URL to navigate to: ⓘ ⚙️

Action Output

Store New Web Browser Instance into: ⓘ

2

URL to navigate to:

This text box allows you to either enter directly the URL you want to navigate to (for example https://en.wikipedia.org/wiki/Domain_controller) or to simply press the interactive gear control element that will allow you to select a variable as this property's value.

Properties of 'Create New Tab' action

Create New Tab
Creates a new tab and navigates to the given url.

General | Advanced | Exception Handling

Action Input

Web Browser Instance: ⓘ

URL to navigate to: ⓘ ⚙️

Action Output

Store New Web Browser Instance into:

This action is Enabled

More Info

Select a variable for property "URL to navigate to":

Available Variables:

- %Browser% (WebBrowser Instance)
- %Browser2% (WebBrowser Instance)
- %Browser3% (WebBrowser Instance)
- %NewBrowser% (WebBrowser Instance)

OK Cancel

3

Start New Web Browser Instance into:

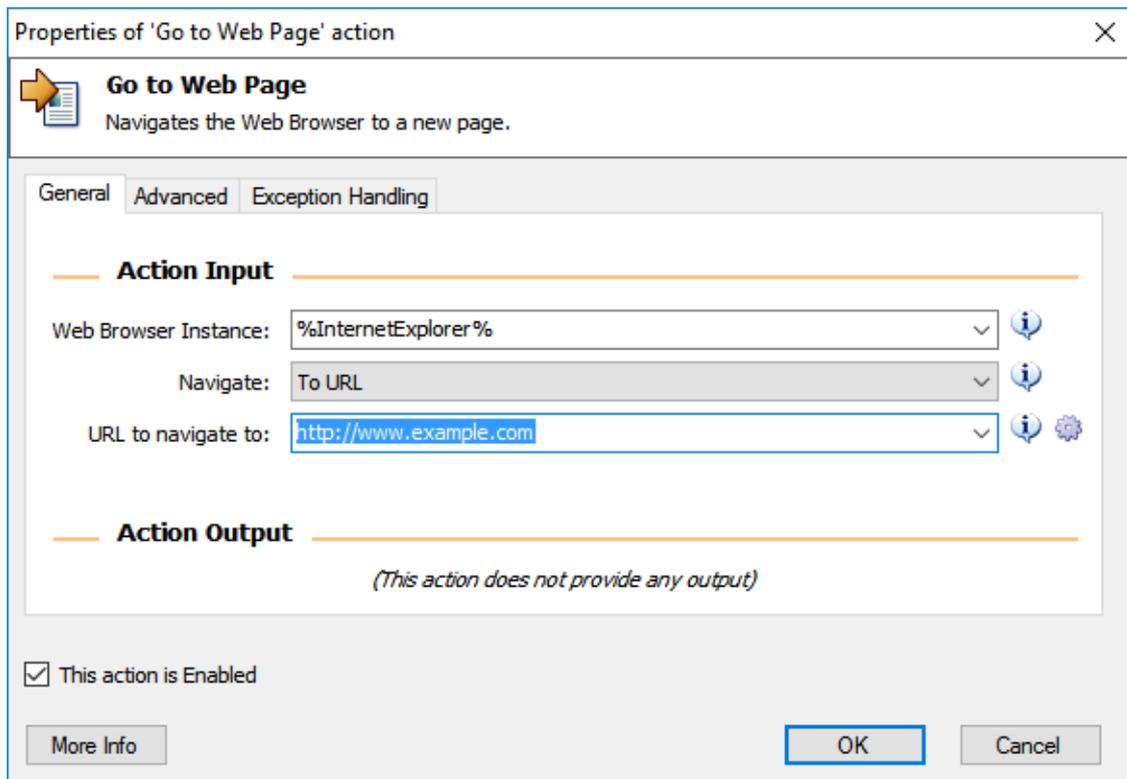
Here you have not but to enter a name as the name of the Variable that will store the resultant web browser instance (the new Tab you have just created) for later use:

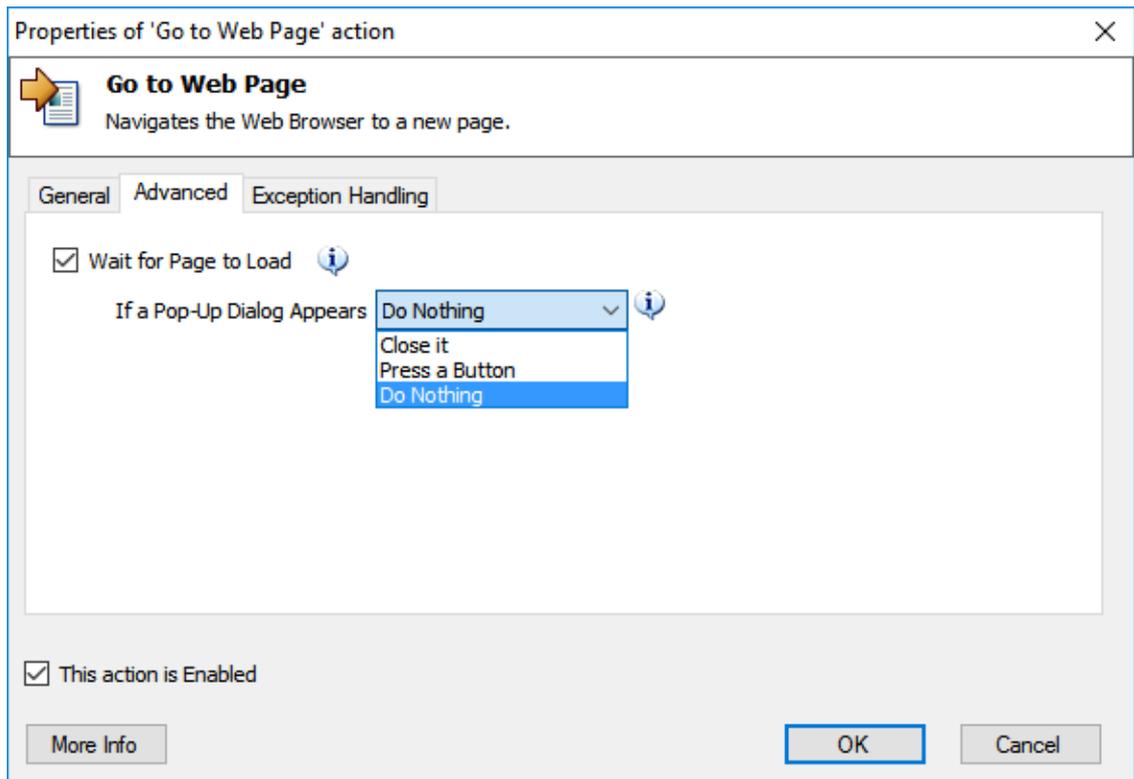


3.11.13.5 Go to Web Page Action

Description:

Navigates the Web Browser to a new page, back, forward or reloads the currently loaded page.





Properties:

Web Browser Instance:

Enter the variable that contains the Web Browser Instance you want to work with. This must be a variable defined by a preceding "[Launch New Internet Explorer](#)"⁷²⁶ action.

Navigate:

Specify whether you want to visit a new URL, refresh the page, move back or forward.

URL to navigate to:

Enter here the URL you want to navigate to.

Wait for Page to Load:

Choose whether you want the Process to wait for the new web page to load completely before proceeding to the next action.

If a pop-up dialog appears:

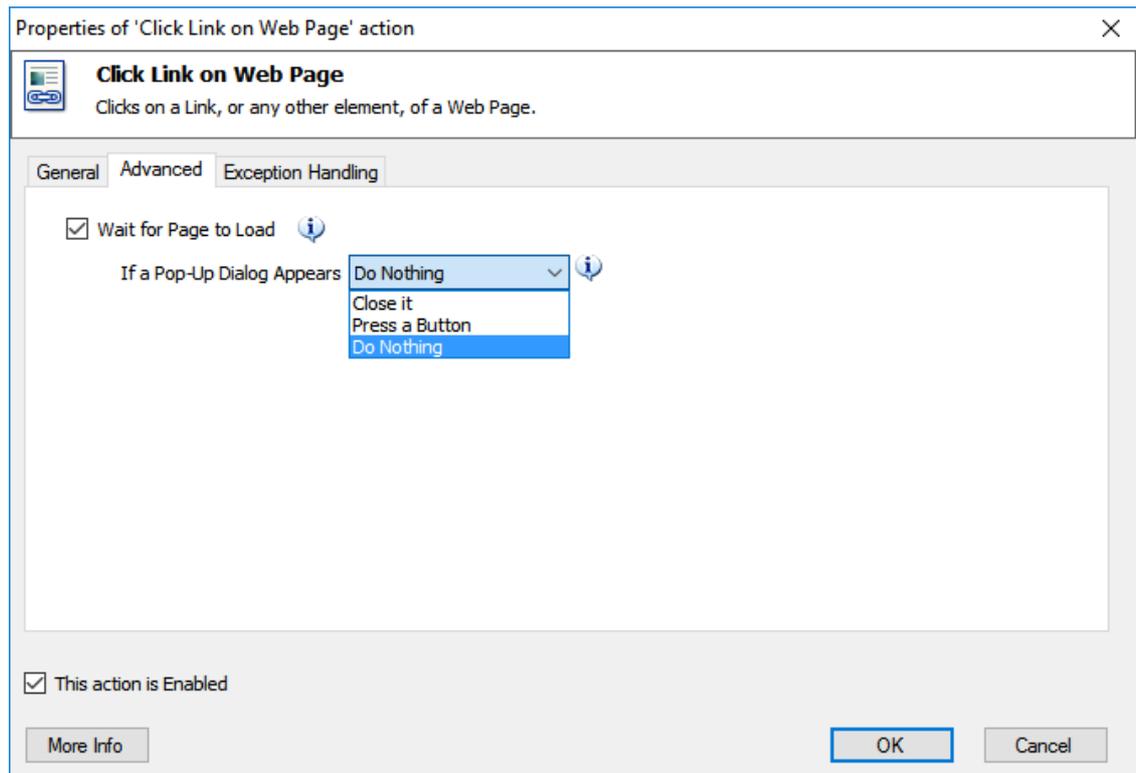
Specify what you want the Process to do if a pop-up dialog appears while loading the web page.

3.11.13.6 Click Link on Web Page Action

Description:

Clicks on a Link, or any other element, of a Web Page.

The screenshot shows the 'Properties of 'Click Link on Web Page' action' dialog box. The title bar reads 'Properties of 'Click Link on Web Page' action'. The main content area is divided into three tabs: 'General', 'Advanced', and 'Exception Handling'. The 'General' tab is selected. Under the 'Action Input' section, there is a 'Web Browser Instance' dropdown menu set to '%InternetExplorer%' and an information icon. Below it is a 'Control' field with the text '(no control selected)' and a camera icon. A 'Select Control From Repository' button is positioned below the 'Control' field. Under the 'Action Output' section, there is a note: '(This action does not provide any output)'. At the bottom left, there is a checked checkbox labeled 'This action is Enabled' and a 'More Info' button. At the bottom right, there are 'OK' and 'Cancel' buttons.



Properties:

Control:

Click on Select Control from Repository button and a box with all controls will appear. Hover over camera icon to preview the control's image.

Wait for Page to Load:

Choose whether you want the Process to wait for the new web page to load completely after clicking on the link.

If a pop-up dialog appears:

Specify what you want the Process to do if a pop-up dialog appears after clicking on the link.

3.11.13.7 Click Download Link on Web Page Action

Description:

Clicks on a link in a Web Page that results in downloading a file.

Properties of 'Click Download Link on Web Page' action

Click Download Link on Web Page
Clicks on a link in a Web Page that results in downloading a file.

General Exception Handling

Action Input

Web Browser Instance: %InternetExplorer%

Control:
(no control selected)

Select Control From Repository

Destination Folder: C:\Users\User\Desktop

Action Output

Store Downloaded File Into: %DownloadedFile%

This action is Enabled

More Info OK Cancel

Properties:

Web Browser Instance:

Enter the variable that contains the Web Browser Instance you want to work with. This must be a variable defined by a preceding "[Launch New Internet Explorer](#)" action.

Control:

Click on Select Control from Repository button and a box with all controls will appear. Hover over camera icon to preview the control's image.

Destination Folder:

Enter the folder where you want the file to be saved.

Store Download File into:

Enter the name to be the variable that will store the file on the disk where the download has been saved. This will be a file path that consists of the download folder as specified above plus the name of the file as provided by the web server.

3.11.13.8 Hover Mouse over element on Web Page Action

Description:

This action hovers the mouse over an element of a Web Page

The screenshot shows the configuration window for the 'Hover Mouse over element on Web Page' action. The window has a title bar with the text 'Properties of 'Hover Mouse over element on Web Page' action' and a close button. Below the title bar is a header section with a mouse cursor icon and the text 'Hover Mouse over element on Web Page' and 'Hovers the mouse over an element of a Web Page.' The main area is divided into two tabs: 'General' (selected) and 'Exception Handling'. Under the 'General' tab, there are sections for 'Action Input' and 'Action Output'. The 'Action Input' section contains a 'Web Browser Instance:' dropdown menu with the value '%InternetExplorer%' and an information icon. Below it is a 'Control:' dropdown menu with the value '(no control selected)', a camera icon, and an information icon. A 'Select Control From Repository' button is located below the 'Control:' dropdown. The 'Action Output' section contains the text '(This action does not provide any output)'. At the bottom of the window, there is a checkbox labeled 'This action is Enabled' which is checked, a 'More Info' button, and 'OK' and 'Cancel' buttons.

Properties:

Web Browser Instance:

Enter the variable that contains the Web Browser Instance you want to close. This must be a variable defined by a preceding "[Launch New Internet Explorer](#)^[726]" action.

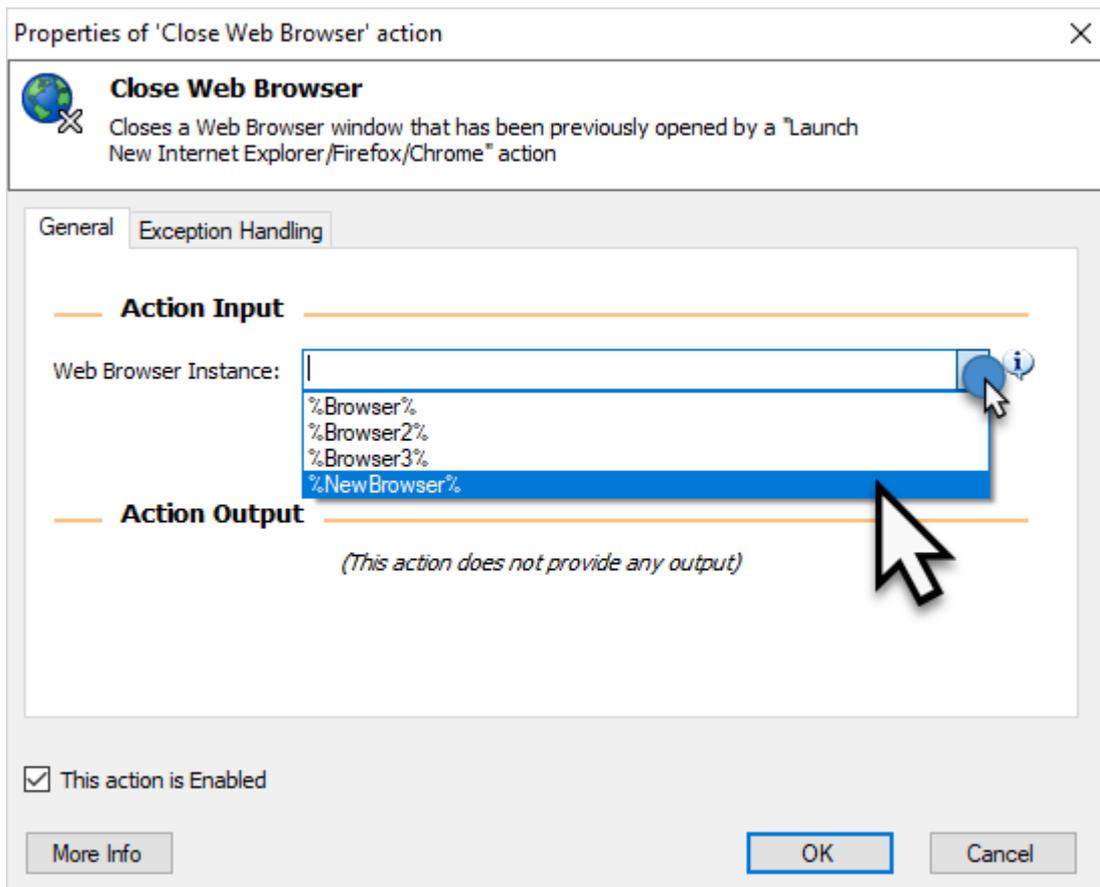
Control:

Click on Select Control from Repository button and a box with all controls will appear. Hover over camera icon to preview the control's image.

3.11.13.9 Close Web Browser

Description:

Closes a Web Browser window that has been previously opened by a "Launch new [Internet Explorer](#)^[726]/[Firefox](#)^[729]/[Chrome](#)^[734]" action.



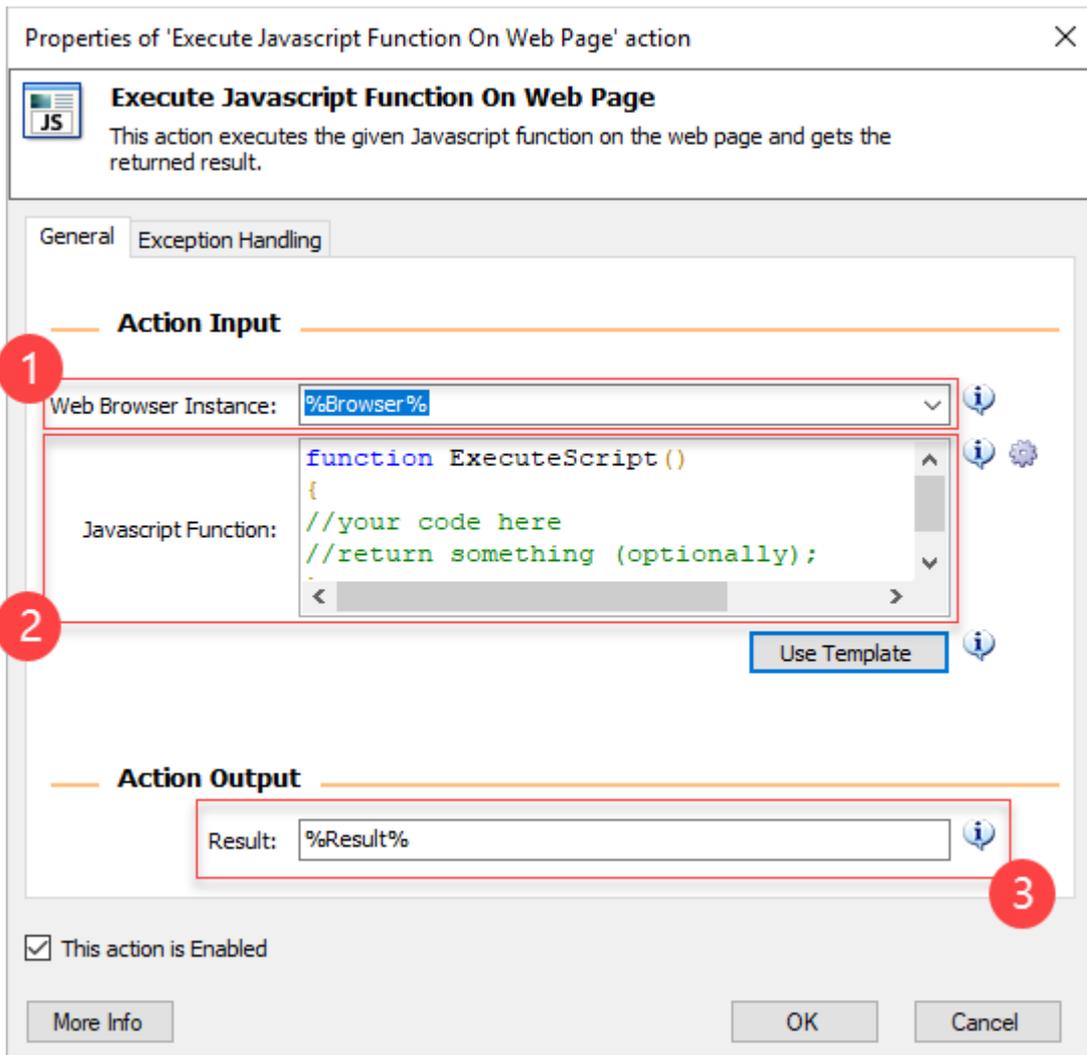
Properties:

Web Browser Instance:

Click on the drop down menu and select the the variable that contains the Web Browser Instance you want to close.

3.11.13.1Execute Javascript Function On Web

This action allows you to execute a given Javascript function [2] on a web page [1] and stores the returned result into a variable for later use [3]:



1 Web Browser Instance:

This drop down menu allows you to enter effortlessly the Browser Instance of your choice, by listing all the variables of that type (web browser instance) in your script:

Action Input

Web Browser Instance:  

JavaScript Function:

```
document.getElementById("#p=...ng-label").
```



Action Output

Result: 

It is worth noting here, that the drop down menu will list even browser instances that haven't yet come up on your workflow, so you might want to be careful about that.

2

JavaScript Function:

The Javascript Function text box, not only allows you to enter your javascript through a variable, using the handy familiar gear next to the info bubble, but it also comes with a little helper on its own, the Use Template button [a]. This button produces the following Javascript template that helps you understand how you should structure your code:

Action Input

Web Browser Instance: %Browser%

Javascript Function:

```
function ExecuteScript ()
{
//your code here
//return something (optionally);
}
```

Use Template

Action Output

Result: %Result%

In this template, you should replace the [single line comments in green](#) with your js code in the

```
function ExecuteScript ()
{
document.getElementById("folderpicker").
}
```

3

Result:

This text box invites you to enter a new or existing variable in order to store in it the result of the Javascript Function from [2] for later use.

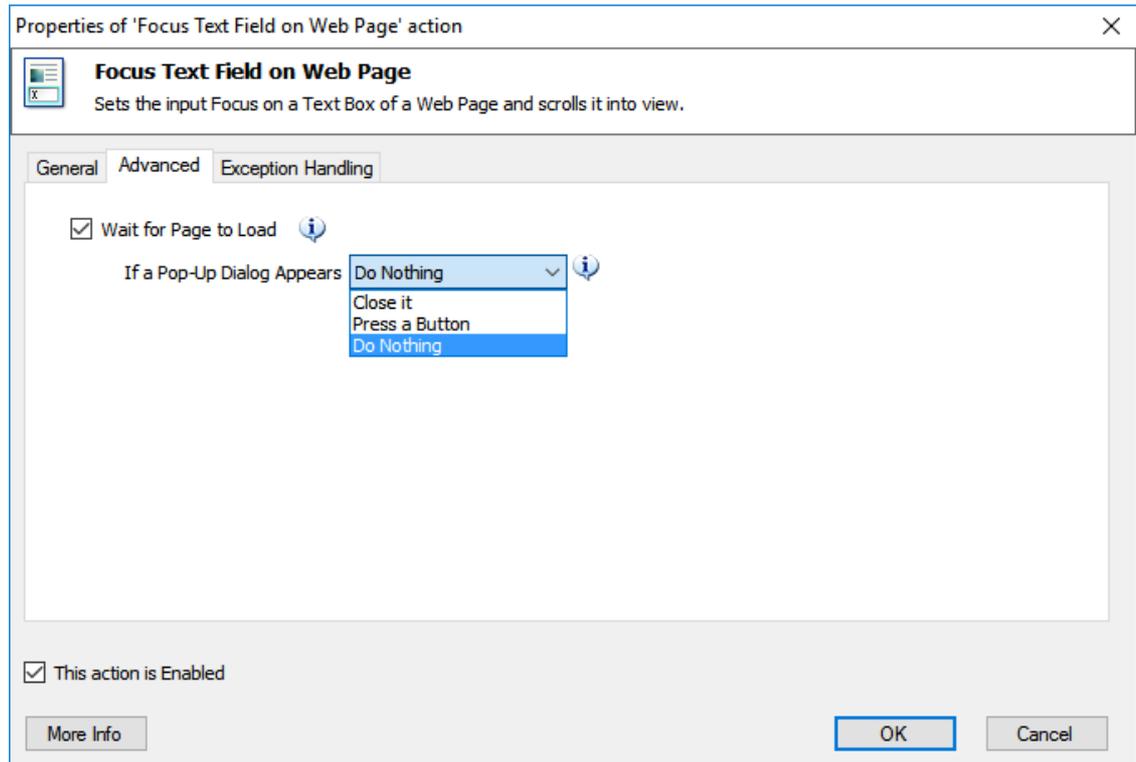
3.11.13.1 Web Forms

3.11.13.1.1 Focus Text Field on Web Page Action

Description:

Sets the input Focus on a Text Box of a Web Page and scrolls it into view.

The screenshot shows the 'Properties of 'Focus Text Field on Web Page' action' dialog box. The title bar reads 'Properties of 'Focus Text Field on Web Page' action'. The main title is 'Focus Text Field on Web Page' with a description: 'Sets the input Focus on a Text Box of a Web Page and scrolls it into view.' Below the title are three tabs: 'General', 'Advanced', and 'Exception Handling'. The 'General' tab is active. Under the 'Action Input' section, there is a 'Web Browser Instance' dropdown menu set to '%InternetExplorer%' with an information icon. Below that is a 'Control' dropdown menu set to '(no control selected)' with a camera icon and an information icon. A 'Select Control From Repository' button is located below the 'Control' dropdown. Under the 'Action Output' section, it states '(This action does not provide any output)'. At the bottom, there is a checkbox labeled 'This action is Enabled' which is checked. There are three buttons at the bottom: 'More Info', 'OK', and 'Cancel'.



Properties:

Web Browser Instance:

Enter the variable that contains the Web Browser Instance you want to work with. This must be a variable defined by a preceding "[Launch New Internet Explorer](#)"⁷²⁶ action.

Control:

Click on Select Control from Repository button and a box with all controls will appear. Hover over camera icon to preview the control's image.

Wait for Page to Load:

Choose whether you want the Process to wait for the new web page to load completely after setting the focus on the text field.

If a pop-up dialog appears:

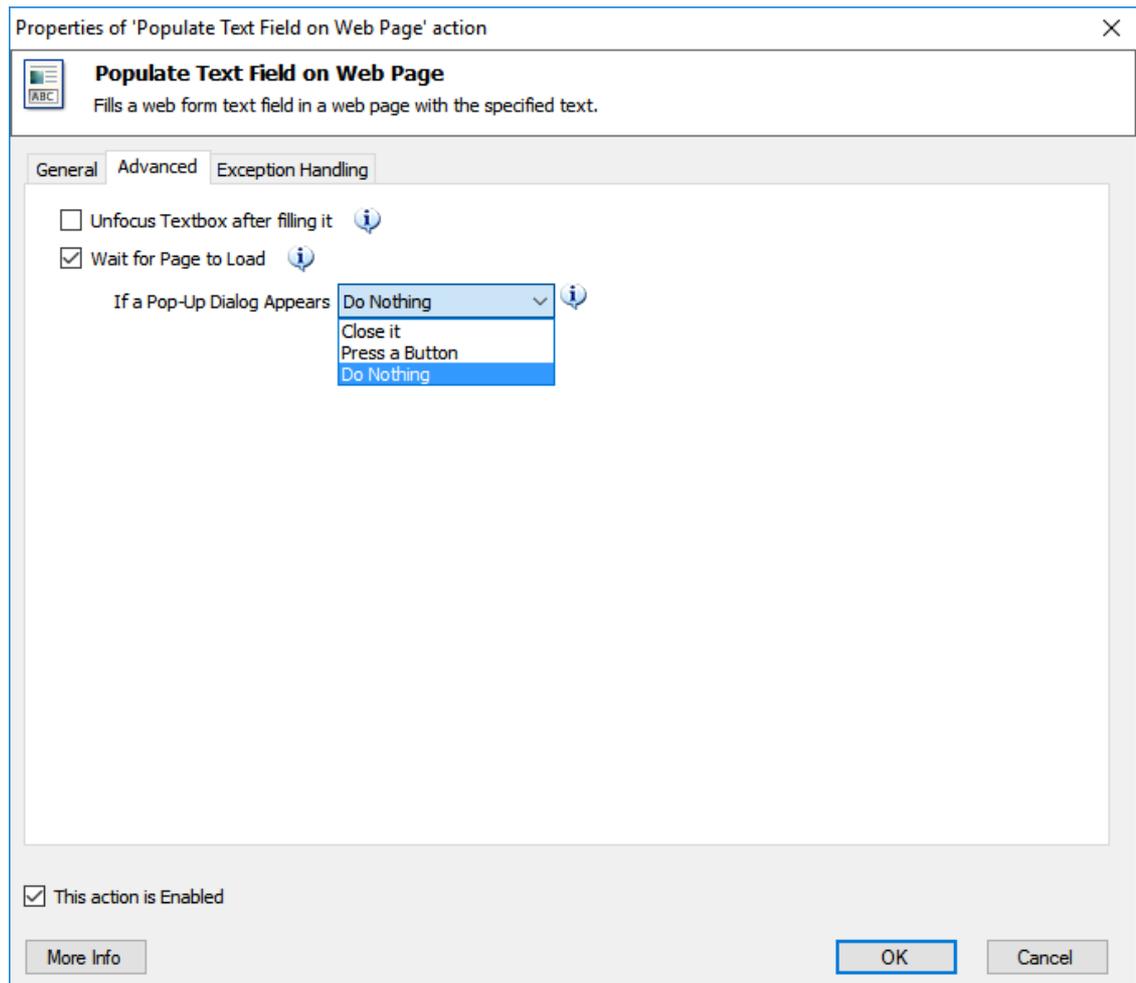
Specify what you want the Process to do if a pop-up dialog appears after setting the focus on the text field.

3.11.13.1 Populate Text Field on Web Page Action

Description:

Fills a web form text field in a web page with the specified text.

The screenshot shows the configuration window for the 'Populate Text Field on Web Page' action. The window title is 'Properties of 'Populate Text Field on Web Page' action'. The main title is 'Populate Text Field on Web Page' with a subtitle 'Fills a web form text field in a web page with the specified text.' The window has three tabs: 'General', 'Advanced', and 'Exception Handling', with 'General' selected. Under the 'Action Input' section, there is a 'Web Browser Instance' dropdown menu set to '%InternetExplorer%' and a 'Control' dropdown menu set to '(no control selected)'. Below the 'Control' dropdown is a 'Select Control From Repository' button. The 'Text to Fill In' field contains 'email@email.com'. There is a checked checkbox for 'Emulate Typing'. Under the 'Action Output' section, it states '(This action does not provide any output)'. At the bottom, there is a checked checkbox for 'This action is Enabled', a 'More Info' button, and 'OK' and 'Cancel' buttons.



Properties:

Web Browser Instance:

Enter the variable that contains the Web Browser Instance you want to work with. This must be a variable defined by a preceding "[Launch New Internet Explorer](#)⁷²⁶" action.

Control:

Click on Select Control from Repository button and a box with all controls will appear. Hover over camera icon to preview the control's image.

Text To Fill-In:

Enter here the text you want to fill in the text field with.

Emulate Typing:

Specify whether you want the text to be filled in at once by simply setting the value of the text box, or emulate a user typing by sending characters one by one. The latter method is slower, but required in some complex web pages.

Unfocus Text after filling it:

Choose whether you want the text field to become unfocused right after this action fills it with the text you've specified.

If you are planning on scraping autocomplete lists, you most probably need to leave this option unchecked.

Wait for Page to Load:

Choose whether you want the Process to wait for the new web page to load completely after populating the text field.

If a pop-up dialog appears:

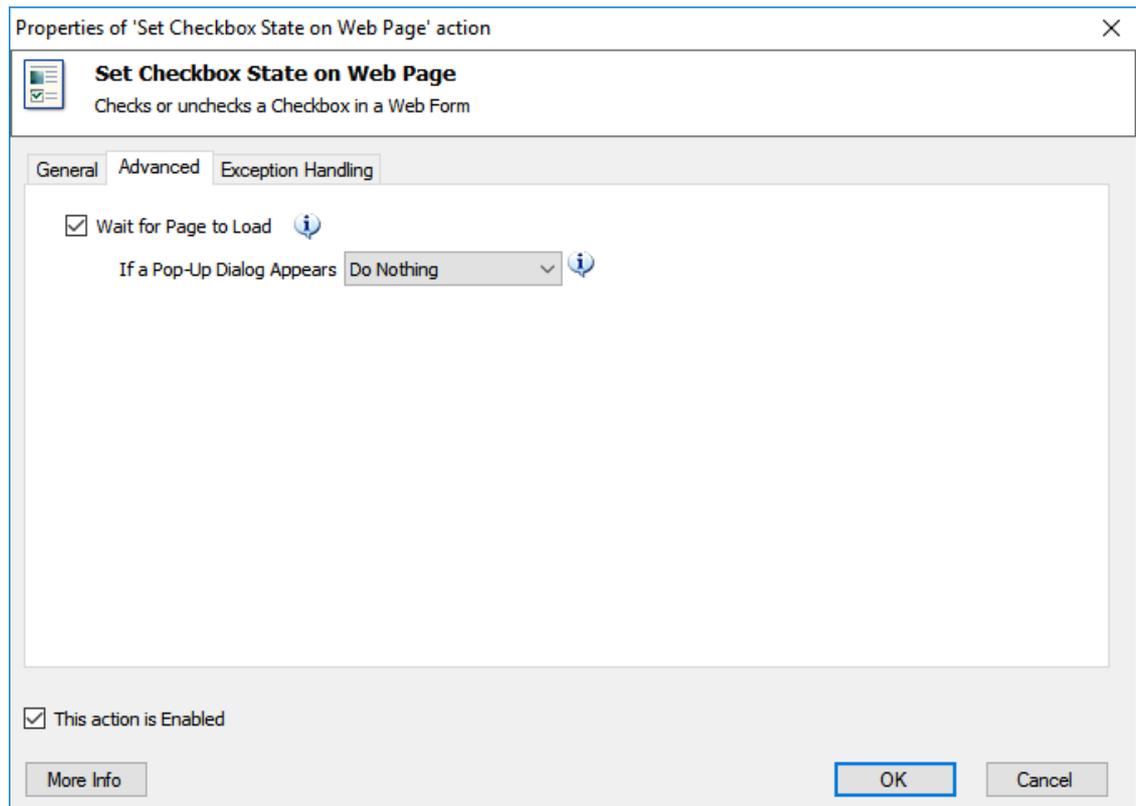
- Specify what you want the Process to do if a pop-up dialog appears after populating the text field.

3.11.13.13 Set Checkbox State on Web Page Action

Description:

Checks or unchecks a Checkbox in a Web Form.

The screenshot shows the configuration window for the 'Set Checkbox State on Web Page' action. The window title is 'Properties of 'Set Checkbox State on Web Page' action'. The main title is 'Set Checkbox State on Web Page' with a subtitle 'Checks or unchecks a Checkbox in a Web Form'. There are three tabs: 'General', 'Advanced', and 'Exception Handling', with 'General' selected. The 'Action Input' section includes a 'Web Browser Instance' dropdown set to '%InternetExplorer%' and a 'Control' dropdown set to '(no control selected)'. Below the 'Control' dropdown is a 'Select Control From Repository' button. The 'Set Checkbox State to:' dropdown is set to 'Checked'. The 'Action Output' section is empty, with the text '(This action does not provide any output)'. At the bottom, there is a checkbox labeled 'This action is Enabled' which is checked, and buttons for 'More Info', 'OK', and 'Cancel'.



Properties:

Web Browser Instance:

Enter the variable that contains the Web Browser Instance you want to work with. This must be a variable defined by a preceding "[Launch New Internet Explorer](#)"⁷²⁶ action.

Control:

Click on Select Control from Repository button and a box with all controls will appear. Hover over camera icon to preview the control's image.

Set Checkbox State to:

Select whether you want the checkbox to become checked or unchecked.

Wait for Page to Load:

Choose whether you want the Process to wait for the new web page to load completely after setting the checkbox state.

If a pop-up dialog appears:

Specify what you want the Process to do if a pop-up dialog appears after setting the checkbox state.

3.11.13.1 Select Radio Button on Web Page Action

Description:

Selects a Radio Button on the Web Page.

Properties of 'Select Radio Button on Web Page' action

Select Radio Button on Web Page
Selects a Radio Button on the Web Page

General | Advanced | Exception Handling

Action Input

Web Browser Instance: %InternetExplorer%

Control:
(no control selected)

Select Control From Repository

Action Output
(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties of 'Select Radio Button on Web Page' action

Select Radio Button on Web Page
Selects a Radio Button on the Web Page

General | Advanced | Exception Handling

Wait for Page to Load

If a Pop-Up Dialog Appears Do Nothing

This action is Enabled

More Info OK Cancel

Properties:**Web Browser Instance:**

Enter the variable that contains the Web Browser Instance you want to work with. This must be a variable defined by a preceding "Launch New Internet Explorer" action.

Control:

Click on Select Control from Repository button and a box with all controls will appear. Hover over camera icon to preview the control's image.

Wait for Page to Load:

Choose whether you want the Process to wait for the new web page to load completely after selecting the radio button.

If a pop-up dialog appears:

Specify what you want the Process to do if a pop-up dialog appears after selecting the radio button.

3.11.13.1 Set DropDown List Value on Web Page Action**Description:**

Set or clears the Selected Option for a Drop Down List in a Web Form.

Properties of 'Set DropDown List Value on Web Page' action

Set DropDown List Value on Web Page

Set or clears the Selected Option for a Drop Down List in a Web Form

General | Advanced | Exception Handling

Action Input

Web Browser Instance: %InternetExplorer%

Control:
(no control selected)

Select Control From Repository

Operation: Clear Selected Options

Action Output
(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties of 'Set DropDown List Value on Web Page' action

Set DropDown List Value on Web Page

Set or clears the Selected Option for a Drop Down List in a Web Form

General | Advanced | Exception Handling

Wait for Page to Load

If a Pop-Up Dialog Appears Do Nothing

This action is Enabled

More Info OK Cancel

Properties:**Web Browser Instance:**

Enter the variable that contains the Web Browser Instance you want to work with. This must be a variable defined by a preceding "[Launch New Internet Explorer](#)"⁷²⁶ action.

Control:

Click on Select Control from Repository button and a box with all controls will appear. Hover over camera icon to preview the control's image.

Operation:

Select whether you want to select a value or clear the selected value of the dropdown list.

Selection Option(s) by:

Choose the method to use in order to select the options: By name or by ordinal position (1 2 3 ...) inside the list.

Option Names:

Enter one or more options (one per line if more than one) you want to be selected in the dropdown list. Multiple options make sense only when working with multi-select lists. If the list is single-selection, then only the first option specified will be used.

Options Indexes:

Type a space-separated list of 1-based indexes (e.g. 1 3 4), denoting which options of the list should be selected.

Wait for Page to Load:

Choose whether you want the Process to wait for the new web page to load completely after setting the value for the dropdown list.

If a pop-up dialog appears:

Specify what you want the Process to do if a pop-up dialog appears after setting the value for the dropdown list.

3.11.13.1 Press Button on Web Page Action**Description:**

Presses a button on a Web Page Form.

Properties of 'Press Button on Web Page' action

Press Button on Web Page
Presses a Web Page Button.

General | Advanced | Exception Handling

Action Input

Web Browser Instance: %InternetExplorer%

Control:
(no control selected)

Select Control From Repository

Action Output
(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties of 'Press Button on Web Page' action

Press Button on Web Page
Presses a Web Page Button.

General | Advanced | Exception Handling

Wait for Page to Load

If a Pop-Up Dialog Appears Do Nothing

This action is Enabled

More Info OK Cancel

Properties:**Web Browser Instance:**

Enter the variable that contains the Web Browser Instance you want to work with. This must be a variable defined by a preceding "[Launch New Internet Explorer](#)"⁷²⁶ action.

Control:

Click on Select Control from Repository button and a box with all controls will appear. Hover over camera icon to preview the control's image.

Wait for Page to Load:

Choose whether you want the Process to wait for the new web page to load completely after the button is pressed.

If a Popup Dialog Appears:

Specify what you want the Process to do if a pop-up dialog appears after pressing the button.

3.11.13.1 Web Data Extraction**3.11.13.1 Extract Data from Web Page Action****Description:**

Extracts Data from specific parts of a Web Page in the form of single values, lists, or tables.

Properties of 'Extract Data from Web Page' action

Extract Data from Web Page

Extracts Data from specific parts of a Web Page in the form of single values, lists, or tables.

General Exception Handling

Action Input

Web Browser Instance: %InternetExplorer%

Synopsis of Data to be Extracted: **No data specified for extraction.**

Specify Web Data to Extract

(bringing an actual Internet Explorer window to the foreground while this dialog is open will automatically activate the live version of this helper.)

Store Extracted Records into

- an Excel Spreadsheet
- an Excel Spreadsheet
- a Variable

Action Output

Store New Excel Instance Into: %ExcelInstance%

This action is Enabled

More Info OK Cancel

Store extracted data into Excel

Properties of 'Extract Data from Web Page' action

Extract Data from Web Page
Extracts Data from specific parts of a Web Page in the form of single values, lists, or tables.

General Exception Handling

Action Input

Web Browser Instance: %InternetExplorer%

Synopsis of Data to be Extracted: **No data specified for extraction.**

Specify Web Data to Extract

(bringing an actual Internet Explorer window to the foreground while this dialog is open will automatically activate the live version of this helper.)

Store Extracted Records into: a Variable

Action Output

Store Extracted Data Into: %DataFromWebPage%

This action is Enabled

More Info OK Cancel

Store extracted data into a Variable

Properties:

Web Browser Instance:

Enter the variable that contains the Web Browser Instance you want to work with. This must be a variable defined by a preceding "[Launch New Internet Explorer](#)⁷²⁶" action.

"Specify Web Data to Extract" button:

To specify which parts of a web page will be extracted you can press this button to launch the "[Data Extraction Web Helper](#)³⁷³". Through this tool you can select the elements that you want to retrieve as well as the form in which you want to retrieve the values. Depending on your selection, the result can be in a form of a list, a table or a single value.

Store Extracted Records into:

Specify whether you want to store the extracted data into a variable for later processing or you want it directly exported into an Excel spreadsheet (you need to have Microsoft Excel installed for the latter option).

Store New Excel Instance Into:

Enter a name to be the variable that will hold the Excel Instance with the Extracted Data. You can use this instance to manipulate the spreadsheet (or save and close it) by using the dedicated Excel Actions.

Store Extracted Data Into:

Enter a name to be the variable that will hold the Extracted Data.

3.11.13.12 Get Details of Web Page Action

Description:

Get a property of a Web Page, such as its title or its source text.

The screenshot shows a dialog box titled "Properties of 'Get Details of Web Page' action". The dialog has a close button (X) in the top right corner. Below the title bar, there is an information icon (i) and the text "Get Details of Web Page" followed by the description "Get a property of a Web Page, such as its title or its source text." The main area of the dialog is divided into sections. The "General" section is active and contains two sub-sections: "Action Input" and "Action Output". Under "Action Input", there are two dropdown menus: "Web Browser Instance" with the value "%InternetExplorer%" and "Operation" with the value "Get Web Page Description". Under "Action Output", there is a text input field labeled "Store Property Into" with the value "%WebPageProperty%". At the bottom left, there is a checkbox labeled "This action is Enabled" which is checked. At the bottom right, there are three buttons: "More Info", "OK", and "Cancel".

Properties:

Web Browser Instance:

Enter the variable that contains the Web Browser Instance you want to work with. This must be a variable defined by a preceding "[Launch New Internet Explorer](#)"⁷²⁶ action.

Operation:

Specify the information of the Web Page that you want to retrieve.

Get Web Page Description
 Get Web Page Meta-Keywords
 Get Web Page Title
 Get Web Page Text
 Get Web Page Source
 Get Web Browser's Current URL Address

Store Property Into:

Set the name of the variable that will hold the retrieved information of the Web Page.

3.11.13.12 Get Details of Element on Web Page Action**Description:**

Gets the value of an element's attribute on a Web Page.

Properties of 'Get Details of Element on Web Page' action

Get Details of Element on Web Page
 Gets the value of an element's attribute on a Web Page

General | Advanced | Exception Handling

Action Input

Web Browser Instance: %InternetExplorer%

Control:
 (no control selected)

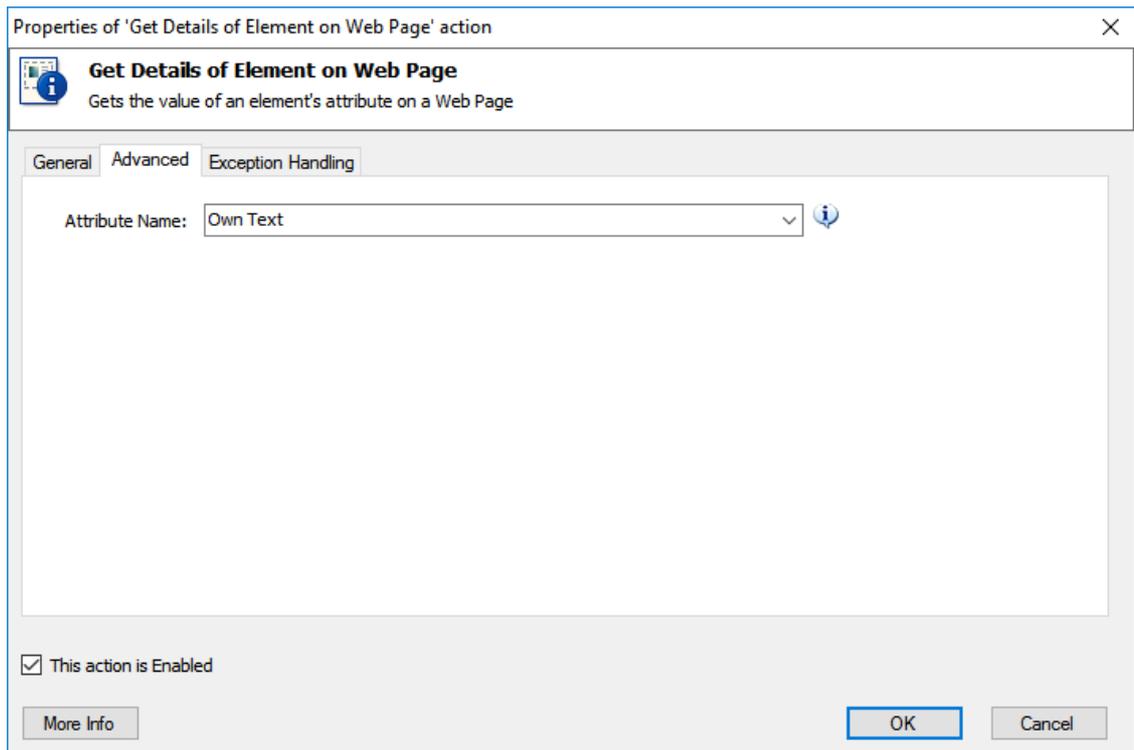
Select Control From Repository

Action Output

Store Web Element Attribute Into: %AttributeValue%

This action is Enabled

More Info | OK | Cancel



Properties:

Web Browser Instance:

Enter the variable that contains the Web Browser Instance you want to work with. This must be a variable defined by a preceding "[Launch New Internet Explorer](#)"⁷²⁶ action.

Control:

Click on Select Control from Repository button and a box with all controls will appear. Hover over camera icon to preview the control's image.

Store Web Element Attribute into:

Enter a name to be the variable that will hold the value of the Web Element's attribute.

3.11.13.12 Take Screenshot of Web Page Action

Description:

Takes a screenshot of the Web Page (or an element of the Web Page) currently displayed in the browser and saves the image into a file or to the clipboard.

Properties of 'Take Screenshot of Web Page' action

Take Screenshot of Web Page
Takes a screenshot of the Web Page (or an element of the Web Page) currently displayed in the browser and saves the image into a file or to the clipboard

General | Exception Handling

Action Input

Web Browser Instance: [%InternetExplorer%] ⓘ

Operation: Capture Specific Element ⓘ

Control:
(no control selected) ⓘ

Select Control From Repository

Save Captured Image to: File ⓘ

Image File: C:\MyData\Pictures\example.png ⓘ ⚙️ 📄

Image Format: Png ⓘ

Action Output
(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties:

Web Browser Instance:

Enter the variable that contains the Web Browser Instance you want to work with. This must be a variable defined by a preceding "[Launch New Internet Explorer](#)⁷²⁶" action.

Operation:

Specify whether you want to capture the entire web page or only a specific element of it.

Control:

Click on Select Control from Repository button and a box with all controls will appear. Hover over camera icon to preview the control's image.

Save Captured Image to:

Choose whether you want to save the image into a file or store it into the clipboard.

Image File:

Set the full path for the file where the image capture will be saved.

Image Format:

Set the format for the image file to be saved.

3.11.13.1: Direct Web Access

3.11.13.1: Download from Web Action

Description:

This action downloads text from a web page, or a file from the web, and stores it.

The screenshot shows the 'Properties of 'Download from Web' action' dialog box. The title bar reads 'Properties of 'Download from Web' action'. The main content area is titled 'Download from Web' and includes the description: 'This action downloads text from a web page, or a file from the web, and stores it.' Below this, there are three tabs: 'General', 'Advanced', and 'Exception Handling'. The 'General' tab is selected. Under the 'Action Input' section, there are three fields: 'URL' with a dropdown set to 'http://' and a text box containing 'www.google.com/search?q=WinAutomation'; 'Method' with a dropdown set to 'GET'; and 'Return Data' with a dropdown set to 'Get text into variable (for web pages)'. Under the 'Action Output' section, there is a text box labeled 'Store Web Page Text into:' containing '%WebPageText%'. At the bottom left, there is a checked checkbox labeled 'This action is Enabled'. At the bottom right, there are 'More Info', 'OK', and 'Cancel' buttons.

Properties of 'Download from Web' action

Download from Web
This action downloads text from a web page, or a file from the web, and stores it.

General | **Advanced** | Exception Handling

Connection Timeout: 30 seconds

Follow Redirection

Clear Cookies

User Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.8.1.21) Gecko

Encoding: Auto-detect

HTTP Authentication

User Name: %Username%

Enter Password: as variable

Password: %Password%

This action is Enabled

More Info OK Cancel

Properties:

URL:

First choose http or https protocol, then enter the remainder of the web address.

Method:

Choose how you want to retrieve a website's information. Choose GET if all information needed is in the URL, and choose POST if you need to enter more information (passwords, etc).

Return Data:

Specify how the returned data will be stored. If you are downloading a webpage, you can choose Get Text so you can access it directly. Otherwise, or if you are downloading a file, choose Save to Disk.

Post Parameters:

Click here to Add, Edit, or Delete details for a Post's Parameter(s).

Connection Timeout:

Specify the time (in seconds) that the agent should wait for a connection to be established with the server, before giving up.

Follow Redirection:

Choose whether you want to allow the web server to redirect you to another web page or website.

Clear Cookies:

Choose whether you want to clear all cookies created by similar actions during this Process before this action. Note that ProcessRobot does not interact with existing cookies, and that old cookies are not used or affected by the Process, and no new cookies are made by the Process. Therefore this only applies to cookies created by other Actions in this Process.

User Agent:

Choose which browser identity you wish to be seen as. Some web servers will not allow you access unless you choose a browser identity.

Encoding:

Choose the encoding used for the web page. If you choose Auto-detect the encoding to be used will be specified by the web server.

File Name:

Choose whether you want to keep the original name of the downloaded file, or you want to specify a new name. In the first case you will need to provide just the destination folder where the file will be saved. In the second case you will need to provide the full path of the downloaded file (destination folder plus the new file name).

Destination File Path:

Enter or choose a file path, or a previously stored variable, to specify the full path (folder plus filename) where the file returned by the web server will be stored.

Destination Folder:

Enter or choose a local folder, or a previously stored variable containing a folder, to specify the folder where the file returned from the web server will be saved.

Store Web Page Text into:

Enter a name to be the variable that will store the Webpage text.

Store Downloaded File into:

Enter a name to be the variable that will store the file object.

HTTP Authentication:

Specify whether or not the web server requires authentication. This property refers to HTTP authentication (that is, when the browser displays a pop-up window asking for user name and password).

User Name:

Enter the User Name for the web server here.

Enter Password:

If you choose 'directly', the password entered in the Password field will be hidden. If you choose 'as variable' it will expect you to enter a variable and treat the '%' as an indicator of a variable, not part of the password.

Password:

Enter the Password here. You may enter either a text, or a variable containing the password depending on the selection of the previous property (Enter Password).

3.11.13.13 Invoke Web Service Action***Description:***

This action invokes a web service by sending data through HTTP and retrieving the response from the web service.

Properties of 'Invoke Web Service' action ✕

 **Invoke Web Service**
This action invokes a web service by sending data and retrieving the response from the web service.

General **Advanced** Exception Handling

Action Input

URL: ⓘ ⚙️

Method: ⓘ

Accept: ⓘ

Content Type: ⓘ

Custom Headers: ⓘ ⚙️

Request Body: ⓘ ⚙️

Return Data: ⓘ

Action Output

Store Headers Into: ⓘ

Store Web Service Response into: ⓘ

Store Status Code: ⓘ

This action is Enabled

Properties of 'Invoke Web Service' action

Invoke Web Service
This action invokes a web service by sending data and retrieving the response from the web service.

General | **Advanced** | Exception Handling

Connection Timeout: 30 seconds

Follow Redirection

Clear Cookies

Fail on Error Status

Encode Request Body

User Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.8.1.21) Gecko

Encoding: Auto-detect

HTTP Authentication

User Name: %Username%

Enter Password: as variable

Password: %Password%

This action is Enabled

More Info OK Cancel

Properties:

URL:

First choose http or https format, then enter the remainder of the web address.

Method:

Choose the HTTP method you want to use to invoke the web service.

Accept:

Specify the acceptable content type for the response of the web service.

Content Type:

Specify the content type of the request that will be sent to the web service.

Custom Headers:

Specify any custom headers to be included in the request that will be sent to the web service.

Request Body:

Specify the body of the request that will be sent to the web service.

Return Data:

Specify how the Web Service response will be stored. You can either store it into a variable or (if the Web Service will return a file) store it into a file by choosing Save to Disk.

File Name:

Choose whether you want to keep the original name of the downloaded file, as provided by the Web Service, or you want to specify a new name. In the first case you will need to provide just the destination folder where the file will be saved. In the second case you will need to provide the full path of the downloaded file (destination folder plus the new file name).

Destination File Path:

Enter or choose a file path, or a previously defined variable, to specify the full path (folder plus filename) where the file returned by the Web Service will be stored.

Destination Folder:

Enter or choose a local folder, or a previously defined variable containing a folder, to specify the folder where the file returned by the Web Service will be saved.

Store Headers Into:

Enter a name to be the variable that will store the HTTP Headers of the response.

Store Web Service Response into:

Enter a name to be the variable that will store the Web Service response.

Store Downloaded File into:

Enter a name to be the variable that will store the file object.

Advanced tab**Connection Timeout:**

Specify the time (in seconds) that the agent should wait for a connection to be established with the server, before giving up.

Follow Redirection:

Choose whether you want to allow the web server to redirect you to another web service.

Clear Cookies:

Choose whether you want to clear all cookies created by similar actions during this Process before this action. Note that ProcessRobot does not interact with existing cookies, and that old cookies are not used or affected by the Process, and no new cookies are made by the Process. Therefore this only applies to cookies created by other Actions in this Process.

Fail on Error Status:

Choose whether you want the responses of the invoked web service that denote errors to be processed as if they were normal responses (suppressing all exceptions) or whether you would like them to result in the related exceptions displayed in the Exception Handling Tab)

Encode Request Body:

Specify whether the body of the request should be url-encoded before invoking.

HTTP Authentication:

Specify whether or not the web server requires authentication. This property refers to HTTP authentication (that is, when the browser displays a pop-up window asking for user name and password).

User Name:

Enter the User Name for the web server here.

Enter Password:

If you choose 'directly', the password entered in the Password field will be hidden. If you choose 'as variable' it will expect you to enter a variable and treat the '%' as an indicator of a variable, not part of the password.

Password:

Enter the Password here. You may enter either a text, or a variable containing the password depending on the selection of the previous property (Enter Password).

3.11.14 FTP**3.11.14.1 Open FTP Connection Action*****Description:***

This action establishes a connection to a remote FTP server

Properties of 'Open FTP Connection' action

Open FTP Connection
This action establishes a specific connection to a remote FTP server, and stores that connection as a variable for later use.

General Exception Handling

Action Input

FTP Server: ⓘ ⚙️

Port: ⓘ

Active Mode ⓘ

User Name: ⓘ ⚙️

Enter Password: ⓘ

Password: ⓘ

Action Output

Store Connection into: ⓘ

This action is Enabled

Properties:**Ftp Server:**

Enter the FTP Server address here.

Port:

Choose which port to use for the FTP Server. Usually, this is Port 25.

Active Mode:

Uncheck this box if you want the connection to be made in passive mode. Default is Active Mode, as that is most common.

User Name:

Enter the User Name for this FTP Server here.

Enter Password:

If you choose 'directly', the password entered in the Password field will be hidden. If you choose 'as variable' it will expect you to enter a variable and treat the '%' as an indicator of a variable, not part of the password.

Password:

Enter the Password here. You may enter either a text, or a variable containing the password depending on the selection of the previous property (Enter Password).

Store Connection into:

Enter a name to be the variable that will store the specific FTP Connection for use with later FTP Actions. This allows you to specify which, of possibly several, FTP Connection to access.

3.11.14.2 Open Secure FTP Connection Action***Description:***

This action establishes a secure connection to a remote FTP server

Properties of 'Open Secure FTP Connection' action

Open Secure FTP Connection
This action establishes a specific secure connection to a remote FTP server, and stores that connection as a variable for later use.

General Exception Handling

Action Input

FTP Server: ⓘ ⚙️

Port: ⓘ

Secure FTP Protocol: ⓘ

Authentication Method: ⓘ

User Name: ⓘ ⚙️

Enter Password: ⓘ

Password: ⓘ

Action Output

Store Connection into: ⓘ

This action is Enabled

Properties:**Ftp Server:**

Enter the FTP Server address here.

Port:

Choose which port to use for the FTP Server.

Secure FTP Protocol:

Choose the FTP protocol you wish to use to encrypt your connection.

Authentication Method:

Choose the method you wish to use to authenticate yourself on the FTP server. You have the choice between using username and password or username and a private key file which may or may not be protected by a passphrase.

User Name:

Enter the User Name for this FTP Server here.

Enter Password:

If you choose 'directly', the password entered in the Password field will be hidden. If you choose 'as variable' it will expect you to enter a variable and treat the '%' as an indicator of a variable, not part of the password.

Password:

Enter the Password here. You may enter either a text, or a variable containing the password depending on the selection of the previous property (Enter Password).

File Path to Private Key:

If the authentication method is set to "Private Key" enter here the path to the file containing the private-key to be used for authentication.

Enter Private Key Pass Phrase:

If you choose 'directly', the pass phrase entered in the Private Key Pass Phrase field will be hidden. If you choose 'as variable' it will expect you to enter a variable and treat the '%' as an indicator of a variable, not part of the password.

Private Key Pass Phrase:

Enter the private key pass phrase here. You may enter either a text, or a variable containing the password depending on the selection of the previous property (Enter Private Key Pass Phrase).

Store Connection into:

Enter a name to be the variable that will store the specific FTP Connection for use with later FTP Actions. This allows you to specify which, of possibly several, FTP Connection to access.

3.11.14.3 Close FTP Connection Action***Description:***

This action closes an open connection to an FTP server

Properties:

FTP Connection to close:

Enter the variable containing the previously opened FTP Connection you wish to close. You must have previously specified this variable in an Open FTP Connection action.

3.11.14.4 List FTP Directory Action

Description:

This action returns the subdirectories and files contained in the current directory of an FTP connection

Properties:

FTP Connection:

Enter the variable containing the previously opened FTP Connection you wish to look in. You must have previously specified this variable in an Open FTP Connection action.

Store Subdirectories into:

Enter a name to be the variable that will store the list of FTP folders in the current FTP location.

Store Files into:

Enter a name to be the variable that will store the list of FTP files in the current FTP location.

3.11.14.5 Change Working Directory Action

Description:

This action sets the current working directory for an FTP connection

Properties:

FTP Connection:

Enter the variable containing the previously opened FTP Connection you wish to alter. You must have previously specified this variable in an Open FTP Connection action.

Set Working Directory to:

Enter the path, or a variable containing the path, that you want to set as the new directory you are working from. When you first connect to an FTP Server, you are always in the root directory. This allows you to work with files and folders in a different directory without specifying a path each time.

3.11.14.6 Download File(s) from FTP Action

Description:

This action downloads one or more files from an FTP server

Properties of 'Download File(s) from FTP' action

Download File(s) from FTP
This action downloads one or more files from an FTP server

General Exception Handling

Action Input

FTP Connection: %FtpConnection%

File(s) to Download: current.log

Download into Folder: C:\Downloads

Transfer Type: Auto

If File Exists: Overwrite

Action Output
(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties:

FTP Connection:

Enter the variable containing the previously opened FTP Connection you wish to download from. You must have previously specified this variable in an Open FTP Connection action.

File(s) to Download:

Enter the name of the file(s), filepath(s), or a variable containing the file(s), that you wish to download.

Download into Folder:

Enter the full path of a folder, or the name of a variable containing a folder, to be the destination of the file(s) you are downloading.

Transfer Type:

Enter ASCII or binary if you wish to specify the method for downloading a single file. If you're not sure what type the file will be, or if you are downloading more than one, choose Auto so

ProcessRobot will follow the transfer rules specified in Tools -> Options -> FTP from the ProcessRobot Console.

If File Exists:

Choose what you want to do if the file already exists. Overwrite writes over the original file so you can't access it any more, and Download with Unique Name adds an underscore and a sequential number to the end.

3.11.14.7 Download Folder(s) from FTP Action

Description:

This action downloads one or more folders from an FTP server

Properties of 'Download Folder(s) from FTP' action

Download Folder(s) from FTP
This action downloads one or more folders from an FTP server

General | Exception Handling

Action Input

FTP Connection: %FtpConnection%

Folder(s) to Download: statdata

Download into Local Folder: C:\Downloads

Action Output

(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties:

FTP Connection:

Enter the variable containing the previously opened FTP Connection you wish to download from. You must have previously specified this variable in an Open FTP Connection action.

File to Download:

Enter the name of the folder(s) path(s), or a variable containing the FTP folder(s), that you wish to download.

Download into Folder:

Enter the full path of a folder, or the name of a variable containing a folder, to be the destination of the FTP Folder(s) you are downloading.

3.11.14.8 Upload File(s) to FTP Action

Description:

This action uploads one or more files to an FTP server

Properties of 'Upload File(s) to FTP' action

Upload File(s) to FTP
This action uploads one or more files to an FTP server

General | Exception Handling

Action Input

FTP Connection: %FtpConnection%

File(s) to Upload:

Transfer Type: Auto

If File Exists: Overwrite

Action Output

(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties:

FTP Connection:

Enter the variable containing the previously opened FTP Connection you wish to upload to. You must have previously specified this variable in an Open FTP Connection action.

File to Upload:

Enter the full path of the file(s), or a variable containing file(s), to upload.

Transfer Type:

Enter ASCII or binary if you wish to specify the method for downloading a single file. If you're not sure what type the file will be, or if you are downloading more than one, choose Auto so ProcessRobot will follow the transfer rules specified in Tools -> Options -> FTP from the ProcessRobot Console.

If File Exists:

Choose what you want to do if the file already exists. Overwrite writes over the original file so you can't access it any more, and Download with Unique Name adds an underscore and a sequential number to the end.

3.11.14.9 Upload Folder(s) to FTP Action***Description:***

This action uploads one or more folders to an FTP server

Properties of 'Upload Folder(s) to FTP' action

Upload Folder(s) to FTP
This action uploads one or more folders to an FTP server

General | Exception Handling

Action Input

FTP Connection: %FtpConnection%

Folder(s) to Upload:

Click here to expand the list of variables eligible for selection

Action Output

(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties:**FTP Connection:**

Enter the variable containing the previously opened FTP Connection you wish to upload to. You must have previously specified this variable in an Open FTP Connection action.

Folder(s) to Upload:

Enter the full path of the folder(s), or a variable containing folder(s), to upload.

3.11.14.1 Delete FTP File Action**Description:**

This action deletes one or more files from an FTP server

Properties of 'Delete FTP File' action

Delete FTP File
This action deletes one or more files from an FTP server

General Exception Handling

Action Input

FTP Connection: %FtpConnection%

File to Delete:

Action Output
(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties:**FTP Connection:**

Enter the variable containing the previously opened FTP Connection you wish to work in. You must have previously specified this variable in an Open FTP Connection action.

File to Delete:

Enter the name of the file, filepath, or a variable containing a file, that you wish to delete.

3.11.14.1 Rename FTP File Action**Description:**

This action renames a file that resides on an FTP server

Properties:

Ftp Connection:

Enter the variable containing the previously opened FTP Connection you wish to work in. You must have previously specified this variable in an Open FTP Connection action.

File to Rename:

Enter the name of the file, filepath, or a variable containing a file, that you wish to rename.

New File Name:

Enter a new name for the file, or a variable containing the new name of the file. If you enter a name with a path, the file will also be moved to that location. If the location doesn't exist, it will throw an exception.

3.11.14.1:Create FTP Directory Action

Description:

This action creates a directory on an FTP server

Properties of 'Create FTP Directory' action

Create FTP Directory
This action creates a directory on an FTP server

General | Exception Handling

Action Input

FTP Connection: %FtpConnection%

New Directory: ImagesDir

Action Output

(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties:

FTP Connection:

Enter the variable containing the previously opened FTP Connection you wish to work in. You must have previously specified this variable in an Open FTP Connection action.

New Directory:

Enter a name, or a variable containing a name, to be the new directory. If this directory already exists, nothing will happen. If you specify a path, all of the new folders leading to the new directory will be created.

3.11.14.1:Delete FTP Directory Action

Description:

This action deletes a directory from an FTP server

Properties:

FTP Connection:

Enter the variable containing the previously opened FTP Connection you wish to work in. You must have previously specified this variable in an Open FTP Connection action.

Directory to Delete:

Enter the name, or a variable containing the name, of a folder to be deleted. Everything in the folder will be deleted as well.

3.11.14.1 Synchronize FTP Directory Action

Description:

Synchronizes the files and subdirectories of a given local folder with a given remote FTP directory. This action can synchronize a local folder to the contents of a remote FTP directory (download updates), or it can synchronize a remote FTP directory to the contents of a local folder (upload updates).

Properties of 'Synchronize FTP Directory' action

Synchronize FTP Directory

Synchronizes the files and subdirectories of a given local folder with a given remote FTP directory. This action can synchronize a local folder to the contents of a remote FTP directory (download updates), or it can synchronize a remote FTP directory to the contents of a local folder (upload updates)

General | Advanced | Exception Handling

Action Input

FTP Connection: %FtpConnection%

Synchronization Direction: Remote -> Local (Download)

Files to Synchronize: Only files matching the file filter

File Filter: *.html

Local Folder: %MySiteFolder%
This will be the TARGET directory

Remote FTP Directory: /
This will be the SOURCE directory

Action Output

Store Added Files into: %FilesAdded%

Store Modified Files into: %FilesModified%

Store Deleted Files into: %FilesDeleted%

This action is Enabled

[More Info](#) [OK](#) [Cancel](#)

Properties of 'Synchronize FTP Directory' action

Synchronize FTP Directory
Synchronizes the files and subdirectories of a given local folder with a given remote FTP directory. This action can synchronize a local folder to the contents of a remote FTP directory (download updates), or it can synchronize a remote FTP directory to the contents of a local folder (upload updates)

General | Advanced | Exception Handling

Delete if source file is absent 

Include Subdirectories 

Time on FTP Server is: 10 hours, 0 minutes ahead 

This action is Enabled

More Info OK Cancel

Properties:**FTP Connection:**

Enter the variable containing the previously opened FTP Connection you wish to work in. You must have previously specified this variable in an Open FTP Connection action.

Synchronization Direction:

Choose the direction of synchronization, i.e., whether the local folder will be synchronized to the remote directory (DOWNLOAD) or the remote directory will be synchronized to the local folder (UPLOAD).

Files to Synchronize:

Choose whether you want to synchronize all files, or you want to use a file filter to include or exclude a specific set of files.

File Filter:

Enter the File-name pattern that controls which files will be included to, or excluded from the synchronization (depending on the setting of the 'Files to Synchronize' property). This allows wild cards, for example "*.txt" or "document?.doc" (without the quotes). If you want to allow for multiple file filters, separate your choices with comma, for example, "*.txt,*.exe".

Local Folder:

Enter the name (or a variable) of the local folder to be synchronized. Depending on the synchronization 'Direction' property, this will be either the source or the target directory.

Remote FTP Directory:

Enter the name (or a variable) of the remote FTP directory to be synchronized. Depending on the synchronization 'Direction' property this will be either the source or the target directory.

Delete if source is absent:

Choose whether you want a file that exists only in the target directory and not in the source directory to be deleted from the target directory during synchronization.

Include Subdirectories:

Choose whether you want the subdirectories of the specified directories to be included in the synchronization process.

Time on FTP Server is:

Enter here the time difference between the FTP server and the local computer. Very often FTP servers are located in different time zones to their clients. The time difference should be used to ensure correct date matching, so that the action will be able to detect which files are out of date and should be synchronized.

Store Added Files into:

Enter the name to be the variable that will hold the list of files that initially existed only in the source directory and after the synchronization were added to the target directory.

If the target directory is the local folder (download direction selected), this variable will hold a list of Files, otherwise, if the target directory is the remote FTP directory (upload direction selected), this variable will hold a list of FTP Files.

Store Modified Files into:

Enter the name to be the variable that will hold the list of files that initially existed in both the source and target directory and were transferred from source to target during synchronization.

If the target directory is the local folder (download direction selected), this variable will hold a list of Files, otherwise, if the target directory is the remote FTP directory (upload direction selected), this variable will hold a list of FTP Files.

Store Deleted Files into:

Enter the name to be the variable that will hold the list of files that initially existed only in the target directory and were deleted during synchronization. Note that files that appear only in the target directory are deleted only if the 'Delete if source file is absent' property is checked.

If the target directory is the local folder (download direction selected), this variable will hold a list of Files, otherwise, if the target directory is the remote FTP directory (upload direction selected), this variable will hold a list of FTP Files.

3.11.14.1 Invoke FTP Command Action

Description:

This action invokes the given literal FTP command on the server.

Properties:

FTP Connection:

Enter the variable containing the previously opened FTP Connection you wish to work in. You must have previously specified this variable in an Open FTP Connection action.

FTP Command:

Enter the command, or a variable containing the command, that you wish to run along with any arguments. For example, you could run FEAT, or CHMOD here.

Valid Reply Code(s):

Enter the code(s) that could be returned by the Command, separated by semi-colon. If the FTP Command returns a code that was not entered here, this action will throw an exception.

Store Reply Code into:

Enter a name to be the Variable that will contain the actual code that was returned.

Store Reply Text into:

Enter a name to be the Variable that will contain any text returned by the FTP Server.

3.11.15 Email

3.11.15.1 Retrieve Emails Action

Description:

This action retrieves email messages from an IMAP server.

Messages can be filtered based on various criteria, such as the sender's address, the IMAP folder (location on server) or even the existence of specific keywords in the body, subject or other fields of the email.

Properties of 'Retrieve Emails' action

 **Retrieve Emails**
This action retrieves email messages from an IMAP server

Email Filters | IMAP Server | Exception Handling

Action Input

Mail Folder: ⓘ ⚙️

Retrieve: ⓘ

...that match the following criteria:

"From" Field Contains: ⓘ ⚙️

"To" Field Contains: ⓘ ⚙️

"Subject" Contains: ⓘ ⚙️

"Body" Contains: ⓘ ⚙️

Attachments: ⓘ

Save Attachments Into: ⓘ ⚙️ 📁

Action Output

Save Emails Into: ⓘ

This action is Enabled

Note that this action will mark the retrieved emails as read on the server. If you wish to keep the emails marked as unread you can change their state back to unread through the Process Emails Action. Only the messages that match **all** of the filters set in the action will be included in the retrieval and marking.

Properties of 'Retrieve Emails' action

 **Retrieve Emails**
This action retrieves email messages from an IMAP server

Email Filters | **IMAP Server** | Exception Handling

IMAP Server: ⓘ ⚙️

Server Port: ⓘ

Enable SSL ⓘ

User Name: ⓘ ⚙️

Enter Password: ⓘ

Password: ⓘ

This action is Enabled

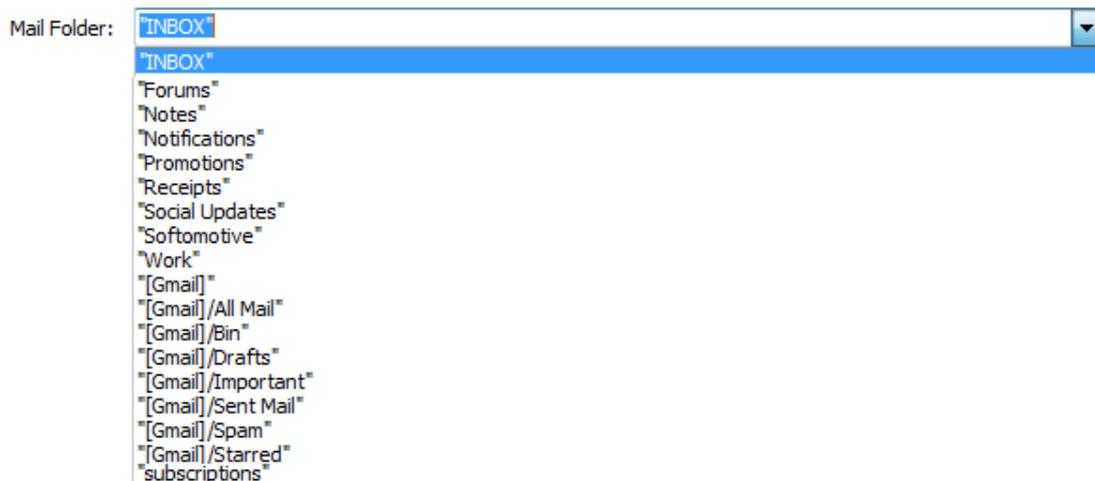
Properties:

Mail Folder:

In the Mail Folder property you can enter the name of the mail folder (also known as 'Mailbox') that you want to retrieve messages from.

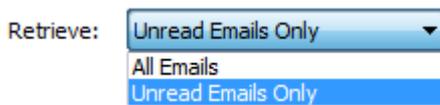
You can select to retrieve emails from your INBOX or from any other folder of your email account. To avoid confusion, you should keep in mind that in this action, by using the term "Mail Folder", we refer to the folders of your email server and not to the folders where you store your files locally, in your PC.

If you have already set some values in the IMAP Server Properties Tab for this or a previous Retrieve Emails action in the Process, the drop-down menu will load and present you all the available folders from your email server. However, even if you have not already set IMAP Server properties, you can specify your preferred Mail Folder by just typing its name in the Mail Folder property (provided that when you execute this Process, the specified folder will exist in the email server).



Retrieve:

In the Retrieve property you can specify whether you want to retrieve all messages from the Mail Folder, or only the unread ones. This could be important especially when the Retrieve Emails Action is to be performed periodically and you want to get feedback on only the updated conversations and unread emails from your mailbox.



From Contains:

Enter the full e-mail address of the Sender whose messages you want to retrieve. Leave this field blank to retrieve all messages regardless of the sender.

To Contains:

Enter the full e-mail address(es) of the Receiver(s) (separated by space if more than one) for the messages you want to retrieve. Leave this field blank to retrieve all messages regardless of the recipient(s).

Subject Contains:

Enter the keywords to be found within the email Subject. Leave this field blank to retrieve all emails regardless of their subject.

Body Contains:

Enter the keywords to be found within the email Body. Leave this field blank to retrieve all emails regardless of their content.

Of course, as with any other property in ProcessRobot, you have the option to define your criteria on all "Contains" properties using Process Variables. You can, for example, set the "From Contains" property to get its value from a variable where you have stored email address

information of the customers that bought a specific product and you want to retrieve any feedback provided by them:

"From" Field Contains:  

With this setting the Retrieve Emails Action will search your specified Mailbox (or your email server inbox if not otherwise specified) for any emails sent by any of the email addresses stored as the values of this Variable (unread or all according to your preference) and retrieve them.

The use of Variables as the values of the Sender's and/or Receiver's properties also gives you the option to include the Retrieve Emails Action in a Loop Section of a Process and iteratively check and retrieve emails for each one of this group of email addresses.

Below we present you a part of a script that uses a variable (here named %Contacts%) as that input of "From Field Contains" property of Retrieve Emails Action, in order to retrieve emails sent by each one of a group of contacts and display a message informing you about the number of the unread messages you have from each one of them.



```
graph TD
    A[Display Select File Dialog] --> B[Read Text from File]
    B --> C[For Each]
    C --> D[Retrieve Emails]
    D --> E[Display Message]
    E --> C
    C --> F[End Loop]
```

Display Select File Dialog
Display the Select File Dialog and store the selected file into %SelectedFile%

Read Text from File
Read content of file %SelectedFile% and store it into %Contacts%

For Each
Loop for each item contained in variable %Contacts% and store the current item into %CurrentContact%

Retrieve Emails
Selectively retrieve e-mails from account: username@example.com

Display Message
Display Message box with message You have %CurrentContactRetrievedEmails.Count% new emails from %CurrentContact%

End Loop

Properties of 'Retrieve Emails' action

 **Retrieve Emails**
This action retrieves email messages from an IMAP server

Email Filters | **IMAP Server** | Exception Handling

Action Input

Mail Folder: "INBOX"  

Retrieve: Unread Emails Only 

...that match the following criteria:

"From" Field Contains: %CurrentContact%  

"To" Field Contains:  

"Subject" Contains: New Order  

"Body" Contains:  

Attachments: Do Not Save attachments 

Action Output

Save Emails Into: %RetrievedEmails% 

This action is Enabled

[More Info](#)

Attachments:

Specify whether you want to save the attachments of the emails retrieved or not.

Save attachments into:

Enter here the folder in which you want the attachments to be saved. ProcessRobot will save all the attachments of the retrieved emails to the same (specified) local folder. In the case that more than one emails have the same name, ProcessRobot will rename them by appending a 4-digit suffix. For example, if there is already a file named *OrderForm.doc* in the destination folder, the next attachment of a retrieved email also named *OrderForm.doc* that is to be saved, will eventually be saved as *OrderForm_0001.doc*.

Furthermore, as you may see below in the Mail Message Variable Properties, each retrieved email variable is accompanied by a list of its attachments that are saved locally (represented by the attachments property) to which you can refer in order to track the email to which each file was attached to.

Save Emails into:

Enter a name to be the variable that will store the retrieved e-mails for later processing. The outcome of the Retrieved Email Action (%RetrievedEmails%) will be a **List of Mail Messages**

variable. As every [list variable](#)^[438], its only property is "Count" that has as value the number of the retrieved emails according to our selections in the Retrieve Emails Action properties fields:

```
☐ %RetrievedEmails% (List of Mail Messages)
    .Count (Numerical Value)
```

Each item of this list will be a [Mail Message](#)^[441] object. Since `%RetrievedEmails%` Variable is a list type variable, each item of the list (i.e. each email) can be described by the following notation:

`%RetrievedEmails[n]%`

with *n* referring to the index of each retrieved email of our action (e.g. 0 for the first retrieved email, 1, for second).

Each **Mail Message** (included as an item in the **List of Mail Messages**) is itself a variable with the [respective properties](#)^[439] (populated by the Retrieve Emails action).

You can retrieve any of each email's properties and use it as input in an action's properties by following the general notation:

`%VariableName.PropertyName%`

For example if you want to retrieve the Subject of the third retrieved email from you action, you should use the notation:

`%RetrievedEmails[2].Subject%`

Below you can see an example of how the Retrieve Emails Action can be used inside a loop. This section of the Process's script will search and retrieve all unread emails from your Inbox that their subject contains the subtext "Order Form" (as specified in the Retrieve Emails Action Properties window). If any of the retrieved emails has attachments, those will be saved locally, inside the specified folder. For every matching email, this Process will append the corresponding emails of the sender(s) to a specific text file.

In other words, the final outcome of this Process will be a text file containing all the sender addresses of messages with a subject containing the text "Order Form"

```

✉ Retrieve Emails
  Selectively retrieve unread e-mails from account: username@example.com

🔄 For Each
  Loop for each item contained in variable %RetrievedEmails% and store the current item into %CurrentEmail%

📄 Write Text to File
  Append %CurrentEmail.From% to C:\Users\t\Documents\WinAutomation\clients_requiring_response.txt

🛑 End Loop
  
```

Properties of 'Retrieve Emails' action

 **Retrieve Emails**
This action retrieves email messages from an IMAP server

Email Filters | **IMAP Server** | Exception Handling

Action Input

Mail Folder: "INBOX"  

Retrieve: Unread Emails Only 

...that match the following criteria:

"From" Field Contains: %CurrentContact%  

"To" Field Contains:  

"Subject" Contains: New Order  

"Body" Contains:  

Attachments: Save attachments 

Save Attachments Into: C:\Users\Documents\WinAutomation\New Orders   

Action Output

Save Emails Into: %RetrievedEmails% 

This action is Enabled

[More Info](#) [OK](#) [Cancel](#)

Note that in order to record only the Sender's address for each of the retrieved emails we used (as presented in the Properties of "Write Text to File" action window below) the notation :

%CurrentEmail.From%

requesting from this action to retrieve the value of the "From" property of the retrieved email used as iterative variable (%CurrentItem%) in each loop.

Properties of 'Write Text to File' action

Write Text to File
This action writes or appends text to a file

General Exception Handling

Action Input

File Path: C:\Users\User\Documents\WinAutomation\clients_requiring_response

Text to Write: 1 %CurrentEmail.From%

Append New Line

If File Exists: Append content

Encoding: Unicode

Action Output

(This action does not provide any output)

This action is Enabled

More Info OK Cancel

IMAP Server:

Enter the IMAP Server address here (e.g., imap.gmail.com).

Port:

Specify the port to be used for the IMAP Server. Usually, this is Port 993.

Enable SSL:

Specify whether you need to use a secure connection to communicate with the IMAP Server.

User Name:

Enter the User Name of the e-mail account you want to access.

Enter Password:

If you choose 'directly', the password entered in the Password field will be hidden. If you choose 'as variable' you must enter a variable containing the password and the '%' character will be treated as an indicator of a variable, not part of the password.

Password:

Enter the Password here. You may enter either a text, or a variable containing the password depending on the selection of the previous property (Enter Password).

3.11.15.2 Process Emails Action

Description:

Moves, deletes or marks as unread an email (or a list of emails) retrieved by a "[Retrieve Emails](#)^[797]" action.

Process Emails Action is one of the three actions under the Email Actions category (the other two being the [Send Email](#)^[808] and [Retrieve Emails](#)^[797]). This particular group of actions provides ProcessRobot with the ability of automating any task typically performed by email clients. With the Process Email Action you can manipulate the retrieved emails from a previous Retrieve Emails action. You may delete the emails from your Server, mark them as (un)read or move them to different folders inside your Mailbox.

Note that when retrieved, messages are automatically marked as "read" on the server.

Properties of 'Process Emails' action ✕

 **Process Emails**
 Moves, deletes or marks as unread an email (or a list of emails) retrieved by a "Retrieve Emails" action.

General | IMAP Settings | Exception Handling

Action Input

Email(s) to process:  

Operation: 

Mail Folder:  

Action Output

(This action does not provide any output)

This action is Enabled

Properties of 'Process Emails' action ✕

 **Process Emails**
 Moves, deletes or marks as unread an email (or a list of emails) retrieved by a "Retrieve Emails" action.

General | IMAP Settings | Exception Handling

IMAP Server:  

Port: 

Enable SSL 

Username:  

Enter Password: 

Password: 

This action is Enabled

Properties:**Email(s) to process:**

Enter a variable containing the email or list of emails to be processed. This should be a variable populated by a "[Retrieve Emails](#)⁷⁹⁷" action.

Operation:

Specify which operation you want to perform on the specified email messages.

Mail Folder:

Enter the name of the mail folder you want to move the emails to.

IMAP Server:

Enter the IMAP Server address here (e.g., imap.gmail.com).

Port:

Specify the port to be used for the IMAP Server. Usually, this is Port 993.

Enable SSL:

Specify whether you need to use a secure connection to communicate with the IMAP Server.

User Name:

Enter the User Name of the e-mail account you want to access.

Enter Password:

If you choose 'directly', the password entered in the Password field will be hidden. If you choose 'as variable' you must enter a variable containing the password and the '%' character will be treated as an indicator of a variable, not part of the password.

Password:

Enter the Password here. You may enter either a text, or a variable containing the password depending on the selection of the previous property (Enter Password).

3.11.15.3 Send Email Action**Description:**

This action creates and sends a new email message

Properties of 'Send Email' action ✕

 **Send Email**
This action creates and sends a new email message

General SMTP Server Exception Handling

Action Input

From:  

Sender Display Name:  

To:  

CC:  

BCC:  

Subject:  

Body:  

Body is HTML 

Attachment(s):   

Action Output

(This action does not provide any output)

This action is Enabled

Properties of 'Send Email' action

Send Email
This action creates and sends a new email message

General SMTP Server Exception Handling

SMTP Server: ⓘ ⚙️

Server Port: ⓘ

Enable SSL ⓘ

SMTP Server needs authentication ⓘ

User Name: ⓘ ⚙️

Enter Password: ⓘ

Password: ⓘ

This action is Enabled

The Send Email Action can be useful in a number of situations, such as:

1. Sending to a large number of recipients (and even personalize each message)
2. Sending emails in specific time intervals
3. Sending messages triggered by specific events, etc.

Properties:

From:

Enter the Sender's email address.

Sender Display Name:

Enter the Sender's display name.

To:

Enter the email(s) of the recipient(s). If you enter more than one email, the list of addresses should be separated by semi-colons.

CC:

Enter the email(s) of the CC recipient(s). If you enter more than one email, the list of addresses should be separated by semi-colons.

BCC:

Enter the email(s) of the BCC (hidden) recipient(s). If you enter more than one email, the list of addresses should be separated by semi-colons.

Subject:

Enter the subject of the email.

Body:

Enter the text of the body.

Body Is HTML:

Choose whether the body of the email is interpreted as HTML coding.

Attachment(s):

Enter or choose the full path of any attachment(s), or choose a variable that contains a file or a list of files. Multiple files should be enclosed in double quotes (") and separated by a space character.

SMTP Server:

Enter the SMTP Server address here.

Server Port:

Choose which port to use for the Server. Usually, this is Port 25.

Enable SSL:

Choose whether or not to communicate with the Server through a secure connection.

SMTP Server need authentication:

Specify whether or not the server requires authentication.

User Name:

Enter the User Name.

Password:

Enter the Password. This entry will be hidden.

3.11.16 Exchange**3.11.16.1 Connect to Exchange Server*****Description:***

This action opens a new connection to an Exchange Server:

Properties of 'Connect to Exchange Server' action

Connect to Exchange Server

This action opens a new connection to an Exchange Server

General | Advanced | Exception Handling

Action Input

Exchange Server Version: Exchange 2013 SP1 

Connection Type: Exchange Server Address 

Server Address:  

Credentials: Exchange Default 

Action Output

Store Exchange Connection into: %ExchangeConnection% 

This action is Enabled

[More Info](#) [OK](#) [Cancel](#)

Properties of 'Connect to Exchange Server' action

Connect to Exchange Server
This action opens a new connection to an Exchange Server

General | Advanced | Exception Handling

Action Input

Exchange Server Version: Exchange 2013 SP1

Connection Type: Exchange Server Address

Server Address:

Credentials: **User Defined**

Domain:

Username:

Enter Password: Directly

Password:

Action Output

Store Exchange Connection into: %ExchangeConnection%

This action is Enabled

More Info OK Cancel

Properties:

Exchange Server Version:

Select the version of the Exchange server that you are using

Connection Type:

Select the way that you want to connect to your Exchange server

Server Address:

Enter your Exchange Server Address

Credentials:

Choose the way that the user's Exchange credentials will be provided to the action

Domain:

Enter your Exchange account Domain. If you leave it empty, action will extract it from the username

Username:

Enter your Exchange account username

Enter Password:

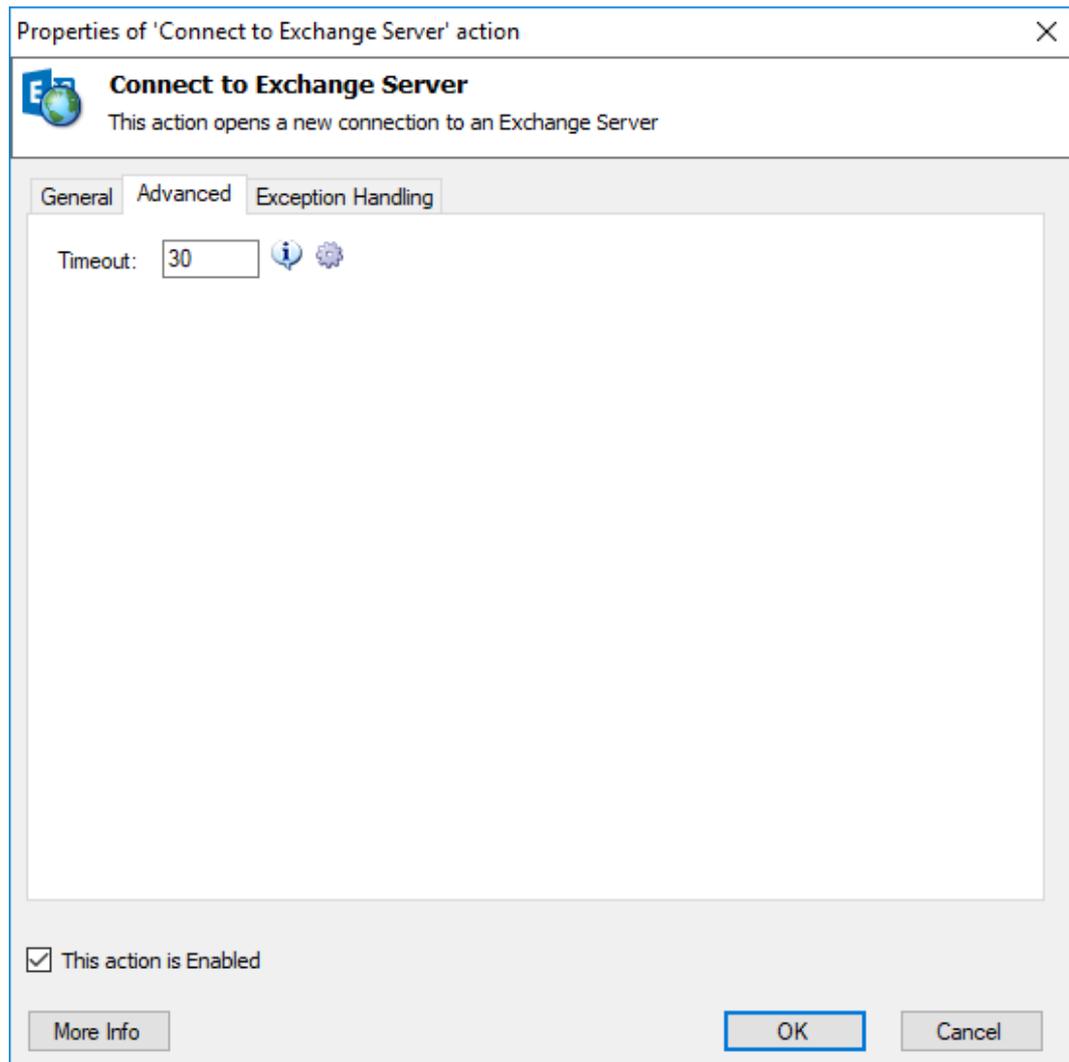
Select the way that you will provide your password.

Password:

Enter your Exchange account password directly

Store Exchange Connection into:

Enter a name to be the variable that will store the specific Exchange Connection for use with later Exchange Actions



Timeout: Set the time in seconds that you want to wait for the connection to be established before the action fails

3.11.16.2 Retrieve Exchange Email Messages

Description:

This action retrieves email messages from an Exchange server.

Properties of 'Retrieve Exchange Email Messages' action

Retrieve Exchange Email Messages

This action retrieves email messages from the specified Exchange server

Email Filters | Exception Handling

Action Input

Exchange Connection: %ExchangeConnection%

Exchange Folders: Inbox

Retrieve: Unread Emails Only

Mark As Read

...that match the following criteria:

"From" Field Contains:

"To" Field Contains:

"Subject" Contains:

"Body" Contains:

Attachments: Do Not Save attachments

Action Output

Save Email Messages Into: %RetrievedEmails%

This action is Enabled

More Info

OK Cancel

Messages can be filtered based on various criteria, such as the sender's address, the Exchange folder or even the existence of specific keywords in the body, subject or other fields of the email.

Note that this action will mark the retrieved emails as read on the server. If you wish to keep the emails marked as unread you can change their state back to unread through the "Process Exchange Email Messages" Action. Only the messages that match **all** of the filters set in the action will be included in the retrieval and marking.

Properties:**Exchange Connection:**

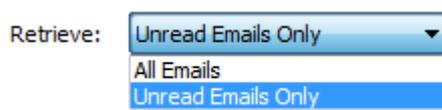
Enter the variable that holds your exchange connection. Exchange connection can be created from \"Connect To Exchange Server\" action.

Exchange Folders:

Select a predefined Exchange Folder to retrieve email messages from or add a custom one.

Retrieve:

In the Retrieve property you can specify whether you want to retrieve all messages from the Mail Folder, or only the unread ones. This could be important especially when the Retrieve Emails Action is to be performed periodically and you want to get feedback on only the updated conversations and unread emails from your mailbox.

**From Contains:**

Enter the full e-mail address of the Sender whose messages you want to retrieve. Leave this field blank to retrieve all messages regardless of the sender.

To Contains:

Enter the full e-mail address(es) of the Receptient(s) (separated by space if more than one) for the messages you want to retrieve. Leave this field blank to retrieve all messages regardless of the recipient(s).

Subject Contains:

Enter the key phrase to be found within the email Subject. Leave this field blank to retrieve all email messages regardless of the their subject.

Body Contains:

Enter the key phrase to be found within the email Body. Leave this field blank to retrieve all email messages regardless of the their content.

Of course, as with any other property in ProcessRobot, you have the option to define your criteria on all "Contains" properties using Process Variables. You can, for example, set the "From Contains" property to get its value from a variable where you have stored email address information of the customers that bought a specific product and you want to retrieve any feedback provided by them:

"From" Field Contains:  

With this setting the Retrieve Action will search your specified Mailbox (or your email server inbox if not otherwise specified) for any emails sent by any of the email addresses stored as the values of this Variable (unread or all according to your preference) and retrieve them.

The use of Variables as the values of the Sender's and/or Receiver's properties also gives you the option to include the Retrieve Email Messages Action in a Loop Section of a Process and iteratively check and retrieve emails for each one of this group of email addresses.

Attachments:

Specify whether you want to save the attachments of the emails retrieved or not.

Save attachments into:

Enter here the folder in which you want the attachments to be saved. ProcessRobot will save all the attachments of the retrieved emails to the same (specified) local folder. In the case that more than one emails have the same name, ProcessRobot will rename them by appending a 4-digit suffix. For example, if there is already a file named *OrderForm.doc* in the destination folder, the next attachment of a retrieved email also named *OrderForm.doc* that is to be saved, will eventually be saved as *OrderForm_0001.doc*.

Furthermore, as you may see below in the Mail Message Variable Properties, each retrieved email variable is accompanied by a list of its attachments that are saved locally (represented by the attachments property) to which you can refer in order to track the email to which each file was attached to.

Save Emails into:

Enter a name to be the variable that will store the retrieved email messages for later processing. The variable will contain a List of Exchange Mail Messages objects.

The outcome of the Retrieved Emails Action (%RetrievedEmails%) will be a **List of Mail Messages** variable. As every [list variable](#),^[436] its only property is "Count" that has as value the number of the retrieved emails according to our selections in the Retrieve Emails Action properties fields:

```
☐ %RetrievedEmails% (List of Mail Messages)
    .Count (Numerical Value)
```

Each item of this list will be a [Mail Message](#)^[441] object. Since %RetrievedEmails% Variable is a list type variable, each item of the list (i.e. each email) can be described by the following notation:

%RetrievedEmails[n]%

with *n* referring to the index of each retrieved email of our action (e.g. 0 for the first retrieved email, 1, for second).

Each **Mail Message** (included as an item in the **List of Mail Messages**) is itself a variable with the [respective properties](#)^[439] (populated by the Retrieve Emails action).

You can retrieve any of each email's properties and use it as input in an action's properties by following the general notation:

%VariableName.PropertyName%

For example if you want to retrieve the Subject of the third retrieved email from you action, you should use the notation:

%RetrievedEmails[2].Subject%

3.11.16.3 Send Exchange Email Message

Description:

This action creates and sends a new email message

Properties of 'Send Exchange Email Message' action

Send Exchange Email Message
This action creates and sends a new email message

General | Exception Handling

Action Input

Exchange Connection: [%ExchangeConnection%] ⓘ

From: sample@aol.com ⓘ ⚙️

Sender Display Name: John Smith ⓘ ⚙️

To: george@aol.com ⓘ ⚙️

CC: maria@gmail.com ⓘ ⚙️

BCC: john@gmail.com; helen@gmail.com; con@yahoo.com ⓘ ⚙️

Subject: Test Subject ⓘ ⚙️

Body: This is a Demo body ⓘ ⚙️

Body is HTML ⓘ

Attachment(s): C:\Users\John\Attachment.pdf ⓘ ⚙️ 📎

Action Output

(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties:

Exchange Connection:

Enter the variable that holds your exchange connection. Exchange connection can be created from \"Connect To Exchange Server\" action.

From:

Enter the Sender's email address.

Sender Display Name:

Enter the Sender's display name.

To:

Enter the email(s) of the recipient(s). If you enter more than one email, the list of addresses should be separated by semi-colons.

CC:

Enter the email(s) of the CC recipient(s). If you enter more than one email, the list of addresses should be separated by semi-colons.

BCC:

Enter the email(s) of the BCC (hidden) recipient(s). If you enter more than one email, the list of addresses should be separated by semi-colons.

Subject:

Enter the subject of the email.

Body:

Enter the text of the body.

Body Is HTML:

Choose whether the body of the email is interpreted as HTML coding.

Attachment(s):

Enter or choose the full path of any attachment(s), or choose a variable that contains a file or a list of files. Multiple files should be enclosed in double quotes (") and separated by a space character.

3.11.16.4 Process Exchange Email Messages

Description:

Moves, deletes or marks as unread an email (or a list of emails) retrieved by a "[Retrieve Exchange Email Messages](#)⁸¹⁶" action.

Properties of 'Process Exchange Email Messages' action

Process Exchange Email Messages
 Moves, deletes or marks as unread an email message (or a list of email messages) retrieved by a "Retrieve Exchange Email Messages" action.

General | Exception Handling

Action Input

Exchange Connection: %ExchangeConnection%

Email Message(s) to process: %CurrentEmail%

Operation: Delete Email Messages from Server

Action Output

(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties of 'Process Exchange Email Messages' action

Process Exchange Email Messages
 Moves, deletes or marks as unread an email message (or a list of email messages) retrieved by a "Retrieve Exchange Email Messages" action.

General | Exception Handling

Action Input

Exchange Connection: %ExchangeConnection%

Email Message(s) to process: %CurrentEmail%

Operation: Move Email Messages to Mail Folder

Exchange Folders: Inbox

Action Output

This action is Enabled

More Info OK Cancel

Properties:**Exchange Connection:**

Enter the variable that holds your exchange connection. Exchange connection can be created from \"Connect To Exchange Server\" action.

Email Message(s) to process:

Enter a variable containing the email or list of emails to be processed. This should be a variable populated by a \"Retrieve Exchange Email Messages\" action.

Operation:

Specify which operation you want to perform on the specified email messages between:

1. Delete Email Messages from Server
2. Mark Email Messages as Unread
3. Move Email Messages to Mail Folder

If the selected Operation is \"Move\" then you will also get the...

Exchange Folders:

Enter the name or the path (e.g.folder1\\folder2) of the Mail-folder that you want to move email messages to.

3.11.17 Cmd Session

3.11.17.1 Open Cmd Session

Description:

This action opens a new command line session. This session will remain open until you close it with a \"Close Cmd Session\". In the meantime you can write commands or read output from the Cmd session. This will be extremely usefully in cases where user input is required on the Cmd or in case that you wish to run something like a batch file.

Properties of 'Open Cmd Session' action

Open Cmd Session
This action opens a new Cmd session

General Exception Handling

Action Input

Working Folder: C:\Users\John\Documents

Action Output

Store Cmd Session into: %CmdSession%

This action is Enabled

More Info OK Cancel

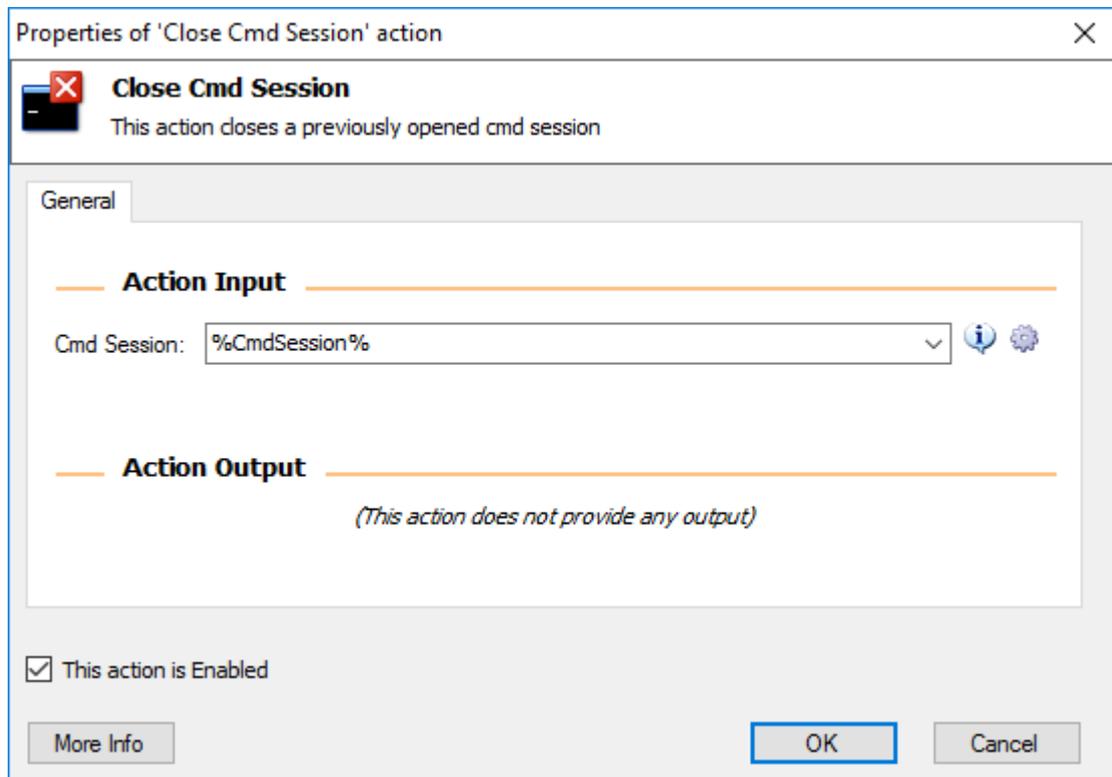
Properties:

Working Folder: Enter the full path of the folder to work out of, if applicable

Store Cmd Session into: Enter a variable name that will hold the Cmd session, for use in later Cmd Session actions.

3.11.17.2 Close Cmd Session

Description: This action closes a previously opened Cmd session.



Properties:

Cmd Session: Enter the variable containing the previously opened Cmd Session you wish to close. You must have previously specified this variable in an "Open Cmd Session" action.

3.11.17.3 Write to Cmd Session

Description: This action executes a command in an open Cmd Action.

Properties of 'Write to Cmd Session' action

Write to Cmd Session
This action executes a command on an open Cmd Session

General | Exception Handling

Action Input

Cmd Session: %CmdSession%

Command: echo username

Action Output
(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties:

Cmd Session: Enter the variable that will contain a previously opened Cmd Session.

Command: Enter the command that you want to execute.

3.11.17.4 Read from Cmd Session

Description: This actions reads the output of a Cmd Session.

Properties of 'Read from Cmd Session' action

Read from Cmd Session
This action reads the output of a Cmd Session

General Exception Handling

Action Input

Cmd Session: %CmdSession%

Action Output

Operation: Read from Output & Error

Store Output: %CmdOutput%

Store Error: %CmdError%

This action is Enabled

More Info OK Cancel

Properties:

Cmd Session: Enter a name to be the variable that contains an open Cmd Session.

Operation: Specify which Read operation you wish to perform on the specified Cmd session. There are three operations that you can choose from. Depending on the operation chosen based on the output that you wish to have, the output variables may be %CmdOutput%, %CmdError%.

Store output: Enter a name to be the variable that will store the Cmd session's standard output

Store Error: Enter a name to be the variable that will store the Cmd session's standard error

3.11.17.5 Wait for Text on Cmd Session

Description: This action waits for a specific text on a previously opened Cmd Session.

Properties of 'Wait for Text on Cmd Session' action

Wait for Text on Cmd Session
This action waits for a specific text on a previously opened Cmd Session

General | Advanced | Exception Handling

Action Input

Cmd Session: %CmdSession%

Text to Wait: Username

Is Regular Expression

Ignore Case

Action Output

(This action does not provide any output)

This action is Enabled

More Info | OK | Cancel

Properties:

Cmd Session: Enter the variable that will contain a previously opened Cmd Session.

Text to Wait: Enter the text or the Regular Expression that you want to wait to appear on standard output or on standard error

Is Regular Expression: Check this box if you want to wait for for a regular expression match instead of a plain text.

Ignore Case: Check this box if you wish to ignore the case of the text to wait.

3.11.18 Cognitive

3.11.18.1 Google

3.11.18.1.1 Vision

Face Detection

Description:

This action invokes the Google Cloud Vision service named Face Detection.

Properties of 'Face Detection' action

Face Detection
This action invokes the Google Cloud Vision service named Face Detection.

General | Advanced | Exception Handling

Action Input

API Key: ⓘ ⚙️

Provide Image: From File ⓘ

Image File: ⓘ ⚙️ 📄
[API Reference](#)

Action Output

Store Response into: ⓘ

Store Status Code: ⓘ

This action is Enabled

More Info | OK | Cancel

Properties:

API Key:

Specify the Google API Key to be used for this API call.

Provide Image:

Select the way that you want to provide the image file. You can either specify the full path of the image or a Google Cloud Storage URI of it.

Image File:

Enter or choose an image file path, or a previously defined variable, to specify the full path (folder plus filename) of the image file that you want to transmit.

GCS Image URI:

Enter the URI of the image residing on Google Cloud Storage.

Store Response into:

Enter a name to be the variable that you want to store the API response results in.

Store Status Code:

Enter a name to be the variable that you want to store the status code returned in.

Label Detection***Description:***

This action invokes the Google Cloud Vision service named Label Detection.

Properties:

API Key:

Specify the Google API Key to be used for this API call.

Provide Image:

Select the way that you want to provide the image file. You can either specify the full path of the image or a Google Cloud Storage URI of it.

Image File:

Enter or choose an image file path, or a previously defined variable, to specify the full path (folder plus filename) of the image file that you want to transmit.

GCS Image URI:

Enter the URI of the image residing on Google Cloud Storage.

Store Response into:

Enter a name to be the variable that you want to store the API response results in.

Store Status Code:

Enter a name to be the variable that you want to store the status code returned in.

Landmark Detection_2_2

Description:

This action invokes the Google Cloud Vision service named Landmark Detection.

The screenshot shows a dialog box titled "Properties of 'Landmark Detection' action". It features a Google Cloud Vision logo and a description: "This action invokes the Google Cloud Vision service named Landmark Detection." The dialog has three tabs: "General", "Advanced", and "Exception Handling". The "General" tab is active and contains the following fields:

- Action Input:**
 - API Key: [Text input field]
 - Provide Image: [Dropdown menu with "From File" selected]
 - Image File: [Text input field]
 - [API Reference](#)
- Action Output:**
 - Store Response into: [%JSONResponse%]
 - Store Status Code: [%StatusCode%]

At the bottom, there is a checkbox labeled "This action is Enabled" which is checked. There are also "More Info", "OK", and "Cancel" buttons.

Properties:

API Key:

Specify the Google API Key to be used for this API call.

Provide Image:

Select the way that you want to provide the image file. You can either specify the full path of the image or a Google Cloud Storage URI of it.

Image File:

Enter or choose an image file path, or a previously defined variable, to specify the full path (folder plus filename) of the image file that you want to transmit.

GCS Image URI:

Enter the URI of the image residing on Google Cloud Storage.

Store Response into:

Enter a name to be the variable that you want to store the API response results in.

Store Status Code:

Enter a name to be the variable that you want to store the status code returned in.

Text Detection***Description:***

This action invokes the Google Cloud Vision service named Text Detection.

Properties:

API Key:

Specify the Google API Key to be used for this API call.

Provide Image:

Select the way that you want to provide the image file. You can either specify the full path of the image or a Google Cloud Storage URI of it.

Image File:

Enter or choose an image file path, or a previously defined variable, to specify the full path (folder plus filename) of the image file that you want to transmit.

GCS Image URI:

Enter the URI of the image residing on Google Cloud Storage.

Store Response into:

Enter a name to be the variable that you want to store the API response results in.

Store Status Code:

Enter a name to be the variable that you want to store the status code returned in.

Logo Detection

Description:

This action invokes the Google Cloud Vision service named Logo Detection.

The screenshot shows a dialog box titled "Properties of 'Logo Detection' action". It features a Google Cloud Vision logo and a description: "This action invokes the Google Cloud Vision service named Logo Detection." The dialog has three tabs: "General", "Advanced", and "Exception Handling". The "General" tab is active and contains the following fields:

- Action Input:**
 - API Key: [Text input field]
 - Provide Image: [Dropdown menu with "From File" selected]
 - Image File: [Text input field]
 - [API Reference](#)
- Action Output:**
 - Store Response into: [%JSONResponse%]
 - Store Status Code: [%StatusCode%]

At the bottom, there is a checkbox labeled "This action is Enabled" which is checked. There are also "More Info", "OK", and "Cancel" buttons.

Properties:

API Key:

Specify the Google API Key to be used for this API call.

Provide Image:

Select the way that you want to provide the image file. You can either specify the full path of the image or a Google Cloud Storage URI of it.

Image File:

Enter or choose an image file path, or a previously defined variable, to specify the full path (folder plus filename) of the image file that you want to transmit.

GCS Image URI:

Enter the URI of the image residing on Google Cloud Storage.

Store Response into:

Enter a name to be the variable that you want to store the API response results in.

Store Status Code:

Enter a name to be the variable that you want to store the status code returned in.

Image Properties Detection***Description:***

This action invokes the Google Cloud Vision service named Image Properties Detection.

Properties of 'Image Properties Detection' action

Image Properties Detection
This action invokes the Google Cloud Vision service named Image Properties Detection.

General | Advanced | Exception Handling

Action Input

API Key: ⓘ ⚙️

Provide Image: From File ⓘ

Image File: ⓘ ⚙️ 📄
[API Reference](#)

Action Output

Store Response into: ⓘ

Store Status Code: ⓘ

This action is Enabled

More Info | OK | Cancel

Properties:

API Key:

Specify the Google API Key to be used for this API call.

Provide Image:

Select the way that you want to provide the image file. You can either specify the full path of the image or a Google Cloud Storage URI of it.

Image File:

Enter or choose an image file path, or a previously defined variable, to specify the full path (folder plus filename) of the image file that you want to transmit.

GCS Image URI:

Enter the URI of the image residing on Google Cloud Storage.

Store Response into:

Enter a name to be the variable that you want to store the API response results in.

Store Status Code:

Enter a name to be the variable that you want to store the status code returned in.

Safe Search Detection

Description:

This action invokes the Google Cloud Vision service named Safe Search Detection.

The screenshot shows a dialog box titled "Properties of 'Safe Search Detection' action". It features a Google Cloud logo and the text "Safe Search Detection" and "This action invokes the Google Cloud Vision service named Safe Search Detection." Below this, there are three tabs: "General", "Advanced", and "Exception Handling". The "General" tab is active and contains the following fields:

- Action Input:**
 - API Key: [Text input field]
 - Provide Image: [Dropdown menu with "From File" selected]
 - Image File: [Text input field]
 - [API Reference](#)
- Action Output:**
 - Store Response into: [%JSONResponse%]
 - Store Status Code: [%StatusCode%]

At the bottom, there is a checkbox labeled "This action is Enabled" which is checked. There are also "More Info", "OK", and "Cancel" buttons.

Properties:

API Key:

Specify the Google API Key to be used for this API call.

Provide Image:

Select the way that you want to provide the image file. You can either specify the full path of the image or a Google Cloud Storage URI of it.

Image File:

Enter or choose an image file path, or a previously defined variable, to specify the full path (folder plus filename) of the image file that you want to transmit.

GCS Image URI:

Enter the URI of the image residing on Google Cloud Storage.

Store Response into:

Enter a name to be the variable that you want to store the API response results in.

Store Status Code:

Enter a name to be the variable that you want to store the status code returned in.

3.11.18.1. Natural Language**Analyze Sentiment*****Description:***

This action invokes the Google Cloud Natural Language service named Analyze Sentiment.

✕
Properties of 'Analyze Sentiment' action

Analyze Sentiment
This action invokes the Google Cloud Natural Language service named Analyze Sentiment.

General
Advanced
Exception Handling

Action Input

API Key: i ⚙

Document Type: Plain Text i

Provide Document: From File i

File Path: i ⚙ 📄

Language: i ⚙

[API Reference](#)

Action Output

Store Response into: i

Store Status Code: i

This action is Enabled

More Info
OK
Cancel

Properties:

API Key:

Specify the Google API Key to be used for this API call.

Document Type:

Select the type of document you want to transmit.

Provide Document:

Select the way that you want to provide the document. You can either specify the full path of the document or a Google Cloud Storage URI of it.

File Path:

Enter or choose a file path, or a previously defined variable, to specify the full path (folder plus filename) of the document that you want to transmit.

GCS Content URI:

Enter the URI of the document residing on Google Cloud Storage.

Language:

(Optional Parameter). Specify the language of the text.

Store Response into:

Enter a name to be the variable that you want to store the API response results in.

Store Status Code:

Enter a name to be the variable that you want to store the status code returned in.

Analyze Entities***Description:***

This action invokes the Google Cloud Natural Language service named Analyze Entities.

✕
Properties of 'Analyze Entities' action



Analyze Entities

This action invokes the Google Cloud Natural Language service named Analyze Entities.

General | Advanced | Exception Handling

Action Input

API Key: ⓘ ⚙️

Document Type: Plain Text ▼ ⓘ

Provide Document: From File ▼ ⓘ

File Path: ⓘ ⚙️ 📄

Language: ⓘ ⚙️

[API Reference](#)

Action Output

Store Response into: ⓘ

Store Status Code: ⓘ

This action is Enabled

More Info
OK
Cancel

Properties:

API Key:

Specify the Google API Key to be used for this API call.

Document Type:

Select the type of document you want to transmit.

Provide Document:

Select the way that you want to provide the document. You can either specify the full path of the document or a Google Cloud Storage URI of it.

File Path:

Enter or choose a file path, or a previously defined variable, to specify the full path (folder plus filename) of the document that you want to transmit.

GCS Content URI:

Enter the URI of the document residing on Google Cloud Storage.

Language:

(Optional Parameter). Specify the language of the text.

Store Response into:

Enter a name to be the variable that you want to store the API response results in.

Store Status Code:

Enter a name to be the variable that you want to store the status code returned in.

Analyze Syntax***Description:***

This action invokes the Google Cloud Natural Language service named Analyze Syntax.

Properties of 'Analyze Syntax' action

 **Analyze Syntax**
This action invokes the Google Cloud Natural Language service named Analyze Syntax.

General | Advanced | Exception Handling

Action Input

API Key: ⓘ ⚙️

Document Type: Plain Text ⓘ

Provide Document: From File ⓘ

File Path: ⓘ ⚙️ 📄

Language: ⓘ ⚙️

[API Reference](#)

Action Output

Store Response into: %JSONResponse% ⓘ

Store Status Code: %StatusCode% ⓘ

This action is Enabled

Properties:**API Key:**

Specify the Google API Key to be used for this API call.

Document Type:

Select the type of document you want to transmit.

Provide Document:

Select the way that you want to provide the document. You can either specify the full path of the document or a Google Cloud Storage URI of it.

File Path:

Enter or choose a file path, or a previously defined variable, to specify the full path (folder plus filename) of the document that you want to transmit.

GCS Content URI:

Enter the URI of the document residing on Google Cloud Storage.

Language:

(Optional Parameter). Specify the language of the text.

Store Response into:

Enter a name to be the variable that you want to store the API response results in.

Store Status Code:

Enter a name to be the variable that you want to store the status code returned in.

3.11.18.2 IBM**3.11.18.2, Visual Recognition****Classify Image*****Description:***

This action invokes the IBM service named Classify.

Properties of 'Classify Image' action

Classify Image
This action invokes the IBM service named Classify.

General | Advanced | Exception Handling

Action Input

API Key: ⓘ ⚙️

Version Date: ⓘ

Provide Image: ⓘ

Image File: ⓘ ⚙️ 📄

Owners: ⓘ ⚙️

Classifier Ids: ⓘ ⚙️

Threshold: ⓘ ⚙️

Language: ⓘ ⚙️

[API Reference](#)

Action Output

Store Response into: ⓘ

Store Status Code: ⓘ

This action is Enabled

Properties:**API Key:**

Specify the API Key to be used for this call.

Version Date:

Specify the release date of the version of the API you want to use.

Provide Image:

Select the way that you want to provide the image. You can either specify the full path of the image or a URL address of it.

Image File Path:

Specify the full file path of the image file (.jpg, or .png) or compressed (.zip) file of images to classify.

Image URL:

Specify the URL of the image that you want to classify.

Owners:

(Optional Parameter). Specify as a comma-separated list, which classifiers to run.

Classifier Ids:

(Optional Parameter). Specify a comma-separated list of the classifier IDs used to classify the images.

Threshold:

(Optional Parameter). Specify as a floating value the minimum score a class must have to be displayed in the response.

Language:

(Optional Parameter). Specify the language of the output.

Store Response into:

Enter a name to be the variable that you want to store the API response results in.

Store Status Code:

Enter a name to be the variable that you want to store the status code returned in.

Detect Faces***Description:***

This action invokes the IBM service named Detect Faces.

Properties of 'Detect Faces' action

Detect Faces
This action invokes the IBM service named Detect Faces.

General | Advanced | Exception Handling

Action Input

API Key: ⓘ ⚙️

Version Date: ⓘ

Provide Image: ⓘ

Image File: ⓘ ⚙️ 📄

[API Reference](#)

Action Output

Store Response into: ⓘ

Store Status Code: ⓘ

This action is Enabled

Properties:**API Key:**

Specify the API Key to be used for this call.

Version Date:

Specify the release date of the version of the API you want to use.

Provide Image:

Select the way that you want to provide the image. You can either specify the full path of the image or a URL address of it.

Image File Path:

Specify the full file path of the image you want to use for this call.

Image URL:

Specify the URL of the image that you want to use for this call.

Store Response into:

Enter a name to be the variable that you want to store the API response results in.

Store Status Code:

Enter a name to be the variable that you want to store the status code returned in.

3.11.18.2 Document Conversion**Convert Document*****Description:***

This action invokes the IBM service named Convert Document.

Properties of 'Convert Document' action

Convert Document
This action invokes the IBM service named Convert Document.

General | Advanced | Exception Handling

Action Input

Username: ⓘ ⚙️

Password: ⓘ ⚙️

Version Date: ⓘ

File: ⓘ ⚙️ 📄

MIME Type: text/html ▼ ⓘ

Conversion Target: answer_units ▼ ⓘ

Answer Units: ⓘ ⚙️

[API Reference](#)

Action Output

Store Response into: %Response% ⓘ

Store Status Code: %StatusCode% ⓘ

This action is Enabled

More Info OK Cancel

Properties:

Username:

Specify the username to be used for this call.

Password:

Specify the password to be used for this call.

Version Date:

Specify the release date of the version of the API you want to use.

File:

Specify the full path of the document that you want to use for this call.

MIME Type:

Specify the MIME type of the file.

Conversion Target:

Specify the output format of the conversion.

Answer Units:

(Optional Parameter). Specify the heading levels as a comma-separated string.

Store Response into:

Enter a name to be the variable that you want to store the API response results in.

Store Status Code:

Enter a name to be the variable that you want to store the status code returned in.

3.11.18.2, Tone Analyzer**Analyze Tone*****Description:***

This action invokes the IBM service named Analyze Tone.

Properties of 'Analyze Tone' action ✕

Analyze Tone
This action invokes the IBM service named Analyze Tone.

General | Advanced | Exception Handling

Action Input

Username: ⓘ ⚙️

Password: ⓘ ⚙️

Version Date: ⓘ

Provide Text: ⓘ

Text: ⓘ ⚙️

Content Type: ⓘ

Tones: ⓘ ⚙️

Sentences: ⓘ ⚙️

[API Reference](#)

Action Output

Store Response into: ⓘ

Store Status Code: ⓘ

This action is Enabled

Properties:**Username:**

Specify the username to be used for this call.

Password:

Specify the password to be used for this call.

Version Date:

Specify the release date of the version of the API you want to use.

Provide Text:

Select the way that you want to provide the text. You can either specify the text directly or by providing the full file path.

File:

Specify the full path of the text file that you want to send.

Text:

Specify the text you want to send.

Content Type:

Select the content type of the text that you want to send.

Tones:

(Optional Parameter). Specify the tone with which you want to filter the results.

Sentences:

(Optional Parameter). Specify if you want to remove the sentence level analysis.

Store Response into:

Enter a name to be the variable that you want to store the API response results in.

Store Status Code:

Enter a name to be the variable that you want to store the status code returned in.

3.11.18.2. Language Translator**Translate*****Description:***

This action invokes the IBM service named Translate.

Properties of 'Translate' action

Translate
This action invokes the IBM service named Translate.

General | Advanced | Exception Handling

Action Input

Username: ⓘ ⚙

Password: ⓘ ⚙

Mode: Model Id ▾ ⓘ

Model Id: ⓘ ⚙

Text: ⓘ ⚙

[API Reference](#)

Action Output

Store Response into: %JSONResponse% ⓘ

Store Status Code: %StatusCode% ⓘ

This action is Enabled

More Info OK Cancel

Properties:

Username:

Specify the username to be used for this call.

Password:

Specify the password to be used for this call.

Mode:

Select the mode to be used for this call. You can either select to provide a translation model id or a source and a target language.

Model Id:

Specify the unique model_id of the translation model that will be used to translate the text.

Source:

Specify the source language of the text.

Target:

Specify the translation target language in 2 or 5 letter language code.

Text:

Specify the text to send.

Store Response into:

Enter a name to be the variable that you want to store the API response results in.

Store Status Code:

Enter a name to be the variable that you want to store the status code returned in.

Identify Language***Description:***

This action invokes the IBM service named Identify Language.

Properties of 'Identify Language' action

Identify Language
This action invokes the IBM service named Identify Language.

General | Advanced | Exception Handling

Action Input

Username: ⓘ ⚙️

Password: ⓘ ⚙️

Text: ⓘ ⚙️

Content Type: ⓘ ⚙️

[API Reference](#)

Action Output

Store Response into: ⓘ

Store Status Code: ⓘ

This action is Enabled

Properties:

Username:

Specify the username to be used for this call.

Password:

Specify the password to be used for this call.

Text:

Specify the text that you want to analyze.

Content Type:

Specify the format of the requested values.

Store Response into:

Enter a name to be the variable that you want to store the API response results in.

Store Status Code:

Enter a name to be the variable that you want to store the status code returned in.

3.11.18.3 Microsoft

3.11.18.3.1 Computer Vision

Analyze Image

Description:

This action invokes the Microsoft Cognitive service named Analyze Image.

Properties of 'Analyze Image' action

Analyze Image
This action invokes the Microsoft Cognitive service named Analyze Image.

General | Advanced | Exception Handling

Action Input

Server Location: West US

Subscription Key:

Provide Image: By File Path

Image File:

Visual Features:

Details:

Language:

[API Reference](#)

Action Output

Store Response into: %JSONResponse%

Store Status Code: %StatusCode%

This action is Enabled

More Info OK Cancel

Properties:**Server Location:**

Specify the server location to be used for this API call.

Subscription Key:

Specify the subscription key to be used for this API call.

Provide Image:

Select the way you will provide the image. You can either specify the full path of the image or a URL address of it.

Image File:

Enter or choose an image file path, or a previously defined variable, to specify the full path (folder plus filename) of the image file that you want to transmit.

Image URL:

Enter the URL address of an image.

Visual Features:

(Optional Parameter). Specify a string indicating what visual feature types to return. Multiple values should be comma-separated. For example: Categories, Tags, Description.

Details:

(Optional Parameter). Specify a string indicating which domain-specific details to return. Multiple values should be comma-separated.

Language:

(Optional Parameter). Specify a string indicating which language to return. The service will return recognition results in specified languages.

Store Response into:

Enter a name to be the variable that you want to store the API response results in.

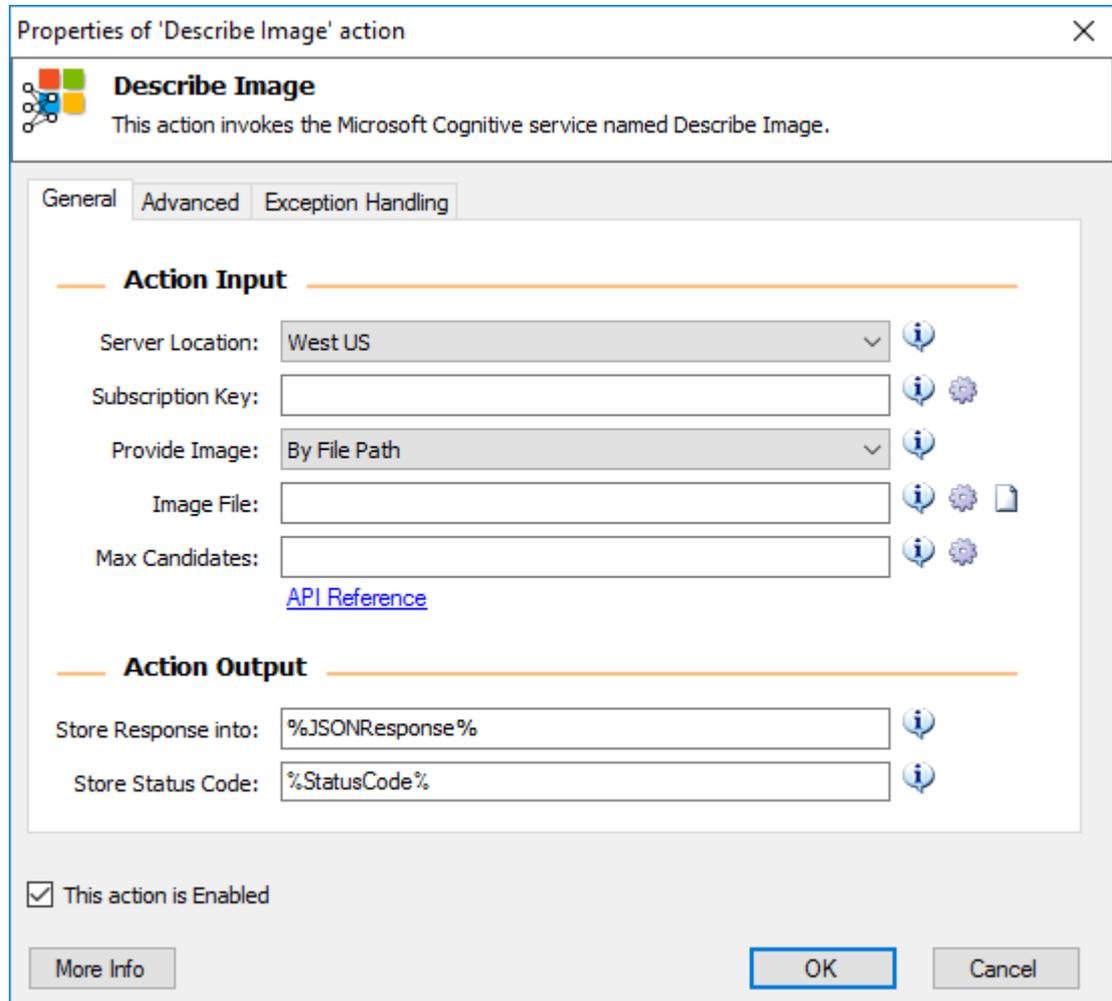
Store Status Code:

Enter a name to be the variable that you want to store the status code returned in.

Describe Image

Description:

This action invokes the Microsoft Cognitive service named Describe Image.



The screenshot shows a configuration window titled "Properties of 'Describe Image' action". The window has a close button (X) in the top right corner. Below the title bar, there is a header section with a logo and the text "Describe Image" and "This action invokes the Microsoft Cognitive service named Describe Image." Below this, there are three tabs: "General", "Advanced", and "Exception Handling". The "General" tab is selected. The "Action Input" section contains the following fields: "Server Location" (a dropdown menu set to "West US"), "Subscription Key" (a text input field), "Provide Image" (a dropdown menu set to "By File Path"), "Image File" (a text input field), and "Max Candidates" (a text input field). Each field has an information icon (i) and a settings icon (gear). Below the "Max Candidates" field is a link labeled "API Reference". The "Action Output" section contains two fields: "Store Response into" (a text input field with the value "%JSONResponse%") and "Store Status Code" (a text input field with the value "%StatusCode%"). Each field has an information icon (i). At the bottom of the window, there is a checkbox labeled "This action is Enabled" which is checked. There are three buttons at the bottom: "More Info", "OK", and "Cancel".

Properties:

Server Location:

Specify the server location to be used for this API call.

Subscription Key:

Specify the subscription key to be used for this API call.

Provide Image:

Select the way you will provide the image. You can either specify the full path of the image or a URL address of it.

Image File:

Enter or choose an image file path, or a previously defined variable, to specify the full path (folder plus filename) of the image file that you want to transmit.

Image URL:

Enter the URL address of an image.

Max Candidates:

(Optional Parameter). Specify the maximum number of candidate descriptions to be returned. The default is 1.

Store Response into:

Enter a name to be the variable that you want to store the API response results in.

Store Status Code:

Enter a name to be the variable that you want to store the status code returned in.

OCR***Description:***

This action invokes the Microsoft Cognitive service named OCR.

Properties of 'OCR' action

OCR
This action invokes the Microsoft Cognitive service named OCR.

General | Advanced | Exception Handling

Action Input

Server Location: West US 

Subscription Key:  

Provide Image: By File Path 

Image File:   

Language:  

Detect Orientation:  

[API Reference](#)

Action Output

Store Response into: 

Store Status Code: 

This action is Enabled

Properties:**Server Location:**

Specify the server location to be used for this API call.

Subscription Key:

Specify the subscription key to be used for this API call.

Provide Image:

Select the way you will provide the image. You can either specify the full path of the image or a URL address of it.

Image File:

Enter or choose an image file path, or a previously defined variable, to specify the full path (folder plus filename) of the image file that you want to transmit.

Image URL:

Enter the URL address of an image.

Language:

(Optional Parameter). Specify the BCP-47 language code of the text that you want to detect in the language.

Detect Orientation:

(Optional Parameter). Specify whether to detect the text orientation in the image.

Store Response into:

Enter a name to be the variable that you want to store the API response results in.

Store Status Code:

Enter a name to be the variable that you want to store the status code returned in.

Tag Image**Description:**

This action invokes the Microsoft Cognitive service named Tag Image.

Properties of 'Tag Image' action

Tag Image
This action invokes the Microsoft Cognitive service named Tag Image.

General | Advanced | Exception Handling

Action Input

Server Location: West US

Subscription Key:

Provide Image: By File Path

Image File:

[API Reference](#)

Action Output

Store Response into: %JSONResponse%

Store Status Code: %StatusCode%

This action is Enabled

More Info OK Cancel

Properties:

Server Location:

Specify the server location to be used for this API call.

Subscription Key:

Specify the subscription key to be used for this API call.

Provide Image:

Select the way you will provide the image. You can either specify the full path of the image or a URL address of it.

Image File:

Enter or choose an image file path, or a previously defined variable, to specify the full path (folder plus filename) of the image file that you want to transmit.

Image URL:

Enter the URL address of an image.

Store Response into:

Enter a name to be the variable that you want to store the API response results in.

Store Status Code:

Enter a name to be the variable that you want to store the status code returned in.

3.11.18.3, Face**Detect Face*****Description:***

This action invokes the Microsoft Cognitive service named Face - Detect.

Properties of 'Detect Face' action

Detect Face
This action invokes the Microsoft Cognitive service named Face - Detect.

General | Advanced | Exception Handling

Action Input

Server Location: West US 

Subscription Key:  

Provide Image: By File Path 

Image File:   

Return Face Id:  

Return Face Landmarks:  

Return Face Attributes:  

[API Reference](#)

Action Output

Store Response into: %JSONResponse% 

Store Status Code: %StatusCode% 

This action is Enabled

Properties:

Server Location:

Specify the server location to be used for this API call.

Subscription Key:

Specify the subscription key to be used for this API call.

Provide Image:

Select the way you will provide the image. You can either specify the full path of the image or a URL address of it.

Image File:

Enter or choose an image file path, or a previously defined variable, to specify the full path (folder plus filename) of the image file that you want to transmit.

Image URL:

Enter the URL address of an image.

Return Face Id:

(Optional Parameter). Specify whether to return the detected faces or not. The default value is true.

Return Face Landmarks:

(Optional Parameter). Specify whether to return face landmarks of the detected faces or not. The default value is false.

Return Face Attributes:

(Optional Parameter). Specify the face attributes you wish to return. For multiple attributes provide a comma-separated string.

Store Response into:

Enter a name to be the variable that you want to store the API response results in.

Store Status Code:

Enter a name to be the variable that you want to store the status code returned in.

Find Similar Face**Description:**

This action invokes the Microsoft Cognitive service named Face - Find Similar.

Properties of 'Find Similar Face' action

Find Similar Face
This action invokes the Microsoft Cognitive service named Face - Find Similar.

General | Advanced | Exception Handling

Action Input

Server Location: West US 

Subscription Key:  

Face Id:  

Provide Face Ids: By Face List Id 

Face List Id:  

Max Num Of Candidates Returned:  

Mode:  

[API Reference](#)

Action Output

Store Response into: %JSONResponse% 

Store Status Code: %StatusCode% 

This action is Enabled

Properties:**Server Location:**

Specify the server location to be used for this API call.

Subscription Key:

Specify the subscription key to be used for this API call.

Face Id:

Specify the faceld of the query to be used.

Face Ids:

Specify a candidate faceld or an array of candidate facelds.

Provide Face Ids:

Select the way that you want to provide the face ids. You can either specify the id of a face-list or the ids of the faces.

Face List Id:

Specify an existing user-specified unique candidate face-list.

Max Num Of Candidates Returned:

(Optional Parameter). Specify the number of the max similar faces that will be returned.

Mode:

(Optional Parameter). Specify the similar face searching mode.

Store Response into:

Enter a name to be the variable that you want to store the API response results in.

Store Status Code:

Enter a name to be the variable that you want to store the status code returned in.

3.11.18.3, Text Analytics**Detect Language*****Description:***

This action invokes the Microsoft Cognitive service named Text Analytics - Detect Language.

Properties of 'Detect Language' action

Detect Language
This action invokes the Microsoft Cognitive service named Text Analytics - Detect Language.

General | Advanced | Exception Handling

Action Input

Server Location: West US 

Subscription Key:  

Text:  

Number of Languages to Detect:  

[API Reference](#)

Action Output

Store Response into: %JSONResponse% 

Store Status Code: %StatusCode% 

This action is Enabled

[More Info](#)

Properties:

Server Location:

Specify the server location to be used for this API call.

Subscription Key:

Specify the subscription key to be used for this API call.

Text:

Specify the text you want to analyze.

Number of Languages to Detect:

(Optional Parameter). Specify the number of languages to detect.

Store Response into:

Enter a name to be the variable that you want to store the API response results in.

Store Status Code:

Enter a name to be the variable that you want to store the status code returned in.

Key Phrases

Description:

This action invokes the Microsoft Cognitive service named Text Analytics - Key Phrases.

The screenshot shows a dialog box titled "Properties of 'Key Phrases' action". It features a header with the action name and a description: "This action invokes the Microsoft Cognitive service named Text Analytics - Key Phrases." Below the header are three tabs: "General", "Advanced", and "Exception Handling". The "General" tab is active and contains the following fields:

- Action Input:**
 - Server Location: A dropdown menu set to "West US".
 - Subscription Key: A text input field.
 - Text: A text input field.
 - Language: A text input field.
 - A blue link labeled "API Reference" is positioned below the Language field.
- Action Output:**
 - Store Response into: A text input field containing "%JSONResponse%".
 - Store Status Code: A text input field containing "%StatusCode%".

At the bottom of the dialog, there is a checkbox labeled "This action is Enabled" which is checked. There are also three buttons: "More Info", "OK", and "Cancel".

Properties:

Server Location:

Specify the server location to be used for this API call.

Subscription Key:

Specify the subscription key to be used for this API call.

Text:

Specify the text or the list of texts you want to analyze.

Language:

Specify the language of the text(s).

Store Response into:

Enter a name to be the variable that you want to store the API response results in.

Store Status Code:

Enter a name to be the variable that you want to store the status code returned in.

Sentiment***Description:***

This action invokes the Microsoft Cognitive service named Text Analytics - Sentiment.

Properties of 'Sentiment' action

Sentiment
This action invokes the Microsoft Cognitive service named Text Analytics - Sentiment.

General | Advanced | Exception Handling

Action Input

Server Location: West US 

Subscription Key:  

Text:  

Language:  

[API Reference](#)

Action Output

Store Response into: %JSONResponse% 

Store Status Code: %StatusCode% 

This action is Enabled

Properties:**Server Location:**

Specify the server location to be used for this API call.

Subscription Key:

Specify the subscription key to be used for this API call.

Text:

Specify the text or a list of texts that you want to analyze.

Language:

Specify the 2 letter ISO 639-1 representation of the language of the text(s).

Store Response into:

Enter a name to be the variable that you want to store the API response results in.

Store Status Code:

Enter a name to be the variable that you want to store the status code returned in.

3.11.18.3, Bing Spell Check

Spell Check

Description:

This action invokes the Microsoft Cognitive service named Bing Spell Check.

Properties of 'Spell Check' action

Spell Check
This action invokes the Microsoft Cognitive service named Bing Spell Check.

General | Advanced | Exception Handling

Action Input

Subscription Key: ⓘ ⚙️

Text: ⓘ ⚙️

Mode: ⓘ ⚙️

Mkt: ⓘ ⚙️

[API Reference](#)

Action Output

Store Response into: ⓘ

Store Status Code: ⓘ

This action is Enabled

More Info | OK | Cancel

Properties:

Subscription Key:

Specify the subscription key to be used for this API call.

Text:

Specify the text or the list of texts you want to send.

Mode:

(Optional Parameter). Specify the similar face searching mode.

Mkt:

(Optional Parameter). For proof mode, only support en-us, es-es, pt-br. For spell mode, support all language codes.

Store Response into:

Enter a name to be the variable that you want to store the API response results in.

Store Status Code:

Enter a name to be the variable that you want to store the status code returned in.

3.11.19 Database

3.11.19.1 Open SQL Connection

Description:

This action opens a new connection to a database.

Properties:

Connection String: Enter how to connect with the database. The button on the right (with ellipses) will allow you to build the connection string.

Store SQL Connection into: Enter a name to be the variable that will store the specific SQL connection for use with later SQL statements.

3.11.19.2 Execute SQL Statement Action

Description:

This action connects to a database and executes a SQL statement

Properties of 'Execute SQL Statement' action

Execute SQL Statement
This action connects to a database and executes a SQL statement

General | Exception Handling

Action Input

Connection String:

SQL Statement:

```
Select * from Table
Where Name=Value
```

Timeout: seconds

Action Output

Store Query Result into:

This action is Enabled

Properties:

Connection String:

Enter how to connect with the database. The button on the right (with ellipses) will allow you to build the connection string.

SQL Statement:

Enter the SQL Statement to be issued to the database here

Timeout:

The Execute SQL Statement action waits for a result from the database. Choose a maximum amount of time that the action will wait.

Store Query Result into:

Enter a name to be the variable that will store the result from the database in the form of a data table, with rows and columns.

More about connecting to a database and executing SQL Queries on it, please visit:

[Connecting to a Database](#)^[472]

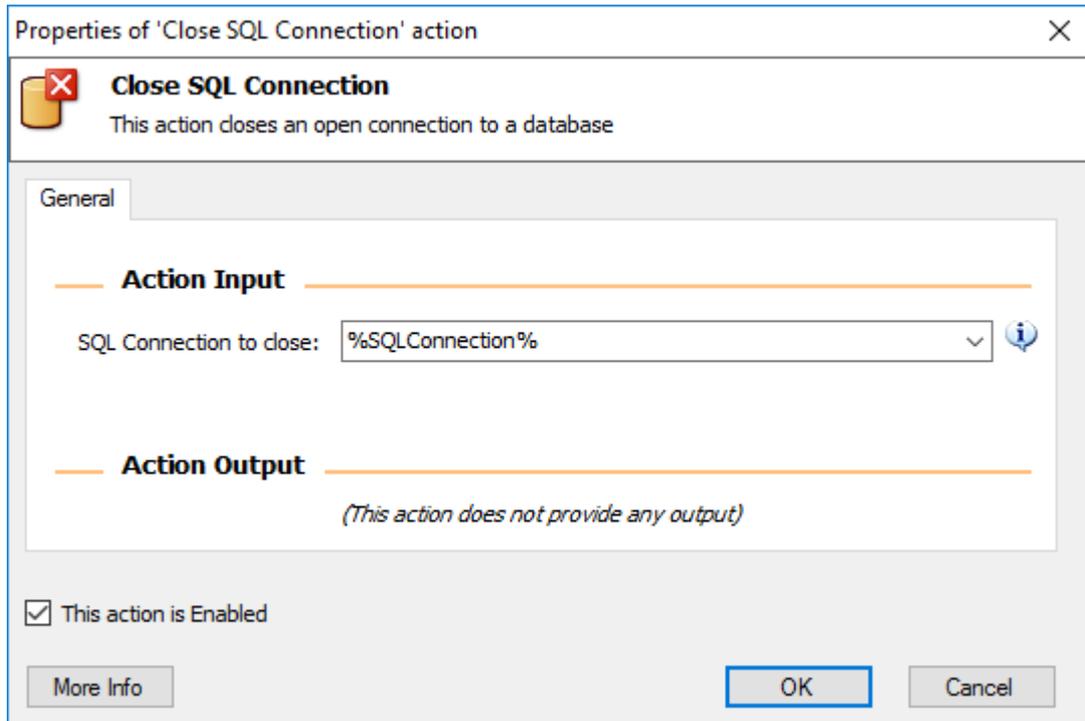
[Connect to Excel or Access Database](#)^[473]

[Connection Strings and Drivers](#)^[473]

3.11.19.3 Close SQL Connection

Description:

This action closes an open connection to a database.



Properties:

SQL Connection to close: Enter the variable containing the previously opened SQL connection you wish to close. You must have previously specified this variable in an Open SQL connection action.

3.11.20 Excel

3.11.20.1 Launch Excel Action

Description:

This action launches a new Excel Instance and opens an Excel document

Properties of 'Launch Excel' action

Launch Excel
This action launches a new Excel Instance and opens an Excel document

General | Advanced | Exception Handling

Action Input

Launch Excel: and open the following document

Document Path: C:\Users\me\Desktop\My Excel.xlsx

Make Instance Visible

Action Output

Store Excel Instance into: %ExcelInstance%

This action is Enabled

More Info | OK | Cancel

Properties:

Launch Excel:

Choose whether you want to open a New Excel document, or an existing one.

Document Path:

Enter the full path of the existing Excel document you wish to open

Make Instance Visible:

Choose whether you want to make the Excel Window visible, or hide it. This doesn't limit ProcessRobot's ability to use Excel, just the whether the user sees it.

Store Excel Instance into:

Enter a name to be the variable that will store the specific Excel Instance for use with later Excel actions. This allows you to specify which, of possibly several, Excel spreadsheet to access.

Cautions:

For any Excel-related action to work correctly, Microsoft Excel must be installed on the computer where the Process will run.

3.11.20.2 Attach to Running Excel Action

Description:

Attached to an Excel Document that's already open.

Properties:

Document Name:

Enter either the name or the path of the Excel file you wish to attach to.

Store Excel Instance into:

Enter the name to be the variable that will hold the Excel Instance this action has attached to, for use with later Excel actions.

3.11.20.3 Read from Excel Worksheet

Description:

This action reads the value of a cell or a range of cells from the active worksheet of a previously launched Excel Instance:

Properties of 'Read from Excel Worksheet' action

Read from Excel Worksheet

This action reads the value of a cell or a range of cells from the active worksheet of a previously launched Excel Instance

General | Advanced | Exception Handling

Action Input

Excel Instance: %ExcelInstance% 

Retrieve: Values from a Range of Cells 

Range Starts At:

Column: 1  

Row: 1  

Range Ends At:

Column: 4  

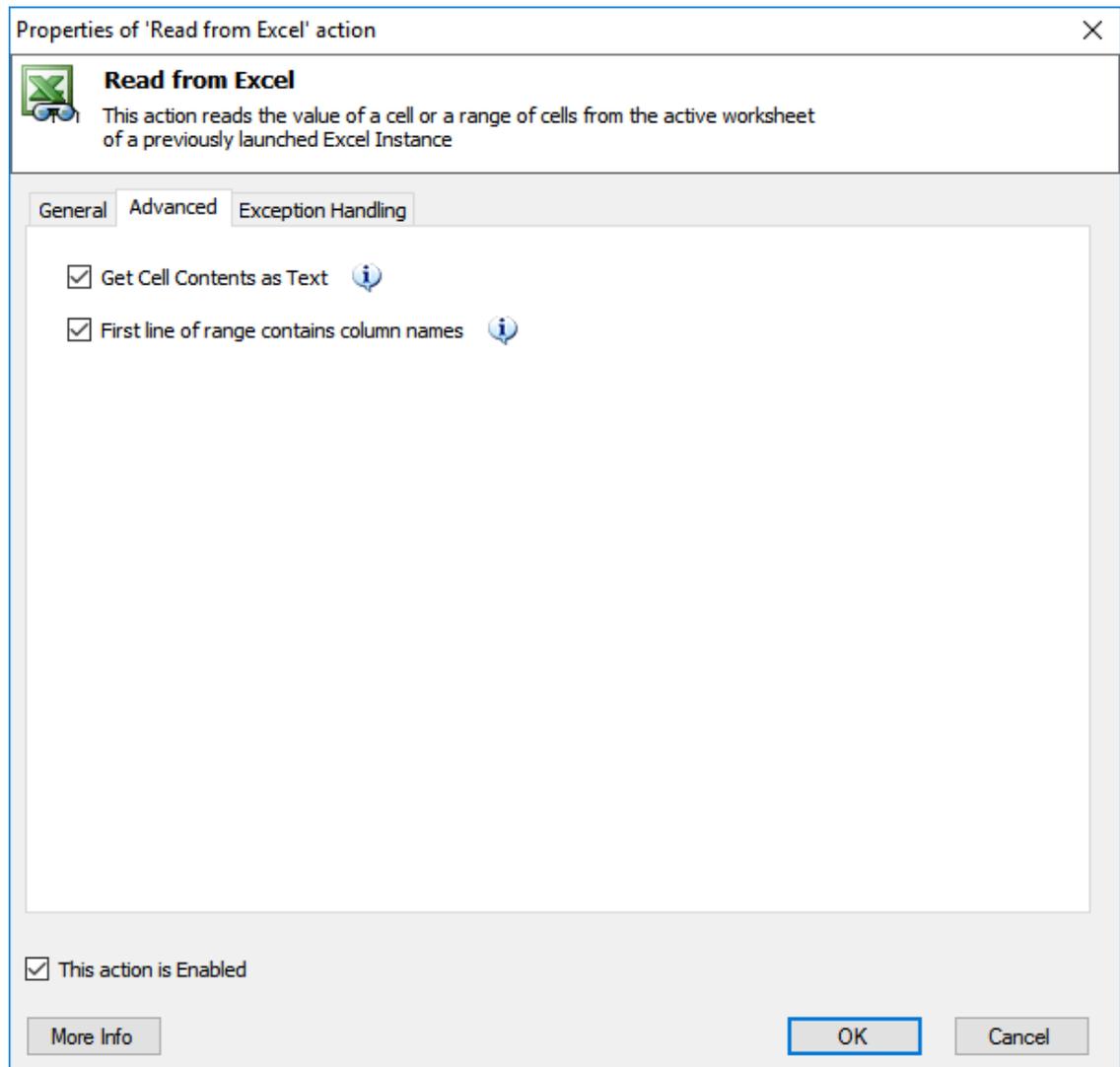
Row: %FirstFreeRow-1%  

Action Output

Store Cell Value(s) into: %ExcelData% 

This action is Enabled

[More Info](#) [OK](#) [Cancel](#)



Properties:

Excel Instance:

Enter the variable that contains the Excel Instance you want to work with. You must have previously specified this variable in a Launch Excel action.

Retrieve:

Choose whether to retrieve the value of a single cell or a table from a range of cells.

Start Column:

Enter a numeric value to be the cell column or starting column number. This must be a number. For example, Column F requires the entry of '6'.

Start Row:

Enter a numeric value to be the cell row or starting row number.

End Column:

Enter a numeric value to be the ending column number. This must be a number. For example, Column J requires the entry of '10'.

End Row:

Enter a numeric value to be the ending row number.

Advanced Properties Tab - Get Cell Contents as Text:

Check this to retrieve the content of the cell(s) purely as text. Leave this option unchecked if you want the action's output variable type to match (as closely as possible) the source cell's data type. In that case, a date in Excel will be stored as DateTime variable in ProcessRobot, numbers as numeric variables etc.

Advanced Properties Tab - First line of range contains column names:

Choose whether this action considers the first row as column names. If it does, the names won't be read as data into the table, and later actions can search the data by column names.

Store Cell Value(s) into:

Enter a name to be the variable that will store the value of the single cell or a table from the range of cells.

3.11.20.4 Save Excel***Description:***

This action saves a previously launched Excel Instance

Properties:

1. Excel Instance:

Enter the variable that contains the Excel Instance you want to work with. You must have previously specified this variable in a Launch Excel action.

2. Save Mode:

Choose how to save the document of this instance.

3.11.20.5 Write to Excel Worksheet

Description:

This action writes some value or the contents of a variable into a cell or a range of cells of a previously launched Excel Instance:

Properties of 'Write to Excel Worksheet' action

 **Write to Excel Worksheet**
This action writes some value or the contents of a variable into a cell or a range of cells of a previously launched Excel Instance

General | Exception Handling

Action Input

Excel Instance: [%ExcelInstance%] ⓘ

Value to Write: [%CurrentDateTime%] ⓘ ⚙️

Write Mode: On Specified Cell ⓘ

Write Value Into Cell At:

Column: 1 ⓘ ⚙️

Row: [%FirstFreeRow%] ⓘ ⚙️

Action Output

(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties:

Excel Instance:

Enter the variable that contains the Excel Instance you want to work with. You must have previously specified this variable in a Launch Excel action.

Value to Write:

Enter the text, number, or a previously stored variable, to be inserted. If the variable contains a table, it will fill in cells to the right and below, writing over other cell data if need be, and a list will fill in cells below.

Write mode:

This drop down menu allows you to write either On Specified Cell or On the Cell that is Currently Active. This can be quite useful as you might not know in advance the Cell that you want to write in.

Cell Column:

Enter a numeric value to be the column number for the cell this action will write to. This must be a number. For example, Column F requires the entry of '6'.

Cell Row:

Enter a number to be the row for the cell this action will write to.

Cautions:

For any Excel-related action to work correctly, Microsoft Excel must be installed on the computer where the Process will run.

3.11.20.6 Close Excel Action

Description:

This action closes a previously launched Excel Instance

Properties of 'Close Excel' action

Close Excel
This action closes a previously launched Excel Instance

General | Exception Handling

Action Input

Excel Instance: %ExcelInstance%

Before Closing Excel: Save documents as

Document Path: C:\Users\MyDocument.xlsx

Action Output

(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties:

Excel Instance:

Enter the variable that contains the Excel Instance you want to close. You must have previously specified this variable in a Launch Excel action.

Before Closing Excel:

Choose whether and how to save the document of this Instance before closing that Instance.

Document Path:

Enter or choose the full path you want to save the document as.

Cautions:

For any Excel-related action to work correctly, Microsoft Excel must be installed on the computer where the Process will run.

3.11.20.7 Set Active Worksheet Action

Description:

This action activates a specific worksheet of a previously launched Excel Instance

Properties of 'Set Active Worksheet' action

Set Active Worksheet
This action activates a specific worksheet of a previously launched Excel Instance

General | Exception Handling

Action Input

Excel Instance: %ExcelWorksheet%

Choose Worksheet by: Index

Worksheet Index: 3

Action Output
(This action does not provide any output)

This action is Enabled

More Info | OK | Cancel

Properties:

Excel Instance:

Enter the variable that contains the Excel Instance you want to work with. You must have previously specified this variable in a Launch Excel action.

Choose Worksheet by:

Choose to find the worksheet by name or index.

Worksheet Index:

Enter the Index number of the Worksheet you wish to make active. The numbering starts from 1, meaning that the index of the first worksheet is 1, of the second is 2, etc.

Worksheet Name:

Enter the name of the Worksheet you wish to make active.

Cautions:

For any Excel-related action to work correctly, Microsoft Excel must be installed on the computer where the Process will run.

3.11.20.8 Add New Worksheet Action

Description:

This action adds a new worksheet to the document of a previously launched Excel Instance

Properties of 'Add New Worksheet' action

Add New Worksheet
This action adds a new worksheet to the document of a previously launched Excel Instance

General Exception Handling

Action Input

Excel Instance: %ExcelInstance%

New Worksheet Name: Summary

Add Worksheet as: First Worksheet

Action Output
(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties:

Excel Instance:

Enter the variable that contains the Excel Instance you want to work with. You must have previously specified this variable in a Launch Excel action.

New Worksheet Name:

Enter the text, or a previously defined text variable, to be used as the name of the new Worksheet.

Add Worksheet As:

Choose whether the new Excel Worksheet will be added before or after the existing Worksheets.

Cautions:

For any Excel-related action to work correctly, Microsoft Excel must be installed on the computer where the Process will run.

3.11.20.9 Delete Excel Worksheet Action***Description:***

This action deletes a specific worksheet from a previously launched excel Instance.

Properties of 'Delete Excel Worksheet' action ✕



Delete Excel Worksheet

This action deletes a specific worksheet from a previously launched Excel Instance

General
Exception Handling

Action Input

Excel Instance: i

Delete Worksheet with: i

Worksheet Name: i ⚙

Action Output

(This action does not provide any output)

This action is Enabled

More Info
OK
Cancel

Properties:

Excel Instance: Enter the variable that contains the Excel Instance you want to work with . You must have previously specified this variable in a Launch Excel action.

Delete Worksheet with: Choose whether you want to find the worksheet by Name or Index

Worksheet Name/Index: Enter the Name/Index of the worksheet you wish to delete.

3.11.20.1 Rename Excel Worksheet Action

Description:

This action renames a specific worksheet from a previously launched excel Instance.

Properties of 'Rename Excel Worksheet' action

Rename Excel Worksheet
This action renames a specific worksheet of a previously launched Excel Instance

General | Exception Handling

Action Input

Excel Instance: %ExcelInstance% 

Rename Worksheet with: Name 

Worksheet Name: Sheet1  

Worksheet New Name: Bookings  

Action Output

(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties:

Excel Instance: Enter the variable that contains the Excel Instance you want to work with . You must have previously specified this variable in a Launch Excel action.

Rename Worksheet with: Choose whether you want to find the worksheet by Name or Index

Worksheet Name/Index: Enter the Name/Index of the worksheet you wish to delete. (this will depend on what you selected in the previous option)

Worksheet New Name: Enter the new name of the worksheet

3.11.20.1 Get Active Excel Worksheet Action

Description:

This action retrieves an Excel document's active worksheet.

The screenshot shows a dialog box titled "Properties of 'Get Active Excel Worksheet' action". The dialog has a close button (X) in the top right corner. Below the title bar, there is a green checkmark icon and the text "Get Active Excel Worksheet" followed by "This action retrieves an Excel document's active worksheet." The dialog is divided into two tabs: "General" (selected) and "Exception Handling". Under the "General" tab, there are two sections: "Action Input" and "Action Output". In the "Action Input" section, there is a dropdown menu labeled "Excel Instance:" with the value "%ExcelInstance%" and an information icon (i). In the "Action Output" section, there are two text input fields: "Active Worksheet Name" with the value "%SheetName%" and "Active Worksheet Index" with the value "%SheetIndex%", both with information icons (i). At the bottom left, there is a checked checkbox labeled "This action is Enabled". At the bottom right, there are three buttons: "More Info", "OK", and "Cancel".

Properties:

Excel Instance: Enter the variable that contains the Excel Instance you want to work with . You must have previously specified this variable in a Launch Excel action

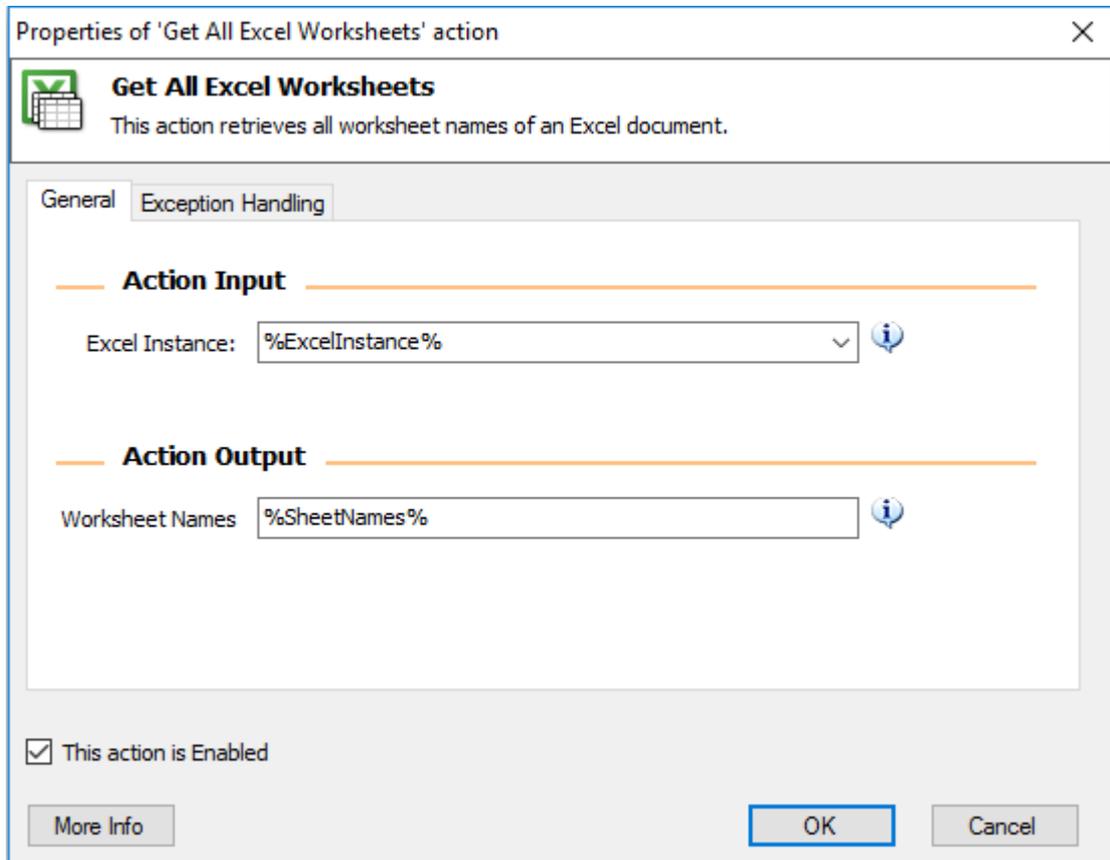
Active Worksheet Name: Enter a name to be the variable that will store the name of the active worksheet

Active Worksheet Index: Enter a name to be the variable that will store the index of the active worksheet

3.11.20.1: Get All Excel Worksheets Action

Description:

This action retrieves all worksheet names of an Excel document.



The screenshot shows a dialog box titled "Properties of 'Get All Excel Worksheets' action". The dialog has a close button (X) in the top right corner. Below the title bar, there is a green checkmark icon and the text "Get All Excel Worksheets" followed by "This action retrieves all worksheet names of an Excel document." Below this, there are two tabs: "General" (selected) and "Exception Handling". The "General" tab contains two sections: "Action Input" and "Action Output". Under "Action Input", there is a label "Excel Instance:" followed by a dropdown menu containing "%ExcelInstance%" and an information icon (i). Under "Action Output", there is a label "Worksheet Names" followed by a text input field containing "%SheetNames%" and an information icon (i). At the bottom left, there is a checkbox labeled "This action is Enabled" which is checked. At the bottom right, there are three buttons: "More Info", "OK", and "Cancel".

Properties:

Excel Instance: Enter the variable that contains the Excel Instance you want to work with . You must have previously specified this variable in a Launch Excel action

Worksheet Names: Enter a name to be the variable that will store the name of the all worksheets

3.11.20.1: Activate Cell in Excel Worksheet Action

Description:

This action activates a cell in the active worksheet of a previously launched Excel Instance, given column, row and offset.

Properties of 'Activate Cell in Excel Worksheet' action

Activate Cell in Excel Worksheet
This action activates a cell in the active worksheet of a previously launched Excel Instance, given column, row and offset

General | Exception Handling

Action Input

Excel Instance: %ExcelInstance%

Activate: Absolutely specified Cell

Cell Location:

Column: []

Row: []

Action Output

(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties:

Excel Instance:

Enter the variable that contains the Excel Instance you want to work with. You must have previously specified this variable in a Launch Excel action.

Activate:

Enter a name to be the variable that will store the numeric value of the first fully empty column. This will be a number. For example, if column F is the first empty column, it will be stored as '6'.

Cell Location Column: Enter a numeric value or letter to be the cell column.

Cell Location Row: Enter a numeric value to be the cell row. The numbering starts from 1.

3.11.20.1 Select Cells in Excel Worksheet Action

Description: This action selects a range of cells in the active worksheet of a previously launched Excel Instance

Properties of 'Select Cells in Excel Worksheet' action

Select Cells in Excel Worksheet
This action selects a range of cells in the active worksheet of a previously launched Excel Instance

General Exception Handling

Action Input

Excel Instance: [%ExcelInstance%] ⓘ

Select: Range of Cells ⓘ

Range Starts At:

Column: ⓘ ⚙

Row: ⓘ ⚙

Range Ends At:

Column: ⓘ ⚙

Row: ⓘ ⚙

Action Output

(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties of 'Select Cells in Excel Worksheet' action

 **Select Cells in Excel Worksheet**
This action selects a range of cells in the active worksheet of a previously launched Excel Instance

General | Exception Handling

Action Input

Excel Instance: ⓘ

Select: ⓘ

Range Ends At

X Axis Direction: ⓘ

Number of Cells: ⓘ ⚙️

Y Axis Direction: ⓘ

Number of Cells: ⓘ ⚙️

Action Output

(This action does not provide any output)

This action is Enabled

Properties:

Excel Instance: Enter the variable that contains the Excel Instance you want to work with. You must have previously specified this variable in a Launch Excel action.

Select: Choose whether to select an explicitly specified range of cells or a range of cells relatively to the currently active cell.

Range Start at Column: Enter a numeric value or letter to be the first column.

Range Starts at Row: Enter a numeric value to be the first row number.

Range Ends at Column: Enter a numeric value or letter to be the last column.

Range Ends at Row: Enter a numeric value or letter to be the last row. The numbering starts from 1.

Range Ends at X Axis direction: Select x-axis offset direction. Select where to look along the horizontal axis, based on currently activated cell's position.

Number of Cells: Enter a numeric value to be the X axis offset

Range Ends at Y Axis direction: Select y-axis offset direction. Select where to look along the vertical axis, based on currently activated cell's position.

Number of Cells: Enter a numeric value to be the Y axis offset

3.11.20.1!Get Selected Cell Range From Excel Worksheet Action

Description:

This action retrieves the selected range of cells in a structure consisting of first column, first row, last column and last row.

Properties of 'Get Selected Cell Range from Excel Worksheet' action

 **Get Selected Cell Range from Excel Worksheet**
 This action retrieves the selected range of cells in a structure consisting of first column, first row, last column and last row.

General | Exception Handling

Action Input

Excel Instance: 

Action Output

First Column Index: 

First Row Index: 

Last Column Index: 

Last Row Index: 

This action is Enabled

Properties:

First Column Index: Enter a name to be the variable that will store the numeric value of the range's first column.

First Row Index: Enter a name to be the variable that will store the numeric value of the range's first row.

Last Column Index: Enter a name to be the variable that will store the numeric value of the range's last column.

Last Row Index: Enter a name to be the variable that will store the numeric value of the range's last row.

3.11.20.1 Copy Cells in Exel Worksheet Action

This action copies a range of cells from the active worksheet of a previously launched Excel Instance

Properties of 'Copy Cells from Excel Worksheet' action

Copy Cells from Excel Worksheet
This action copies a range of cells from the active worksheet of a previously launched Excel Instance

General | Exception Handling

Action Input

Excel Instance: %ExcelInstance% ⓘ

Copy Mode: Single Cell's Value ⓘ

Cell Location

Column: ⓘ ⚙️

Row: ⓘ ⚙️

Action Output

(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Operation: Single Cell Value

Properties of 'Copy Cells from Excel Worksheet' action ✕



Copy Cells from Excel Worksheet

This action copies a range of cells from the active worksheet of a previously launched Excel Instance

General Exception Handling

Action Input

Excel Instance: 

Copy Mode: 

Range Starts At:

Column:  

Row:  

Range Ends At:

Column:  

Row:  

Action Output

(This action does not provide any output)

This action is Enabled

Operation: Value from a Range of Cells

Properties of 'Copy Cells from Excel Worksheet' action

Copy Cells from Excel Worksheet
This action copies a range of cells from the active worksheet of a previously launched Excel Instance

General | Exception Handling

Action Input

Excel Instance: %ExcelInstance% 

Copy Mode: Selection 

Action Output

(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Operation: Selection

Excel Instance: Enter the variable that contains the Excel Instance you want to work with. You must have previously specified this variable in a Launch Excel action.

Copy Mode: Choose whether to copy a single cell, a range of cells or the current selection of cells.

(The screenshots above show how the actions properties change depending on the operation that you will choose.)

Column: Enter a numeric value or letter to be the cell column.

Row: Enter a numeric value to be the cell row.

First Column: Enter a numeric value or letter to be the cell column or first column.

First Row: Enter a numeric value to be the cell row or first row number.

Last Column: Enter a numeric value or letter to be the last column.

Last Row: Enter a numeric value to be the last row number.

3.11.20.1 Paste Cells to Excel Worksheet Action

Description:

This action pastes a range of cells to the active worksheet of a previously launched Excel Instance

Properties of 'Paste Cells to Excel Worksheet' action

Paste Cells to Excel Worksheet
This action pastes a range of cells to the active worksheet of a previously launched Excel Instance

General Exception Handling

Action Input

Excel Instance: %ExcelInstance%

Paste Mode: On Specified Cell

Cell Location:

Column:

Row:

Action Output

(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties:

Excel Instance: Enter the variable that contains the Excel Instance you want to work with. You must have previously specified this variable in a Launch Excel action.

Paste Mode: Choose whether to paste on a specified cell or the currently active cell.

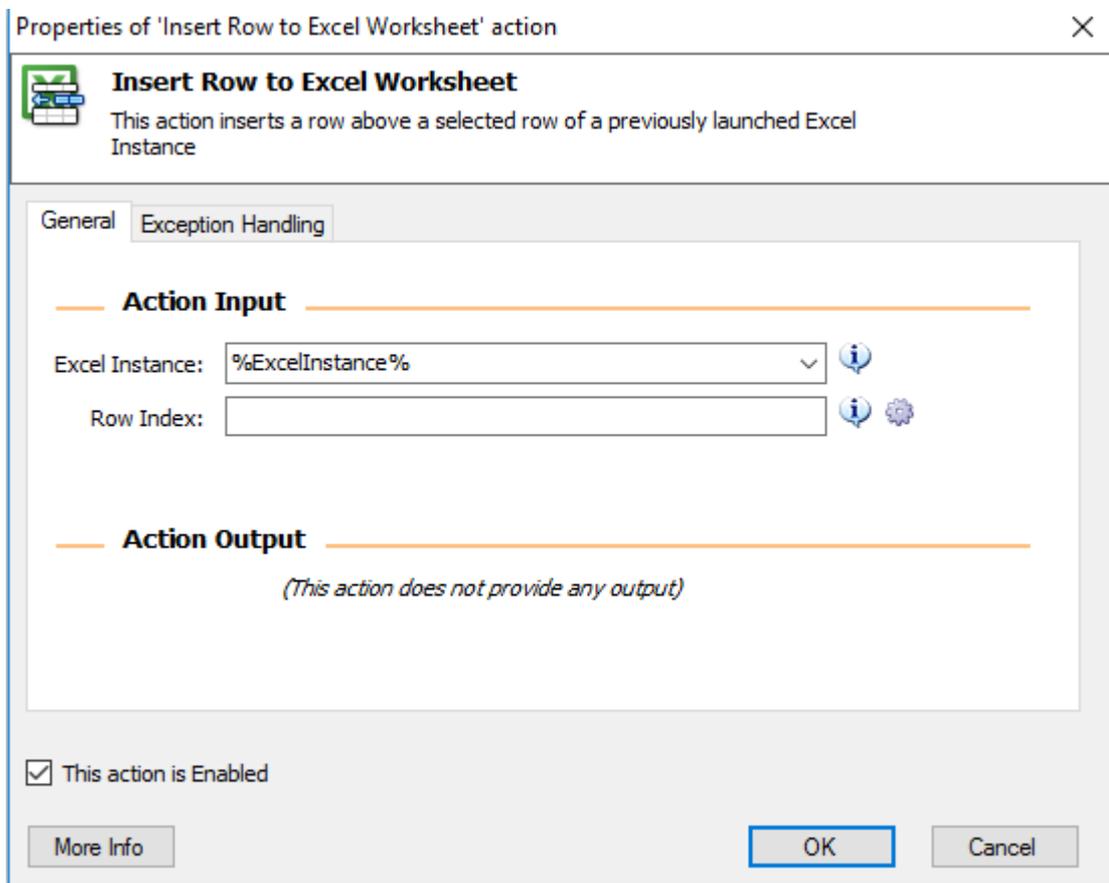
Column: Enter a numeric value or letter to be the cell column.

Row: Enter a numeric value to be the row number.

3.11.20.1 Insert Row in Excel Worksheet Action

Description:

This action inserts a row above a selected row of a previously launched Excel Instance



The screenshot shows a dialog box titled "Properties of 'Insert Row to Excel Worksheet' action". The dialog has a close button (X) in the top right corner. Below the title bar, there is a header section with an Excel icon and the text "Insert Row to Excel Worksheet" and "This action inserts a row above a selected row of a previously launched Excel Instance". The main area has two tabs: "General" (selected) and "Exception Handling". Under the "General" tab, there is a section titled "Action Input" with two input fields: "Excel Instance:" with a dropdown menu containing "%ExcelInstance%" and an information icon, and "Row Index:" with a text input field and information and settings icons. Below this is a section titled "Action Output" with the text "(This action does not provide any output)". At the bottom left, there is a checkbox labeled "This action is Enabled" which is checked. At the bottom right, there are three buttons: "More Info", "OK", and "Cancel".

Properties:

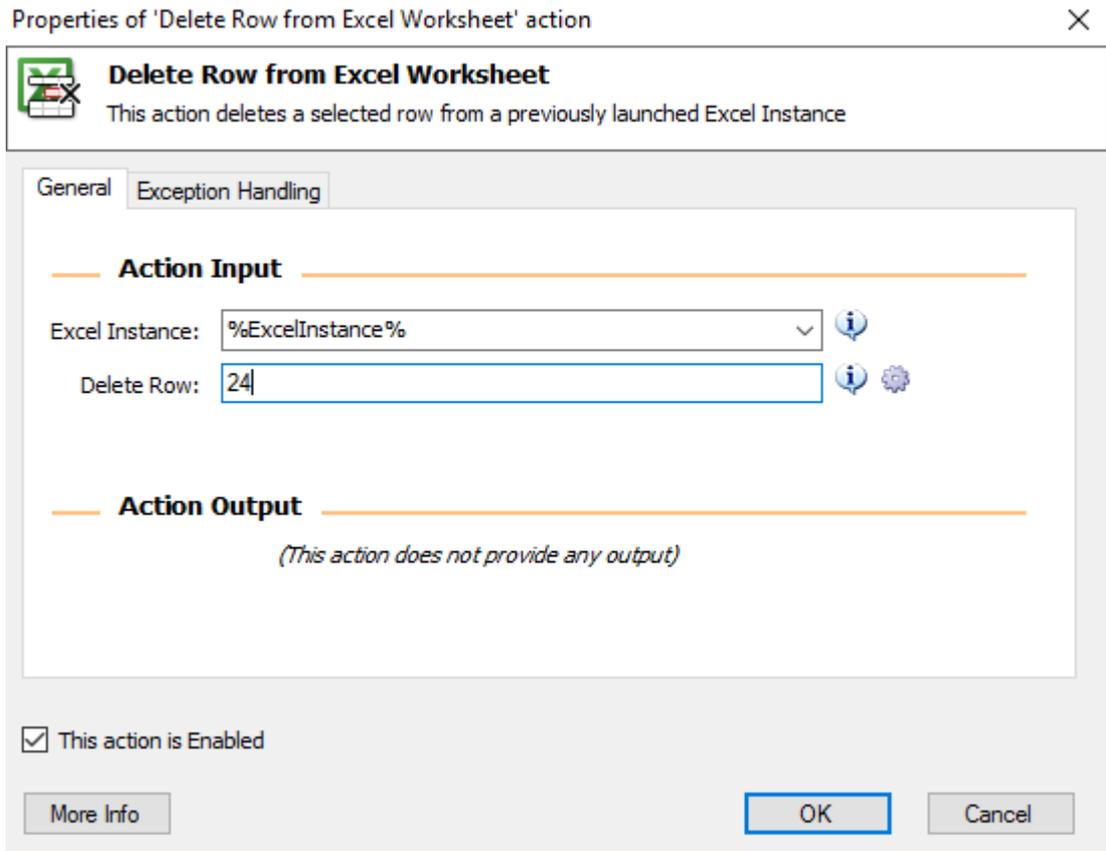
Excel Instance: Enter the variable that contains the Excel Instance you want to work with. You must have previously specified this variable in a Launch Excel action.

Row Index: Enter the Index number of the row you wish to add a new row above. The numbering starts from 1.

3.11.20.1!Delete Row from Excel Worksheet Action

Description:

This action inserts a row above a selected row of a previously launched Excel Instance



The screenshot shows the configuration window for the 'Delete Row from Excel Worksheet' action. The window title is 'Properties of 'Delete Row from Excel Worksheet' action'. The main title is 'Delete Row from Excel Worksheet' with a sub-description: 'This action deletes a selected row from a previously launched Excel Instance'. There are two tabs: 'General' (selected) and 'Exception Handling'. Under 'Action Input', there are two fields: 'Excel Instance' with a dropdown menu set to '%ExcelInstance%' and 'Delete Row' with a text input field containing '24'. Both fields have information and settings icons. Under 'Action Output', there is a note: '(This action does not provide any output)'. At the bottom, there is a checkbox 'This action is Enabled' which is checked, and three buttons: 'More Info', 'OK', and 'Cancel'.

Properties:

Excel Instance: Enter the variable that contains the Excel Instance you want to work with. You must have previously specified this variable in a Launch Excel action.

Row Index: Enter the Index number of the row you wish to delete. The numbering starts from 1.

3.11.20.2 Insert Column to Excel Worksheet Action

Description:

This action inserts a column to the left of a selected column of a previously launched Excel Instance

Properties:

Excel Instance: Enter the variable that contains the Excel Instance you want to work with. You must have previously specified this variable in a Launch Excel action.

Column: Enter a column's index number or letter. A new column will appear on the left side of the column indicated.

3.11.20.2 Delete Column from Excel Worksheet Action

Description:

This action deletes a selected column from a previously launched Excel Instance

Properties:

Excel Instance: Enter the variable that contains the Excel Instance you want to work with. You must have previously specified this variable in a Launch Excel action.

Delete Column: Enter the Index number or letter of the column you wish to delete.

3.11.20.2 Get First Free Column/Row Action

Description:

This action retrieves the first free column and/or row of the active worksheet. This is useful for adding new data into a worksheet that already has data in it.

Properties of 'Get First Free Column/Row' action

Get First Free Column/Row
This action retrieves the first free column and/or row of the active worksheet. This is useful for adding new data into a worksheet that already has data in it.

General Exception Handling

Action Input

Excel Instance: %ExcelInstance%

Click here to expand the list of variable

Action Output

Store First Free Column into: %FirstFreeColumn%

Store First Free Row into: %FirstFreeRow%

This action is Enabled

More Info OK Cancel

Properties:**Excel Instance:**

Enter the variable that contains the Excel Instance you want to work with. You must have previously specified this variable in a Launch Excel action.

Store First Free Column into:

Enter a name to be the variable that will store the numeric value of the first fully empty column. This will be a number. For example, if column F is the first empty column, it will be stored as '6'.

Store First Free Row into:

Enter a name to be the variable that will store the numeric value of the first fully empty row.

3.11.20.2: Get First Free Row on Column from Excel Worksheet Action**Description:**

This action retrieves the first free row, given the column of the active worksheet.

Properties of 'Get First Free Row On Column from Excel Worksheet' action

 **Get First Free Row On Column from Excel Worksheet**
This action retrieves the first free row, given the column of the active worksheet.

General | Exception Handling

Action Input

Excel Instance: [%ExcelInstance%] ⓘ

Column: [B] ⓘ ⚙️

Action Output

Store Column's First Free Row into: [%FirstFreeRowOnColumn%] ⓘ

This action is Enabled

Properties:

Excel Instance:

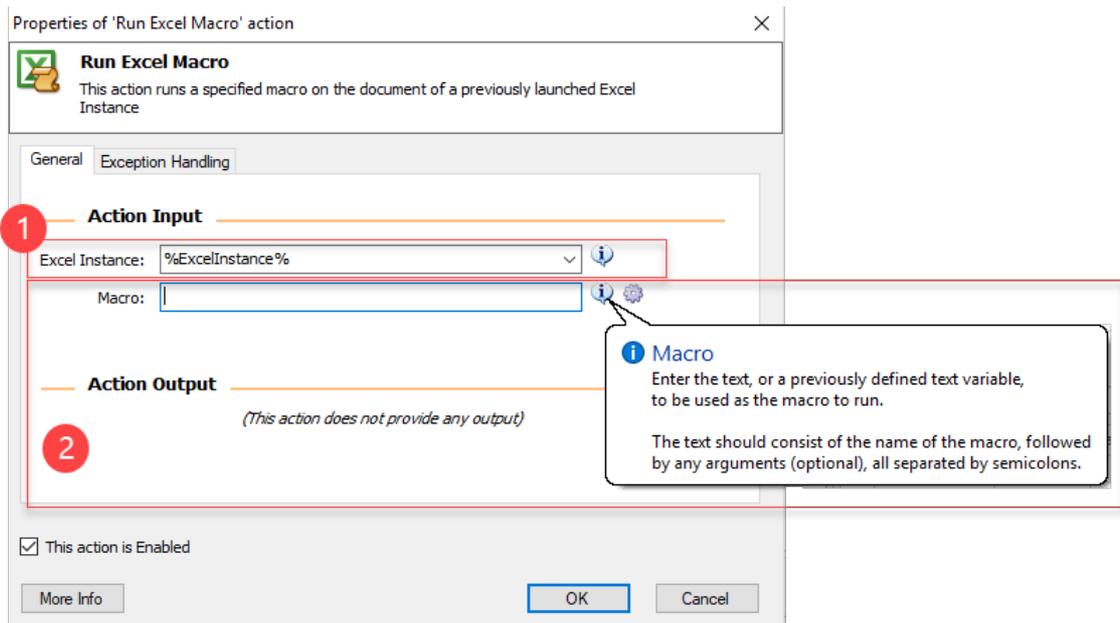
Enter the variable that contains the Excel Instance you want to work with. You must have previously specified this variable in a Launch Excel action.

Column: Enter the index or the letter that identifies the column. Column numbering starts from index 1.

Store Column's First Free Row into: Enter a name to be the variable that will store the numeric value of the given column's first fully empty row.

3.11.20.2 Run Excel Macro

This action runs a specified macro [2] on the document of a previously launched [1] Excel Instance:



1. Excel Instance:

Enter the variable that contains the Excel Instance you want to work with. You must have previously specified this variable in a Launch Excel action.

2. Macro:

As the help bubble says, enter the text or a previously defined text variable, to be used as the macro to run. The text should consist of the name of the macro, followed by any arguments (optional), all separated by semicolons.

3.11.21 PDF

3.11.21.1 Extract Text from PDF

Description: This actions extracts the text from a PDF file.

Properties of 'Extract Text From PDF' action

Extract Text From PDF
This action extract text from a PDF file

General | Advanced | Exception Handling

Action Input

PDF File: C:\Users\John\Desktop\7255582734.pdf

Page(s) To Extract: Range

From Page Number: 12

To Page Number: 20

Action Output

Store Extract Text into: %ExtractedPDFText%

This action is Enabled

More Info OK Cancel

General

The screenshot shows a dialog box titled "Properties of 'Extract Text From PDF' action". The dialog has three tabs: "General", "Advanced", and "Exception Handling". The "Advanced" tab is selected. Inside the "Advanced" tab, there is a checkbox labeled "Use Password" which is checked. Below this, there are two input fields: "Enter PDF Password:" with a dropdown menu set to "Directly", and "Password:" with an empty text box. Both input fields have an information icon to their right. At the bottom of the dialog, there is a checkbox labeled "This action is Enabled" which is also checked. There are three buttons at the bottom: "More Info", "OK", and "Cancel".

Advanced

Properties:

PDF File: Enter or choose the PDF file, whose text will be extracted. It can be a file path, a variable containing a file or a text path.

Page(s) To Extract: Choose how many pages will be extracted. All, Single or Range of pages

Single Page Number: Set the number of the single page that text will be extracted.

From Page Number: Set the first page number from the range of pages that text will be extracted

To Page Number: Set the last page number from the range of pages that text will be extracted

Store Extracted Text into: Enter a name to be the variable that will store the extracted text.

Use Password: Choose whether you want to use password for PDF that are password protected.

Enter PDF Password: If you choose 'directly', the password entered in the Password field will be hidden. If you choose 'as variable' you must enter a variable containing the password and the '%' character will be treated as an indicator of a variable, not part of the password.

Password: Enter the Password here. The password will be hidden.

Password: Enter a variable containing the password here.

3.11.21.2 Extract Text From PDF With OCR

Description: This action extracts text from a PDF file using OCR.

Properties of 'Extract Text From PDF With OCR' action

 **Extract Text From PDF With OCR**
This action extract text from a PDF file using OCR

General | Advanced | Exception Handling

Action Input

OCR Engine: %OCREngine% 

PDF File: C:\Users\aj\Downloads\ProcessRobot-RPA-DataSheet.pdf   

Page(s) To Extract: Range 

From Page Number: 12  

To Page Number: 21  

Action Output

Store Extract Text into: %ExtractedPDFTextWithOCR% 

This action is Enabled

General Tab

Properties of 'Extract Text From PDF With OCR' action

Extract Text From PDF With OCR
This action extract text from a PDF file using OCR

General | **Advanced** | Exception Handling

Use Password ⓘ

Enter PDF Password: Directly ⓘ

Password: ⓘ

This action is Enabled

More Info OK Cancel

Advanced Tab

Properties:**General Tab's Properties:**

OCR Engine: This text field with drop down menu options invites you to enter the OCR Engine instance you want to work with.

PDF File: Enter or choose the PDF file, whose text will be extracted. It can be a file path, a variable containing a file or a text path.

Page(s) To Extract: Choose how many pages will be extracted using OCR. All, Single or Range of pages

Single Page Number: Set the value of the page you want to extract text using OCR.

The following two options will be available if you have chosen Range on the "Page(s) To Extract" Property Value:

From Page Number: Set the first page number from the range of pages that text will be extracted using OCR.

To Page Number: Set the last page number from the range of pages that text will be extracted using OCR.

Store Extracted Text into: Enter a name to be the variable that will store the extracted text using OCR.

Advanced Tab's Properties:

Use Password: Choose whether you want to work with PDFs that are password protected.

Enter PDF Password: If you choose 'directly', the password entered in the Password field will be hidden. If you choose 'as variable' you must enter a variable containing the password and the '%' character will be treated as an indicator of a variable, not part of the password.

Password (Directly): Enter the Password here. The password will be hidden.

Password (as Variable): Enter a variable containing the password here.

3.11.21.3 Extract Images From PDF

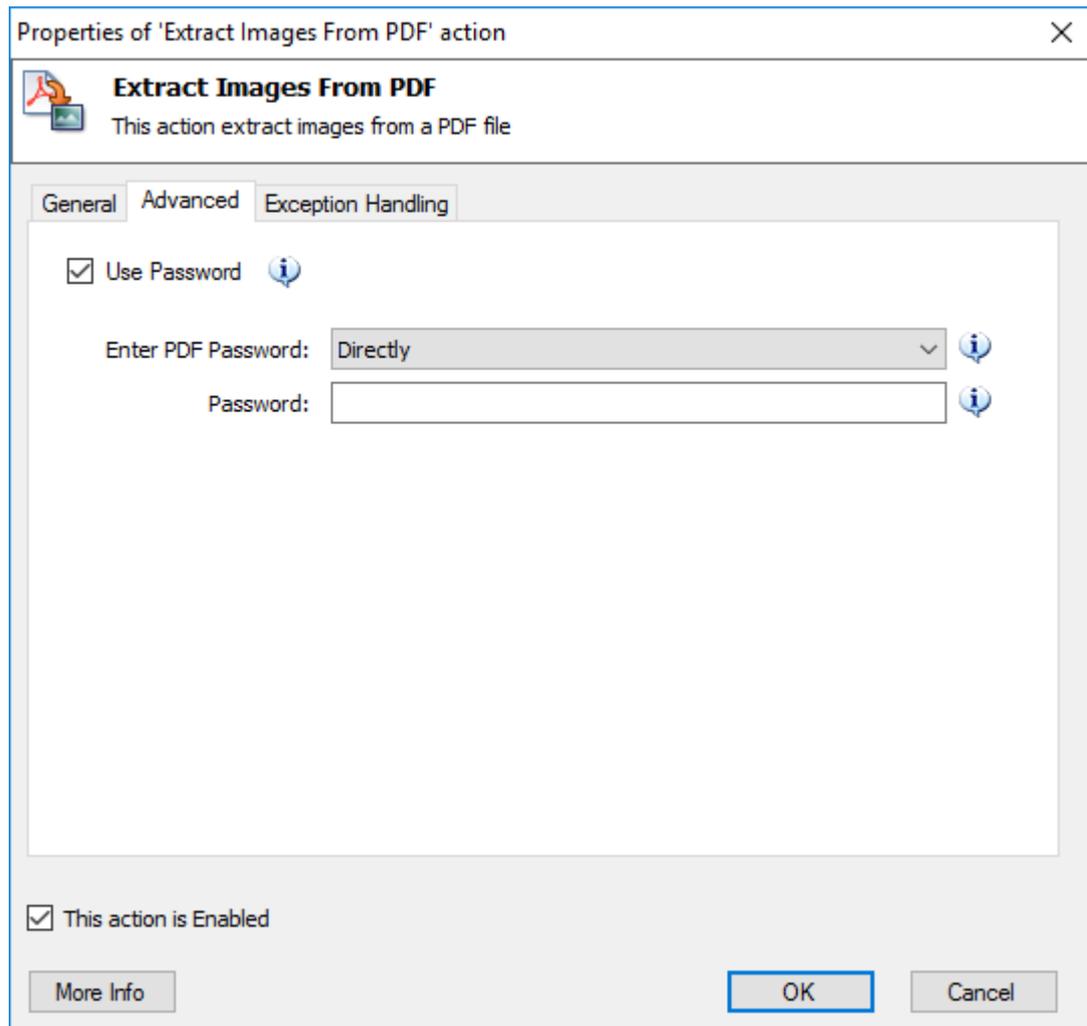
Description: This action extract images from a PDF file.

The screenshot shows a dialog box titled "Properties of 'Extract Images From PDF' action". The dialog has three tabs: "General", "Advanced", and "Exception Handling". The "General" tab is selected. The dialog contains the following fields and controls:

- Action Input:**
 - PDF File: C:\Users\John\Desktop\7255582734.pdf
 - Page(s) To Extract: Range
 - From Page Number: 12
 - To Page Number: 20
 - Image(s) Name: Image_1
 - Save Image(s) to: C:\Users\John\Desktop
- Action Output:**

(This action does not provide any output)
- This action is Enabled
- Buttons: More Info, OK, Cancel

General Tab



Advanced Tab

Properties:

PDF File: Enter or choose the PDF file, whose text will be extracted. It can be a file path, a variable containing a file or a text path.

Page(s) To Extract: Choose how many pages will be extracted using OCR.. All, Single or Range of pages

Single Page Number: Set the number of the single page that images will be extracted.

From Page Number: Set the first page number from the range of pages that images will be extracted.

To Page Number: Set the last page number from the range of pages that images will be extracted.

Image(s) Name: Enter how the image(s) will start. Extracted image(s) name example: GivenName_1, GivenName_2

Save Image(s) To: Enter or choose a local folder, or a previously defined variable containing a folder, to specify the folder where the extracted images from PDF will be saved into as png files.

Use Password: Choose whether you want to use password for PDF that are password protected.

Enter PDF Password: If you choose 'directly', the password entered in the Password field will be hidden. If you choose 'as variable' you must enter a variable containing the password and the '%' character will be treated as an indicator of a variable, not part of the password.

Password: Enter the Password here. The password will be hidden.

Password: Enter a variable containing the password here.

3.11.21.4 Extract PDF Pages to New PDF

Description: This action extracts pages from a PDF file to a new PDF file.

Properties:

PDF File: Enter or choose the PDF file, whose text will be extracted. It can be a file path, a variable containing a file or a text path.

Page Selection: Set the Index numbers of the pages you want to keep e.g. (1,3,17-24)

Extracted PDF Path: Enter a path where the extracted PDF will be stored

If File Exists: Choose what to do in case Output PDF file already exists

Store Extracted PDF File into: Select a variable where the new PDF File will be stored

3.11.21.5 Merge PDF Files

Description: This action merges a list of PDF files into a new one

Properties:

PDF Files: Enter the files to be merged. Multiple files should be enclosed in double quotes (") and should be separated by a delimiter of your choice. You can also use a variable that contains a list of files

Merged PDF Path: Enter the path to the file that this action will use to store the PDF file.

If File Exists: Choose what to do in case Output PDF file already exists

Store Merged PDF File into: Enter the Name of the Variable that will contain the PDF File

3.11.22 Terminal Emulation

3.11.22.1 Open Terminal Session

Description:

This action opens a new terminal session. This session will remain open until you close it with a "Close Terminal Session". In the meantime you can write commands, or read and manipulate the output from the Terminal Session.

Properties of 'Open Terminal Session' action

Open Terminal Session
This action opens a new terminal session

General | Exception Handling

Action Input

Provider: Micro Focus Reflection

Installation Path:

Configuration: Existing Profile

Profile:

Action Output

Store Session into: %TerminalSession%

This action is Enabled

More Info OK Cancel

Properties:

Provider: Select the terminal emulation provider you will be using.

Micro Focus Reflection

Installation Path: Select the installation path of the provider's application on your file system. By default, the path is *C:\Program Files (x86)\Micro Focus\Reflection*

Configuration: If you choose Existing Profile, you have to select a file that contains a preconfigured terminal emulation connection.

If you choose Specify Connection, you have to choose the Host Type of the connection and also the Host Address and Port.

Profile: Specify the file that contains the preconfigured connection. By default this file can be found in *C:\Users\“UserName”\Documents\Micro Focus\Reflection*.

Host Type: Specify the host type of the connection.

Host Address: Specify the host's address to connect to.

Port: Specify the port to be used for this connection.

HLLAPI

HLLAPI DLL Path: Select the HLLAPI DLL of your provider:

- RocketSoftware BlueZone default name: 'ehlapi64.dll'.
- IBM Personal Communications default name: 'EHLAPI32.dll'.
- MicroFocus Rumba default path: 'System/ehlapi32.Dll'.
- Cybelesoft zScope default name: 'zHllap32.dll'.

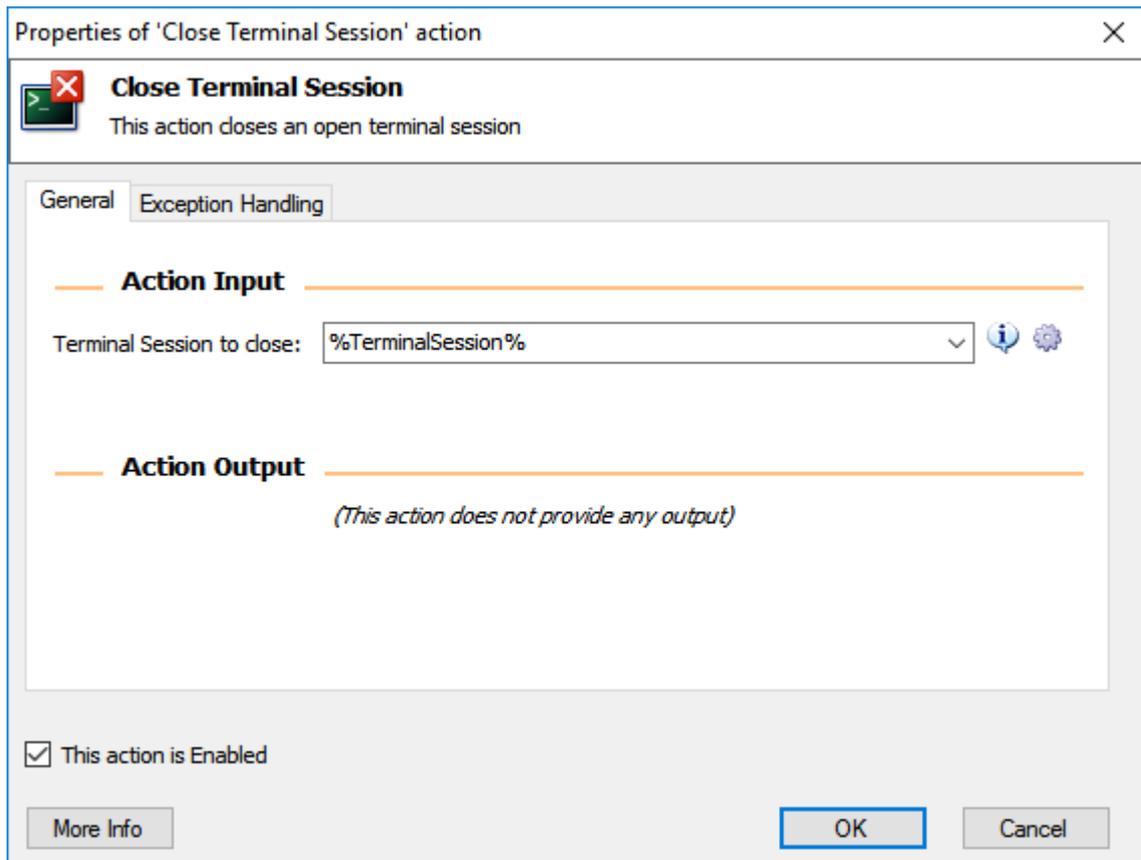
Session Name: Select the session name to connect to. Session names are one-letter (A-Z) and start from A. So type A if is the first time you open the emulator. Each provider has a setting where you can see or modify this value.

Store Terminal Session into: Enter a name to be the variable that will store the specific Terminal Session for use with later Terminal Emulation Commands.

3.11.22.2 Close Terminal Session

Description:

This action closes an open terminal session.



Properties:

Terminal Session to close: Enter the variable containing the previously opened terminal session.

3.11.22.3 Move Cursor on Terminal Session

Description:

This action moves the terminal's cursor on the specified position

Properties:

Terminal Session: Enter the variable containing the previously opened terminal session.

Row: Specify the vertical position of the cursor on the screen

Column: Specify the horizontal position of the cursor on the screen

3.11.22.4 Get Text from Terminal Session

Description:

This action gets text from a terminal session.

Properties of 'Get Text from Terminal Session' action

Get Text from Terminal Session
This action gets text from a terminal session

General Exception Handling

Action Input

Terminal Session: %TerminalSession%  

Get Text from: Field 

Get Field by: Label 

Label:  

Action Output

Store Text into: %TerminalText% 

This action is Enabled

More Info OK Cancel

Properties:

Terminal Session: Enter the variable containing the previously opened terminal session.

Get Text from: Specify the location to get the text from.

Text Length: Enter the length of the text you want to receive.

Get Field By: Specify the way by which to look for the field.

Field Label: Specify the label of the field to look for.

Field Index: Specify the index of the field on the screen.

Field Row: Specify the vertical position of the field on the screen.

Field Column: Specify the horizontal position of the field on the screen.

Store Text into: Enter a name to be the variable that will store the text.

3.11.22.5 Set Text on Terminal Session

Description:

This action sets text on a terminal session.

Properties of 'Set Text on Terminal Session' action

Set Text on Terminal Session
This action sets text on a terminal session

General | Exception Handling

Action Input

Terminal Session: ⓘ ⚙️

Set Text:

on Location: ⓘ

Get Field by: ⓘ

Label:

Action Output

(This action does not provide any output)

This action is Enabled

Properties:

Terminal Session: Enter the variable containing the previously opened terminal session.

Set Text: Enter the text you want to set on the specified location

Text Location: Specify the location to set the text on..

Get Field By: Specify the way by which to look for the field.

Field Label: Specify the label of the field to look for.

Field Index: Specify the index of the field on the screen.

Field Row: Specify the vertical position of the field on the screen.

Field Column: Specify the horizontal position of the field on the screen.

3.11.22.6 Send Key to Terminal Session

Description:

This action sends a control key to a terminal session.

Properties of 'Send Key to Terminal Session' action

Send Key to Terminal Session
This action sends a control key to a terminal session

General Exception Handling

Action Input

Terminal Session: %TerminalSession%

Control Key: Transmit

Action Output

(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties:

Terminal Session: Enter the variable containing the previously opened terminal session.

Control Key: Select the key to submit to the session.

3.11.22.7 Wait for Text on Terminal Session**Description:**

This action waits for a specific text to appear on a terminal session

The screenshot shows a dialog box titled "Properties of 'Wait for Text on Terminal Session' action". The dialog has a close button (X) in the top right corner. Below the title bar, there is a header section with a small icon of a terminal window and the text "Wait for Text on Terminal Session" and "This action waits for a specific text to appear on a terminal session".

The main area of the dialog is divided into three tabs: "General", "Advanced", and "Exception Handling". The "General" tab is selected. Under the "Action Input" section, there are three fields:

- "Terminal Session:" with a dropdown menu containing "%TerminalSession%" and an information icon.
- "Text to Wait for:" with a text input field, an information icon, and a gear icon.
- "Is Regular Expression" with an unchecked checkbox and an information icon.

 Below these is the "on Location:" field with a dropdown menu set to "Screen" and an information icon.

Under the "Action Output" section, there is a note: "(This action does not provide any output)".

At the bottom of the dialog, there is a checkbox labeled "This action is Enabled" which is checked. There are three buttons: "More Info", "OK", and "Cancel".

Properties: Depending on the location you will get different inputs and outputs.

Terminal Session: Enter the variable containing the previously opened terminal session.

Wait For Text Location: Specify the location to wait for the text to appear on.

Get Field By: Specify the way by which to look for the field.

Field Label: Specify the label of the field to look for.

Field Index: Specify the index of the field on the screen.

Field Row: Specify the vertical position of the field on the screen.

Field Column: Specify the horizontal position of the field on the screen.

Text to Wait for: Enter the text or the regular expression that you want to wait to appear on the terminal screen or on a specified field.

Regular Expression: Check this box if you want to wait for a regular expression instead of plain text.

Timeout: The Wait for Text action waits for specific text on the terminal session. Choose a maximum amount of time that the action will wait.

3.11.23 OCR

3.11.23.1 Create Tesseract OCR Engine

Process Robot allows you to work with a number of different OCR Engines through an equal number of different actions. Each of these actions has different Properties and property Values according to each respective Engine's capabilities.

This action allows you to work with the [Tesseract OCR Engine](#) in order to extract text from Image Files, in combination with the [Extract Text With OCR](#) ^[940] action.

Please Note: Tesseract is the only OCR Engine that it comes ready to use with Process Robot without the need to install it.

Tesseract Engine allows you to detect a number of different languages [1] and this Action is giving you the option to select any of them. It also allows you to rescale your image; you can resize width and height [2] of your image independently from one another through the use of multipliers, since Tesseract works best on images which have a DPI (Dots Per Inch) [of at least 300](#). The Action returns an Ocr Engine Data Type [3] stored within a variable:

Properties of 'Create Tesseract OCR Engine' action

Create Tesseract OCR Engine
This action creates a Tesseract OCR Engine

General Exception Handling

Action Input

Tesseract Language: English

Image Width Multiplier: 1

Image Height Multiplier: 1

Action Output

OCR Engine: %OCREngine%

This action is Enabled

More Info OK Cancel

1 Tesseract Language:

This drop down menu allows you to select the language of the image's text that Tesseract will detect:

Action Input

Tesseract Language: ⓘ

Image Width Multiplier: ⓘ ⚙️

Image Height Multiplier: ⓘ ⚙️

English
 German
 Spanish
 French
 Italian
 Other

Action Output

OCR Engine: ⓘ

2

Width & Height Multipliers:

These multipliers allow you to rescale an image in order to help the OCR Engine read the text in it. Tesseract is known to require a DPI of at least 300 in order to work, however since there is plenty of confusion among non-experts regarding DPI, PPI and optimal Image settings for OCR Text Extraction we invite you to feel comfortable to experiment/play with the available options (try for example 2, 3 or 4).

3

OCR Engine:

This text field invites you to set the variable that will hold the value of the [Ocr Engine Data Type](#)⁴³⁶ produced from this action.

3.11.23.2 Create MODI OCR Engine

Process Robot allows you to work with a number of different OCR Engines through an equal number of different actions. Each of these actions has different Properties and property Values according to each respective Engine's capabilities.

This action allows you to work with the [MODI \(Microsoft Office Document Imaging\) OCR Engine](#) in order to extract text from Image Files, in combination with the [Extract Text With OCR](#) action.

Please Note: MODI requires additional installations. [Tesseract](#) is the only OCR Engine that it comes ready to use with Process Robot without the need to install it.

MODI allows you to detect a number of different languages [1] and this Action is giving you the option to select any of them. It also allows you to rescale your image; you can resize width and height [2] of your image independently from one another through the use of multipliers, since that might help MODI read the text we want to extract. The Action returns an Ocr Engine Data Type [3] stored within a variable:

Properties of 'Create MODI OCR Engine' action

Create MODI OCR Engine
This action creates a MODI OCR Engine

General Exception Handling

Action Input

MODI Language: English

Image Width Multiplier: 1

Image Height Multiplier: 1

Action Output

OCR Engine: %OCREngine2%

This action is Enabled

More Info OK Cancel

1

MODI Language:

This drop down menu allows you to select the language of the image's text that MODI will detect:

The screenshot shows a configuration window for MODI. It is divided into two main sections: **Action Input** and **Action Output**.
 In the **Action Input** section:
 - **MODI Language:** A dropdown menu is open, showing a list of languages: Chinese Simplified, Chinese Traditional, Czech, Danish, Dutch, English (highlighted), Finnish, French, German, Greek, Hungarian, Italian, Japanese, Korean, Norwegian, Polish, Portuguese, Russian, Spanish, Swedish, and Turkish. An information icon (i) is to the right.
 - **Image Width Multiplier:** An empty text input field with an information icon (i) and a gear icon (settings).
 - **Image Height Multiplier:** An empty text input field with an information icon (i) and a gear icon (settings).
 In the **Action Output** section:
 - **OCR Engine:** An empty text input field with an information icon (i).
 At the bottom left, there is a checkbox labeled "This action is Enabled" which is checked, and a "More Info" button.
 At the bottom right, there are "OK" and "Cancel" buttons.

2

Width & Height Multipliers:

These multipliers allow you to rescale an image in order to help the OCR Engine read the text in it. Resizing an Image is known to help OCR read text in them and since there is plenty of confusion among non-experts regarding DPI, PPI and optimal Image settings for OCR Text Extraction we invite you to feel comfortable to experiment/play with the available options (try for example 2, 3 or 4).

3

OCR Engine:

This text field invites you to set the variable that will hold the value of the [Ocr Engine Data Type](#)⁴³⁶ produced from this action.

3.11.23.3 Get Data with ABBYY FlexiCapture

Description:

This action analyzes a list of images using the [ABBY FlexiCapture](#).

Properties of 'Get Data With ABBYY FlexiCapture' action ✕

 **Get Data With ABBYY FlexiCapture**
This action analyzes a list of images with ABBYY FlexiCapture

General Exception Handling

Action Input

Document Definition Files:   

Images:   

Export Type: XML Variable 

Action Output

Store XML Variable into: 

Store Erroneous Images Files into: 

This action is Enabled

More Info OK Cancel

Properties of 'Get Data With ABBYY FlexiCapture' action ✕

 **Get Data With ABBYY FlexiCapture**
This action analyzes a list of images with ABBYY FlexiCapture

General Exception Handling

Action Input

Document Definition Files: i ⚙️ 📄

Images: i ⚙️ 📄

Export Type: Excel Document i

XML Variable

Excel Document

DBF Database

CSV File

XML File

Export Path: i ⚙️ 📁

Action Output

Store Exported Results Files into: i

Store Erroneous Images Files into: i

This action is Enabled

More Info
OK
Cancel

Export Type: File

Properties:

Document Definition Files: Select the ABBYY Flexi Capture Document Definition files that will be used to analyze the images against to.

Images: Select the image's filenames that will be analyzed.

Export Type: Select how the results will be exported.

Export Path: Select the folder that the results will be exported to.

Store XML Variable into/Store Exported Results Files into: Select a name for the XML variable to export the results to. / Select a name for the variable to assign a list of the exported result files.

Store Erroneous Images Files into: Select a name for the variable to assign a list of the images that failed the analysis.

3.11.23.4 Extract Text With OCR

Description: This action extracts text from the image-source specified: A region of the screen, a window, an image file on the disk and so on.

Properties of 'Extract Text With OCR' action

Extract Text With OCR
This action extracts text from the image-source specified: A region of the screen, a window or an image file on the disk

General | Advanced | Exception Handling

Action Input

OCR Engine: %OCREngine%

OCR Source: Screen

Search Mode: Whole of specified source

Action Output

Scanned Text: %OCRText%

This action is Enabled

More Info | OK | Cancel

Search Mode: Whole of specified source

Properties of 'Extract Text With OCR' action

Extract Text With OCR
This action extracts text from the image-source specified: A region of the screen, a window or an image file on the disk

General | **Advanced** | Exception Handling

Action Input

OCR Engine: %OCREngine% ⓘ

OCR Source: Screen ⓘ

Search Mode: Specific subregion only ⓘ

X1: ⓘ X2: ⓘ

Y1: ⓘ Y2: ⓘ

Select Subregion ⓘ

Action Output

Scanned Text: %OCRText% ⓘ

This action is Enabled

More Info OK Cancel

Search Mode: Specific Subregion only

Properties of 'Extract Text With OCR' action

Extract Text With OCR
This action extracts text from the image-source specified: A region of the screen, a window or an image file on the disk

General | **Advanced** | Exception Handling

Action Input

OCR Engine: [%OCREngine%] ⓘ

OCR Source: [Image on disk] ⓘ

Path to image file: [C:\Users\j\Desktop\New folder (3)\Test.prp] ⓘ ⚙️ 📄

Search Mode: [Subregion relative to image] ⓘ

Image: [Empty list box] ⓘ

Add Image From Images Repository ⓘ

Delete Image From List

Tolerance: [10] ⓘ

Subregion relative to image: X1: [] ⚙️ X2: [] ⚙️ ⓘ

Y1: [] ⚙️ Y2: [] ⚙️ ⓘ

Action Output

Scanned Text: [%OCRText%] ⓘ

This action is Enabled

More Info | OK | Cancel

Search Mode: Subregion relative to image

Properties:

OCR Engine: Select the Engine you want to use for the OCR operation.

OCR Source: Choose whether you want to search for the specified Image in the foremost Window only, or the entire visible screen. Neither choice will find the Image if it is not clearly visible on the screen.

Search Mode: Specify whether you want to scan the entire screen (or window) to find the supplied image or only a narrowed down subregion of it.

X1 Y1 X2 Y2: Specify the subregion of the screen/window/image to narrow down the scan to.

Image: Specify the image to use as a Image for narrowing the scan down to a subregion which is positionally relative to said Image.

Tolerance: Specify a value for how much the Image searched for can differ from the originally chosen Image.

Sub-region relative to Image: Specify the subregion (relative to the top left corner of the Image image you've specified) which you would like to have it scanned for its text.

Scanned Text: Enter a name to be the variable that will store the text which got extracted using OCR.

OCR Language: Select the language of the image that the OCR will be performed

Wait for image to appear: Choose whether you want the action to wait if the image is not found on the screen or foreground window. If this property is not checked and the image is not found the action will through an exception, otherwise the action will wait until the image appears. In the next property you can specify the maximum number of seconds (timeout) to wait.

Fail if image does not appear within x seconds: Specify whether the action should wait indefinitely for the image to appear or throw an exception after a set number of seconds (if the specified Image does not get found).

3.11.24 CaptureFast

3.11.24.1 Connect to CaptureFast Account

Description:

This action connects to the specified CaptureFast Account.

Properties:

Username:

Specify the username of your CaptureFast account.

Password Mode:

This dropdown menu allows you to select the way you will provide your password; either directly or as a variable.

Password:

Specify your CaptureFast password.

Store Connection into:

This text field invites you to set the variable that will hold your connection with CaptureFast.

3.11.24.2 Upload Document to CaptureFast

Description:

This action uploads a document to the account of an established CaptureFast Connection.

Properties of 'Upload Document to CaptureFast' action

Upload Document to CaptureFast
This action uploads a document to the account of an established CaptureFast Connection.

General | Advanced | Exception Handling

Action Input

CaptureFast Connection:

Team Id:

Template Id:

File Path:

Action Output

Store Response into:

Store Status Code:

Document Id:

This action is Enabled

More Info | OK | Cancel

Properties:

CaptureFast Connection:

Enter the variable that holds your CaptureFast connection.

Team Id:

Specify your Team Id if any.

Template Id:

Specify the Template Id to use for the document to be uploaded.

File Path:

Enter or choose a file path, or a previously defined variable, to specify the full path of the document that you want to upload.

Store Response into:

Set the name of the variable that will store the API response result in.

Store Status Code:

Set the name of the variable that will store the status code returned in.

Document Id:

Set the name of the variable that will store the Id of the Document to be uploaded.

3.11.24.3 Get Results From CaptureFast Document

Description:

This action gets all the results from an uploaded document in CaptureFast.

Properties of 'Get Results from CaptureFast Document' action

Get Results from CaptureFast Document
This action gets all the results from an uploaded document in CaptureFast.

General | Advanced | Exception Handling

Action Input

CaptureFast Connection:  

Document Id:  

Action Output

Store Response into: 

This action is Enabled

Properties:

CaptureFast Connection:

Enter the variable that holds your CaptureFast connection.

Document Id:

Enter the Id of the document that you want to get results from.

Store Response into:

Set the name of the variable that will store the API response result.

3.11.25 CyberArk

3.11.25.1 Get Password From CyberArk

Description:

This action retrieves a password for a specific application from CyberArk.

Properties of 'Get Password From CyberArk' action ✕

 **Get Password From CyberArk**
This action retrieves a password for a specific application from CyberArk

General Advanced Exception Handling

Action Input

Server Address: ⓘ ⚙️

Application Id: ⓘ ⚙️

Safe: ⓘ ⚙️

Folder: ⓘ ⚙️

Object: ⓘ ⚙️

Extra data: ⓘ ⚙️

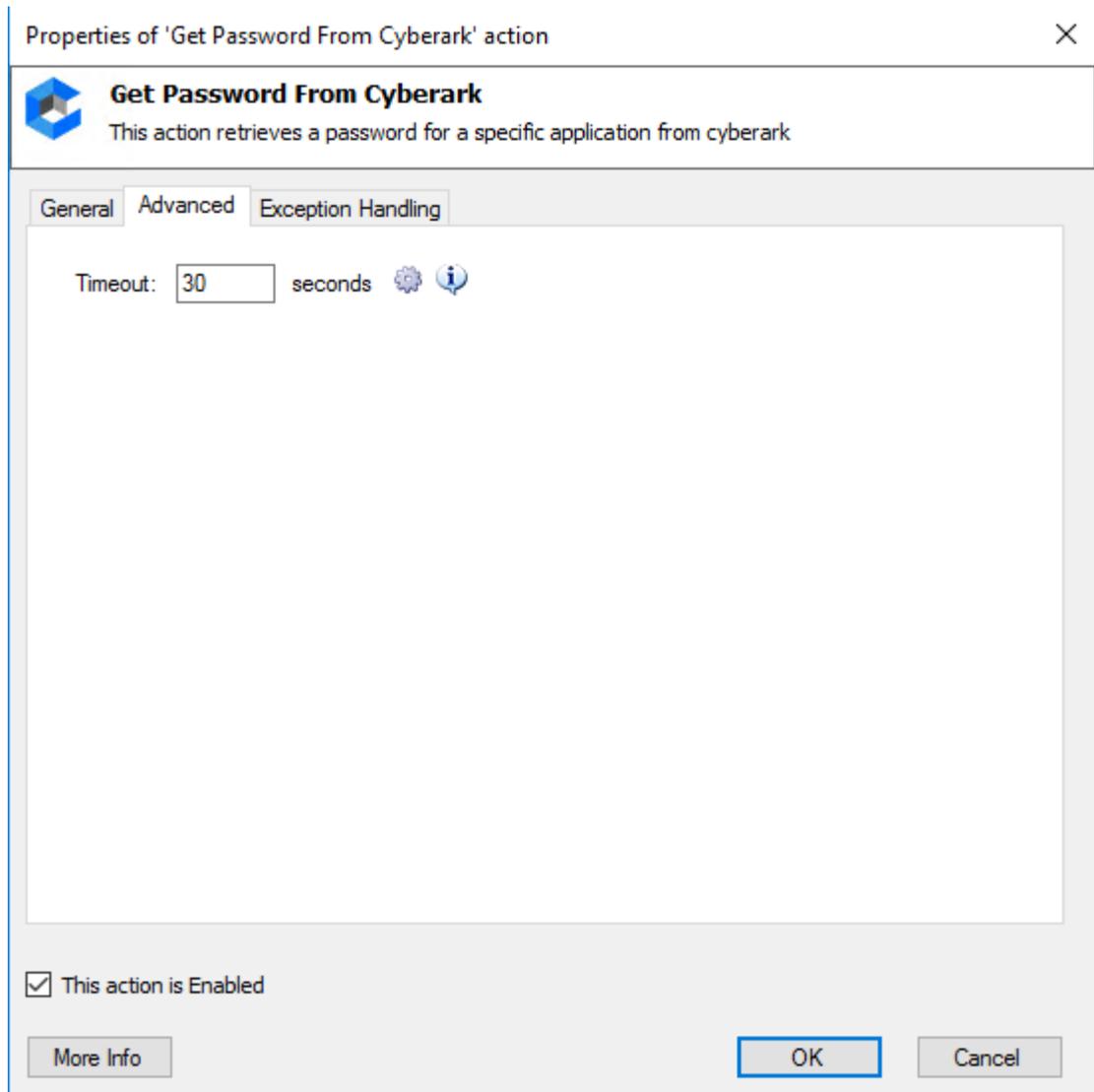
Certificate Location: ⓘ

Action Output

Cyberark Password: ⓘ

Store Response into: ⓘ

This action is Enabled



Properties:

Server Address:

Specify the base URI for the web request.

Application Id:

Specify the Application Id to use for the web request.

Safe:

Specify the Safe on Cyberark in which the Application belongs.

Folder:

Specify the folder necessary for the web request query.

Object:

Specify the object necessary for the web request query.

Extra data:

Specify the extra data if any for the web request's query.

Certificate Location:

If you need a certificate for the request select the desired way to load it.

When choosing 'Load Certificate from Windows Store', the 'Certificate Path' field appears, along with the 'Use Only Valid Certificate' option, and the 'Find Certificate' button.

When choosing 'Load Certificate from File', the 'Certificate file path' and 'File Password' fields appear.

Use Only Valid Certificate:

Check this if you want only valid certificates to be loaded from the store.

Find Certificate button:

Opens the Certificate Selector window, where a list of the certificates available in the store is displayed.

Certificate Path:

This shows the Path of the certificate selected from the store.

Certificate file path:

Enter the path to your certificate.

File Password:

Enter the password for your certificate file.

Cyberark Password:

Enter a name to be the variable that holds the password retrieved from CyberArk.

Store Response into:

Enter a name to be the variable that you want to store the API response result in.

Timeout:

Specify the time (in seconds) that the action should wait for getting results from Cyberark, before it fails.

3.11.26 Enate

3.11.26.1 Login to Enate

Description:

This Action logs into Enate and returns an Enate Instance.

Properties:

Enate URL: Specify the URL to use for the subsequent calls.

Username: Enter the username for the Enate account.

Enter Password: If you choose 'directly', the password entered in the Password field will be hidden. If you choose 'As Variable', you must enter a variable containing the password and the '%' character will be treated as an indicator of a variable and not part of the password.

Password: Enter the password for the Enate account. The password will be hidden.

Enate Instance: Enter a name for the variable which will store the Enate Instance.

Status Code: Enter a name for the variable which will the returned status code.

Cautions:

For any Enate-related Action to work correctly, the User must have access to an Enate account.

3.11.26.2 Is Work Available on Enate**Description:**

This Action checks if there is work available on Enate.

Properties:

Enate Instance: Enter the variable that contains the Enate Instance you want to work with. You must have previously specified this variable in a Login to Enate Action.

Is Work Available: Enter a name for the variable which will store the Boolean value that refers to whether or not there is available work.

Status Code: Enter a name for the variable which will store the returned status code.

3.11.26.3 Get More Work from Enate

Description:

This Action retrieves a Work Item from Enate.

Properties of 'Get More Work from Enate' action

Get More Work from Enate
This Action retrieves a Work item from Enate

General | Advanced | Exception Handling

Action Input

Enate Instance: [%EnateInstance%]

Action Output

Enate Work: [%EnateWork%]
Status Code: [%StatusCode%]

This action is Enabled

More Info | OK | Cancel

Properties:

Enate Instance: Enter the variable that contains the Enate Instance you want to work with. You must have previously specified this variable in a Login to Enate Action.

Enate Work: Enter a name for the variable which will store the retrieved work.

Status Code: Enter a name for the variable which will store the returned status code.

3.11.26.4 Create Case on Enate

Description:

This Action creates a Case on Enate.

Properties of 'Create Case on Enate' action X

Create Case on Enate
This Action creates a Case on Enate

General **Advanced** Exception Handling

Action Input

Enate Instance:

Customer:

Contract:

Service:

Process:

Case Title:

Email Address:

Return Created Case:

Action Output

Created Case:

Status Code:

This action is Enabled

Properties:

Enate Instance: Enter the variable that contains the Enate Instance you want to work with. You must have previously specified this variable in a Login to Enate Action.

Customer: Specify the context customer for this Case.

Contract: Specify the context contract for this Case.

Service: Specify the context service for this Case.

Process: Specify the context process for this Case.

Case Title: Specify a title for the Case to create.

Email Address: Specify an email address to look for and to set as a contact.

Return Created Case: Check to retrieve the created Case.

Created Case: Enter a name for the variable which will store the created Case.

Status Code: Enter a name for the variable which will store the returned status code.

3.11.26.5 Get Case from Enate

Description:

This Action retrieves a Case from Enate.

Properties of 'Get Case from Enate' action

Get Case from Enate
This Action retrieves a Case from Enate

General | Advanced | Exception Handling

Action Input

Enate Instance: [%EnateInstance%] ⓘ

Case GUID: [] ⓘ ⚙

Existing Packet Activity GUID: [] ⓘ ⚙

Action Output

Case: [%EnateCase%] ⓘ

Status Code: [%StatusCode%] ⓘ

This action is Enabled

More Info | OK | Cancel

Properties:

Enate Instance: Enter the variable that contains the Enate Instance you want to work with. You must have previously specified this variable in a Login to Enate Action.

Case GUID: Specify the unique identifier of the Case to retrieve.

Existing Packet Activity GUID: (Optional) Specify the unique identifier of an existing Packet Activity record.

Case: Enter a name for the variable which will store the retrieved Case.

Status Code: Enter a name for the variable which will store the returned status code.

3.11.26.6 Update Case on Enate

Description:

This Action updates a Case on Enate.

Properties of 'Update Case on Enate' action

Update Case on Enate
This Action updates a Case on Enate

General | Advanced | Exception Handling

Action Input

Enate Instance: %EnateInstance%

Enate Case:

Return Updated Case:

Action Output

Updated Case: %EnateUpdatedCase%

Status Code: %StatusCode%

This action is Enabled

More Info OK Cancel

Properties:

Enate Instance: Enter the variable that contains the Enate Instance you want to work with. You must have previously specified this variable in a Login to Enate action.

Enate Case: Specify the Case to update.

Return Updated Case: Check to retrieve the updated Case.

Updated Case: Enter a name for the variable which will store the updated Case.

Status Code: Enter a name for the variable which will store the returned status code.

3.11.26.7 Create Ticket on Enate

Description:

This Action creates a Ticket on Enate.

✕
Properties of 'Create Ticket on Enate' action

▶
Create Ticket on Enate
This Action creates a Ticket on Enate

General
Advanced
Exception Handling

Action Input

Enate Instance: i

Customer: i ⚙

Contract: i ⚙

Service: i ⚙

Process: i ⚙

Ticket Title: i ⚙

Email Address: i ⚙

Category Level 1: i ⚙

Category Level 2: i ⚙

Category Level 3: i ⚙

Return Created Ticket: i

Action Output

Created Ticket: i

Status Code: i

This action is Enabled

More Info
OK
Cancel

Properties:

Enate Instance: Enter the variable that contains the Enate Instance you want to work with. You must have previously specified this variable in a Login to Enate Action.

Customer: Specify the context customer for this ticket.

Contract: Specify the context contract for this ticket.

Service: Specify the context service for this ticket.

Process: Specify the context process for this ticket.

Ticket Title: Specify a title for the ticket you want to create.

Email Address: Specify an email address to look for and to set it as a contact.

Category Level 1: Specify the name of the first level ticket attribute.

Category Level 2: Specify the name of the second level ticket attribute.

Category Level 3: Specify the name of the third level ticket attribute.

Return Created Ticket: Check to retrieve the created ticket.

Created Ticket: Enter a name for the variable which will store the created ticket.

Status Code: Enter a name for the variable which will store the returned status code.

3.11.26.8 Get Ticket from Enate

Description:

This Action retrieves a Ticket from Enate.

Properties of 'Get Ticket from Enate' action

Get Ticket from Enate
This Action retrieves a Ticket from Enate

General | Advanced | Exception Handling

Action Input

Enate Instance: [%EnateInstance%] ⓘ

Ticket GUID: [] ⓘ ⚙️

Existing Packet Activity GUID: [] ⓘ ⚙️

Action Output

Ticket: [%EnateTicket%] ⓘ

Status Code: [%StatusCode%] ⓘ

This action is Enabled

More Info OK Cancel

Properties:

Enate Instance: Enter the variable that contains the Enate Instance you want to work with. You must have previously specified this variable in a Login to Enate Action.

Ticket GUID: Specify the unique identifier of the Ticket you wish to retrieve.

Existing Packet Activity GUID: (Optional) Specify the unique identifier of an existing Packet Activity record.

Ticket: Enter a name for the variable which will store the retrieved ticket.

Status Code: Enter a name for the variable which will store the returned status code.

3.11.26.9 Update Ticket on Enate

Description:

This Action updates a Ticket on Enate.

Properties:

Enate Instance: Enter the variable that contains the Enate Instance you want to work with. You must have previously specified this variable in a Login to Enate Action.

Enate Ticket: Specify the ticket to update.

Return Updated Ticket: Check to retrieve the updated ticket.

Updated Ticket: Enter a name for the variable which will store the updated ticket.

Status Code: Enter a name for the variable which will store the returned status code.

3.11.26.1 Get Action from Enate

Description:

This Action retrieves an Action from Enate.

Properties:

Enate Instance: Enter the variable that contains the Enate Instance you want to work with. You must have previously specified this variable in a Login to Enate Action.

Action GUID: Specify the unique identifier of the Action to retrieve.

Existing Packet Activity GUID: (Optional) Specify the unique identifier of an existing Packet Activity record.

Action: Enter a name for the variable which will store the retrieved Action.

Status Code: Enter a name for the variable which will store the returned status code.

3.11.26.1 Update Action on Enate

Description:

This Action updates an Action on Enate.

Properties:

Enate Instance: Enter the variable that contains the Enate Instance you want to work with. You must have previously specified this variable in a Login to Enate Action.

Enate Action: Specify the Action to update.

Return Updated Action: Check to retrieve the updated Action.

Updated Action: Enter a name for the variable which will store the updated Action.

Status Code: Enter a name for the variable which will store the returned status code.

3.11.26.1: Update Checklist on Enate

Description:

This Action updates a checkbox in a Checklist on Enate.

Properties:

Enate Instance: Enter the variable that contains the Enate Instance you want to work with. You must have previously specified this variable in a Login to Enate Action.

Enate Check: Specify the Enate checkbox to update.

Status Code: Enter a name for the variable which will store the returned status code.

3.11.26.1: Reject Work from Enate

Description:

This Action rejects Work from Enate.

Properties:

Enate Instance: Enter the variable that contains the Enate Instance you want to work with. You must have previously specified this variable in a Login to Enate Action.

Enate Packet GUID: Enter the unique identifier of the Enate Packet to reject.

Reject Reason: Specify the reason for rejecting the work.

Status Code: Enter a name for the variable which will store the returned status code.

3.11.26.1 Add File to Enate Item

Description:

This Action adds a File to an Enate Item.

Properties of 'Add File to Enate Item' action

Add File to Enate Item
This Action adds a File to an Enate Item

General Exception Handling

Action Input

Enate Item: ⓘ ⚙

File: ⓘ ⚙ 📄

File Note: ⓘ ⚙

Action Output

(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties:

Enate Item: Specify the item to which to add the file.

File: Specify the file or the path containing the file, to add to the Enate item.

File Note: Specify a note for the file.

3.11.26.1 Save File from Enate**Description:**

This Action saves a file from an Enate Item.

Properties of 'Save File from Enate' action

Save File from Enate
This Action saves a file from an Enate item

General | Advanced | Exception Handling

Action Input

Enate Instance: [%EnateInstance%] ⓘ

Packet GUID: [] ⓘ ⚙

File GUID: [] ⓘ ⚙

Path to Save: [] ⓘ ⚙ 📄

Action Output

File: [%File%] ⓘ

Status Code: [%StatusCode%] ⓘ

This action is Enabled

More Info | OK | Cancel

Properties:

Enate Instance: Enter the variable that contains the Enate Instance you want to work with. You must have previously specified this variable in a Login to Enate Action.

Packet GUID: Specify the packet unique identifier which holds the desired file.

File GUID: Specify the unique identifier of the desired file.

Path to Save: Enter the path to save the desired file locally.

File: Enter a name for the variable which will store the file.

Status Code: Enter a name for the variable which will store the returned status code.

3.11.26.1 Logout from Enate

Description:

This Action logs out from Enate.

Properties of 'Logout from Enate' action

Logout from Enate
This Action Logs out from Enate

General | Advanced | Exception Handling

Action Input

Enate Instance: %EnateInstance%

Action Output

Status Code: %StatusCode%

This action is Enabled

More Info OK Cancel

Properties:

Enate Instance: Enter the variable that contains the Enate Instance you want to work with. You must have previously specified this variable in a Login to Enate Action.

Status Code: Enter a name for the variable which will store the returned status code.

3.11.27 Cryptography

3.11.27.1 Encrypt Text

Description:

This action encrypts a string, using a key, a specified algorithm and encoding format.

Properties of 'Encrypt Text' action ✕

Encrypt Text
Encrypts a string, using a key, a specified algorithm and encoding format

General | Advanced | Exception Handling

Action Input

Encoding: ⓘ

Encryption Algorithm: ⓘ

Text to Encrypt: ⓘ ⚙️

Enter Encryption Key: ⓘ

Encryption Key: ⓘ

Action Output

Encrypted Text: ⓘ

This action is Enabled

Properties of 'Encrypt Text' action

Encrypt Text
Encrypts a string, using a key, a specified algorithm and encoding format

General Advanced Exception Handling

Cipher Mode: CBC

Padding: PKCS7

Block Size: 128

Key Size: 256

Use Salt: None

Use IV: None

This action is Enabled

More Info OK Cancel

Properties:

Encoding:

Select the encoding of the text to encrypt.

Encryption Algorithm:

Select the algorithm to be used for the encryption.

Text to Encrypt:

Enter the text to encrypt.

Enter Encryption Key:

When choosing 'directly', the value entered in the Encryption Key field will be hidden.

When choosing 'as variable', a variable that stores the encryption key must be entered in this field. The '%' character will be treated as an indicator of the variable, not part of the key.

Encryption Key:

Enter the Encryption Key here. The encryption key value will be hidden.

Encrypted Text:

Enter a name for the output variable that will store the encrypted text for later processing.

Cipher Mode:

Select the cipher mode to use for the selected encryption algorithm.

Padding:

Select the padding to use for the selected encryption algorithm.

Block Size:

Select the block size in bits to use for the encryption.

Key Size:

Select the size of the key in bits to use for the encryption.

Use Salt:

Select whether to use Salt for encryption.

When choosing to use a randomly generated Salt, its value will be stored in an output variable in the form of a base64 string.

Use IV (Initialization Vector):

Select whether to use an Initialization Vector.

When choosing to use a randomly generated Initialization Vector, its value will be stored in an output variable in the form of a base64 string.

3.11.27.2 Decrypt Text***Description:***

This action decrypts a string based on a specified key, encoding and algorithm.

Properties of 'Decrypt Text' action ✕

Decrypt Text
Decrypts a string based on a specified key, encoding and algorithm

General | Advanced | Exception Handling

Action Input

Encoding: ⓘ

Encryption Algorithm: ⓘ

Text to Decrypt: ⓘ ⚙️

Enter Encryption Key: ⓘ

Encryption Key: ⓘ

Action Output

Decrypted Text: ⓘ

This action is Enabled

Properties of 'Decrypt Text' action

Decrypt Text
Decrypts a string based on a specified key, encoding and algorithm

General | Advanced | Exception Handling

Cipher Mode: CBC

Padding: PKCS7

Block Size: 128

Key Size: 256

Use Salt: None

Use IV: None

This action is Enabled

More Info OK Cancel

Properties:

Encoding:

Select the encoding of the text to decrypt.

Encryption Algorithm:

Select the algorithm to be used for the decryption.

Text to Decrypt:

Enter the text to decrypt. The text must be in the form of a base64 string.

Enter Decryption Key:

When choosing 'directly', the value entered in the Decryption Key field will be hidden.

When choosing 'as variable', a variable that stores the decryption key must be entered in this field. The '%' character will be treated as an indicator of the variable, not part of the key.

Decryption Key:

Enter the Decryption Key here. The decryption key value will be hidden.

Decrypted Text:

Enter a name for the output variable that will store the decrypted text for later processing.

Cipher Mode:

Select the cipher mode to use for the selected decryption algorithm.

Padding:

Select the padding to use for the selected decryption algorithm.

Block Size:

Select the block size in bits to use for the decryption.

Key Size:

Select the size of the key in bits to use for the decryption.

Use Salt:

Select whether to use Salt for the decryption.

Salt:

Enter the Salt to use for the decryption. The Salt value must be in the form of a base64 string.

Use IV (Initialization Vector):

Select whether to use an Initialization Vector.

IV (Initialization Vector):

Enter the Initialization Vector to use for the decryption. The Initialization Vector must be in the form of a base64 string.

3.11.27.3 Encrypt From File

Description:

Encrypts the contents of a file, using a key, specified algorithm and encoding.

Properties of 'Encrypt From File' action ✕

 **Encrypt From File**
Encrypts the contents of a file, using a key a specified algorithm and encoding format

General Advanced Exception Handling

Action Input

Encoding: 

Encryption Algorithm: 

File to Encrypt:   

Enter Encryption Key: 

Encryption Key: 

Action Output

Encrypted Text: 

This action is Enabled

Properties of 'Encrypt From File' action

Encrypt From File
Encrypts the contents of a file, using a key a specified algorithm and encoding format

General Advanced Exception Handling

Cipher Mode: CBC ⓘ
 Padding: PKCS7 ⓘ
 Block Size: 128 ⓘ
 Key Size: 256 ⓘ
 Use Salt: None ⓘ
 Use IV: None ⓘ

This action is Enabled

More Info OK Cancel

Properties:

Encoding:

Select the encoding of the text to encrypt.

Encryption Algorithm:

Select the algorithm to be used for the encryption.

File to Encrypt:

Select the text file that stores the contents to encrypt.

Enter Encryption Key:

When choosing 'directly', the value entered in the Encryption Key field will be hidden.

When choosing 'as variable', a variable that stores the encryption key must be entered in this field. The '%' character will be treated as an indicator of the variable, not part of the key.

Encryption Key:

Enter the Encryption Key here. The encryption key value will be hidden.

Encrypted Text:

Enter a name for the output variable that will store the text of the encrypted file for later processing.

Cipher Mode:

Select the cipher mode to use for the selected encryption algorithm.

Padding:

Select the padding to use for the selected encryption algorithm.

Block Size:

Select the block size in bits to use for the encryption.

Key Size:

Select the size of the key in bits to use for the encryption.

Use Salt:

Select whether to use Salt for the encryption.

When choosing to use a randomly generated Salt, its output will be stored in an output variable in the form of a base64 string.

Salt:

Enter the Salt to use for the decryption. The Salt value must be in the form of a base64 string.

Use IV (Initialization Vector):

Select whether to use an Initialization Vector.

When choosing to use a randomly generated Initialization Vector, its value will be stored in an output variable in the form of a base64 string.

3.11.27.4 Decrypt To File**Description:**

This action decrypts a string to a file based on a specified key, encoding and algorithm.

Properties of 'Decrypt To File' action ✕

 **Decrypt To File**
Decrypts a string to a file based on a specified key, encoding and algorithm

General **Advanced** Exception Handling

Action Input

Encoding: ⓘ

Decryption Algorithm: ⓘ

Text to Decrypt: ⓘ ⚙️

Enter Decryption Key: ⓘ

Decryption Key: ⓘ

Decrypt to File: ⓘ ⚙️ 📄

If File Exists: ⓘ

Action Output

Decrypted File: ⓘ

This action is Enabled

Properties of 'Decrypt To File' action

Decrypt To File
Decrypts a string to a file based on a specified key, encoding and algorithm

General Advanced Exception Handling

Cipher Mode: CBC

Padding: PKCS7

Block Size: 128

Key Size: 256

Use Salt: None

Use IV: None

This action is Enabled

More Info OK Cancel

Properties:

Encoding:

Select the encoding of the text to decrypt.

Decryption Algorithm:

Select the algorithm to be used for the decryption.

Text to Decrypt:

Enter the text to decrypt. The text must be in the form of a base64 string.

Enter Decryption Key:

When choosing 'directly', the value entered in the Decryption Key field will be hidden.

When choosing 'as variable', a variable that stores the decryption key must be entered in this field. The '%' character will be treated as an indicator of the variable, not part of the key.

Decryption Key:

Enter the Decryption Key here. The decryption key value will be hidden.

Decrypt to File:

Select the file in which to save the decrypted text.

If File Exists:

Choose what to do in case the destination file already exists.

Decrypted File:

Enter a name for the output variable which will store the decrypted file for later processing.

Cipher Mode:

Select the cipher mode to use for the selected decryption algorithm.

Padding:

Select the padding to use for the selected decryption algorithm.

Block Size:

Select the block size in bits to use for the decryption.

Key Size:

Select the size of the key in bits to use for the decryption.

Use Salt:

Select whether to use Salt for the decryption.

Salt:

Enter the Salt to use for the decryption. The Salt value must be in the form of a base64 string.

Use IV (Initialization Vector):

Select whether to use an Initialization Vector.

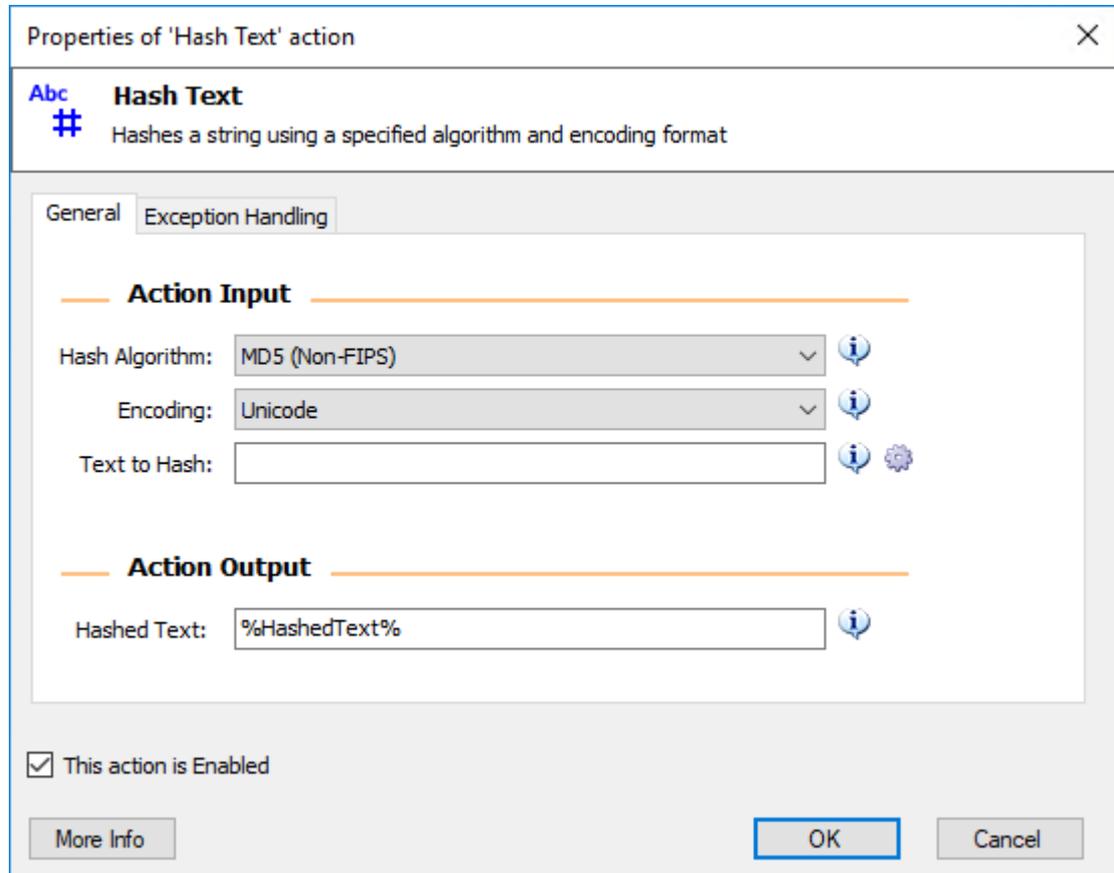
IV (Initialization Vector):

Enter the Initialization Vector to use for the decryption. The Initialization Vector must be in the form of a base64 string.

3.11.27.5 Hash Text

Description:

This action hashes a string using a specified algorithm and encoding format.



The screenshot shows the 'Properties of 'Hash Text' action' dialog box. It has a title bar with a close button (X). Below the title bar, there is a header section with 'Hash Text' and a description: 'Hashes a string using a specified algorithm and encoding format'. The main area is divided into two tabs: 'General' (selected) and 'Exception Handling'. Under the 'General' tab, there are two sections: 'Action Input' and 'Action Output'. In the 'Action Input' section, there are three fields: 'Hash Algorithm' (set to 'MD5 (Non-FIPS)'), 'Encoding' (set to 'Unicode'), and 'Text to Hash' (an empty text box). In the 'Action Output' section, there is one field: 'Hashed Text' (set to '%HashedText%'). At the bottom of the dialog, there is a checkbox labeled 'This action is Enabled' which is checked. There are also three buttons: 'More Info', 'OK', and 'Cancel'.

Properties:

Hash Algorithm:

Select the algorithm to be used for hashing.

Encoding:

Select the encoding of the text to hash.

Text to Hash:

Enter the text to hash.

Hashed Text:

Enter a name for the output variable that will store the hashed text for later processing.

3.11.27.6 Hash From File

Description:

This action hashes the contents of a file using a specified algorithm and encoding format.

Properties:

Hash Algorithm:

Select the algorithm to be used for hashing.

Encoding:

Select the encoding of the text to hash.

File to Hash:

Select the file that contains the text to hash.

Hashed Text:

Enter a name for the output variable that will store the hashed text for later processing.

3.11.27.7 Hash Text With Key

Description:

This action hashes a string with a key, using a specified algorithm and encoding format.

The screenshot shows a dialog box titled "Properties of 'Hash Text With Key' action". The dialog has a close button (X) in the top right corner. Below the title bar, there is a header area with "Hash Text With Key" and a description: "Hashes a string with a key, using a specified algorithm and encoding format". The main content area is divided into two tabs: "General" (selected) and "Exception Handling". Under the "General" tab, there are two sections: "Action Input" and "Action Output".

Action Input:

- Hash Algorithm: HMAC MD5 (Non-FIPS) (dropdown menu)
- Encoding: Unicode (dropdown menu)
- Text to Hash: (text input field)
- Enter Hash Key: Directly (dropdown menu)
- Hash Key: (text input field)

Action Output:

- Hashed Text: %HashedText% (text input field)

At the bottom of the dialog, there is a checkbox labeled "This action is Enabled" which is checked. There are three buttons: "More Info", "OK", and "Cancel".

Properties:

Hash Algorithm:

Select the algorithm to be used for hashing.

Encoding:

Select the encoding of the text to hash.

Text to Hash:

Enter the text to hash.

Enter Hash Key:

When choosing 'directly', the hash key entered in the Hash Key field will be hidden.

When choosing 'as variable', enter a variable containing the hash key. The '%' character will be treated as an indicator of a variable, not part of the key.

Hash Key:

Enter the Hash Key here. The hash key will be hidden.

Hashed Text:

Enter a name for the output variable that will store the hashed text for later processing.

3.11.27.8 Hash From File With Key***Description:***

This action hashes the contents of a file with a key, using a specified algorithm and encoding format.

Properties of 'Hash From File With Key' action

Hash From File With Key
Hashes the contents of a file with a key, using a specified algorithm and encoding format

General Exception Handling

Action Input

Hash Algorithm: HMACMD5 (Non-FIPS) ⓘ

Encoding: Unicode ⓘ

File to Hash: ⓘ ⚙️ 📄

Enter Hash Key: Directly ⓘ

Hash Key: ⓘ

Action Output

Hashed Text: %HashedText% ⓘ

This action is Enabled

More Info OK Cancel

Properties:

Hash Algorithm:

Select the algorithm to be used for hashing.

Encoding:

Select the encoding of the file to hash.

File to Hash:

Select the file that contains the text to hash.

Enter Hash Key:

When choosing 'directly', the hash key entered in the Hash Key field will be hidden.

When choosing 'as variable', enter a variable containing the hash key. The '%' character will be treated as an indicator of a variable, not part of the key.

Hash Key:

Enter the Hash Key here. The hash key will be hidden.

Hashed Text:

Enter a name for the output variable that will store the hashed text for later processing.

3.11.28 Outlook**3.11.28.1 Launch Outlook Action****Description:**

This action launches Outlook and creates a new Outlook Instance

Properties of 'Launch Outlook' action

Launch Outlook
This action launches Outlook and creates a new Outlook Instance

General Exception Handling

Action Input
(This action does not accept any input)

Action Output
Store Outlook Instance into: ⓘ

This action is Enabled

More Info OK Cancel

Properties:**Store Outlook Instance into:**

Enter a name to be the variable that will store the specific Outlook Instance for use with later Outlook actions.

Cautions:

For any Outlook-related action to work correctly, Microsoft Outlook must be installed on the computer where the robot will run.

3.11.28.2 Retrieve Email Messages From Outlook Action**Description:**

This action retrieves email messages from an Outlook account

Properties of 'Retrieve Email Messages From Outlook' action


Retrieve Email Messages From Outlook
This action retrieves email messages from an Outlook account

Email Filters Exception Handling

Action Input

Outlook Instance: ⓘ

Account: ⓘ ⚙️

Mail Folder: ⓘ ⚙️

Retrieve: ⓘ

Mark As Read ⓘ

...that match the following criteria:

"From" Field Contains: ⓘ ⚙️

"To" Field Contains: ⓘ ⚙️

"Subject" Contains: ⓘ ⚙️

"Body" Contains: ⓘ ⚙️

Attachments: ⓘ

Action Output

Store Messages into:

This action is Enabled

Properties:**Outlook Instance:**

Enter the variable that contains the Outlook Instance you want to work with. You must have previously specified this variable in a "Launch Outlook" action.

Account:

Enter the e-mail address of the Outlook account you wish to work with.

Mail Folder:

Enter the name of the folder that you want to retrieve messages from. Enter the full folder path for subfolders (ex. Inbox\Work).

Retrieve:

Specify whether you want to retrieve all messages in the folder, or only the unread ones.

Mark As Read:

Specify whether you want to mark as read all the unread messages retrieved.

"From" Field Contains:

Enter the full e-mail address of the Sender whose messages you want to retrieve. Leave this field blank to retrieve all messages regardless of the sender.

"To" Field Contains:

Enter the full e-mail address(es) of the Receptient(s) (separated by space or semicolon if more than one) for the messages you want to retrieve. Leave this field blank to retrieve all messages regardless of the recipient(s).

"Subject" Contains:

Enter the key phrase to be found within the email Subject. Leave this field blank to retrieve all email messages regardless of the their subject.

"Body" Contains:

Enter the key phrase to be found within the email Body. Leave this field blank to retrieve all email messages regardless of the their content.

Attachments:

Specify whether you want to save the attachments of the email messages retrieved or not.

Save Attachments into:

Enter here the folder in which you want the attachments to be saved.

Store Messages into:

Enter a name to be the variable that will store the retrieved email messages for later processing. The variable will contain a List of Outlook Message objects.

Cautions:

For any Outlook-related action to work correctly, Microsoft Outlook must be installed on the computer where the robot will run.

3.11.28.3 Send Email Through Outlook Action

Description:

This action creates and sends a new email message through Outlook

Properties of 'Send Email Through Outlook' action ✕


Send Email Through Outlook
 This action creates and sends a new email message through Outlook

General Exception Handling

Action Input

Outlook Instance: i ⚙

Account: i ⚙

To: i ⚙

CC: i ⚙

BCC: i ⚙

Subject: i ⚙

Body: i ⚙

Body is HTML i

Attachment(s): i ⚙ 📎

Action Output

(This action does not provide any output)

This action is Enabled

Properties:

Outlook Instance:

Enter the variable that contains the Outlook Instance you want to work with. You must have previously specified this variable in a "Launch Outlook" action.

Account:

Enter the e-mail address of the Outlook account you wish to work with.

To:

Enter the email address(es) of the recipient(s). If you enter more than one email address, the list of addresses should be separated by spaces or semicolons.

CC:

Enter the email address(es) of the CC recipient(s). If you enter more than one email address, the list of addresses should be separated by spaces or semicolons.

BCC:

Enter the email address(es) of the BCC (hidden) recipient(s). If you enter more than one email address, the list of addresses should be separated by spaces or semicolons.

Subject:

Enter the subject of the email.

Body:

Enter the text of the body.

Body Is HTML:

Choose whether the body of the email is interpreted as HTML coding.

Attachment(s):

Enter or choose the full path of any attachment(s), or choose a variable that contains a file or a list of files. Multiple files should be enclosed in double quotes (") and separated by a space character.

Cautions:

For any Outlook-related action to work correctly, Microsoft Outlook must be installed on the computer where the robot will run.

3.11.28.4 Process Email Messages in Outlook Action

Description:

Moves or deletes an email (or a list of email messages) retrieved by a "Retrieve Email Messages From Outlook" action.

Properties of 'Process Email Messages in Outlook' action ✕

 **Process Email Messages in Outlook**
 Moves or deletes an email (or a list of email messages) retrieved by a "Retrieve Emails From Outlook" action.

General Exception Handling

Action Input

Outlook Instance: i ⚙

Account: i ⚙

Email Message(s) To Process: i ⚙

Operation: i

Mail Folder: i ⚙

Action Output

(This action does not provide any output)

This action is Enabled

Properties:

Outlook Instance:

Enter the variable that contains the Outlook Instance you want to work with. You must have previously specified this variable in a "Launch Outlook" action.

Account:

Enter the e-mail address of the Outlook account you wish to work with.

Email messages to process:

Enter a variable containing the email or list of email messages to be processed. This should be a variable populated by a "Retrieve Email Messages From Outlook" action.

Operation:

Specify which operation you want to perform on the specified email messages.

Mail Folder:

Enter the name of the folder that you want to move e-mail messages to. Enter the full folder path for subfolders (ex. Inbox\Work).

Cautions:

For any Outlook-related action to work correctly, Microsoft Outlook must be installed on the computer where the robot will run.

3.11.28.5 Save Outlook Email Messages Action**Description:**

This action saves Outlook email messages given an account

Properties:**Outlook Instance:**

Enter the variable that contains the Outlook Instance you want to work with. You must have previously specified this variable in a "Launch Outlook" action.

Account:

Enter the e-mail address of the Outlook account you wish to work with.

Email Message(s) To Save:

Enter a variable containing the email message or list of email messages to be saved. This should be a variable populated by a "Retrieve Email Messages From Outlook" action.

File Name:

Specify whether you want to save the messages using the default name (Subject) or provide your own.

Save Format:

Specify the format by which to save the messages.

Save As:

Specify custom name for messages' name which will differ from message to message by an automatically added suffix.

Save Email Message(s) to:

Enter here the folder to which you want the messages to be saved.

Stored Message(s) Files:

Enter a name to be the variable that will store the file paths of the saved email messages for later processing.

Cautions:

For any Outlook-related action to work correctly, Microsoft Outlook must be installed on the computer where the robot will run.

3.11.28.6 Respond To Outlook Email Message Action***Description:***

This action provides a response to an Outlook Message, by replying, replying to all or forwarding

Properties of 'Respond To Outlook Mail Message' action ×

 **Respond To Outlook Mail Message**
This action provides a response to an Outlook Message, by replying, replying to all or forwarding

General Exception Handling

Action Input

Outlook Instance: i ⚙

Account: i ⚙

Email Message: i ⚙

Response Action: i

Body: i ⚙

Attachment(s): i ⚙ 📎

Action Output

(This action does not provide any output)

This action is Enabled

Properties:

Outlook Instance:

Enter the variable that contains the Outlook Instance you want to work with. You must have previously specified this variable in a "Launch Outlook" action.

Account:

Enter the e-mail address of the Outlook account you wish to work with.

Response Action:

Choose whether to reply (to sender or all) with a message or forward the received message.

To:

Enter the email address(es) of the recipient(s). If you enter more than one email address, the list of addresses should be separated by spaces or semicolons.

CC:

Enter the email address(es) of the CC recipient(s). If you enter more than one email address, the list of addresses should be separated by spaces or semicolons.

BCC:

Enter the email address(es) of the BCC (hidden) recipient(s). If you enter more than one email address, the list of addresses should be separated by spaces or semicolons.

Mail Message:

Enter the variable that holds the mail message you wish to act upon. This should be a variable populated by a "Retrieve Email Messages From Outlook" action.

Body:

Enter the HTML text of the body.

Attachment(s):

Enter or choose the full path of any attachment(s), or choose a variable that contains a file or a list of files. Multiple files should be enclosed in double quotes (") and separated by a space character.

Cautions:

For any Outlook-related action to work correctly, Microsoft Outlook must be installed on the computer where the robot will run.

3.11.28.7 Close Outlook Action

Description:

This action closes a previously launched Outlook Instance

Properties of 'Close Outlook' action

Close Outlook
This action closes a previously launched Outlook Instance

General Exception Handling

Action Input

Outlook Instance: %OutlookInstance%

Action Output

(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties:

Outlook Instance:

Enter the variable that contains the Outlook Instance you want to close. You must have previously specified this variable in a "Launch Outlook" action.

Cautions:

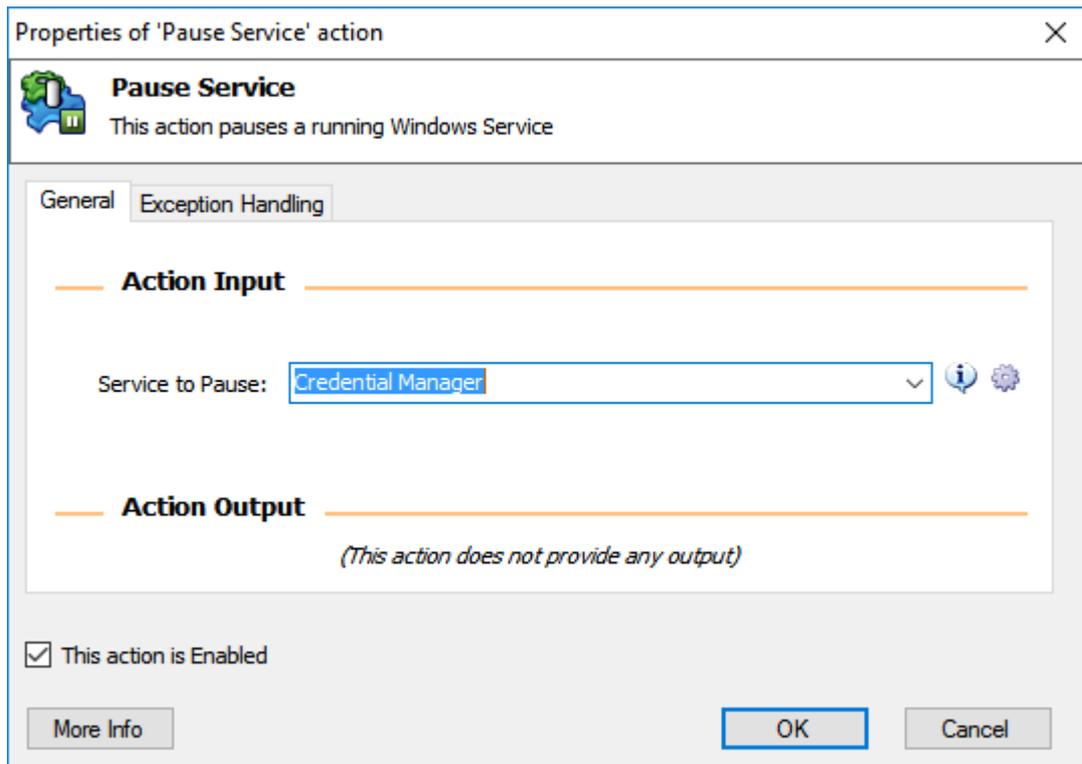
For any Outlook-related action to work correctly, Microsoft Outlook must be installed on the computer where the robot will run.

3.11.29 Services

3.11.29.1 Start Service Action

Description:

This action starts a stopped Windows Service



Properties:

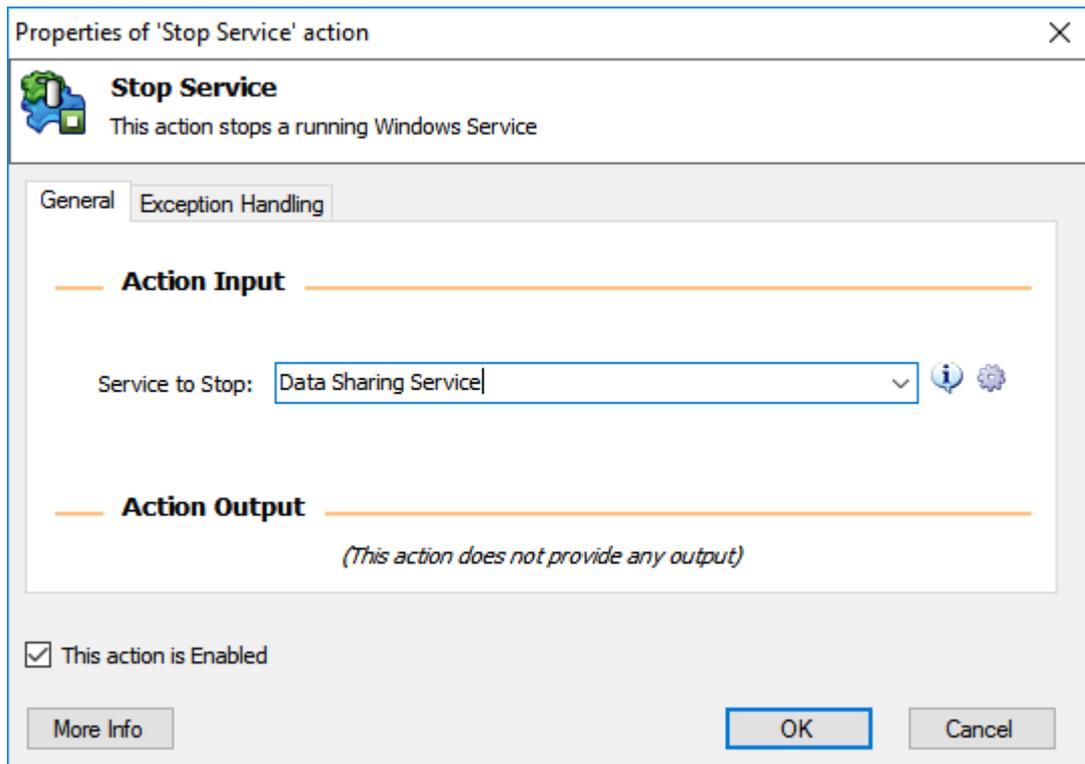
Service to Start:

Choose or enter the name of a Service to start.

3.11.29.2 Stop Service Action

Description:

This action stops a running Windows Service



Properties:

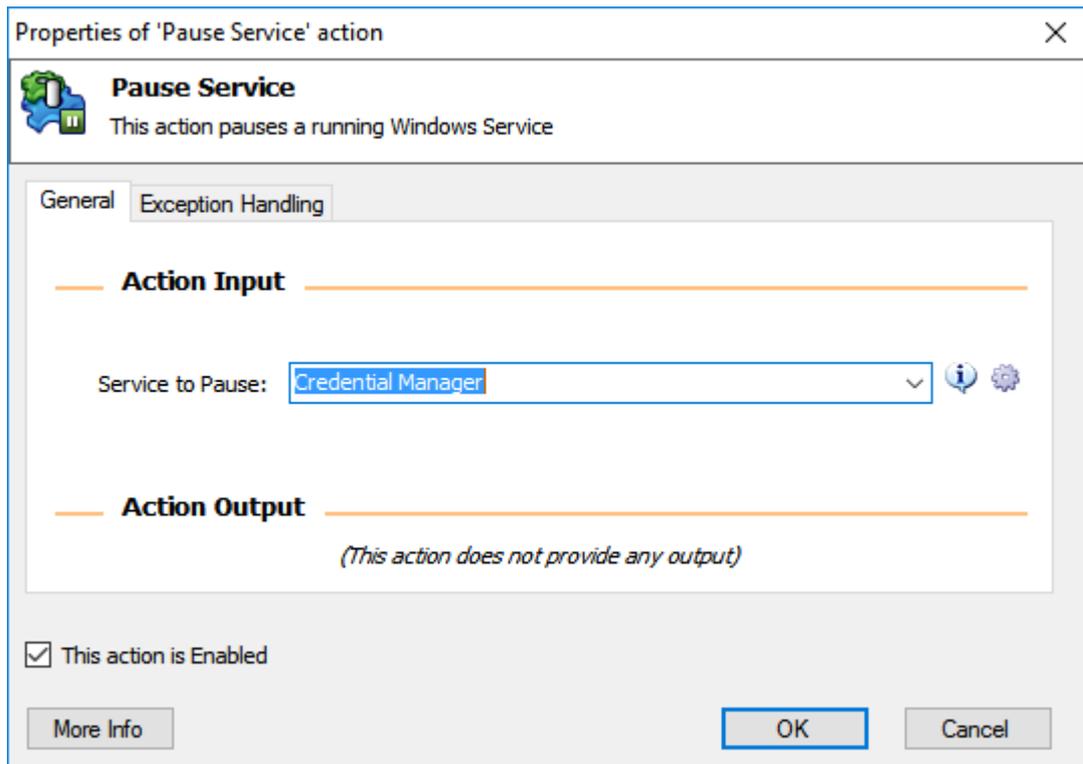
Service to Stop:

Choose or enter the name of a Service to stop.

3.11.29.3 Pause Service Action

Description:

This action pauses a running Windows Service



Properties:

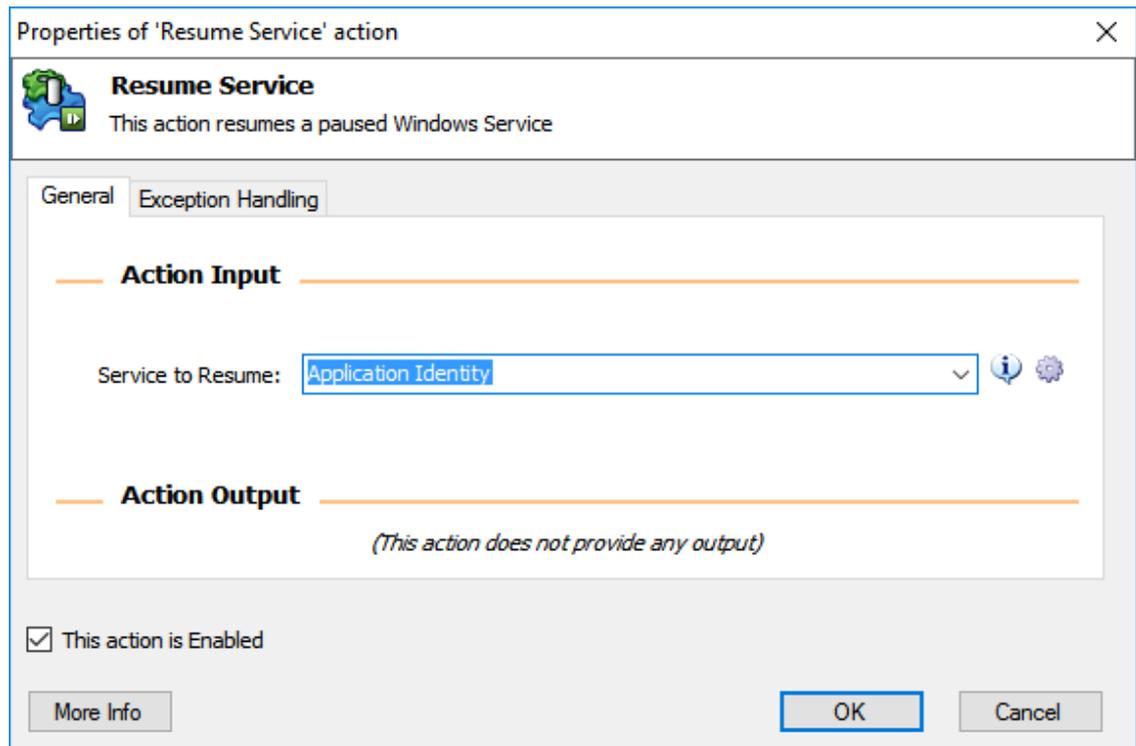
Service to Pause:

Choose or enter the name of a Service to pause.

3.11.29.4 Resume Service Action

Description:

This action resumes a paused Windows Service



Properties:

Service to Resume:

Choose or enter the name of a paused Service to resume.

3.11.30 ProcessRobot Actions

3.11.30.1 Start Process Action

Description:

This action executes a specific Process

Properties:

Process to Run:

Choose or enter the name of a ProcessRobot Process. Usually, it will be a Process other than the current one.

Wait for Process to Complete:

Check this box if you want to pause this Process until the called Process is complete. Otherwise both will run simultaneously.

Continue Anyway:

Choose whether the Process continues after a set number of seconds, regardless of whether the called Process has finished or not.

3.11.30.2 Get Command Line Arguments Action

Description:

This action retrieves the command line arguments that can be provided if the Process is compiled or started through the command line

Properties of 'Get Command Line Arguments' action

Get Command Line Arguments
This action retrieves the command line arguments that can be provided if the robot is compiled or started through the command line

General

Action Input
(This action does not accept any input)

Action Output
Store Command Line Arguments into: %CommandLineArguments%

This action is Enabled

More Info OK Cancel

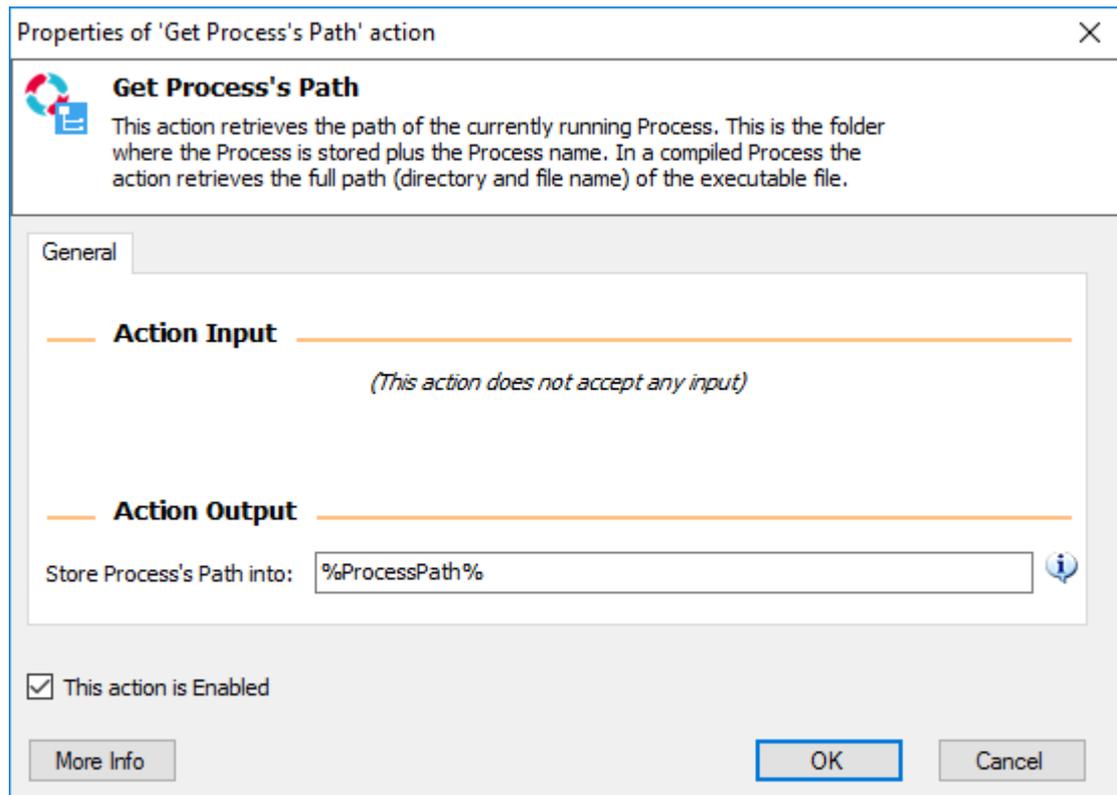
Properties:**Store Command Line Arguments into:**

Enter a name to be the variable that will store the Command Line Arguments entered by the user. This will be a list of text values.

Note: A backslash (\) character will "escape" the next control character, including quotes (") and itself. Therefore, when you need to include literal backslashes into one of the parameters (e.g. "C:\My Data\etc\"), please remember to escape them first ("C:\\My Data\\etc\\").

3.11.30.3 Get Process' Path Action**Description:**

This action retrieves the path of the currently running Process. This is the folder where the Process is stored plus the Process name. In a compiled Process the action retrieves the full path (directory and file name) of the executable file.



Properties:

Store Process's Path into:

Enter a name to be the variable that will store the path of the current Process.

3.11.30.4 Log Message Action

Description:

This action creates an custom Log Entry in the [ProcessRobot Event Log](#)¹⁷⁸.

Properties of 'Log Message' action

Log Message
This action creates a custom Log Entry in the ProcessRobot Event Log.

General

Action Input

Message to Log: Backup Completed Successfully

Event Type: Informational

Action Output
(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties:

Message to Log:

Enter the message you want to log.

Event Type:

Specify the event type that will be logged (informational, warning or error).

3.11.30.5 Get Last Exception Action

Description:

This action retrieves the last occurred exception.

Properties of 'Get Last Exception' action

 **Get Last Exception**
This action retrieves the last occurred exception

General

Action Input

(This action does not accept any input)

Action Output

Store exception into: 

This action is Enabled

[More Info](#) [OK](#) [Cancel](#)

Properties:

Store Exception into:

Enter a name to be the variable that will store the last exception occurred and its relevant info.

3.11.30.6 Update KPI Action

Description: This action updates Key Performance Indicators

Properties of 'Update KPI' action

Update KPI
This action updates Key Performance Indicators.

General

Action Input

KPI Name:  

Update KPI Value by:  

Dimension #1:  

Dimension #2:  

Dimension #3:  

Dimension #4:  

Dimension #5:  

Action Output

(This action doesn't provide any output)

This action is Enabled

Properties:

KPI Name: Enter the name of the Key Performance Indicator you wish to log the state of.

Update KPI Value by: Specify the delta value you want the KPI to be (de) incremented by.

Dimension #1: Specify the value you want to set the first Dimension of the KPI to.

Dimension #2: Specify the value you want to set the second Dimension of the KPI to.

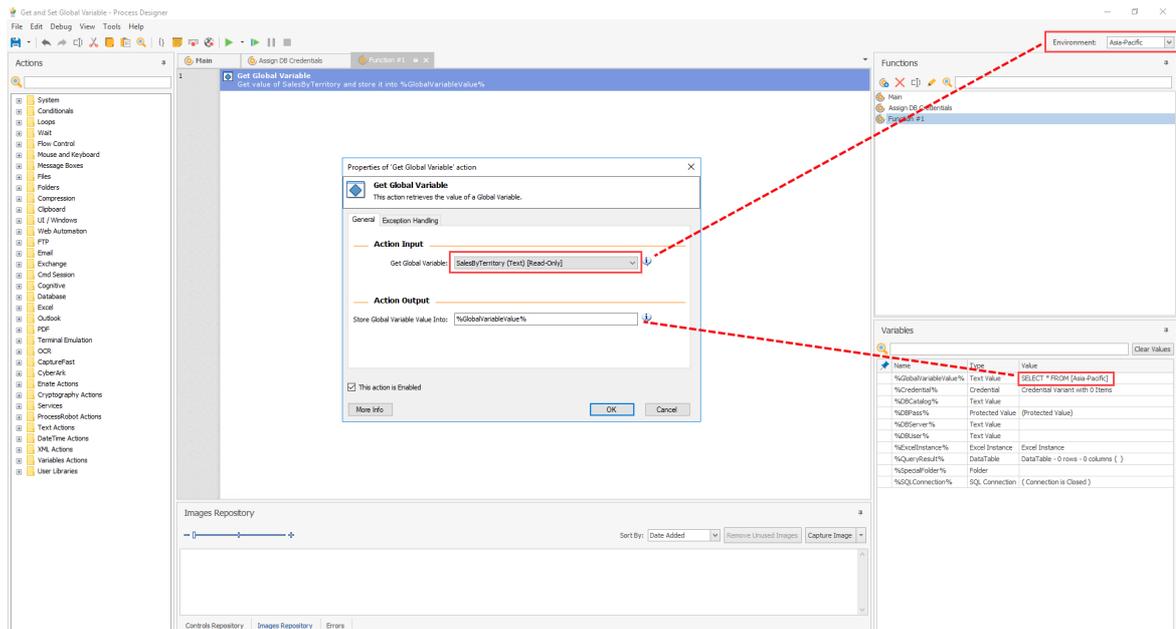
Dimension #3: Specify the value you want to set the third Dimension of the KPI to.

Dimension #4: Specify the value you want to set the fourth Dimension of the KPI to.

Dimension #5: Specify the value you want to set the fifth Dimension of the KPI to.

3.11.30.7 Get Global Variable

Description: This actions gets a Global Variable's Value



Properties:

Get Global Variable: Set the Global Variable you wish to get the value of. If no Global Variable is available then go to Settings > Global Variables to create one.

Store Global Variable into: Enter a name to be the Variable that will hold the Global Variable's Value.

3.11.30.8 Set Global variable

Description: This actions sets a Global Variable's Value

Properties:

Set Global Variable: Select the name of Global Variable that you want to set. If no global variable is available, contact Process Robot administrator.

With Value: Enter the value to be stored in the Global Variable.

3.11.30.9 Enter Locked Region Action

Description:

This action marks the begging of a locked region which only one process will be allowed to enter according the Locker Name and Lock Type

Properties of 'Enter Locked Region' action

 **Enter Locked Region**
This action marks the beginning of a locked region which only one process will be allowed to enter according to the Locker Name and Lock Type

General

Action Input

Locker Name:  

Lock Type: 

Action Output

(This action does not provide any output)

This action is Enabled

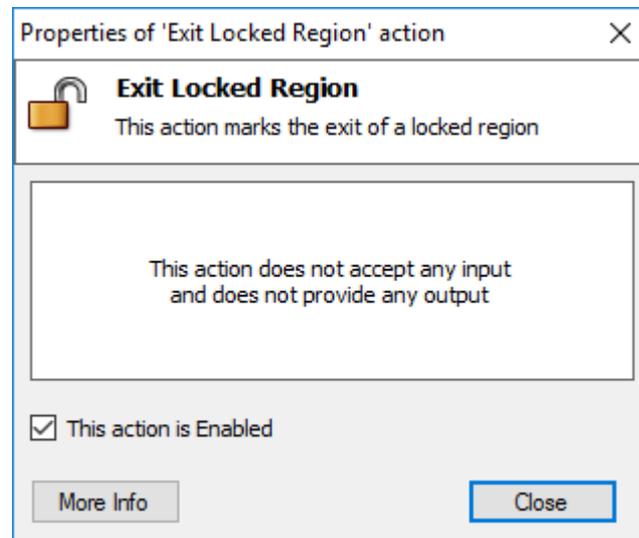
Properties:

Region Locker Name: Enter the name of locker for the Locker Region. Characters '!' or ' ' are invalid for locker name.

Lock Type: Select the lock type for the specified Locked Region.

3.11.30.1 Exit Locked Region Action

Description: This action marks the exit of a locked region



3.11.30.1 Add Queue Item in Queue Action

Description: This action adds a Queue Item into a Queue:

Properties of 'Add Queue Item In Queue' action

Add Queue Item In Queue
This action adds a Queue Item into a Queue.

General | Advanced | Exception Handling

Action Input

Add Into Queue: Support Tickets (Text)

Queue Item: %CurrentItem%

Action Output

(This action does not provide any output)

This action is Enabled

More Info OK Cancel

General tab Properties:

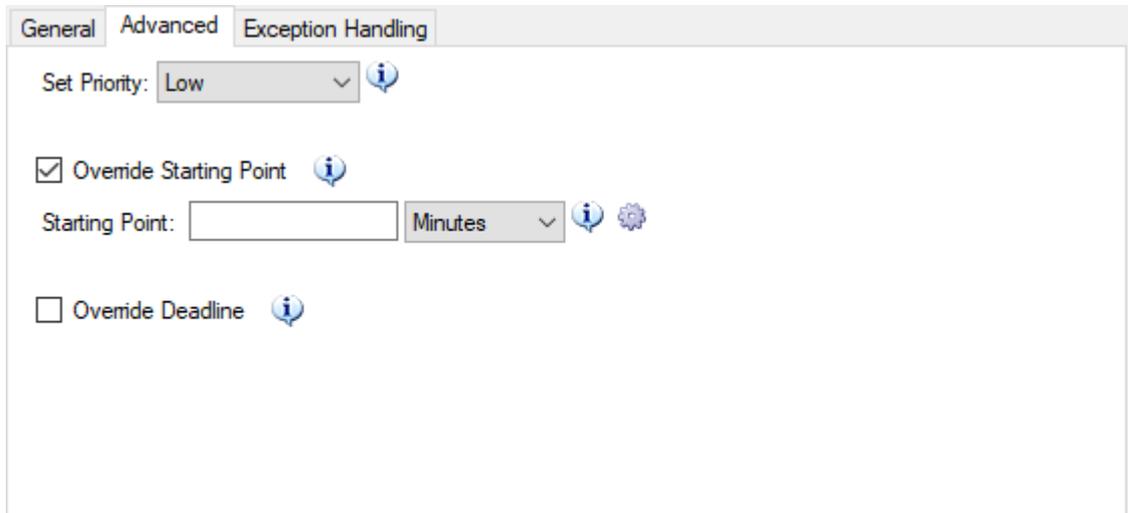
Add into Queue: Select the Queue in which you wish to add a Queue Item.

Queue Item: Enter a literal value or a variable that is holding a value in order to add it into the Queue. Please bear in mind that the Queue Item should be of the same Data Type that has been defined in the Queue Settings.

Advanced tab Properties:

Set Priority: This drop down menu allows you to set the Priority Status of the given Item.

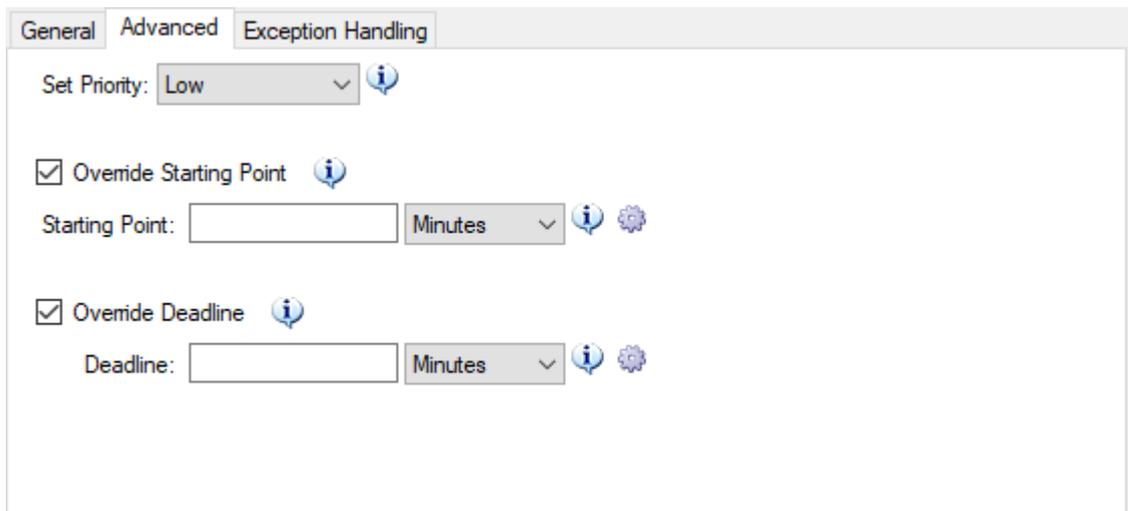
Override Starting Point: Ticking this check box produces an extra Starting Point property that allows you to override the default SLA Settings set in the [Settings > Queues](#) ^[220].



The screenshot shows a settings window with three tabs: 'General', 'Advanced', and 'Exception Handling'. The 'Exception Handling' tab is active. It contains the following elements:

- 'Set Priority: Low' with a dropdown arrow and an information icon.
- A checked checkbox for 'Override Starting Point' with an information icon.
- 'Starting Point: [text input] Minutes' with a dropdown arrow, an information icon, and a gear icon.
- An unchecked checkbox for 'Override Deadline' with an information icon.

Override Deadline: Ticking this check box produces an extra Deadline property that allows you to override the default SLA Settings set in the [Settings > Queues](#) ^[220].



The screenshot shows the same settings window as above, but with the 'Override Deadline' checkbox also checked. It now includes:

- 'Deadline: [text input] Minutes' with a dropdown arrow, an information icon, and a gear icon.

3.11.30.1: Get Queue Item from Queue Action

Description: This action retrieves the first Queue Item from a Queue or the Queue Item that initiated this process:

The screenshot shows a dialog box titled "Properties of 'Get Queue Item from Queue' action". The dialog has a close button (X) in the top right corner. Below the title bar, there is a header section with a blue icon of a hand pointing to a queue and the text "Get Queue Item from Queue". Below this, a description reads: "This action retrieves the first Queue Item from a Queue or the Queue Item that initiates this process." The dialog is divided into two tabs: "General" (selected) and "Exception Handling". Under the "General" tab, there are two sections: "Action Input" and "Action Output". In the "Action Input" section, there are two dropdown menus: "Get:" with the value "First Queue Item Of Queue" and "Select Queue:" with the value "Support Tickets (Text)". In the "Action Output" section, there is a text input field labeled "Store Queue Item Into:" with the value "%QueueItemValue%". At the bottom of the dialog, there is a checkbox labeled "This action is Enabled" which is checked. There are three buttons at the bottom: "More Info", "OK", and "Cancel".

Get: First working item in Queue

Properties of 'Get Queue Item from Queue' action

Get Queue Item from Queue
This action retrieves the first Queue Item from a Queue or the Queue Item that initiates this process.

General Exception Handling

Action Input

Get: Queue Item That Initiated This Process

Action Output

Store Queue Item Into: %QueueItemValue%

This action is Enabled

More Info OK Cancel

Get: Working Item that Initiated this Process

Properties:

Get: There are two ways to get Queue Items from a Queue depending on whether we want Queue Items automatically distributed or not.

Use [First Queue Item of Queue](#) if you do not want to use automatic distribution.

Use [Queue Item That Initiated This Process](#) if you want to use automatic distribution.

If you select [First Queue Item of Queue](#) an additional Property "Select Queue" will be produced allowing you to select readily which one from your Queues you want to Get the Item from.

The screenshot shows a configuration window with two tabs: 'General' and 'Exception Handling'. The 'General' tab is active. It is divided into two sections: 'Action Input' and 'Action Output'.
Under 'Action Input':
- 'Get:' is a dropdown menu set to 'First Queue Item Of Queue'.
- 'Select Queue:' is a dropdown menu with a list of options: 'Support Tickets (Text)', 'Client Directory (Custom Object)', 'Numeric ID (Numeric)', and 'Pending Request (Does not exist on server)'. The 'Client Directory (Custom Object)' option is currently selected.
Under 'Action Output':
- 'Store Queue Item Into:' is a text input field containing the placeholder text '%QueueItemValue%'.

Store Queue Item Into: Enter a Name to be the variable that will hold the Queue Item.

3.11.30.1: Get Credential

This action retrieves the specified Credential from the Control Desk > Settings > Credential Manager [a] and stores them inside a variable [b] for later use.

Properties of 'Get Credential' action

Get Credential
This action retrieves the specified Credential from the Server.

General Exception Handling

Action Input

Get Credential: Credential1
Credential1

Action Output

Store Credential Into: %Credential%

This action is Enabled

More Info OK Cancel

a Get Credential:

Select the name of the Credential you wish to get. If no Credentials are available, contact Process



Store Credential into:

Enter a name to be the variable that will hold the Credential.

-If you do not understand how to work with credentials please read the

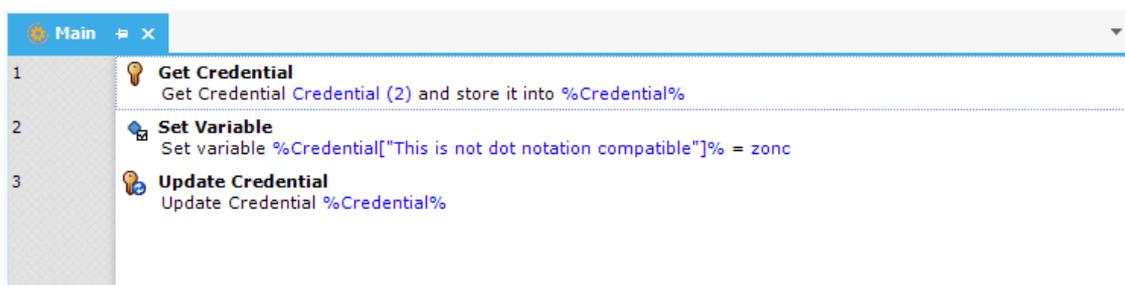
a) How to Update Credentials within a Process:

To update any given credential stored in a Variable, a minimum of three actions is required in the

a) Get Credential: this action will allow you to store the credential you wish to update, within a Variable.

b) Set Variable: Set Variable or any other action that will give you the opportunity to access (see **How to work with Credentials** section later on) and set the value of a Credential Property; effectively creating an updated Credential.

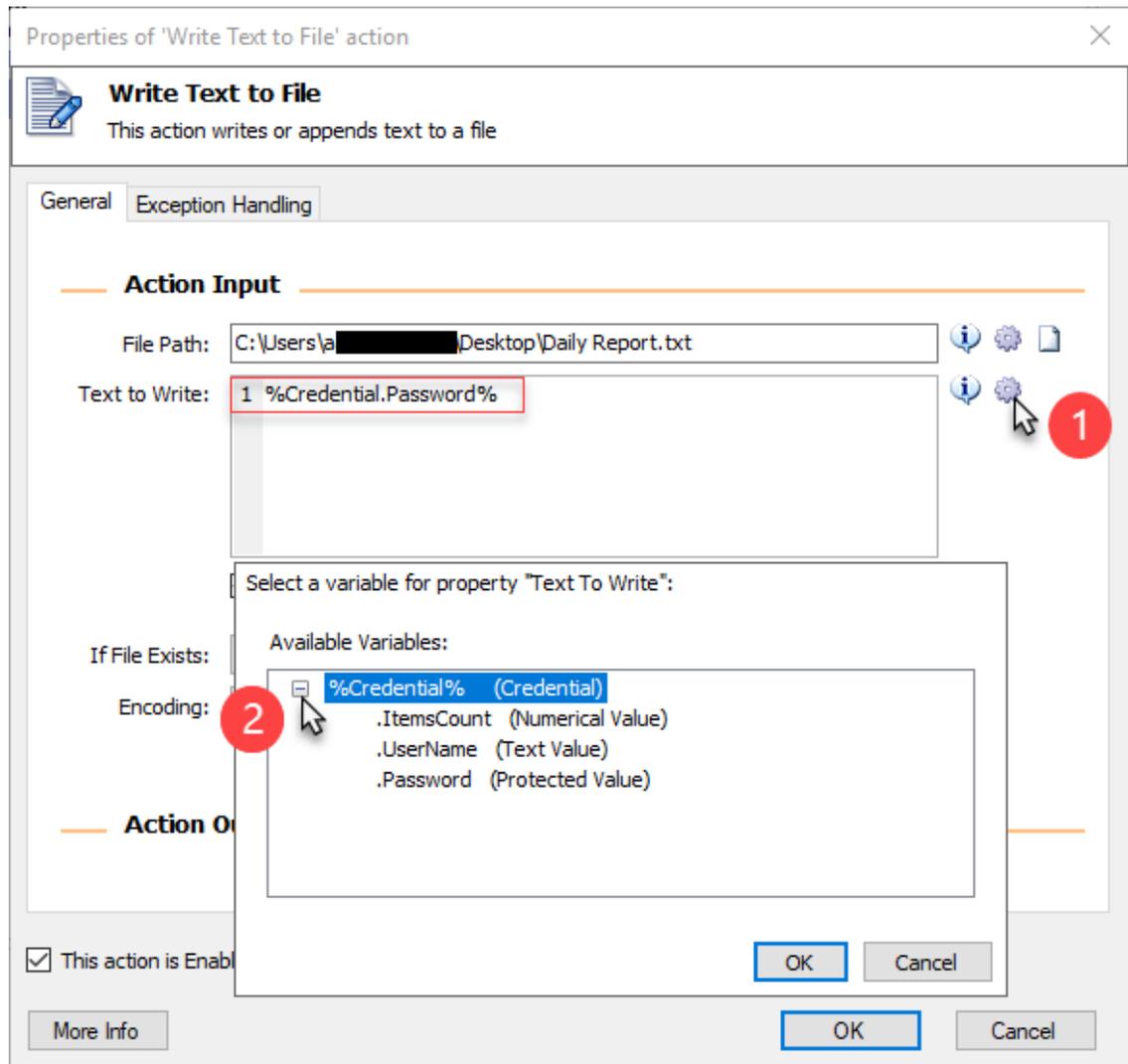
c) The Update Credential action (this action): that allows you to save the changes of the updated Credential.



The Screenshot above is showing you how these actions combine in the Workspace in order to Update Credentials within a Process.

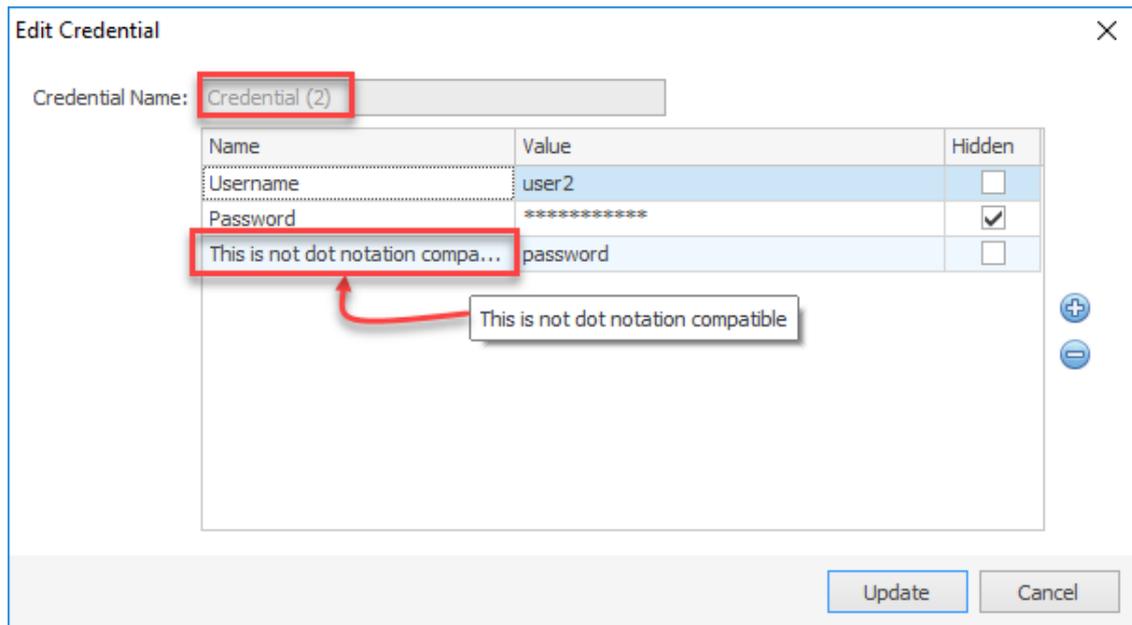
b) How to work with Credentials:

Credentials are a special data type on Process Robot that allows you to store and access safely, sensitive information like passwords and addresses. After selecting the Credential you want to work with through the drop down menu of Get Credential and store it into the Variable provided in the Store Credential Into, you can access any of that Credential's Values as a regular Data type Property, readily available to you via the Gear [1] Icon and the resulting treeview [2] control element:

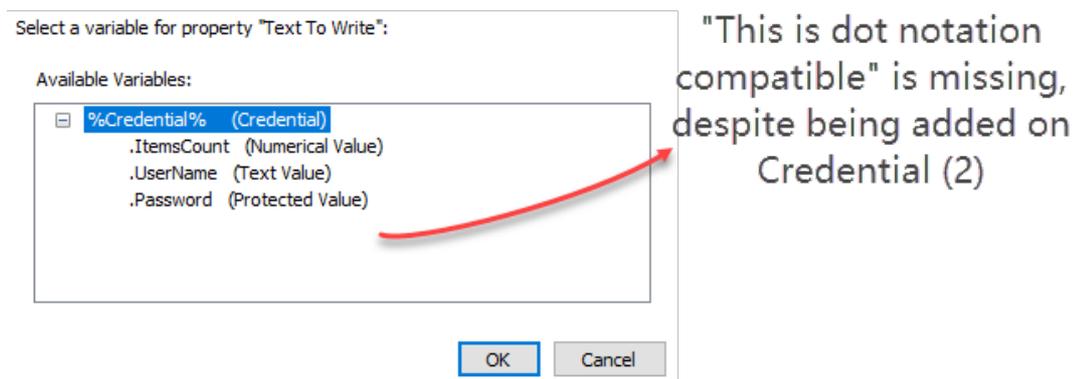


You can also access the Value you are interested in using bracket [""] notation in case you have used an irregular, custom named Credential Property that does not follow the syntax rules necessary to enable access via dot (.) notation.

For example, the added "This is not dot notation compatible" **Credential (2)** Property below:



it won't be accessible via the Gear Icon options, since it is not following valid dot notation syntax:



To access this Value, we should use bracket notation in the following manner:

Properties of 'Write Text to File' action

Write Text to File

This action writes or appends text to a file

General Exception Handling

Action Input

File Path: C:\Users\j... Desktop\Daily Report.txt

Text to Write: 1 %Credential["This is not dot notation compatible"]%

Append New Line

If File Exists: Append content

Encoding: Unicode

Action Output

(This action does not provide any output)

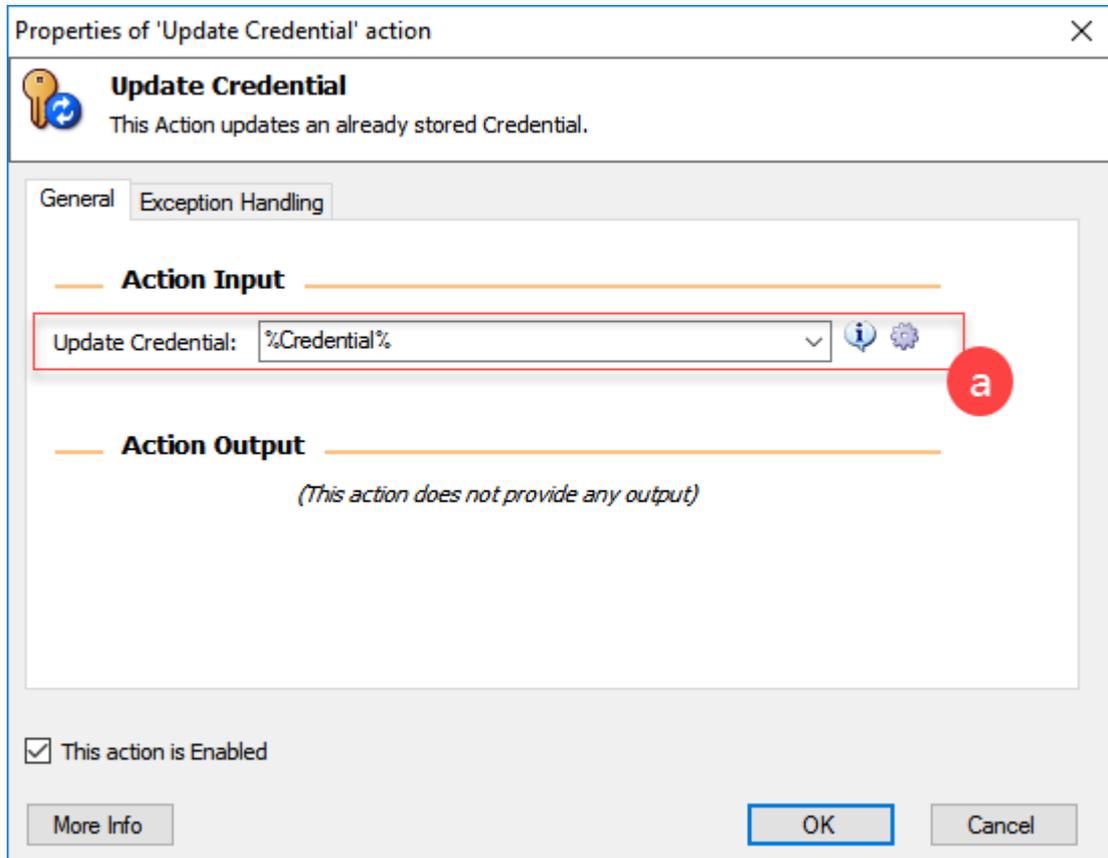
This action is Enabled

More Info OK Cancel

This will indeed append the value of that Credential Property on the selected text file.

3.11.30.1 Update Credential

This Action saves the changes implemented to an existing Credential stored [a] within a Variable.



The screenshot shows the 'Properties of 'Update Credential' action' dialog box. The title bar reads 'Properties of 'Update Credential' action'. The main heading is 'Update Credential' with a key icon and a refresh icon, followed by the description 'This Action updates an already stored Credential.' Below this, there are two tabs: 'General' (selected) and 'Exception Handling'. Under the 'General' tab, there are two sections: 'Action Input' and 'Action Output'. The 'Action Input' section contains a text field labeled 'Update Credential:' with a dropdown menu showing '%Credential%' and a red circle 'a' next to it. The 'Action Output' section is empty and contains the text '(This action does not provide any output)'. At the bottom, there is a checkbox labeled 'This action is Enabled' which is checked. There are three buttons: 'More Info', 'OK', and 'Cancel'.



Update Credential:

This text field with drop down menu options invite you to enter the Variable that holds the updated

-If you do not understand how to work with credentials please read the

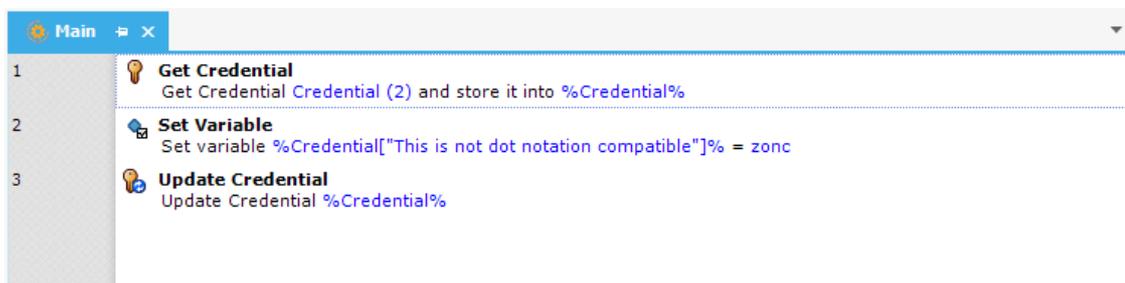
a) How to Update Credentials within a Process:

To update any given credential stored in a Variable, a minimum of three actions is required in the

a) Get Credential: this action will allow you to store the credential you wish to update, within a Variable.

b) Set Variable: Set Variable or any other action that will give you the opportunity to access (see **How to work with Credentials** section later on) and set the value of a Credential Property; effectively creating an updated Credential.

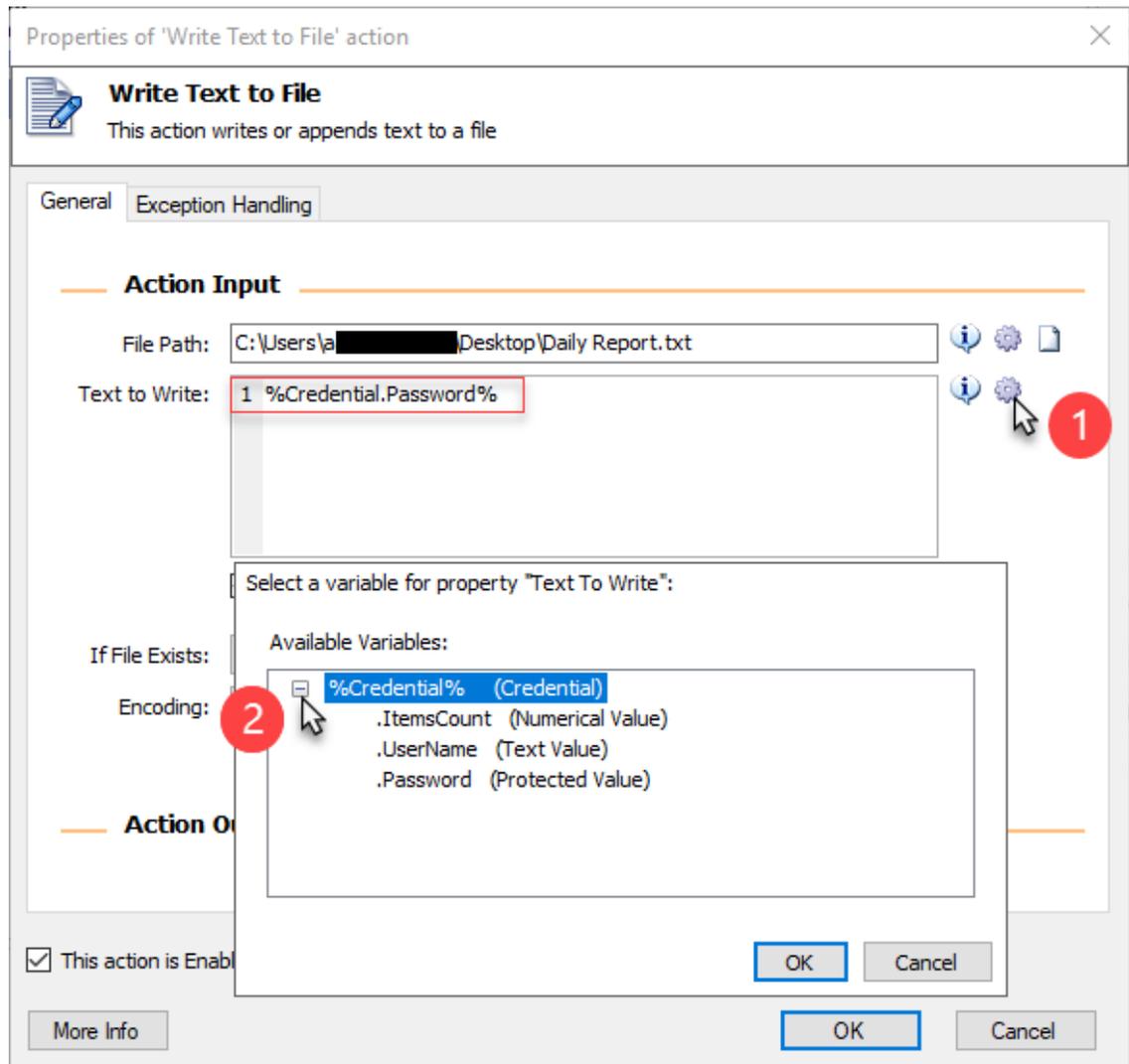
c) The Update Credential action (this action): that allows you to save the changes of the updated Credential.



The Screenshot above is showing you how these actions combine in the Workspace in order to Update Credentials within a Process.

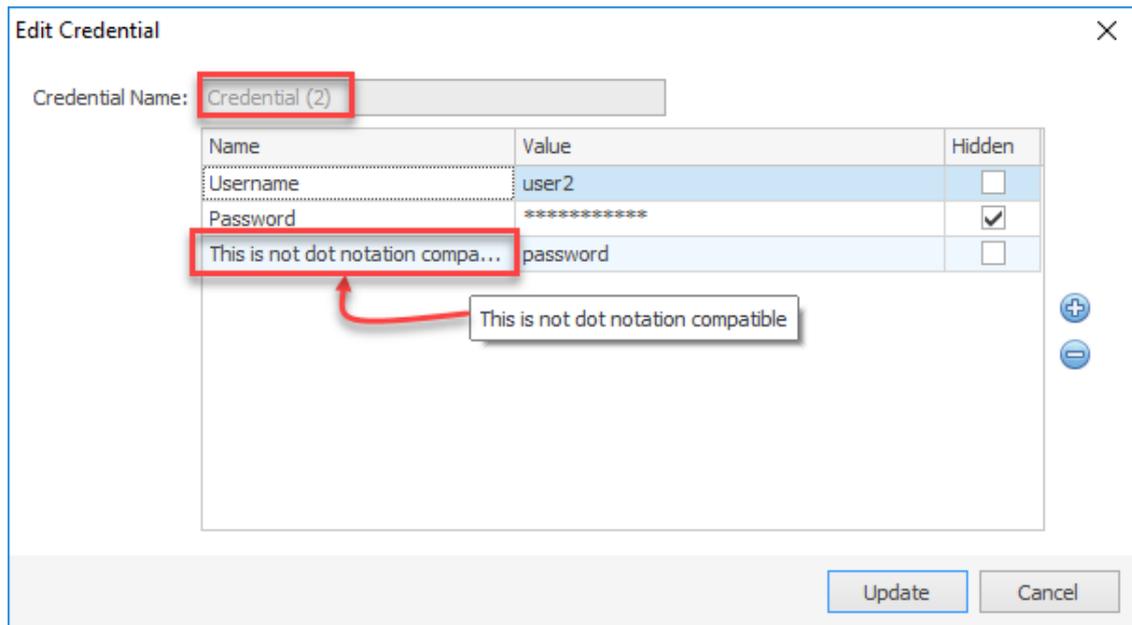
b) How to work with Credentials:

Credentials are a special data type on Process Robot that allows you to store and access safely, sensitive information like passwords and addresses. After selecting the Credential you want to work with through the drop down menu of Get Credential and store it into the Variable provided in the Store Credential Into, you can access any of that Credential's Values as a regular Data type Property, readily available to you via the Gear [1] Icon and the resulting treeview [2] control element:

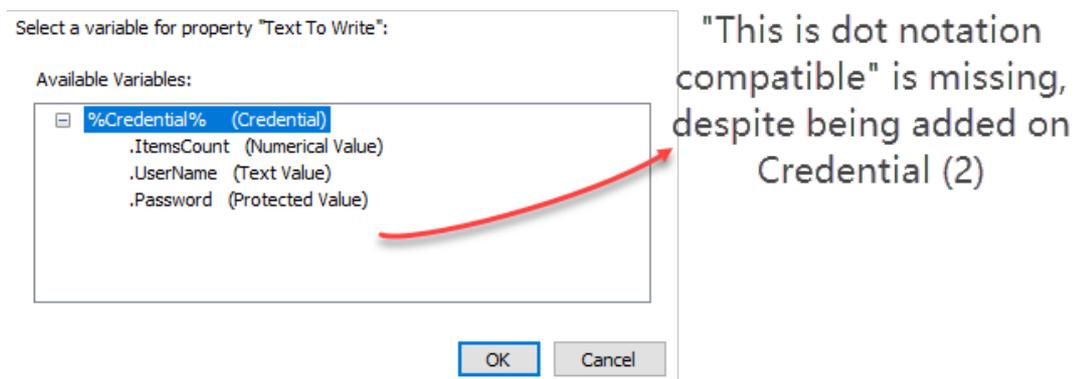


You can also access the Value you are interested in using bracket [""] notation in case you have used an irregular, custom named Credential Property that does not follow the syntax rules necessary to enable access via dot (.) notation.

For example, the added "This is not dot notation compatible" **Credential (2)** Property below:



it won't be accessible via the Gear Icon options, since it is not following valid dot notation syntax:



To access this Value, we should use bracket notation in the following manner:

Properties of 'Write Text to File' action

Write Text to File
This action writes or appends text to a file

General | Exception Handling

Action Input

File Path: C:\Users\... Desktop\Daily Report.txt

Text to Write: 1 %Credential["This is not dot notation compatible"]%

Append New Line

If File Exists: Append content

Encoding: Unicode

Action Output
(This action does not provide any output)

This action is Enabled

More Info OK Cancel

This will indeed append the value of that Credential Property on the selected text file.

3.11.31 Text

3.11.31.1 Get Text Length Action

Description:

This action retrieves the length (in characters) of a text value or variable

The screenshot shows a dialog box titled "Properties of 'Get Text Length' action". The dialog has a close button (X) in the top right corner. Below the title bar, there is a small icon with the text "abcde" and a hand cursor, followed by the text "Get Text Length" and a description: "This action retrieves the length (in characters) of a text value or variable".

The dialog is divided into a "General" tab. Under the "Action Input" section, there is a text box labeled "Text to Measure:" containing the value "%UserInput%". To the right of this text box are an information icon (i) and a settings icon (gear). Under the "Action Output" section, there is a text box labeled "Store Length into:" containing the value "%TextLength%". To the right of this text box is an information icon (i).

At the bottom of the dialog, there is a checked checkbox labeled "This action is Enabled". Below the checkbox are three buttons: "More Info", "OK", and "Cancel".

Properties:

Text to Measure:

Enter the text, or a previously stored text variable, to be measured.

Store Length into:

Enter a name to be the variable that will store the text's length as a number.

3.11.31.2 Append Line to Text Action

Description:

This action appends a new line of text to a text value or variable

Properties of 'Append Line to Text' action

Append Line to Text
This action appends a new line of text to a text value or variable

General

Action Input

Original Text: %AllText%

Line to Append: %UserInput%

Action Output

Store Result into: %Result%

This action is Enabled

More Info OK Cancel

Properties:**Original Text:**

Enter the original text or a previously stored text variable.

Line to Append:

Enter the text, or a previously stored text variable, to add on as a new line.

Store Result into:

Enter a name to be the variable that will store the new text. As elsewhere, this can be the name of the original variable, overwriting the original text with the new text.

3.11.31.3 Get Subtext Action**Description:**

This action retrieves a subtext from a text value or variable

Properties of 'Get Subtext' action

Get Subtext
This action retrieves a subtext from a text value or variable

General | Exception Handling

Action Input

Original Text: ⓘ ⚙️

Start Index: ⓘ

Character Position: ⓘ ⚙️

Length: ⓘ

Number of Chars: ⓘ ⚙️

Action Output

Store Subtext into: ⓘ

This action is Enabled

Properties:

Original Text:

Enter the text, or a previously stored text variable, that contains the section of text you want to retrieve.

Start Index:

Choose how you want to find the starting point for text retrieval.

Character Position:

Set the position of the first character you will retrieve. This is a zero-based index, counting from zero for the first character. For example, the tenth character would be 9.

Length:

Choose whether the subtext continues to the end of the text, or includes only a certain number of characters.

Number of Chars:

Set the number of characters to be retrieved.

Store Subtext into:

Enter a name to be the variable where the subtext will be stored. As elsewhere, this can be the name of the original variable, overwriting the original text with the new text.

3.11.31.4 Pad Text Action

Description:

This action creates a fixed length text by adding characters to the left or to the right of an existing text

Properties of 'Pad Text' action

Pad Text
This action creates a fixed length text by adding characters to the left or to the right of an existing text

General

Action Input

Text to Pad: %Sum%

Pad: Left

Char or Text for padding: 0

Total Length: 5 characters

Action Output

Store Result into: %Sum%

This action is Enabled

More Info OK Cancel

Properties:

Text to Pad:

Enter the text, or a previously stored text variable, to be lengthened.

Pad:

Choose whether to add characters to the left or right of the existing text.

Char or Text for Padding:

Enter the character or text that will be added to lengthen the original text.

Total Length:

Set the total character length of the final padded text. This means that the Char or Text for Padding will be repeatedly added until the final text is the right length. If it is already the right length, no characters will be added.

Store Result into:

Enter a name to be the variable that will store the new, padded Text. As elsewhere, this can be the name of the original variable, overwriting the original text with the new text.

3.11.31.5 Trim Text Action***Description:***

This action removes all occurrences of white space characters (such as space, tab, or new line) from the beginning and/or end of an existing text.

Properties of 'Trim Text' action

Trim Text
This action removes all occurrences of white space characters (such as space, tab, or new line) from the beginning and/or end of an existing text.

General

Action Input

Text to Trim: ⓘ ⚙️

What to Trim: ⓘ

Action Output

Store Trimmed Text into: ⓘ

This action is Enabled

Properties:**Text to Trim:**

Enter the text, or a previously stored text variable, that you want to have trimmed.

What to Trim:

Choose where white space characters will be removed from.

Store Trimmed Text into:

Enter a name to be the variable that will store the new, trimmed Text. As elsewhere, this can be the name of the original variable, overwriting the original text with the new text.

3.11.31.6 Change Text Case Action**Description:**

This action changes the casing of a text to uppercase, lowercase, title case or sentence case.

Properties:**Text to Convert:**

Enter the text, or a previously stored text variable, that you want to convert.

Convert to:

Choose which text case style you want to use. Examples are given in their names.

Store Text with New Case into:

Enter a name to be the variable that will store the new, converted Text. As elsewhere, this can be the name of the original variable, overwriting the original text with the new text.

3.11.31.7 Convert Text to Number Action**Description:**

This action converts a text representation of a number to a variable that contains a numeric value

The screenshot shows a dialog box titled "Properties of 'Convert Text to Number' action". It features a header with a small icon and the title "Convert Text to Number", followed by a description: "This action converts a text representation of a number to a variable that contains a numeric value". Below this are two tabs: "General" (selected) and "Exception Handling". The "General" tab contains two sections: "Action Input" and "Action Output". The "Action Input" section has a label "Text to Convert:" and a text box containing "%UserInput%". The "Action Output" section has a label "Store Result into:" and a text box containing "%TextAsNumber%". At the bottom left, there is a checked checkbox labeled "This action is Enabled". At the bottom right, there are three buttons: "More Info", "OK", and "Cancel".

Properties:

Text to Convert:

Enter a previously stored text variable containing only a number, that will be converted to a numeric value variable. Spaces will be ignored, but non-number text will throw an exception.

Store Result into:

Enter a name to be the variable that will store the new, numeric value.

3.11.31.8 Convert Number to Text Action

Description:

This action converts a number to text using a specific format that you determine.

Properties of 'Convert Number to Text' action

Convert Number to Text
This action converts a number to text using a specific format that you determine.

General

Action Input

Number to Convert: ⓘ ⚙️

Decimal Places: ⓘ

Use 1000 Separator ⓘ

Example ⓘ

Action Output

Store Result into: ⓘ

This action is Enabled

Properties:**Number to Convert:**

Enter a number, or a previously stored numeric variable, that will be converted to text.

Decimal Places:

Choose the number of decimal places that will be included before truncation. Zeros can also be added to the end to pad the text in this way.

Use 1000 Separator:

Choose whether or not to use punctuation as a 1000 separator.

Example:

Here you can see an example of what your textual number will look like.

Store Result into:

Enter a name to be the variable that will store the formatted number as Text.

3.11.31.9 Convert Text to DateTime Action

Description:

This action converts a text representation of a Date and/or Time value to a variable that contains a DateTime value

Properties of 'Convert Text to DateTime' action

Convert Text to DateTime
This action converts a text representation of a Date and/or Time value to a variable that contains a DateTime value

General Exception Handling

Action Input

Text to Convert: 20172103

Date is represented in custom format

Custom Format: yyyyddMM

Action Output

Store Result into: %TextAsDateTime%

This action is Enabled

More Info OK Cancel

Properties:

Text to Convert:

Enter the text, or a previously stored text variable, to be converted to a DateTime value. This text must be in a recognizably DateTime value format.

Date is represented in custom format:

Use this option if the text to be converted contains a representation of the Date Time in a non-standard, non-recognizable format. In this case you will also need to provide the custom format in which the date is represented.

Custom Format:

Enter the format in which the date is stored in the text. You can express a custom format as, for example, yyyyMMdd for date, and hhmmss for time. For more information on how to specify a custom format see [Custom Date Formats](#)^[444].

Store Result into:

Enter a name to be the variable that will store the DateTime value.

3.11.31.1 Convert DateTime to Text Action

Description:

This action converts a DateTime value to text using a specific format

Properties of 'Convert DateTime to Text' action

Convert DateTime to Text
This action converts a DateTime value to text using a specific format that you determine.

General

Action Input

DateTime to Convert: %CurrentDateTime%

Format to Use: Custom

Custom Format: yyyy-MM-dd hh:mm:ss

Sample: 2017-03-21 06:50:55

Action Output

Store Result into: %FormattedDateTime%

This action is Enabled

More Info OK Cancel

Properties:

DateTime to Convert:

Enter a DateTime value, or a previously stored DateTime variable, that will be converted to text.

Format to Use:

Choose whether to use a standard DateTime format, or create a custom one.

Standard Format:

Choose which standard DateTime format this action will use to display the DateTime value.

Custom Format:

Set the custom Format this action will use to display the DateTime value. You can express a DateTime as, for example, MM/dd/yyyy for date, and hh:mm:ss. For more information, see the [Custom Date Formats](#) topic.

Example:

This is an example of how the action will display the DateTime value.

Store Result into:

Enter a name to be the variable that will store the formatted DateTime as a text value.

3.11.31.1 Create Random Text Action**Description:**

This action generates a text of specified length consisting of random characters. This can be useful for generating passwords.

Properties:

Characters to Use:

Choose which characters may be included in the generated Text.

Minimum Length:

Choose a minimum length for the random Text. If you want a certain length of Text, set the minimum and maximum values to that number.

Maximum Length:

Choose a maximum length for the random Text. If you want a certain length of Text, set the minimum and maximum values to that number.

Store Random Text into:

Enter a name to be the variable that will store the random Text.

3.11.31.1 Join Text Action

Description:

This action converts a list into a text value by separating its items with a specified delimiter.

Properties:

List to Join:

Enter a previously stored list variable, to be converted to Text.

Delimiter to separate list items:

Choose whether to use no Delimiter, a standard Delimiter or a custom one.

Standard Delimiter:

Choose the Delimiter and how many repetitions from the list.

Custom Delimiter:

Enter the character(s) to be used as Delimiter. Spaces at the beginning and the end will be trimmed, unless in the format of %"xxx"% , where xxx is the text, and can contain spaces anywhere.

Store Result into:

Enter a name to be the variable that will store the new, delimited Text.

3.11.31.1 Split Text Action

Description:

This action creates a list containing the substrings of a text that are separated by a specified delimiter or by regular expression

Properties of 'Split Text' action

Split Text
This action creates a list containing the substrings of a text that are separated by a specified delimiter or by regular expression

General

Action Input

Text to Split: %csvLine%

Delimiter that separates text elements: Custom Delimiter

Custom Delimiter: |

Is Regular Expression

Action Output

Store Result into: %TextList%

This action is Enabled

More Info OK Cancel

Properties:

Text to Split:

Enter the text that contains delimiters, or a previously stored text variable, to be split.

Delimiter that separates text elements:

Choose whether the used Delimiter is of a standard or custom format.

Standard Delimiter:

Choose the Delimiter used from the list.

Custom Delimiter:

Enter the character(s) that were used as Delimiter.

Is Regular Expression:

Check this box if your Delimiter is a Regular Expression. A Regular Expression creates a range of possibilities for the Delimiter. For example \d means that the Delimiter could be any digit.

Store Result into:

Enter a name to be the variable that will store the new List.

3.11.31.1 Parse Text Action***Description:***

This action parses a text to find the first or all occurrences of a specified subtext or a regular expression pattern

Properties of 'Parse Text' action

Parse Text
This action parses a text to find the first or all occurrences of a specified subtext or a regular expression pattern

General | Exception Handling

Action Input

Text to Parse: ⓘ ⚙

Text to Find: ⓘ ⚙

Is Regular Expression ⓘ

Start Parsing at Position: ⓘ ⚙

Find First Occurrence Only ⓘ

Ignore Case ⓘ

Action Output

Store Position(s) of found text into: ⓘ

Store Match(es) into: ⓘ

This action is Enabled

More Info OK Cancel

Properties:**Text to Parse:**

Enter the text, or a previously stored text variable, to be parsed.

Text to Find:

Enter the subtext, a previously stored text variable, or a Regular Expression, to search for.

Is Regular Expression:

Check this box if your subtext is a Regular Expression. A Regular Expression creates a range of possibilities to be the subtext. For example `\d` means that the subtext could be any digit.

Start Parsing at Position:

Enter the position where this action will begin to look for the Text to Find. The first position is zero, so use 0 to start from the beginning.

First Occurrence Only:

Choose if you want this action to find the first occurrence only, or each occurrence of the Text to Find.

Ignore Case:

Choose whether you want this action to find the specified text using case-sensitive or case-insensitive matching.

Store Position(s) of found text into:

Enter a name to be the variable that will store the position(s) of the "Text to Find" into the "Text to Parse". If you have not checked Find First Occurrence, the result(s) will be a list of numerical values; otherwise it will be a single numerical value. If the text you are searching for is not found within the original text, this variable will hold the value -1.

Store Match(es) into:

Enter a name to be the variable that will store the result(s) that match your Regular Expression. If you have not checked Find First Occurrence, the result(s) will be a list of text values; otherwise it will be a single text value.

3.11.31.1!Replace Text Action***Description:***

This action replaces all occurrences of a specified subtext, with another specified text. It can also be used with Regular Expressions

Properties of 'Replace Text' action

Replace Text
This action replaces all occurrences of a specified subtext, with another specified text. It can also be used with Regular Expressions

General

Action Input

Text to Parse: %FileContents%

Text to Find: %OldText%

Use Regular Expressions for find

Ignore Case

Replace With: %NewText%

Activate Escape Sequences

Action Output

Store result text into: %Replaced%

This action is Enabled

More Info OK Cancel

Properties:**Text to Parse:**

Enter the text, or a previously stored text variable, to be parsed.

Text to Find:

Enter the subtext, a previously stored text variable, or a Regular Expression, to search for.

Use Regular Expressions for find and replace:

Check this box if the subtexts are Regular Expressions. A Regular Expression creates a range of possibilities to be the subtext. For example \d means that the subtext could be any digit.

Ignore Case:

Choose whether you want this action to find the subtext to be replaced using case-sensitive or case-insensitive matching.

Replace With:

Enter the text, a previously stored text variable, or a Regular Expression, to replace found text. For example '\t' in the replacement text will be interpreted as a tab.

Activate Escape Sequences:

Check this box if you want special sequences.

Store Result Text into:

Enter a name to be the variable that will store the new, updated Text.

3.11.31.1 Escape Text for Regular Expression Action

Description:

This action escapes a minimal set of characters (\, *, +, ?, |, {, [, (, ^, \$, ., # and white spaces) by replacing them with their escape codes of an existing test.

Properties of 'Escape Text For Regular Expression' action

Escape Text For Regular Expression
 This action escapes a minimal set of characters (\, *, +, ?, |, {, [, (, ^, \$, ., #, and white space) by replacing them with their escape codes of an existing test.

General

Action Input

Text to Escape

Action Output

Store Escaped Text into:

This action is Enabled

More Info OK Cancel

Properties:

Text to Escape:

Enter the text, or a previously stored text variable, to be escaped.

Store Escaped Text into:

Enter a name to be the variable that will store the escaped text.

3.11.32 DateTime**3.11.32.1 Get Current Date and Time Action****Description:**

This action retrieves the current date or the current date and time

Properties of 'Get Current Date and Time' action

Get Current Date and Time
This action retrieves the current date or the current date and time

General

Action Input

Retrieve: Current Date and Time

Action Output

Store Retrieved DateTime into: %CurrentDateTime%

This action is Enabled

More Info OK Cancel

Properties:**Retrieve:**

Choose whether to gather the date and time, or just the date. If you choose to gather only the Date, the time value will be stored as midnight (0:00:00).

Store Retrieved DateTime into:

Enter a name to be the variable that will store the current DateTime value.

3.11.32.2 Add to DateTime Action

Description:

This action adds (or subtracts) a specific number of seconds, minutes hours or days to a DateTime value stored into a variable

Properties of 'Add to DateTime' action

Add to DateTime
This action adds (or subtracts) a specific number of seconds, minutes hours or days to a DateTime value stored into a variable

General

Action Input

Add: ⓘ ⚙️

To Variable: ⓘ ⚙️

Action Output

Store the result into: ⓘ

This action is Enabled

Properties:

Add:

Enter a numeric value, or a previously stored numeric variable, and choose the unit of time it represents. To subtract a time, this value would be negative. For example, add -7 days here to go back one week.

To Variable:

Enter the name of a variable with a DateTime value that you wish to alter.

Store the result into:

Enter a name to be the variable that will store the new, altered DateTime value. As elsewhere, this can be the name of the original variable, overwriting the original DateTime with the new DateTime.

3.11.32.3 Subtract Dates Action

Description:

This action finds the time difference between two given dates in days, hours, minutes, or seconds.

Properties of 'Subtract Dates' action

Subtract Dates
This action finds the time difference between two given dates in days, hours, minutes, or seconds.

General

Action Input

Subtract Date 1: ⓘ ⚙️

From Date 2: ⓘ ⚙️

Get Difference in: ⓘ

Action Output

Store Result into: ⓘ

This action is Enabled

Properties:

Subtract Date 1:

Enter a DateTime or the name of a variable that holds a DateTime value.

From Date 2:

Enter a DateTime or the name of a variable that holds a DateTime value. This will be the base DateTime, so generally put the later date/time in this field.

Get Difference in:

Choose what unit of time to express the difference in.

Store Result into:

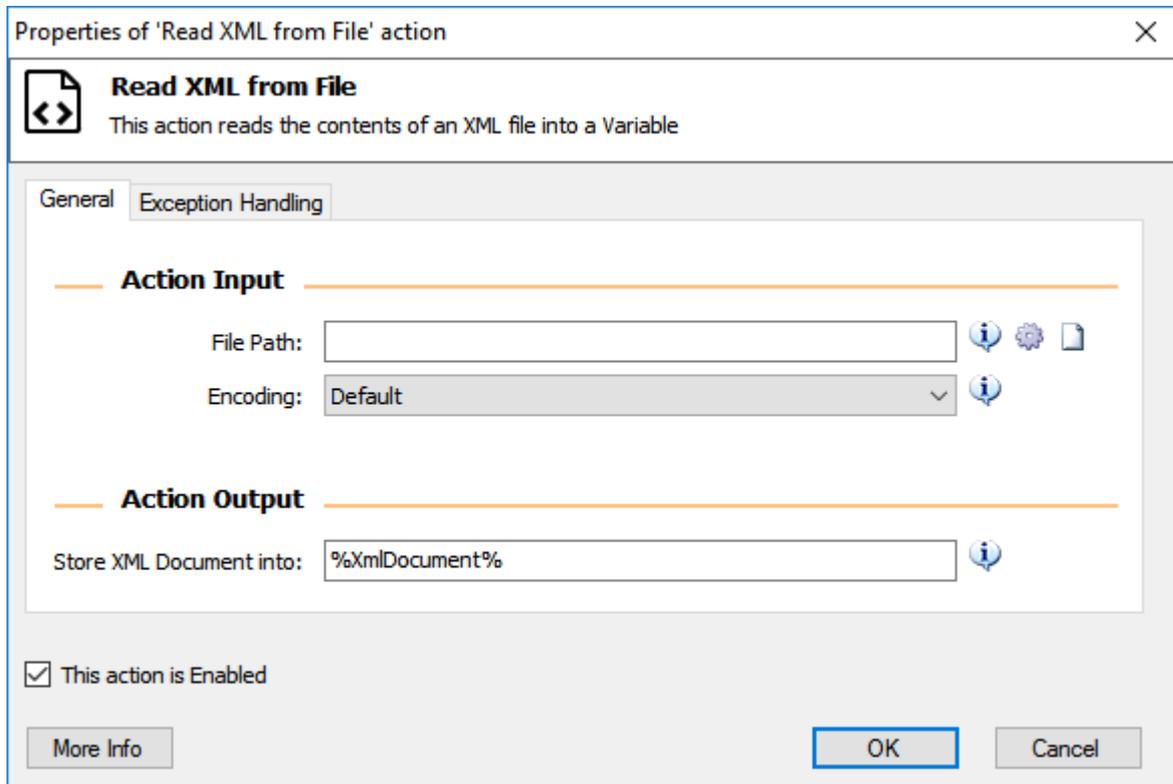
Enter a name to be the variable that will hold the difference in time as a numeric value.

3.11.33 XML

3.11.33.1 Read XML from File Action

Description:

This action reads the contents of an XML file into a variable.



The screenshot shows the 'Properties of 'Read XML from File' action' dialog box. The title bar reads 'Properties of 'Read XML from File' action'. The main area has a header 'Read XML from File' with a sub-header 'This action reads the contents of an XML file into a Variable'. Below this are two tabs: 'General' (selected) and 'Exception Handling'. The 'General' tab is divided into two sections: 'Action Input' and 'Action Output'. Under 'Action Input', there is a 'File Path:' text box with a file icon, an 'Encoding:' dropdown menu set to 'Default' with an information icon, and an 'Action Output' section with a 'Store XML Document into:' text box containing '%XmlDocument%' and an information icon. At the bottom, there is a checked checkbox 'This action is Enabled', a 'More Info' button, and 'OK' and 'Cancel' buttons.

Properties:

File path:

Enter or choose the XML document to be read. This can be a file path, or a variable containing a file or a textual path.

Encoding:

Select the encoding used for the specified file.

Store XML Document into:

Enter the name of the variable that will store the XML Document read from the file.

3.11.33.2 Write XML to File Action

Description:

This action writes the contents of an XML Node variable into a file.

Properties:

File path:

Enter or choose the file where the XML document to be written into.

XML to write:

Enter the variable that contains the XML Node or document to be written that you want to write into the file.

Encoding:

Select the encoding used for the specified file.

3.11.33.3 Execute XPath Expression Action

Description:

This action extracts values from an XML Document (a variable containing data in XML format) based on the XPath Query provided.

The screenshot shows the configuration window for the 'Execute XPath Expression' action. The window title is 'Properties of 'Execute XPath Expression' action'. It features a 'General' tab and an 'Exception Handling' tab. The 'General' tab is active and contains the following fields:

- Action Input:**
 - 'XML Document to Parse': A dropdown menu with the value '%XmlDocument%' and an information icon.
 - 'XPath Query': A text input field with an information icon and a settings gear icon.
 - 'Get First Value Only': A checkbox that is currently unchecked, with an information icon.
- Action Output:**
 - 'Store Extracted Value(s) into': A text input field with the value '%XPathResult%' and an information icon.

At the bottom of the dialog, there is a checked checkbox labeled 'This action is Enabled', a 'More Info' button, and 'OK' and 'Cancel' buttons.

Properties:**XML Document to Parse:**

This action displays an inobtrusive message through the notification pop-up

XPath Query:

Enter the XPath expression to execute against the XML Document.

Get First Value Only:

Specify whether you want to retrieve a single value (The first value only) or all the values that match the provided XPath expression.

Store Extracted Value(s) into:

Enter the name of the variable that will store the extracted node(s) as an XML Node or a list of XML Nodes (depending on your choice in "Get First Value Only" property)

3.11.33.4 Get XML Element Attribute Action**Description:**

This action gets the value of an attribute of an XML Element.

Properties of 'Get XML Element Attribute' action

Get XML Element Attribute
This action gets the value of an attribute of an XML Element

General Exception Handling

Action Input

XML Document: %XmlDocument% 

XPath Query:  

Attribute Name:  

Get Value as: Text Value 

Action Output

Store the XML Attribute Value into: %XmlAttributeValue% 

This action is Enabled

[More Info](#)

Properties:

XML Document:

Enter the variable that contains the XML Document or XML Element whose attribute you want to retrieve.

XPath Query:

Enter the XPath Expression to locate the sub-element whose attribute you want to retrieve.

Attribute Name:

Enter the name of the attribute whose value you want to retrieve.

Get Value as:

Select the data type for the attribute value.

Set the XML Attribute Value into:

Enter the name of the variable that will store the retrieved value of the XML Attribute. The type of the variable will depend on the selection of the “Get Value as” property.

3.11.33.5 Set XML Element Attribute Action

Description:

This action sets the value of an attribute of an XML Element.

The screenshot shows the configuration window for the 'Set XML Element Attribute' action. The window title is 'Properties of 'Set XML Element Attribute' action'. The main title is 'Set XML Element Attribute' with a description: 'This action sets the value of an attribute of an XML Element'. There are two tabs: 'General' (selected) and 'Exception Handling'. The 'Action Input' section contains four fields: 'XML Document' (a dropdown menu with '%XmlDocument%' selected), 'XPath Query' (an empty text box), 'Attribute Name' (an empty text box), and 'Attribute Value' (a large empty text area). Each field has an information icon (i) and a gear icon (settings). The 'Action Output' section is empty and contains the text '(This action does not provide any output)'. At the bottom, there is a checkbox labeled 'This action is Enabled' which is checked. There are three buttons: 'More Info', 'OK', and 'Cancel'.

Properties:**Xml Document:**

Enter the variable that contains the XML Document or XML Element whose attribute you want to set.

XPath Query:

Enter the XPath Expression to locate the sub-element whose attribute you want to set.

Attribute Name:

Enter the name of the attribute whose value you want to set.

Attribute Value:

Enter the new value for the attribute.

3.11.33.6 Remove XML Element Attribute Action**Description:**

This action removes an attribute from an XML Element.

Properties of 'Remove XML Element Attribute' action

Remove XML Element Attribute
This action removes an attribute from an XML Element

General Exception Handling

Action Input

XML Document: [%XmlDocument%] ⓘ

XPath Query: ⓘ ⚙

Attribute Name: ⓘ ⚙

Action Output

(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties:**XML Document:**

Enter the variable that contains the XML Document or XML Element whose attribute you want to remove.

XPath Query:

Enter the XPath Expression to locate the sub-element whose attribute you want to remove.

Attribute Name:

Enter the name of the attribute you want to remove.

3.11.33.7 Get XML Element Value Action**Description:**

This action gets the value of an XML Element.

Properties of 'Get XML Element Value' action

Get XML Element Value
This action gets the value of an XML Element

General Exception Handling

Action Input

XML Document: %XmlDocument% 

XPath Query:  

Get Value as: Text Value 

Action Output

Store the XML Element Value into: %XmlElementValue% 

This action is Enabled

[More Info](#)

Properties:

XML Document:

Enter the variable that contains the XML Document or XML Element whose value you want to retrieve.

XPath Query:

Enter the XPath Expression to locate the sub-element whose value you want to retrieve.

Get Value as:

Select the data type for the Element value

Store the XML Element Value into:

Enter the name of the variable that will store the retrieved value of the XML Element. The type of the variable will depend on the selection of the "Get Value as" property.

3.11.33.8 Set XML Element Value Action

Description:

This action sets the value of an XML Element.

Properties:

XML Document:

Enter the variable that contains the XML Document or XML Element whose value you want to retrieve.

XPath Query:

Enter the XPath Expression to locate the sub-element whose value you want to retrieve.

XML Element Value:

Enter the Value for the XML Element

3.11.33.9 Insert XML Element Action

Description:

This action inserts a new XML Element into an XML document

Properties of 'Insert XML Element' action

Insert XML Element
This action inserts a new XML Elements into an XML Document

General Exception Handling

Action Input

XML Document: %XmlDocument%

XPath Query:

XML Element to Insert:

Action Output

(This action does not provide any output)

This action is Enabled

[More Info](#)

Properties:**XML Document:**

Enter the XML Document where you want to insert the new XML Element.

XPath Query:

Enter the XPath Expression to locate the parent XML element into which the new element will be inserted.

XML Element to Insert:

Enter the new XML element you want to insert into the XML document.

3.11.33.1 Remove XML Element Action**Description:**

This action removes one or more XML Elements from an XML Document.

Properties of 'Remove XML Element' action

Remove XML Element
This action removes one or more XML Elements from an XML Document

General Exception Handling

Action Input

XML Document: %XmlDocument%

XPath Query:

Action Output
(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties:**XML Document:**

Enter the XML Document that contains the XML element(s) you want to remove.

XPath Query:

Enter the XPath expression to locate the element(s) you want to remove.

3.11.34 Variables**3.11.34.1 Set Variable Action****Description:**

This action sets the value of a new or existing variable, and can be used to create a new variable or overwrite a previously created variable.

Properties of 'Set Variable' action

Set Variable
This action sets the value of a new or existing variable, and can be used to create a new variable or overwrite a previously created variable.

General

Action Input

Set Value:

Action Output

into Variable:

This action is Enabled

More Info OK Cancel

Properties:**Set Value:**

Enter a value or expression, or a previously stored variable, to be the value of this variable.

into Variable:

Enter a name to be the variable that will hold the value you set. This can be a new or existing variable.

3.11.34.2 Increase Variable Action

Description:

This action increases the value of a variable by a specific amount

Properties of 'Increase Variable' action

Increase Variable
This action increases the value of a variable by a specific amount

General

Action Input

Variable Name: ⓘ

Increase by: ⓘ ⚙️

Action Output

(This action does not provide any output)

This action is Enabled

Properties:

Variable Name:

Enter the name of the numeric variable you wish to increase.

Increase by:

Enter a numeric value, or a previously stored numeric variable, to increase the variable by.

3.11.34.3 Decrease Variable Action

Description:

This action decreases the value of a variable by a specific amount

Properties of 'Decrease Variable' action

Decrease Variable
This action decreases the value of a variable by a specific amount

General

Action Input

Variable Name: %Sum%

Decrease By: 1

Action Output
(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties:

Variable Name:

Enter the name of the numeric variable you wish to decrease.

Decrease By:

Enter a numeric value, or a previously stored numeric variable, to decrease the variable by.

3.11.34.4 Truncate Number Action

Description:

This action accepts a numeric value and returns its integral or fractional digits, or rounds up the value to a specified number of decimal places.

Properties of 'Truncate Number' action

Truncate Number
 This action accepts a numeric value and returns its integral or fractional digits, or rounds up the value to a specified number of decimal places.

General

Action Input

Number to Truncate: ⓘ ⚙️

Operation: ⓘ

Decimal Places: ⓘ

Action Output

Store Truncated Value into: ⓘ

This action is Enabled

Properties:**Number to Truncate:**

Enter the number or the name of the variable that holds the numeric value you want to truncate/round up.

Operation:

Select the operation you want to perform on the given number.

Decimal Places:

Specify the number of decimal places you want to round the given number up. Enter 0, do you want the result to be an integer.

3.11.34.5 Generate Random Number Action**Description:**

Generates a random number or a list of random numbers that fall between a minimum and maximum value

Properties of 'Generate Random Number' action

Generate Random Number
1542 Generates a random number or a list of random numbers that fall between a minimum and maximum value

General

Action Input

Minimum Value: ⓘ ⚙️

Maximum Value: ⓘ ⚙️

Generate Multiple Numbers: ⓘ

How Many Numbers: ⓘ ⚙️

Allow Duplicates ⓘ

Action Output

Store Random Number(s) into: ⓘ

This action is Enabled

Properties:**Minimum Value:**

Enter a numeric value, or a previously stored numeric variable, to specify a lower boundary for the random number(s) to generate.

Maximum Value:

Enter a numeric value, or a previously stored numeric variable, to specify an upper boundary for the random number(s) to generate.

Generate Multiple Numbers:

Specify whether you want to generate a single random number or a list of random numbers.

How Many Numbers:

Enter a numeric value, or a previously stored numeric variable, to specify how many random numbers to generate.

Allow Duplicates:

Specify whether to permit or prevent the same number from appearing more than once in the random numbers list.

Store Random Number(s) into:

Enter a name to be the variable that will store the newly generated random number(s). The variable will hold a numeric value or a list of numeric values, depending on your choice in 'Generate Multiple Numbers'

3.11.34.6 Get Items Count Action

Description:

This action retrieves the count of items of a variable that contains a list or a datatable. A datatable will return the number of rows.

Properties of 'Get Items Count' action

Get Items Count
This action retrieves the count of items of a variable that contains a list or a datatable. A datatable will return the number of rows.

General

Action Input

Variable Name:

Action Output

Store Items Count into:

This action is Enabled

More Info OK Cancel

Properties:

Variable Name:

Enter the name of the list or datatable variable to be counted.

Store Items Count into:

Enter a name to be the variable that will hold the count of items on the list or the number of rows in the datatable.

3.11.34.7 Create New List Action

Description:

This action creates a new empty list and assigns it to a variable

Properties of 'Create New List' action

Create New List
This action creates a new empty list and assigns it to a variable

General

Action Input
(This action does not accept any input)

Action Output
Store new empty list into:

This action is Enabled

More Info OK Cancel

Properties:

Store new empty list into:

Enter a name to be the variable that will hold the list.

3.11.34.8 Clear List Action

Description:

This action removes all items from a List contained into a variable.

Properties of 'Clear List' action

Clear List
This action removes all items from a List contained into a variable.

General

Action Input

List to Clear: ⓘ ⚙

Action Output

(This action does not provide any output)

This action is Enabled

Properties:

List to Clear:

Enter the name of a variable that contains the list you wish to remove its items.

3.11.34.9 Add Item to List Action

Description:

This action appends a new item of the same type to a variable that contains a List.

Properties:

Add Item:

Enter a value, or the name of a variable, to be added. The new value must be a single value and not a list of values. If the list you are adding to has a specific type of elements, say all numeric or all file information, the new value must be of the same type.

into List:

Enter the name of a variable that contains a list you wish to add to.

3.11.34.1 Remove Item from List Action

Description:

This action removes an item at a specified index from a variable that contains a List.

Properties of 'Remove Item from List' action

Remove Item from List
This action removes an item at a specified index from a variable that contains a List.

General | Exception Handling

Action Input

Remove Item at Index: 3

From List stored into: %NewListVar%

Action Output
(This action does not provide any output)

This action is Enabled

More Info OK Cancel

Properties:**Remove Item at Index:**

Enter the index number of the item you wish to remove. This action will shorten the list by that item.

From List stored into:

Enter the name of a variable that contains a list you wish to remove the item from.

3.11.34.1 Sort List Action**Description:**

This action sorts the items of a List contained in a variable. The items on the list must be of the same type.

Properties of 'Sort List' action

Sort List
This action sorts the items of a List contained in a variable. The items on the list must be of the same type.

General

Action Input

List to Sort: ⓘ ⚙

Sort by List Items' properties ⓘ

First Property to Sort by: Ascending ⓘ

Second Property to Sort by: Ascending ⓘ

Third Property to Sort by: Ascending ⓘ

Action Output

(This action does not provide any output)

This action is Enabled

Properties:

List to Sort:

Enter the name of a variable that contains a list you wish to sort.

Sort by list items' properties:

If your list items are objects (such as files, folders, etc) you can choose here to sort the item by a specific property. If you leave this unchecked the items of the list will be sorted by their default property (e.g. file objects will be sorted by their full path).

So, for example, if you have a list of files, using this option you can sort them, say, by extension and then by size. If you don't use the option, the files will be sorted by their full path.

First Property to Sort by:

Enter the name of a property of the items contained in the list that you want to sort by. You can find the properties for some of the data types supported by ProcessRobot [here](#)⁴³⁹.

Second Property to Sort by:

Optional: Enter the name of a second property to sort by.

Third Property to Sort by:

Optional: Enter the name of a second property to sort by.

3.11.34.1: Shuffle List Action

Description:

This action creates a random permutation of a List contained in a variable. The items on the list must be of the same type.

The screenshot shows a dialog box titled "Properties of 'Shuffle List' action". It features a "Shuffle List" icon and a description: "This action creates a random permutation of a List contained in a variable. The items on the list must be of the same type." Below this is a "General" tab with an "Action Input" section containing a text field for "List to Shuffle:" with the value "%NewListVar%". An "Action Output" section below it states "(This action does not provide any output)". At the bottom, there is a checked checkbox "This action is Enabled", a "More Info" button, and "OK" and "Cancel" buttons.

Properties:

List to Shuffle:

Enter the name of a variable that contains a list you wish to shuffle.

3.11.34.1: Merge Lists

Description:

This action merges two lists into one.

Properties:**First List:**

Enter the name of a variable that is the first list of items to be merged.

Second List:

Enter the name of a variable that is the second list of items to be merged.

Output List:

Enter the name of the variable that will contain the merged list. The initial lists will not be affected.

3.11.34.1 Reverse List Action**Description:**

This action reverses the order of the items of a List contained into a variable.

The screenshot shows a dialog box titled "Properties of 'Reverse List' action". At the top, there is a header with a list icon and the text "Reverse List" and "This action reverses the order of the items of a List contained into a variable." Below this is a "General" tab. Under "Action Input", there is a text field labeled "List to Reverse:" containing the value "%OutputList%". To the right of the field are information and settings icons. Under "Action Output", there is a note: "(This action does not provide any output)". At the bottom left, there is a checked checkbox "This action is Enabled". At the bottom right, there are "More Info", "OK", and "Cancel" buttons.

Properties:

List to Reverse:

Enter the name of the variable that contains the List, whose items order you want to reverse.

3.11.34.1: Remove Duplicate Items from List Action

Description:

This action removes the multiple occurrences of items in a list, so that in the resulting list each item will be unique.

The screenshot shows a dialog box titled 'Properties of 'Remove Duplicate Items from List' action'. The dialog has a close button (X) in the top right corner. Below the title bar, there is a small icon of a list and the text 'Remove Duplicate Items from List'. A description follows: 'This action removes the multiple occurrences of items in a list, so that in the resulting list each item will be unique.' Below this is a 'General' tab. Under the 'Action Input' section, there is a text field labeled 'List to Remove Duplicate Items from:' containing the value '%OutputList%'. To the right of the text field are two icons: an information icon (i) and a gear icon. Below the text field is a checked checkbox labeled 'Ignore Text Case while searching for Duplicate Items' with an information icon (i) to its right. Under the 'Action Output' section, there is a note: '(This action does not provide any output)'. At the bottom left, there is a checked checkbox labeled 'This action is Enabled' and a 'More Info' button. At the bottom right, there are 'OK' and 'Cancel' buttons.

Properties:

List to Remove Duplicate Items from:

Enter the name of a variable that contains a list you wish to remove duplicate items from.

3.11.34.1 Find Common List Items Action

Description:

This action compares two lists, and creates a new list with the items that are common to both.

Properties of 'Find Common List Items' action

Find Common List Items
This action compares two lists, and creates a new list with the items that are common to both.

General

Action Input

List 1: %MyList_1%

List 2: %MyList_2%

Action Output

Store Common Items into List: %IntersectionList%

This action is Enabled

More Info OK Cancel

Properties:**List 1:**

Enter the name of a variable containing a list, to be compared.

List 2:

Enter the name of a variable containing a different list, to be compared.

Store Common Items into List:

Enter a name to be the variable that will hold the new list of common items.

3.11.34.1 Subtract Lists Action**Description:**

This action compares two lists, and creates a new list with the items that are in the first list but not in the second.

Properties of 'Subtract Lists' action

Subtract Lists
This action compares two lists, and creates a new list with the items that are in the first list but not in the second.

General

Action Input

List 1: ⓘ ⚙️

List 2: ⓘ ⚙️

Action Output

Store difference as a new List into: ⓘ

This action is Enabled

Properties:**List 1:**

Enter the name of a variable containing a list to be compared.

List 2:

Enter the name of a variable containing a different list to be subtracted from the first list. Subtracted items will shorten the resultant list. If two identical lists are compared, the result will be an empty list.

Store difference as new List into:

Enter a name to be the variable that will hold the new, resulting list.

3.11.34.1 Retrieve DataTable Column into List Action**Description:**

This action converts the contents of a DataTable column into a List for further processing

Properties of 'Retrieve DataTable Column into List' action

Retrieve DataTable Column into List
This action converts the contents of a DataTable column into a List for further processing

General Exception Handling

Action Input

DataTable: %ExcelTable%  

Column Name or Index: 0  

Action Output

Store column contents as new List into: %ColumnAsList% 

This action is Enabled

[More Info](#)

Properties:

DataTable:

Enter the name of a variable containing a data table.

Column Name or Index:

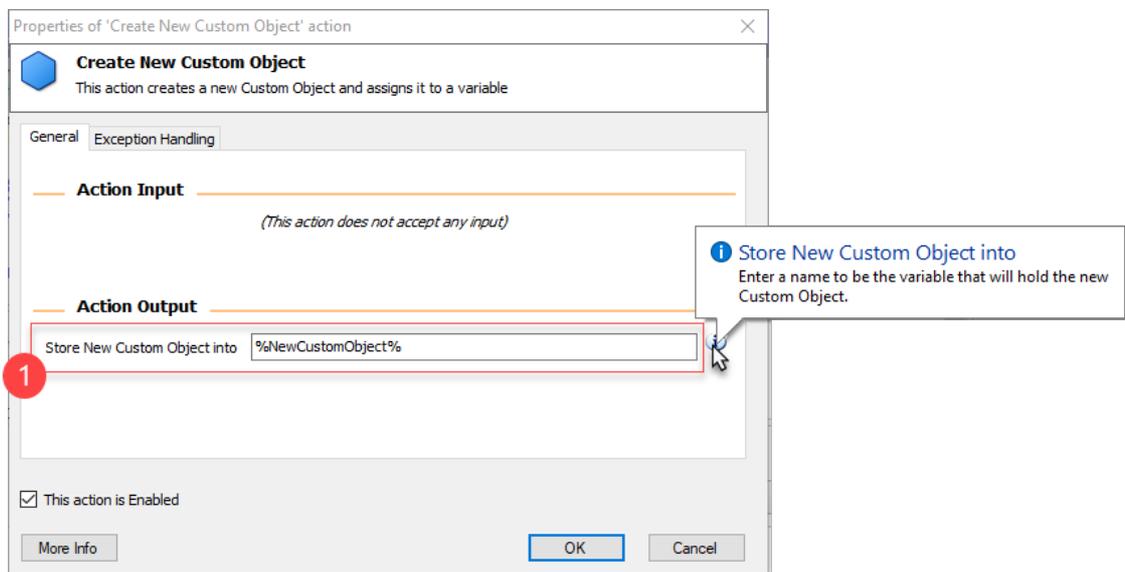
Enter the column name, if you have defined column names, or the index number of the column you wish to retrieve. Note that the index is 0 based.

Store column contents as new list into:

Enter a name to be the variable that will hold the new list. This list will hold the contents of the specified data table.

3.11.34.1 Create New Custom Object

This action creates a new [Custom Object](#)^[446] and stores it **[1]** into a variable for later use:



This action does not accept any input.

1

Store New Custom Object into:

This text field invites you to type the variable you wish to hold the new [Custom Object](#)^[446].

3.11.34.2 Add Property To Custom Object

This action allows you to add *pairs* of custom object Property Names **[2]** and Property Values **[3]** to an existing [Custom Object](#)^[446]. What is noteworthy is that the format of the action allows you to work with a number of available custom objects through the drop down menu of the Custom Object **[1]** action property:

Properties of 'Add Property To Custom Object' action

Add Property To Custom Object
This action adds a new property on the given Custom object.

General Exception Handling

Action Input

1 Custom Object: %NewCustomObject%

2 Property Name: servers

3 Property Value: %NewListVar%

Action Output
(This action does not provide any output)

This action is Enabled

More Info OK Cancel

This action does not provide any output.

1 Custom Object:

This is a text-field with drop down menu options. It allows you to enter the Name of the Custom Object you want to work with, or simply find and select it effortlessly from the drop down menu that lists all available custom objects in your script.

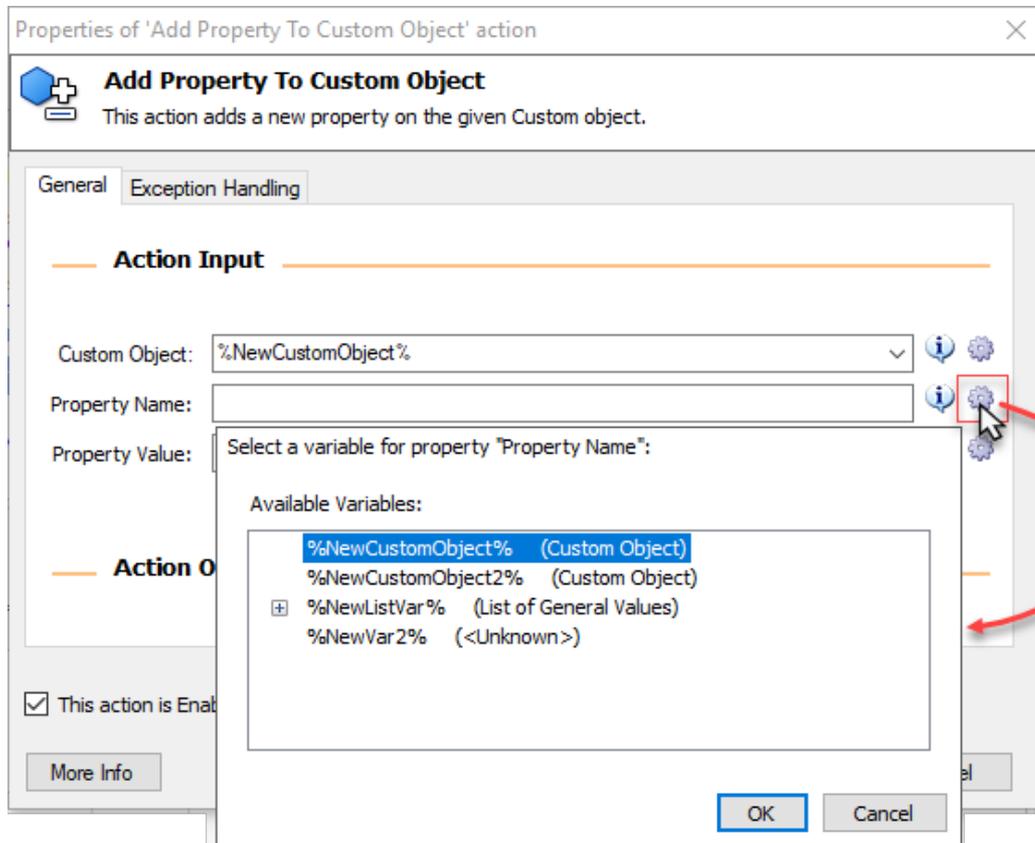
Custom Object: %NewCustomObject%

%NewCustomObject%

%NewCustomObject2%

2 Property Name:

This is a text-field that allows you to enter the new Property Name of the Custom Object of your choice [1]. If you have any questions on what is the Property Name of a Custom Object, please refer to the *Custom Object format* of the following [article](#)^[446]. Please also know, that you can always use the Gear Icon to enter Property Name as a variable:



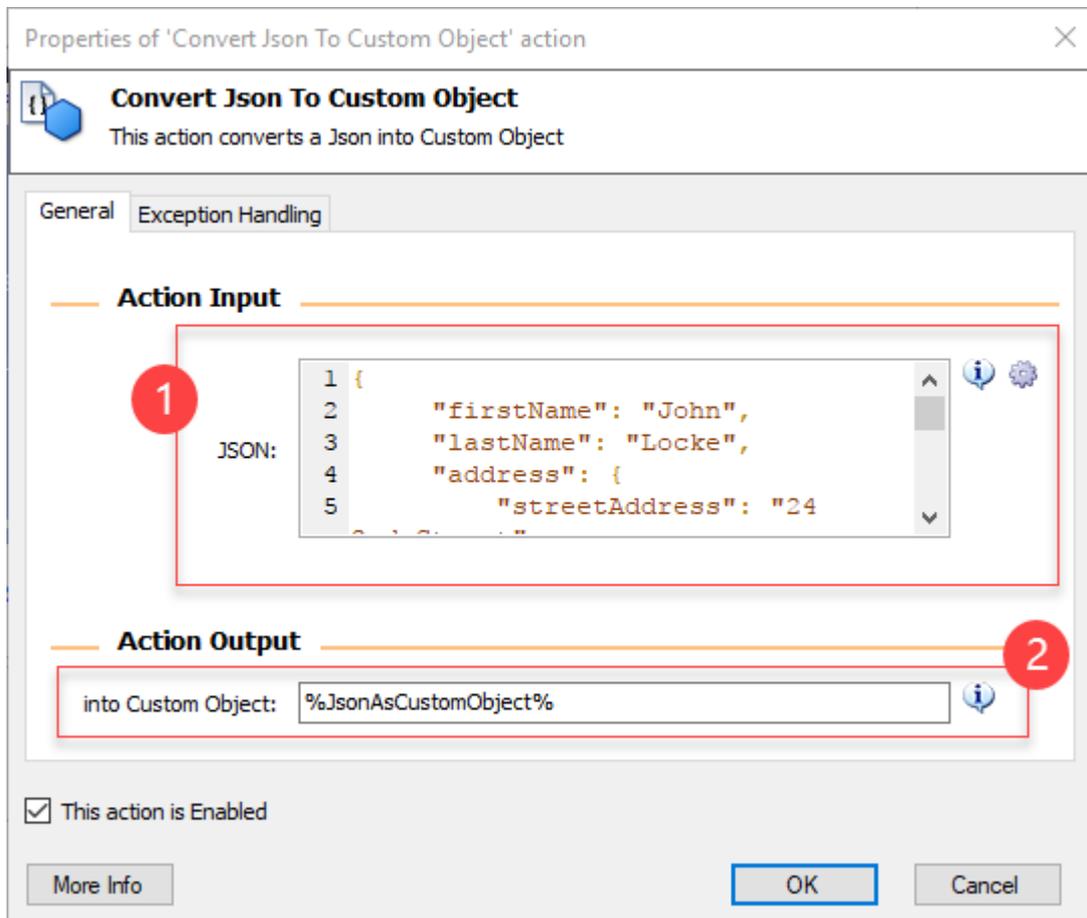
3

Property Value:

This is a text-field that allows you to enter the Property Value of the previously filled Property Name [2]. If you have any questions on what is Property Name or Property Value of a Custom Object, please refer to the *Custom Object format* of the following [article](#)^[446]. Please also know, that you can always use the Gear Icon to enter this Property Value as a variable. This can be very helping in case you want to use a data type that you have constructed earlier on in your script.

3.11.34.2 Convert Json to Custom Object

This action is giving you the ability to translate JSON [1] into a Custom Object and store it inside a new variable for later use [2]:



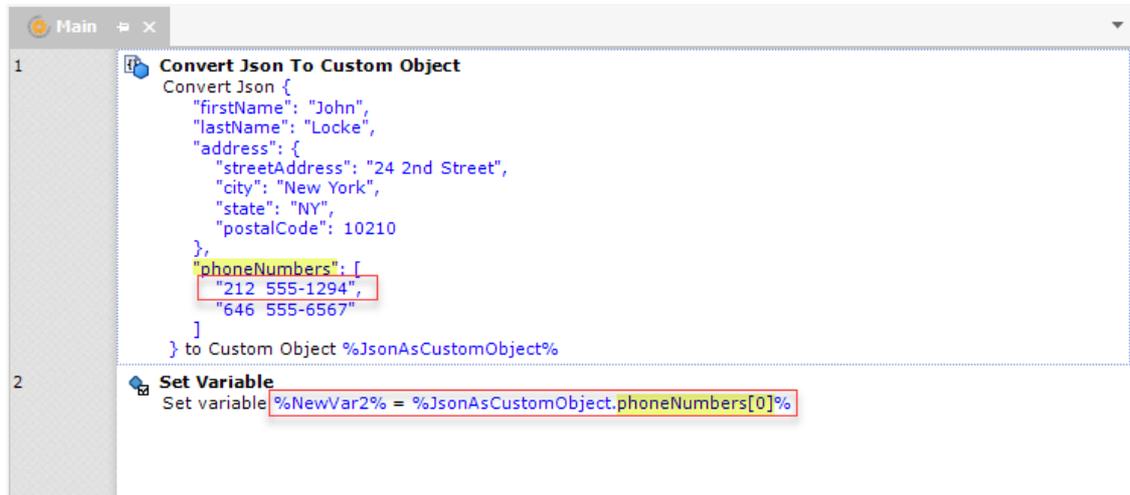
1 JSON:

This action property allows you to enter either a multiple lines long JSON directly or through a Process Robot variable. If you have any questions on what JSON or Custom Objects are, please refer to the appropriate section of the following [topic](#)⁴⁴⁶. Please also know, that you can always use the Gear Icon to enter effortlessly an existing variable.

2 Into Custom Object:

In this text-field you should enter the name of the existing or new variable that you wish to store the Custom Object.

Please note that you can access the values of your custom object almost exactly like if they were JSON. Consider for example the following actions in Process Designer's [Workspace](#) ^[286]:



[Testing this from Process Designer](#) ^[315] will produce the following variable value for `%NewVar2%`:

The Variables panel shows the following table:

Name	Type	Value
<code>%NewVar2%</code>	Text Value	212 555-1294
<code>%CustomObjectAsJson%</code>	Text Value	{ "has_wife": true }
<code>%JsonAsCustomObject%</code>	Custom Object	{ "firstName": "John", "lastName": "Locke", "..."

A red arrow points to the `%NewVar2%` row.

3.11.34.2: Convert Custom Object to Json

This action enables you to convert a Custom Object ^[1] to a JSON and store it into a Process Robot variable ^[2] for later use:

Properties of 'Convert Custom Object To Json' action

Convert Custom Object To Json
This action converts a Custom Object into a Json

General Exception Handling

Action Input

Custom Object: ⓘ ⚙

- %NewCustomObject%
- %JsonAsCustomObject%
- %NewCustomObject%

Action Output

into JSON: ⓘ

This action is Enabled

More Info OK Cancel

1

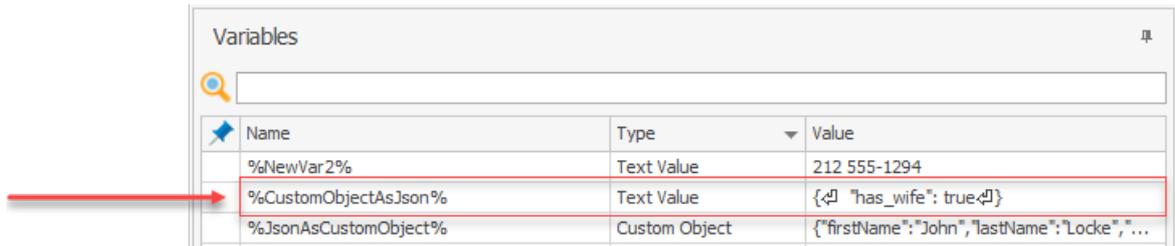
Custom Object:

This is a text-field with drop down menu options. It allows you to enter the Name of the Custom Object you want to work with, but also find it and select it effortlessly from the drop down menu that lists all available custom objects in your script.

2

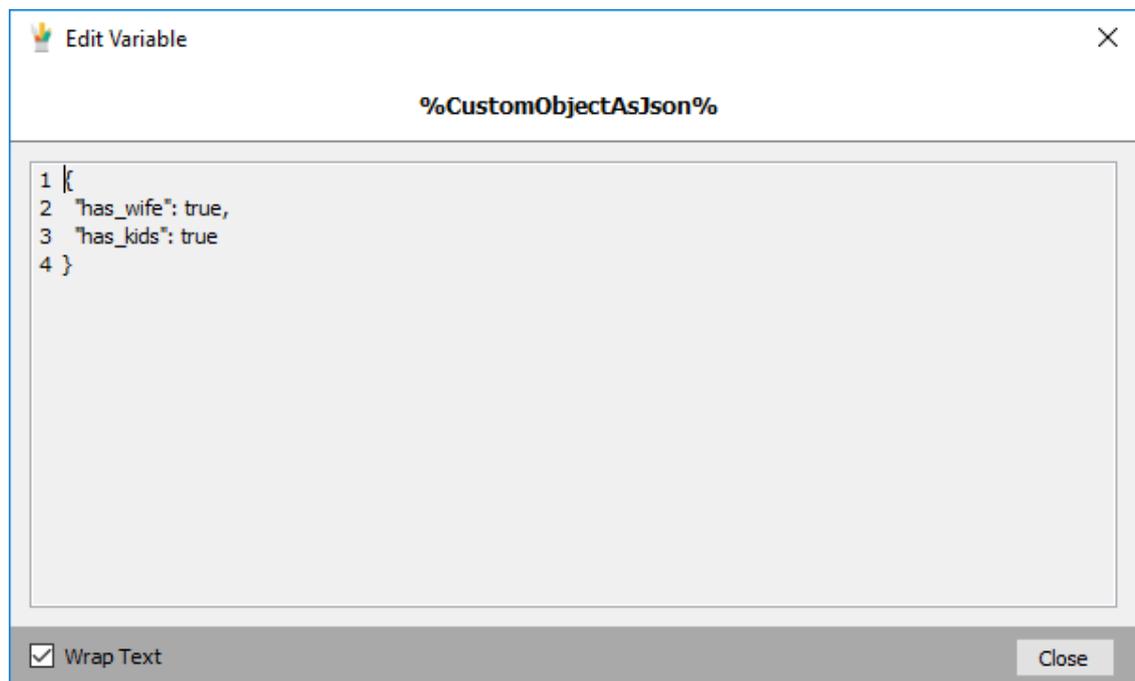
Into JSON:

JSON is a text format, so it should not be a surprise that this action returns a Text Value. This text field however allows you to enter the name of the variable (new or existing) in which you want to store the JSON Text Value. Also, in case you wish to work with that Text Value, please do note that it is JSON formatted, containing line breaks:

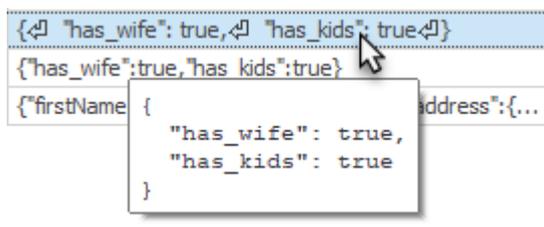


Name	Type	Value
%NewVar2%	Text Value	212 555-1294
%CustomObjectAsJson%	Text Value	{ "has_wife": true, "has_kids": true }
%JsonAsCustomObject%	Custom Object	{ "firstName": "John", "lastName": "Locke", "...

Please also note that **double clicking** on the `{ "has_wife": true, "has_kids": true }` Value on the Image above will produce the following pop-up (which is the [Variables Visualizer Window](#)^[290]), showing us the format of the returned value.



Hovering over the `{ "has_wife": true, "has_kids": true }` Value will produce the following pop-up that also allows you to develop a quick feeling of your variable value, without of course the capacity to edit your variable:



3.11.35 User Actions

If any User Actions have been created through the [User Library Designer](#)³⁹⁴ you should expect to find them here In the sequence of action in the Process Designer..

4. Sidebot

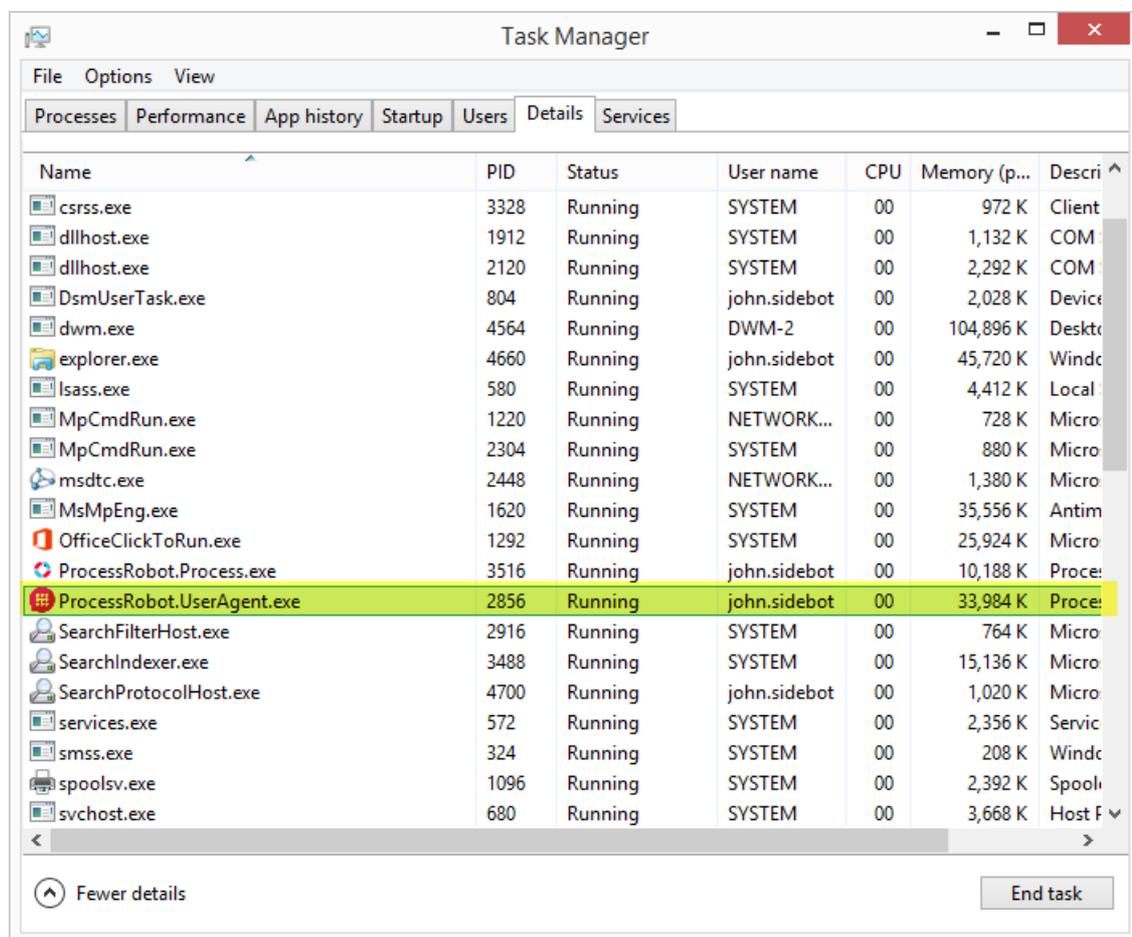
4 Sidebot

4.1 Sidebot Introduction

A Sidebot should be installed on a human user's workstation to run along with the user's presence.

A Process can be invoked by a trigger, a schedule, a Hotkey or manually by the user himself.

On the machine where a Sidebot resides, there is no service installed. As in the Solobot, however, it will be created "C:\Program Files\ProcessRobot\Clients" and while the Sidebot is up and running in the Task Manager you would be able to see the ProcessRobot.UserAgent.exe only.

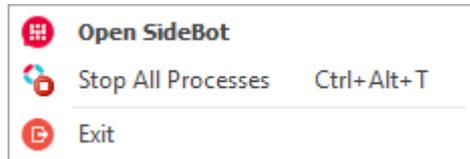


Sidebot Task Manager

On the machines where a Solobot or a Sidebot is installed the icon will be available in the system tray indicating that the Robot is up or not. If it is up it will be red, if not it will be grayed out.



Clicking on the system tray icon on a machine with a Sidebot you can:

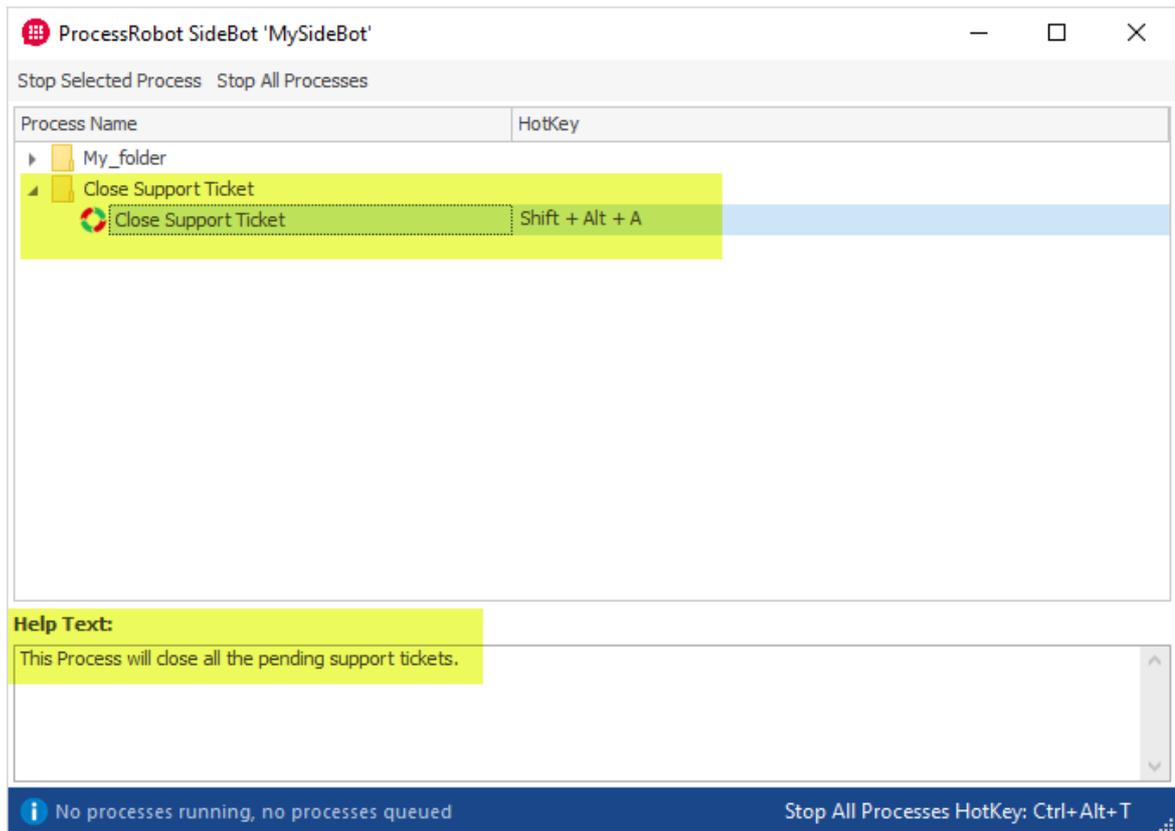


- **"Open Sidebot"**: This will open the [Sidebot Interface window](#)^[1090]
- **"Stop All Processes"**: This option will stop all the processes that are running at this point or hit the **Ctrl+Alt+T** hotkey.
- **"Exit"**: This option will exit the Sidebot and kill the ProcessRobot.UserAgent.

4.2 Sidebot Interface Window

If you double click on the Sidebot icon in the system Tray or if you right click and select "Open Sidebot" then you will have the Sidebot Interface window pop-up.

- 1. Process Name:** List of all the Processes that have been assigned to this specific Sidebot. Clicking on the Process Name column will rearrange the Processes. Double Clicking on a Process it will execute!
- 2. Hotkey:** If the Process has been assigned a specific Hotkey during its [Deployment](#)^[131] then the Hotkey will appear here for the user to see.
- 3. Help Text:** The Help Text that you have written in the [Deployment](#)^[131] tab of the Process Properties will be visible here, so to help the user understand with a bit more details what this Process does.



Sidebot Interface Window

4.3 Deploy a Process to a Sidebot

Why?

Process Robot (pr) is an Enterprise Grade RPA that is easy to scale and fit in your corporate structure. In order to allow employees that might not have access to Process Studio and the Control Desk to use distinct Processes designed to run on specific Pools of Robots, PR is providing every Sidebot with a customized interface that communicates with the Control Desk, the [Sidebot Interface Window](#)¹⁰⁹⁰.

This topic is all about how Administrators (people with access to Control Desk), can deploy processes to the Sidebot(s) Interface Window(s) of one or more users, en masse.

How?

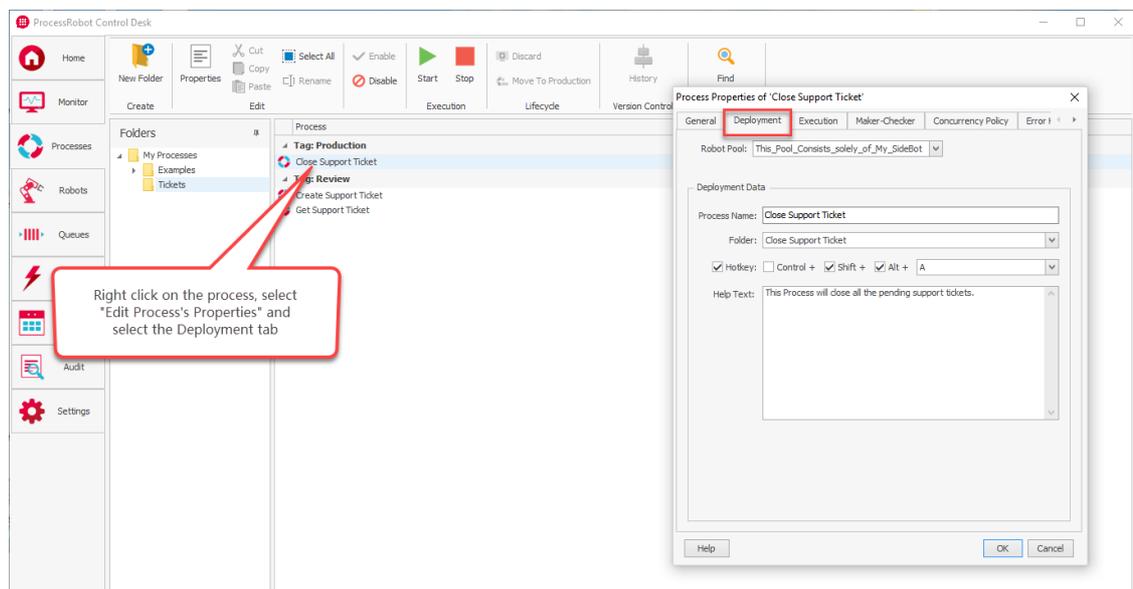
In order to deploy a Process to a (number of) Sidebot(s) first of all you would have to create the appropriate Robot Pool.

For creating a Robot Pool, please refer to the [Robot Pools](#)^[217] topic. Once the Pool that contains all the desired Sidebots is in place, then you can go to the Process you wish to be deployed, in the Control Desk.

The second thing you need to do in order to deploy a Process to a Sidebot, is to make sure that this Process is in Production:

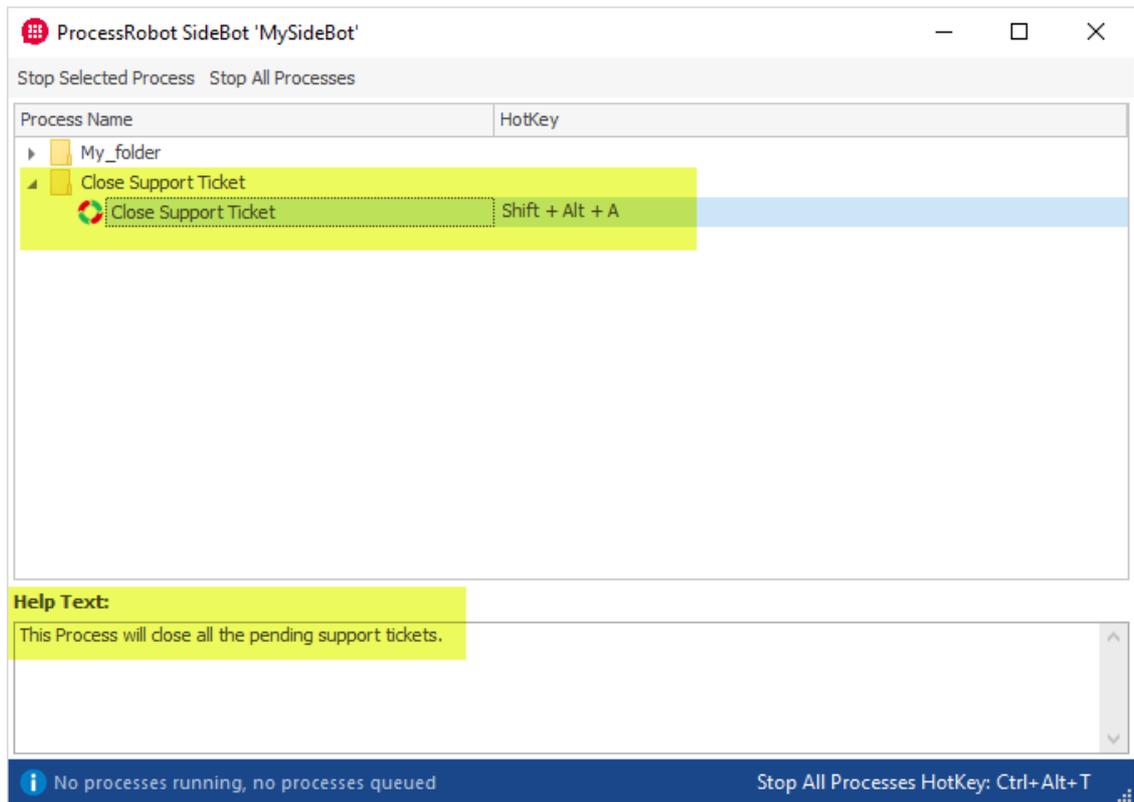
Process	Enabled	Triggers	Erroneous
<ul style="list-style-type: none"> Tag: Production <ul style="list-style-type: none"> Close Support Ticket Tag: Review <ul style="list-style-type: none"> Create Support Ticket Get Support Ticket 	True	None	

Then, right click on the Process and select "Edit Process's Properties". Select the Pool, Process Name, Folder under which it will be organized in the [Sidebot Interface Window](#)^[1090], Hotkey (for lightning fast triggered invocation) and the Help Text related to this Process.



Deploy a Process to Sidebot(s)

Once you click OK, then the Process will appear in all of the Pool's Sidebots.



And now you are ready to run

Please note: A Process can ONLY be deployed to a Pool if it is in the stage of Production. If a Process is [in Review](#)⁴⁵⁵, it can only run from the Control Desk.

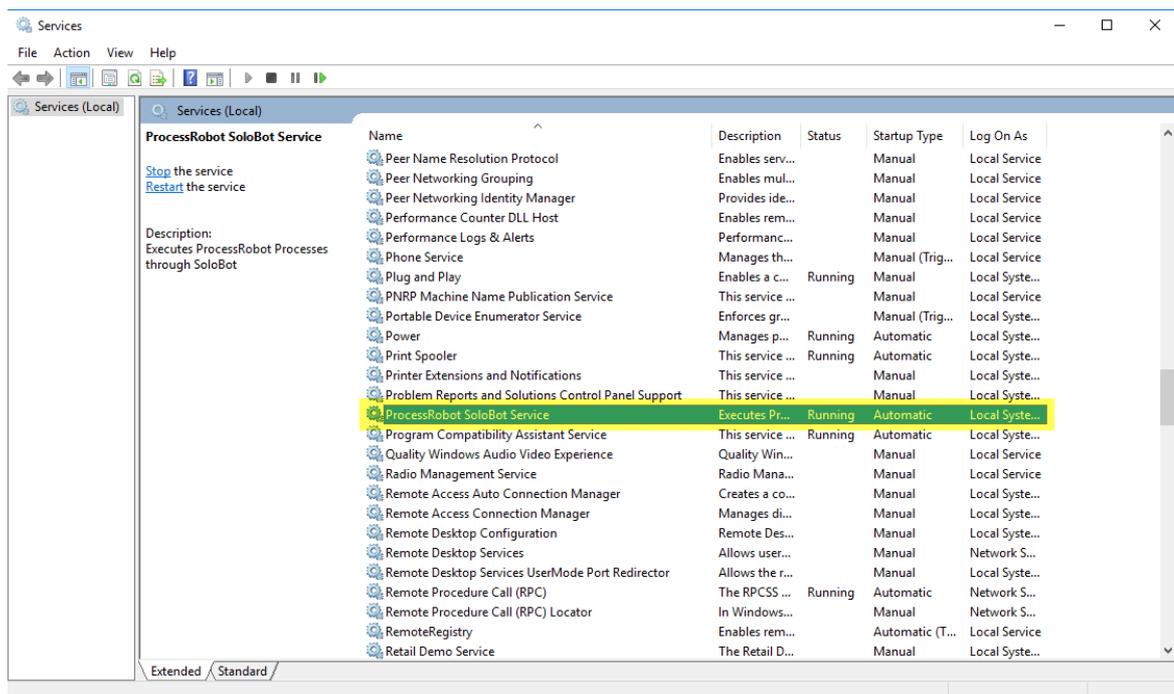
5. Solobot

5 Solobot

5.1 Solobot Introduction

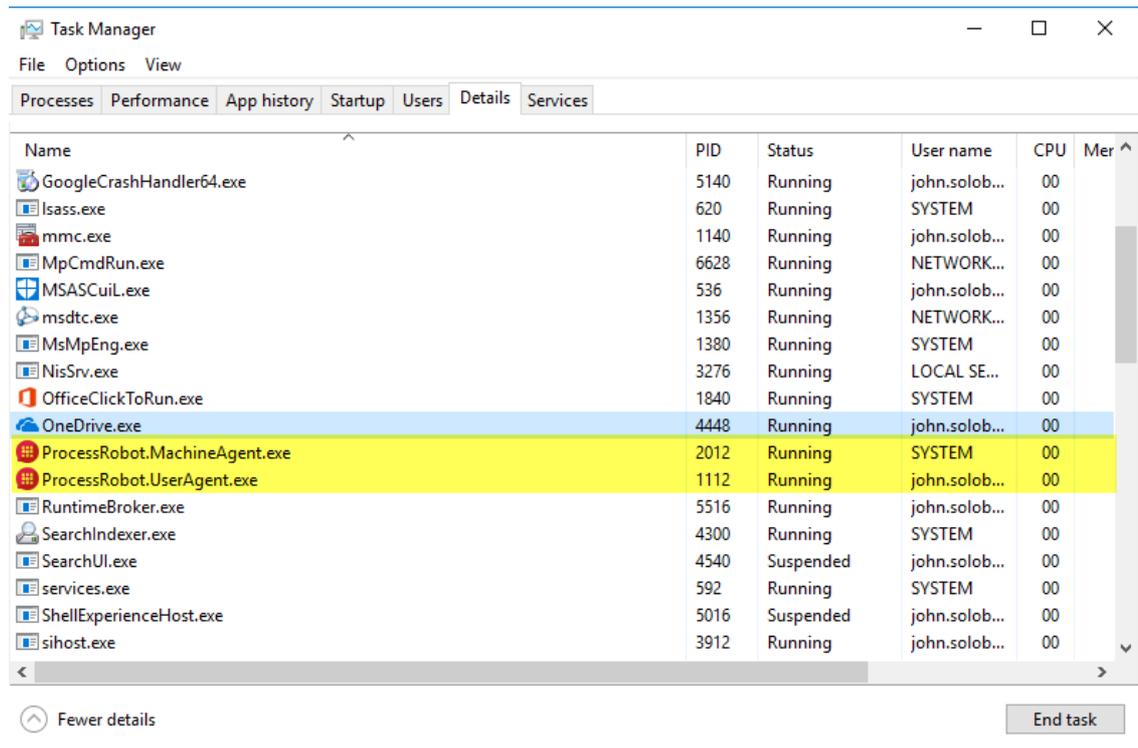
A Solobot is a dedicated machine that will have Processes assigned to run on it. The Solobot resides to the workstation and it is running completely unattended as it is able to login and logout by itself, based on the settings of the Processes. In contrast with a Sidebot, a Solobot has no "Solobot window" interface.

On the machine where a Solobot is installed, in the "Services" window one will be able to see that the ProcessRobot Solobot Service is installed and should be running. Also, on any machine that a Solobot or a Sidebot is installed the following folder will be created "C:\Program Files\ProcessRobot\Clients".



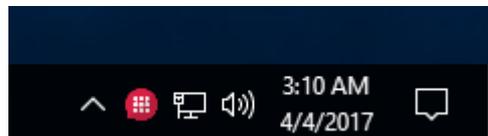
Services on the Solobot Machine

In the Task Manager on a Solobot machine you should see in the details "ProcessRobot.MachineAgent.exe" and "ProcessRobot.UserAgent.exe"



Solobot Task Manager

On the machines where a Solobot or a Sidebot is installed, the icon will be available in the system tray indicating that the Robot is up or not. If it is up it will be red, if not it will be grayed out.



Clicking on the system tray icon you can either "Exit" the Solobot or "Stop all running processes". By exiting the Solobot it is like you are turning it down and no processes will be able to run until its service is up and running again. By stopping all the processes everything that is running at that point will be stopped.

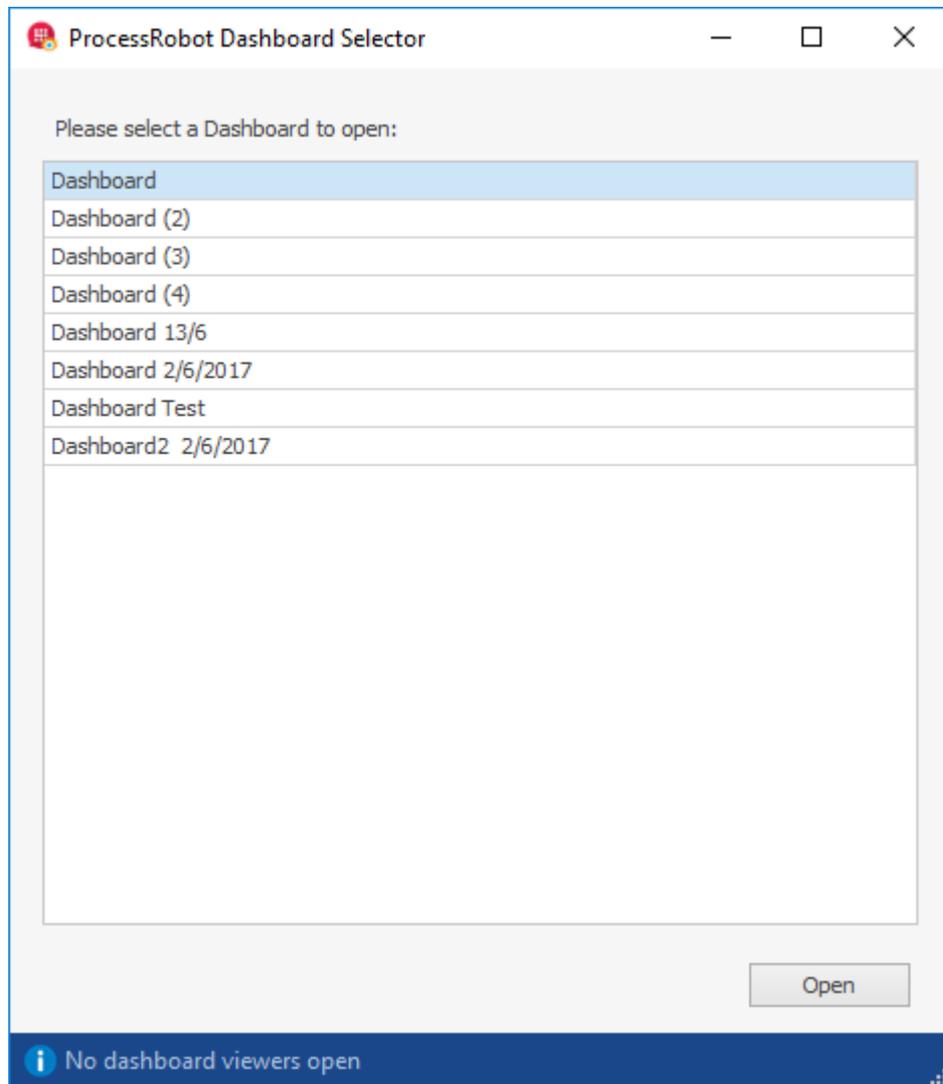
6. Insights Dashboards

6 Insights Dashboards

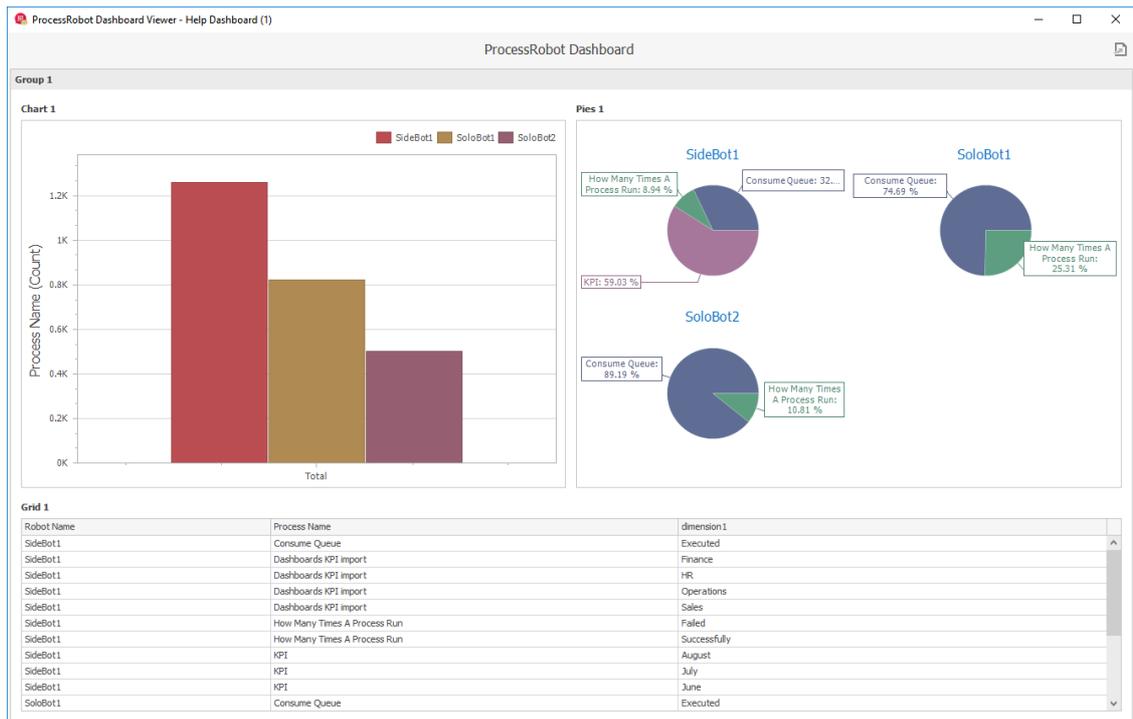
6.1 Dashboard Viewer

6.1.1 Dashboard Viewer Window

All dashboards created through the Dashboard Designer, can be viewed through the Dashboard Viewer.



You can right click on the name of the dashboard to open it.



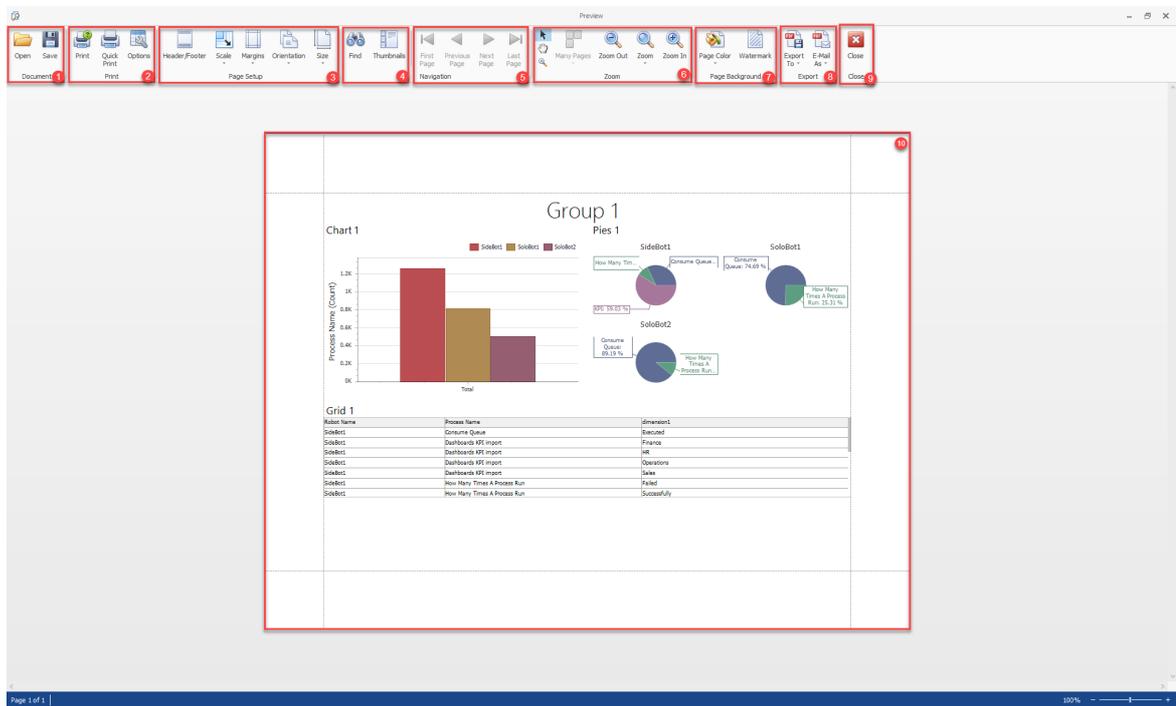
The dashboard shows as designed in the dashboard designer.

By clicking the icon in the top right corner, you can access the print preview as well as to export the dashboard to a PDF or image file.

6.1.2 Print Preview

By right clicking on any element, you can select and access the print preview.

The Print Preview window, has the following menu groups and panes:



1. Document:

In the document tab, you can open a preview document file as well as save the current file that you are working on.

2. Print:

In the print group you can print, quick print and select the options of the printing.

3. Page Setup:

Using the page setup group, you can add headers and footers, as well as to change the scaling, size, orientation of the page and edit the margins of the page.

4. Find - Thumbnails:

In this group, you can search for terms within the page as well as to view thumbnails of all the pages.

5. Navigation:

Using the navigation buttons, you can navigate yourself between the pages.

6. Zoom:

Using the zoom group, you can view the print preview in another scale as well as to select from mouse pointer, hand and magnifier tool for your cursor.

7. Page Background:

In the page background group, you can change the page color as well as to add a Watermark to your page.

8. Export:

The Export group, allows you, to export the page to a PDF or Image file as well as to email the generated PDF or Image file.

9. Close:

You can close the dashboard using the close button.

10. Print Preview pane:

You can view the final result in the print preview pane.

6.2 Dashboard Designer

6.2.1 Dashboard Designer - Introduction

Insights Dashboard, is a business intelligence analytics tool and key component of Softomotive Process Robot Enterprise RPA platform. Insights Dashboard allows for custom KPI tracking and monitoring and therefore it is mainly interesting for Senior Managers, who would like to have a picture of the high-level performance of the RPA system. Data in the Insights Dashboard, are being inserted by the use of the "Update KPI" action in the Process Studio.

The Update KPI action has the field to set your own name. Next, you can find the value by which the KPI will be updated as well as the dimensions with the values that you would like to measure per KPI.

Please keep in mind, that the unique feature here is that you can customize your own KPI's.

Except from the KPI's name and the dimensions, we can visualize the KPI's according to the Environment, the Process, the Robot that were executed. Additionally, we can set the timestamp to view our results over time as well as set custom values.

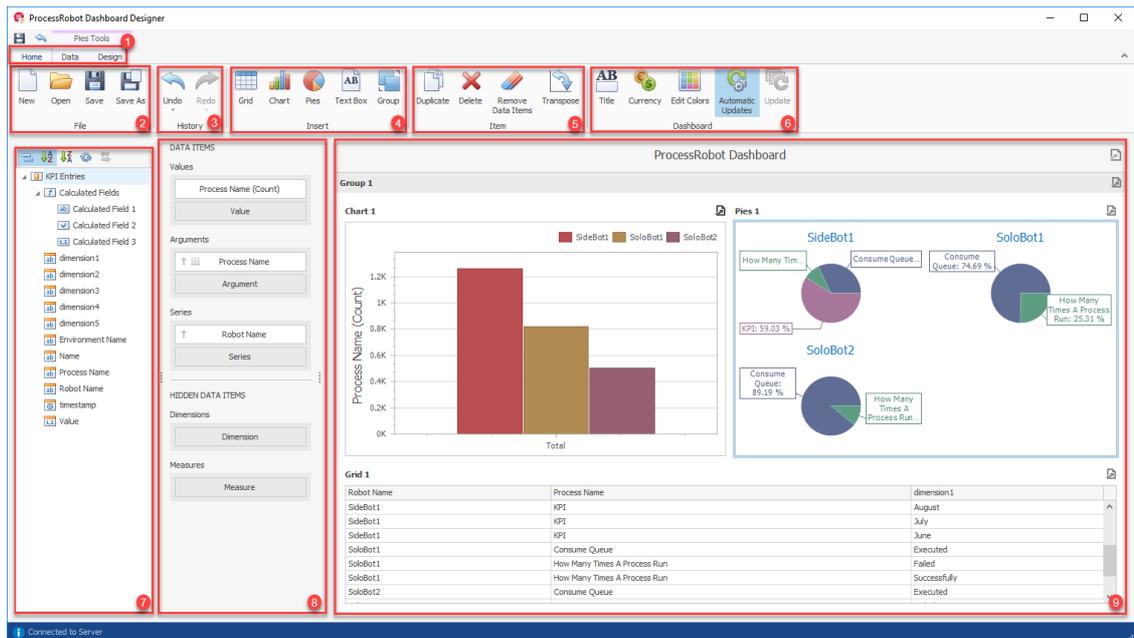
Creating custom KPI graphs with this business intelligence analytics tool, allowing Senior Managers to review trends of these KPIs, as they are retrieved directly from the robots.

Access to performance analytics and real-time monitoring of organizational goals can be achieved and also an ROI calculator is available.

Insights Dashboard can be integrated with other commercially available analytics tools, such as Tableau or Cognos, but of course it can be used as stand-alone tool as well.

6.2.2 Dashboard Designer Window

The Dashboard Designer has the following menu groups and panes:



1. Main Menu Tabs:

In the main menu tabs, you can go and select if you view the Click, the Data or the Design menus.

2. File:

In the File group you can create a new dashboard, open an existing one, save/save as a dashboard that you are developing.

3. History:

Using the history group, you can undo and redo changes on the dashboards that you are developing.

4. Insert:

In the insert group you can insert different kind of available items.

5. Item:

In this group, you can duplicate items, delete them, remove their data items and transpose the data.

6. Dashboard:

Using the dashboard group, you can insert titles, change the currency, edit colors as well as set the dashboard to update automatically.

7. KPI Entries:

In the KPI entries pane, you can view and select which KPI entries you will use.

8. Data Items:

In the Data Items pane, you can select the KPI entries as values, dimensions and series you use within the items.

9. Process Dashboard:

In the Process Dashboard pane, you develop your final dashboards.

6.2.3 Dashboard Elements

You can pick between four type of elements to insert into your dashboards.

You can add grids, charts, pies, and textboxes. All these, can be added in groups so that you can design your dashboards accordingly.

6.2.3.1 Grids

6.2.3.1,1 Data Tools

After you have inserted a Grid in your Dashboard, you are able to filter the results according to your preference.

Filtering

You can edit filters of the data shown in the grid. Filtering can be done according to the values items used in the columns, Sparkline and dimensions of the grid.

Pressing the clear button, you can remove all the filters being used.

Interactivity

You can use the single master filter, to make all dashboard items, show data corresponding to the element you have selected.

Similarly, you can use the multiple master filter, to make the dashboard items show data corresponding to multiple selected elements.

The Drill Down option, enables you to click an element within the dashboard item and display the detailed data related to this element.

Interactivity Settings

Using the cross-data-source filtering, you can allow the master filter to affect dashboard items that display data from other data sources.

In this instance, filtering is performed if full names of data source fields match.

You can select the option to ignore master filters in the dashboard.

6.2.3.1,2 Design Tools

After you have inserted a Grid in your Dashboard, you are able to edit the design of it through the design tab.

Common

You can select to show or hide the caption (title of grid)

Using the edit names option, you are able to change the title of the grid as well as the name of every column's header.

Style

In the style section, you are able to show or hide the vertical and horizontal lines as well as show banded rows of the results.

Layout

In the merge cells, you may merge cells with identical data. Additionally, you may show or hide the column's headers as well as enable and disable word wrapping in multiple line cells.

Column Width Mode

In the column width mode section, you may auto fit the columns according to contents or the grid as well as enable the option to edit the width of the columns manually.

6.2.3.1,3 Data Items

The Grid Element, has four types of data items.

Columns

Firstly, the columns data item, is set so that to add KPI entries to be viewed as columns of the grid.

By clicking the icon next to the entry, you can select the column type between a dimension, a measure, a delta, or a Sparkline.

In the left side of each entry, you can select to sort the results of the grid in ascending or descending order.

Sparkline

The Sparkline contains the argument. For the argument, you can select the time-stamp or the value in order to sort the items in the grid in ascending or descending order.

Dimensions

By adding entries in the dimensions, you can set filters for values that are not included in the grid.

Measures

Adding entries in the measures item, allows you to set minimum, maximum values and other calculations to show or not show within the grid.

6.2.3.2 Charts

6.2.3.2,1 Data Tools

After you have inserted a Chart in your Dashboard, you are able to filter the results according to your preference.

Filtering

You can edit filters of the data shown in the chart. Filtering can be done according to the values items used in the arguments, series and dimensions of the chart.

Pressing the clear button, you can remove all the filters being used.

Interactivity

You can use the single master filter, to make all dashboard items, show data corresponding to the element you have selected.

Similarly, you can use the multiple master filter, to make the dashboard items show data corresponding to multiple selected elements.

The Drill Down option, enables you to click an element within the dashboard item and display the detailed data related to this element.

Interactivity Settings

Using the cross-data-source filtering, you can allow the master filter to affect dashboard items that display data from other data sources.

In this instance, filtering is performed if full names of data source fields match.

You can select the option to ignore master filters in the dashboard.

6.2.3.2,2 Design Tools

After you have inserted a Chart in your Dashboard, you are able to edit the design of it through the design tab.

Common

You can select to show or hide the caption (title of chart)

Using the edit names option, you are able to change the title of the chart.

Diagram

In the diagram section, you are able to rotate the diagram at 90 degrees. Also, you may change the settings of the X and Y axis. You can reverse, enable zooming, show and hide axis and their titles as well as limit the visible points.

Legend

Pressing the legend button, you can show and hide the legend.

Series Type

In the series type section, you are able to change the series type of the chart.

Coloring

In the coloring section, you are able to set global and local colors as well as to edit the colors used in the chart.

6.2.3.2,3 Data Items

The Charts Element, has five types of data items.

Values

Firstly, the values data item, is set so that to add KPI entries to be viewed as charts.

By clicking the icon next to the entry, you can select the chart's type and options.

You can add new charts using the icon above the Values .

Arguments

Adding entries in the arguments, allows you to view the items in an order according to the selected entry.

Series

The series field, allows you to show the values according to other values as well which can be added from the KPI Entries section.

Dimensions

By adding entries in the dimensions, you can set filters for values that are not included in the chart.

Measures

Adding entries in the measures item, allows you to set minimum, maximum values and other calculations to show or not show within the chart.

6.2.3.3 Pies

6.2.3.3.1 Data Tools

After you have inserted a Pie in your Dashboard, you are able to filter the results according to your preference.

Filtering

You can edit filters of the data shown in the pie. Filtering can be done according to the values items used in the arguments, series and dimensions of the pie.

Pressing the clear button, you can remove all the filters being used.

Interactivity

You can use the single master filter, to make all dashboard items, show data corresponding to the element you have selected.

Similarly, you can use the multiple master filter, to make the dashboard items show data corresponding to multiple selected elements.

The Drill Down option, enables you to click an element within the dashboard item and display the detailed data related to this element.

Interactivity Settings

Using the cross-data-source filtering, you can allow the master filter to affect dashboard items that display data from other data sources.

In this instance, filtering is performed if full names of data source fields match.

You can select the option to ignore master filters in the dashboard.

Target Dimensions

You can select the arguments and series in order to perform master filtering and drill down.

Points, are being used to perform interactivity actions.

6.2.3.3,2 Design Tools

After you have inserted a Pie in your Dashboard, you are able to edit the design of it through the design tab.

Common

You can select to show or hide the caption (title of pie)

Using the edit names option, you are able to change the title of the pie as well as the values.

Content Arrangement

In the content arrangement section, you are able to arrange the pies in columns or rows as well as to auto arrange the pies.

Labels

In the labels section, you are able to set the format of the data labels, tool-tips and whether or not to show the pies captions.

Style

In the style section, you are able to select the style of the pie between a pie and a donut.

Coloring

In the coloring section, you are able to set global and local colors as well as to edit the colors used in the pie.

6.2.3.3,3 Data Items

The Pie Element, has five types of data items.

Values

Firstly, the values data item, is set so that to add KPI entries to be viewed as pies.

Arguments

Adding entries in the arguments, allows you to view the proportion of the value according to the selected entry.

Series

The series field, allows you to show different pie charts for each of the items of the entry entered.

Dimensions

By adding entries in the dimensions, you can set filters for values that are not included in the pies.

Measures

Adding entries in the measures item, allows you to set minimum, maximum values and other calculations to show or not show within the pies.

6.2.3.4 Text Boxes

6.2.3.4,1 Design Tools

After you have inserted a Text Box in your Dashboard, you are able to edit the design of it through the design tab.

Common

You can select to show or hide the caption (title of text box)

Using the edit names option, you are able to change the title of the text box.

6.2.3.5 Groups

6.2.3.5,1 Data Tools

After you have inserted a Group in your Dashboard, you are able to filter the results according to your preference.

Interactivity

You can use the single master filter, to make all dashboard items, show data corresponding to the element you have selected.

Similarly, you can use the multiple master filter, to make the dashboard items show data corresponding to multiple selected elements.

6.2.3.5,2 Design Tools

After you have inserted a Group in your Dashboard, you are able to edit the design of it through the design tab.

Common

You can select to show or hide the caption (title of group)

Using the edit names option, you are able to change the title of the group.

6.2.4 KPI Entries

In the KPI Entries pane, you view the entries that you can select from in order to set them as data items in the elements of your dashboard.

By right clicking on the pane, you can add custom fields.

Using the buttons above the KPI Entries pane, you can group the entries by type, sort them alphabetically and refresh the field.

6.2.4.1 Entries

The default Entries from which you can select from are:

dimension 1: Value set in the dimension 1 of the "Update KPI" action.

dimension 2: Value set in the dimension 2 of the "Update KPI" action.

dimension 3: Value set in the dimension 3 of the "Update KPI" action.

dimension 4: Value set in the dimension 4 of the "Update KPI" action.

dimension 5: Value set in the dimension 5 of the "Update KPI" action.

Environment Name: Environment on which the process was executed.

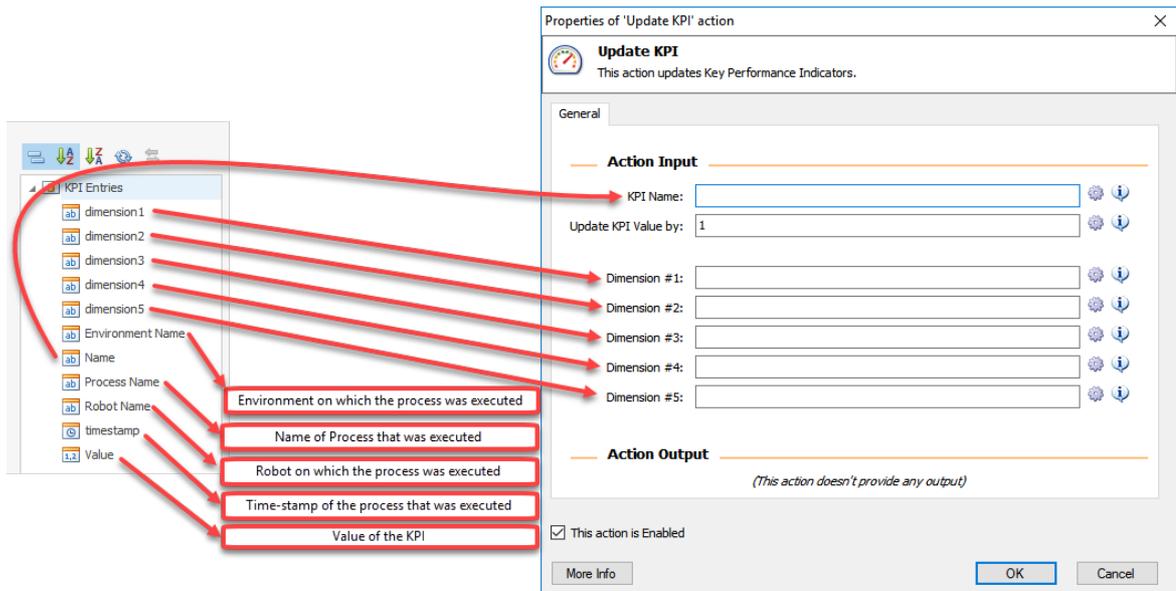
Name: Name of the KPI that was updated.

Process Name: Name of the process that was executed.

Robot Name: Robot on which the process was executed.

timestamp: Time-stamp of the process that was executed.

Value: Value of the KPI.



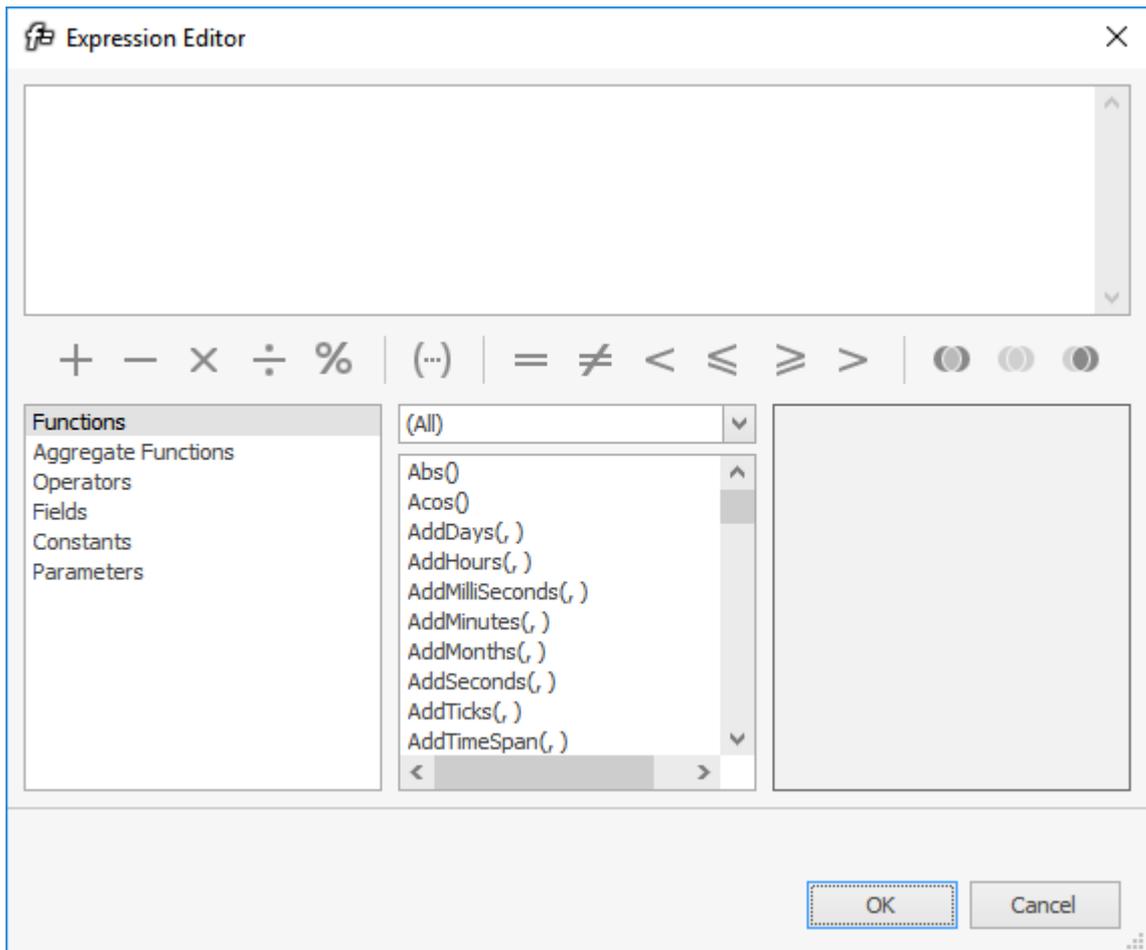
6.2.4.2 Calculated Fields

The Dashboard Designer provides the capability to create calculated fields that allow you to apply complex expressions to data fields that are obtained from the dashboard's data source.

You can use these fields in data visualizations as regular data source fields.

To create a calculated field, right-click the entries pane and select Add Calculated Field in the context menu.

This invokes the Expression Editor dialog, which allows you to specify an expression that will be used to obtain calculated field values. Here, you can construct the required expression.



In your expressions you can use functions, aggregate functions, operators, fields, constants and parameters.

After the expression has been specified you can click OK to add the calculated field in the entries list.

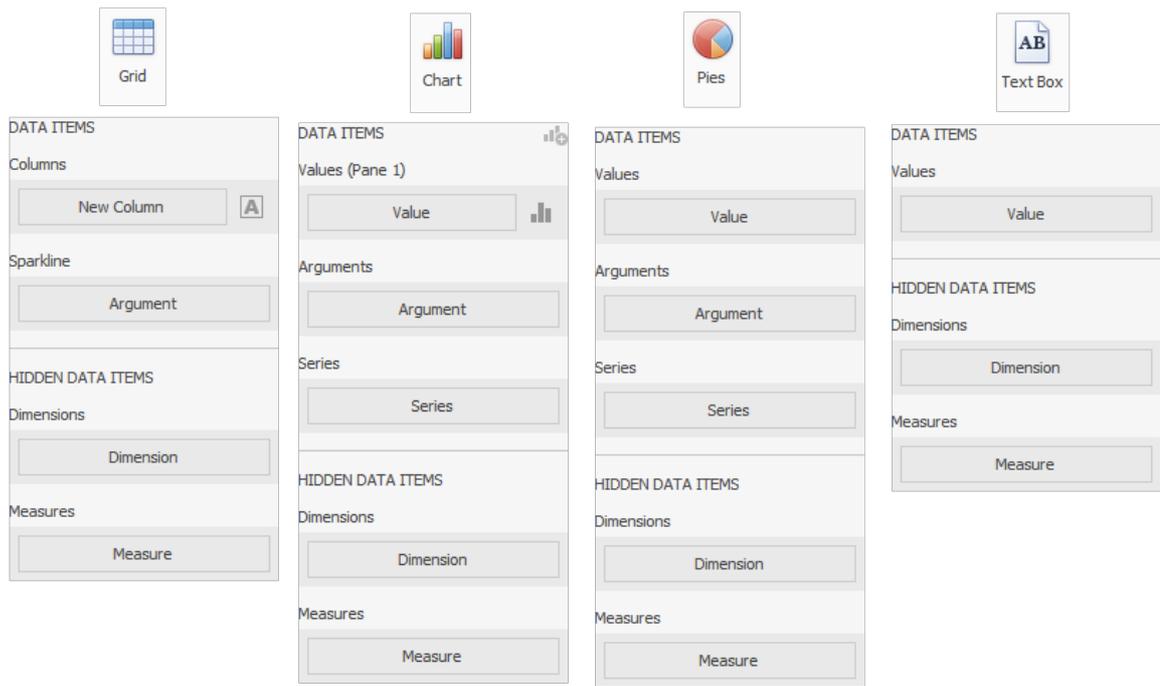
You can right click the Calculated field to change it's name, type as well as to edit and delete it.

The types from which you can select are, String, Integer, Decimal, Double, Boolean and DateTime.

6.2.5 Data Items

In the Data Items pane, you can import the entries from the KPI entries pane into the dashboard.

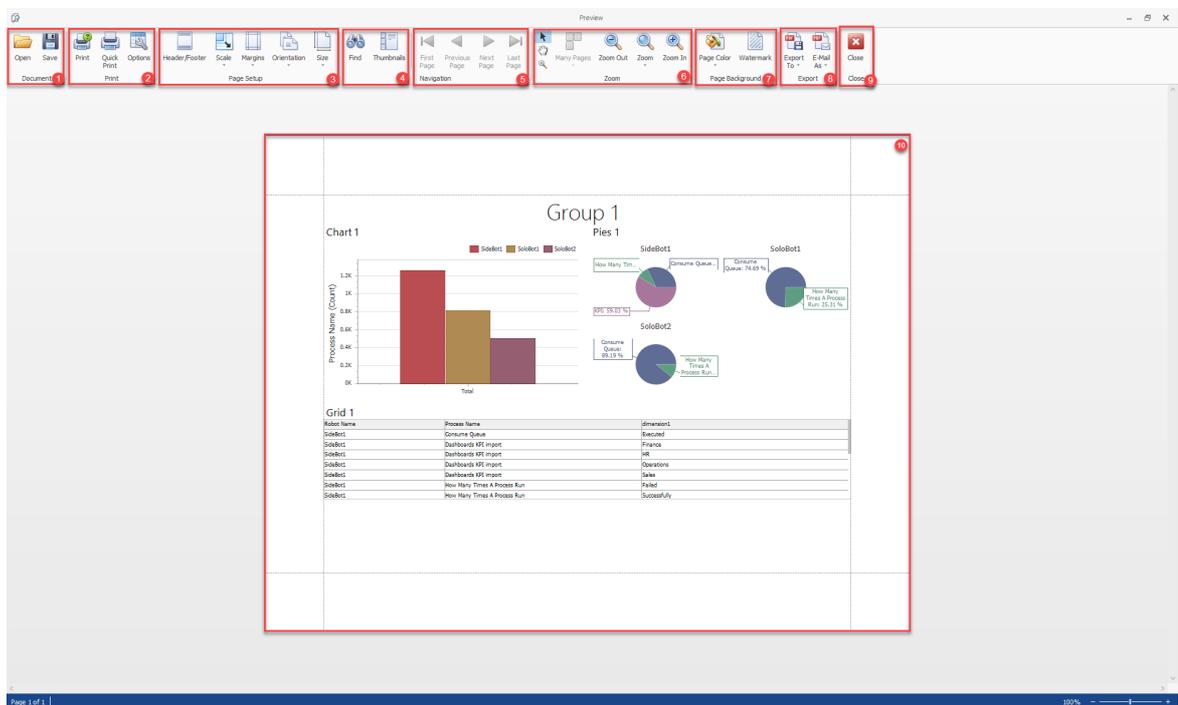
Each Element, requires different kind of Items.



6.2.6 Print Preview

By right clicking on any element, you can select and access the print preview.

The Print Preview window, has the following menu groups and panes:



1. Document:

In the document tab, you can open a preview document file as well as save the current file that you are working on.

2. Print:

In the print group you can print, quick print and select the options of the printing.

3. Page Setup:

Using the page setup group, you can add headers and footers, as well as to change the scaling, size, orientation of the page and edit the margins of the page.

4. Find - Thumbnails:

In this group, you can search for terms within the page as well as to view thumbnails of all the pages.

5. Navigation:

Using the navigation buttons, you can navigate yourself between the pages.

6. Zoom:

Using the zoom group, you can view the print preview in another scale as well as to select from mouse pointer, hand and magnifier tool for your cursor.

7. Page Background:

In the page background group, you can change the page color as well as to add a Watermark to your page.

8. Export:

The Export group, allows you, to export the page to a PDF or Image file as well as to email the generated PDF or Image file.

9. Close:

You can close the dashboard using the close button.

10. Print Preview pane:

You can view the final result in the print preview pane.

6.3 Creating a Sample Dashboard

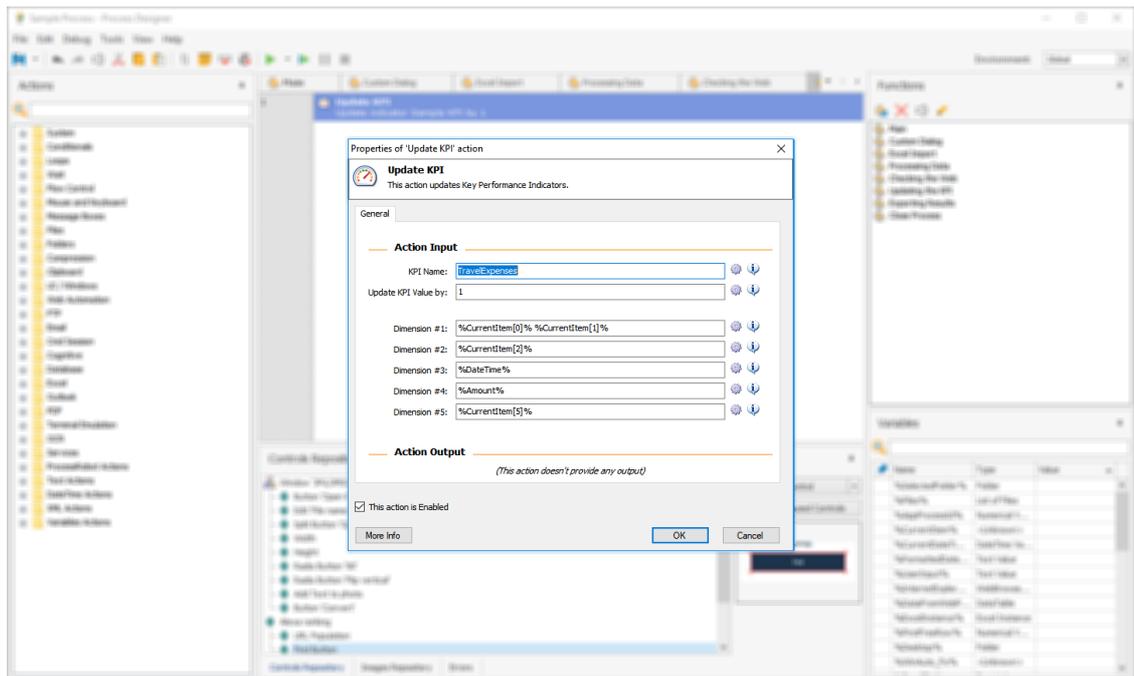
6.3.1 Updating the KPI through the Process

In this example, we are going to execute a process that imports information about travel expenses of employees in a multinational company.

Every Year, an excel file as the one shown below, is being imported in a process and according to the records of that excel file, the "TravelExpenses" KPI is being updated.

This process, updates the KPI so that the upper management can have full overview on the travel expenses of their employees.

	A	B	C	D	E	F
1	First Name	Surname	Location	Date of Transaction	Amount(USD)	Category
2	Melissa	Scott	Singapore	05-01-16	1737.00	Fees & Charges
3	Jane	Smith	New York	13-01-16	1111.00	Travel Expenses
4	Maria	Hall	New Delhi	22-01-16	952.00	Food & Dining
5	Melissa	Scott	Singapore	28-01-16	1953.00	Food & Dining
6	Jane	Smith	New York	30-01-16	1111.00	Food & Dining
7	Melissa	Scott	Singapore	30-01-16	1796.00	Travel Expenses
8	Melissa	Scott	Singapore	02-02-16	1810.00	Food & Dining
9	Sarah	Jones	Sao Paulo	04-02-16	1315.00	Travel Expenses
10	Nick	Barton	Athens	08-02-16	1059.00	Food & Dining
11	Sarah	Jones	Sao Paulo	09-02-16	1282.00	Food & Dining
12	Robert	Williams	London	12-02-16	1683.00	Food & Dining
13	Max	Green	Berlin	16-02-16	1833.00	Fees & Charges



As shown in the above screen-shot, when the process "Expenses" is being executed, it updates the "TravelExpenses" setting as dimensions the following.

- Dimension 1: Name of employee
- Dimension 2: City of transaction
- Dimension 3: DateTime
- Dimension 4: Amount
- Dimension 5: Category of expense

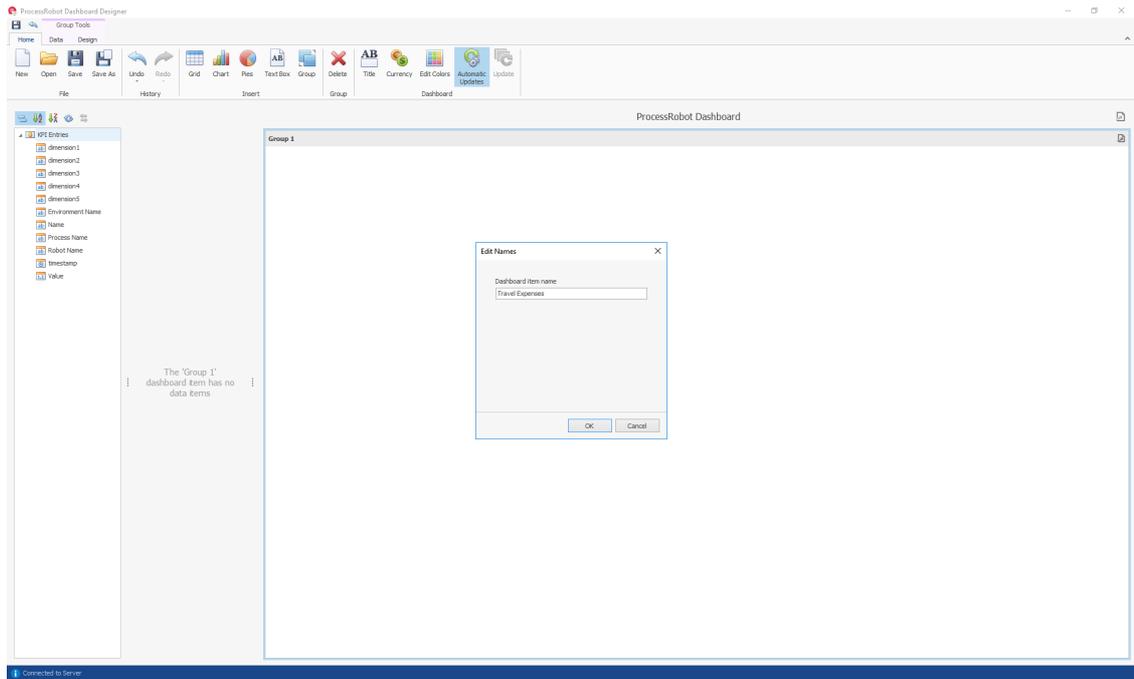
Automatically, once the Update KPI action is being executed, the following information are also updated on each entry.

- Process Name
- Robot Name
- Environment Name
- timestamp

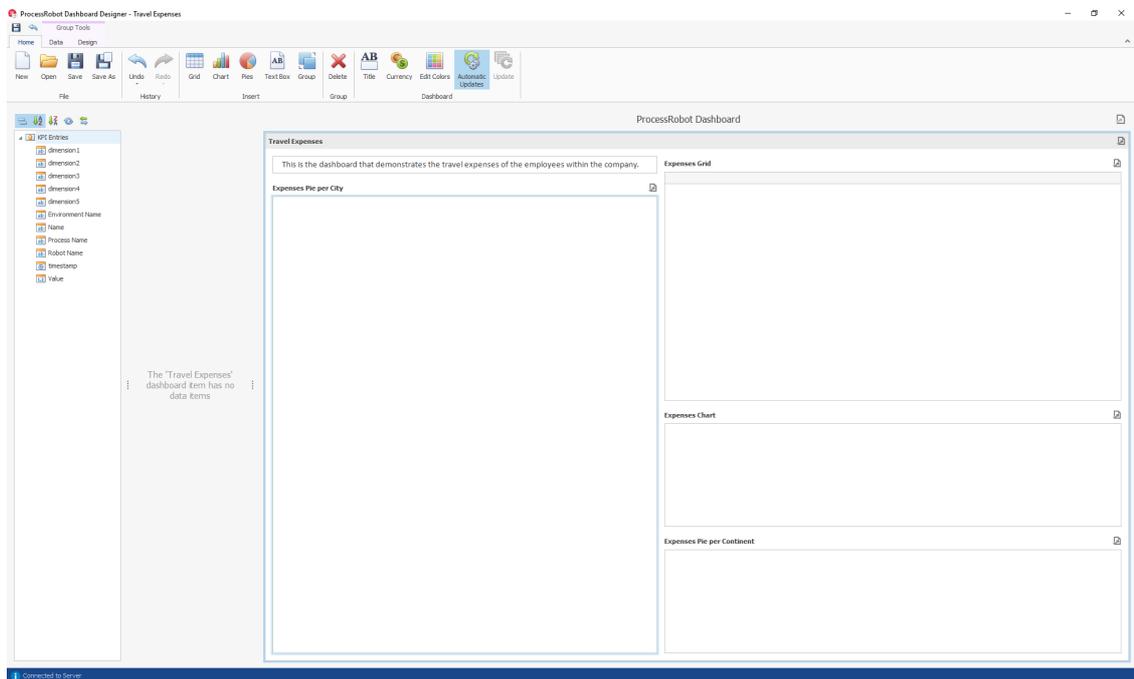
6.3.2 Creating The Dashboard

Once the user opens the Dashboard Designer, he is able to insert elements.

In this example, we will first insert a group in order to insert all other elements in one group. By right clicking, we will select "Edit Names" to rename the Group to "Travel Expenses".



We will add a Title to the dashboard by clicking on the title button and then add one of each elements. By dragging and dropping them, we will include them in the group. By right clicking in each element and editing the name, we can reach to the below result.



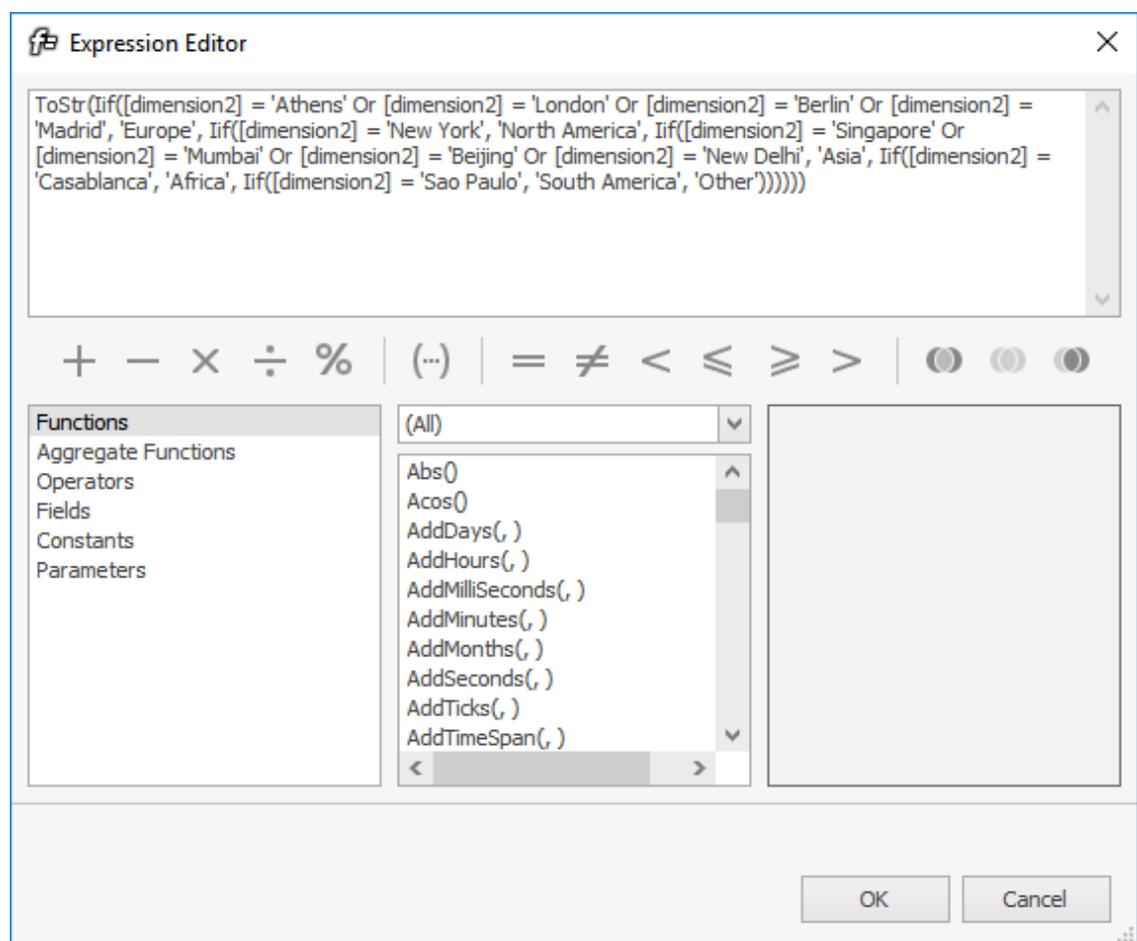
6.3.3 Configuring a Calculated Fields

In this example, we will create three calculated fields that we will use later.

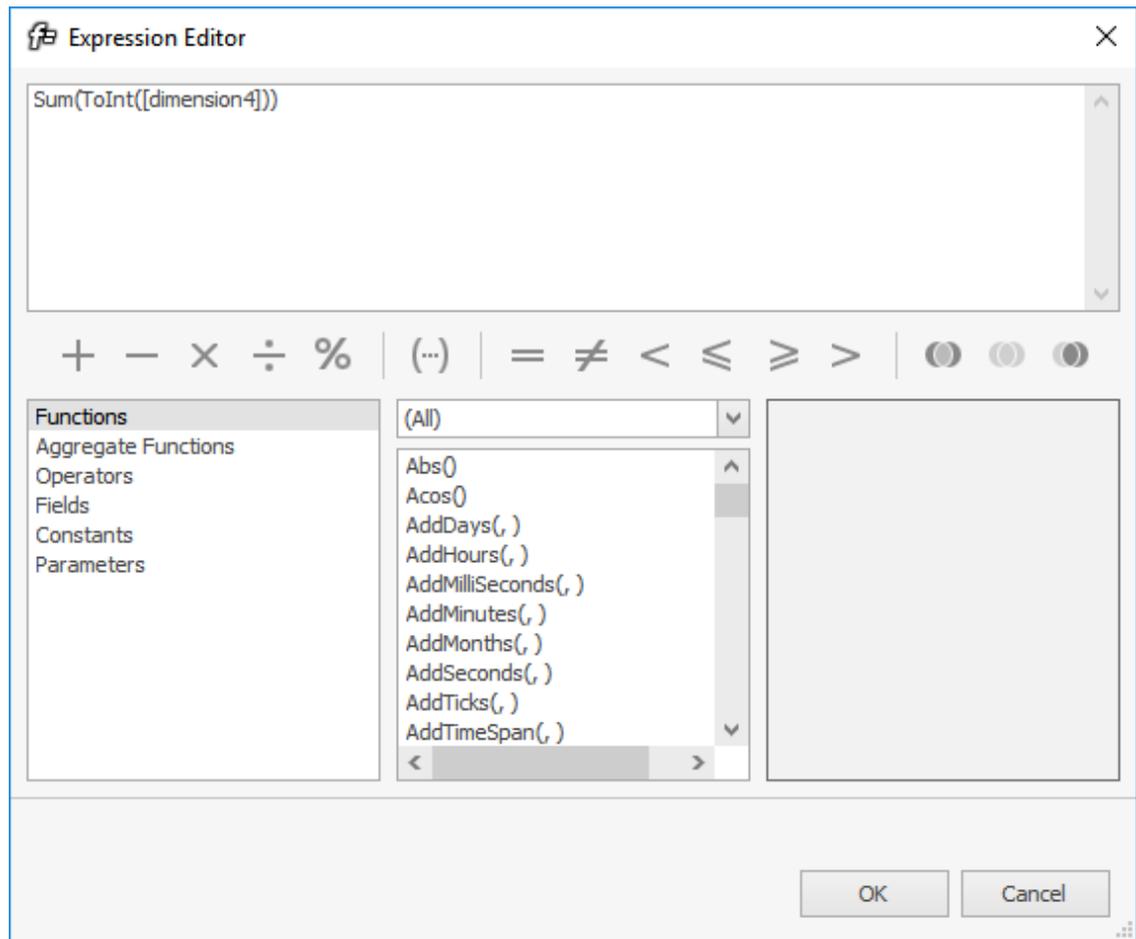
By right clicking on the entries pane, we are going to create a calculated field so that to view the Continent in which the expense in each KPI was updated. Note, that the continent was not in any of the dimensions.

As we know the cities that are being used (Dimension 2), we will set an expression so that to deduce the continent according to every city.

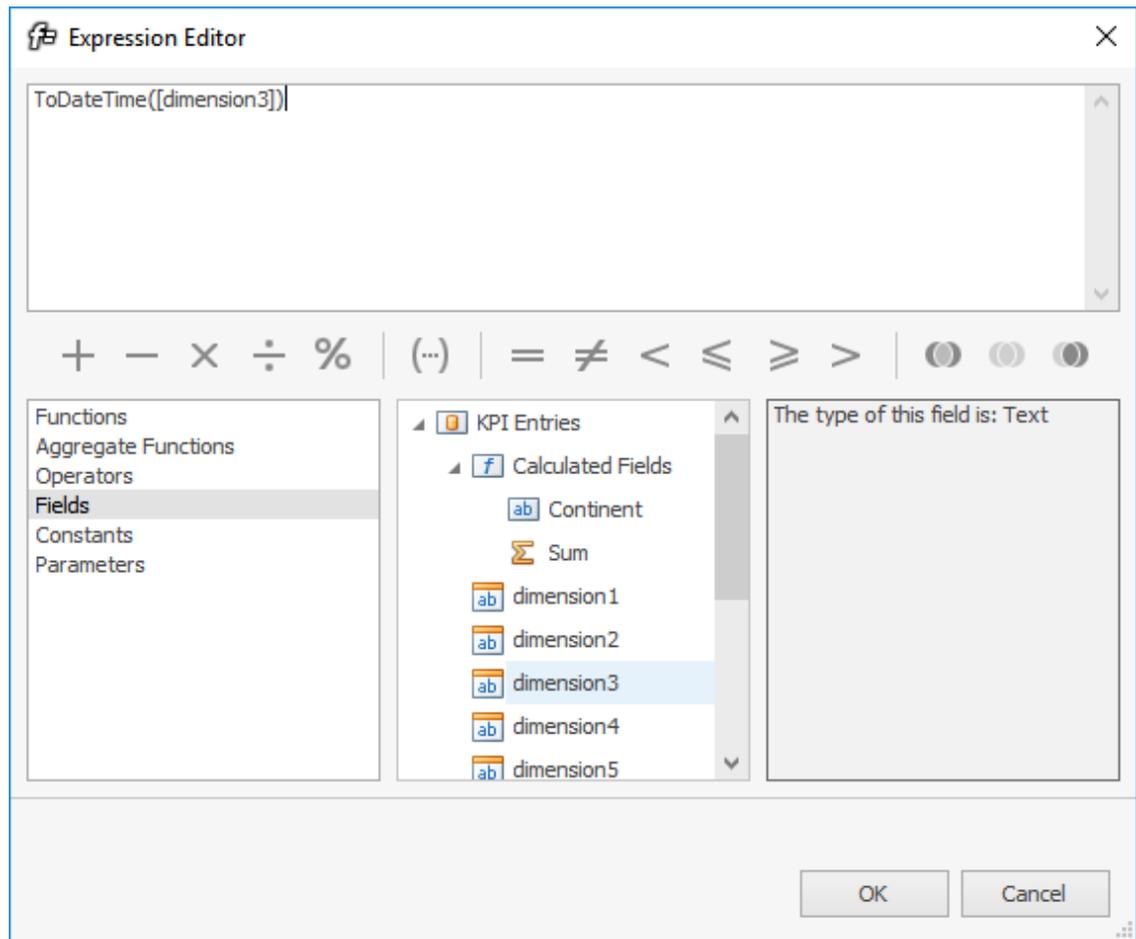
For example, if the city is Athens or Berlin, the Continent is Europe while if the country is Sao Paulo, the Continent is South America.



Except from the Continent, we will create the "Sum" calculated field so that to be able to add the amounts (Dimension 4) of each transaction.



Last but not least, we will create the "Date" calculated field so that to be able to get the details of the date according to the datetime (Dimension 3) imported.



6.3.4 Developing the Grid

After selecting the grid, we can now set all the items shown in each of the columns. Moreover, we can right click on the columns to rename them.

In this case, we will add the following columns.

- Dimension 1 (Full Name)
- Dimension 2 (City)
- Continent
- Date
- Date (Day Of Week)
- Date (Month)
- Dimension 4 (Amount)
- Dimension 5 (Category)

In order to view only the results of this KPI, we will add the Name of the KPI as the dimension so that we are able to filter it using the filter editor.

The screenshot shows the ProcessRobot Dashboard Designer interface. The main dashboard area displays a table titled 'Expenses Grid' with the following columns: Full Name, City, Continent, Date, Date (Day of Week), Date (Month), Amount, and Category. The table contains 20 rows of expense data. A 'Filter Editor' dialog box is open in the center, showing a filter rule: '[Name] Equals Expenses'. The left sidebar shows 'KPI Entries' with a tree view including 'Calculated Fields', 'Sum', and several 'Dimension' and 'Measure' items. The bottom status bar indicates 'Connected to Server'.

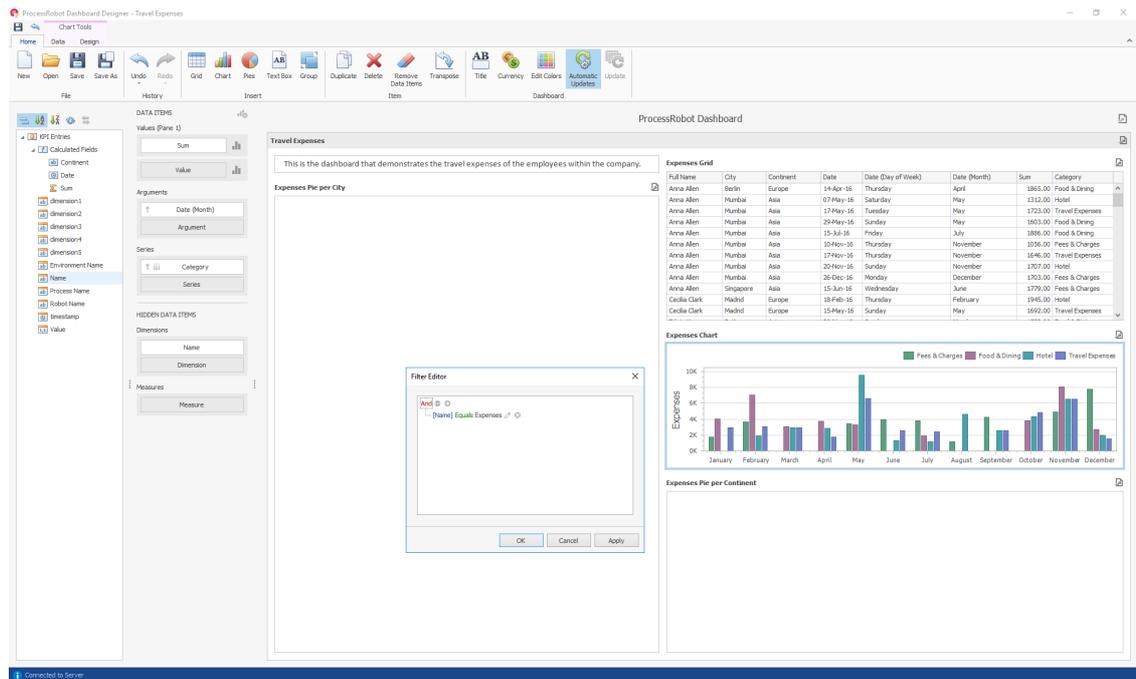
The final result, is a grid showing each expense with the name of the employee, the city, the continent, exact date, day and month that the transaction took place as well as the amount and the category of the expense.

6.3.5 Developing the Chart

After selecting the chart, we can now set all the items shown in the chart by setting them as arguments, series and values. Moreover, we can right click on the items to rename them.

In this case, we will set as the series the category of the expense, as the argument the month and the amount as the value.

In order to be able to view only the results of this KPI, we will add the Name of the KPI as the dimension so that we are able to filter it using the filter editor.



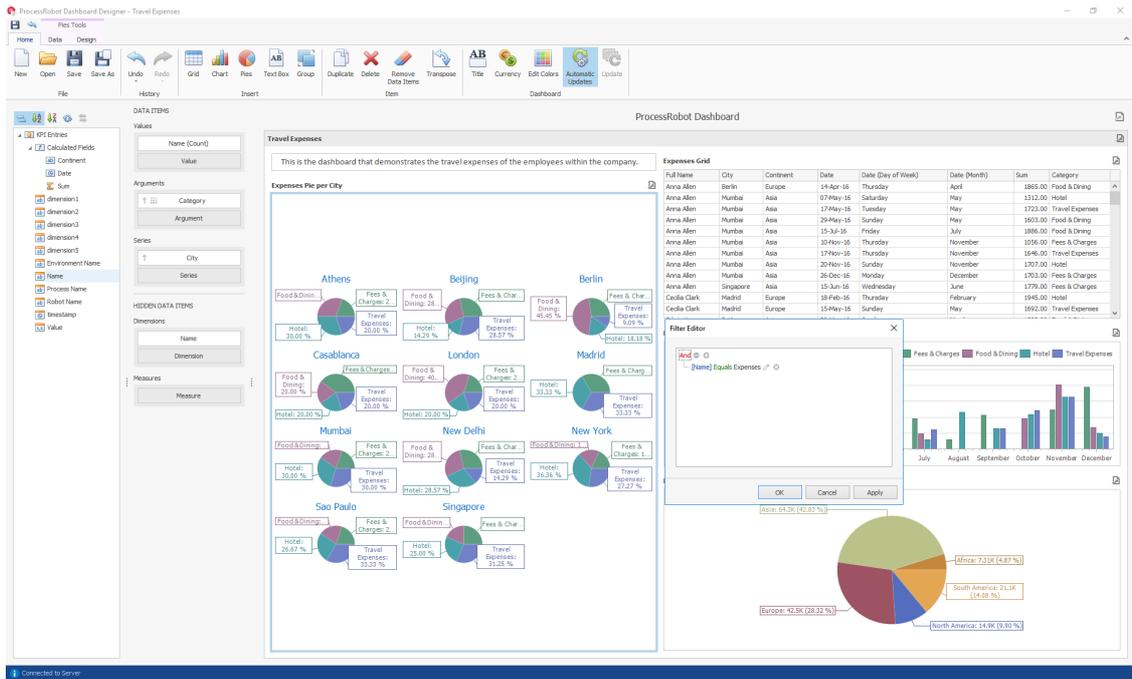
The final result, is a chart showing each month's total transactions according to its category.

6.3.6 Developing the Pie

After selecting the pie, we can now set all the items shown in the chart by setting them as arguments, series and values. Moreover, we can right click on the items to rename them.

Similar to the chart, we will set as the series the City, as the argument the Category and the Name that we are going to filter the results with, as the value.

Once again order to be able to view only the results of this KPI, we will add the Name of the KPI as the dimension so that we are able to filter it using the filter editor.



The final result, is a series of pies showing the records for the categories, according to each city.

In the same Dashboard, we added one more pie that demonstrates the amounts spent per Continent. To come to this result, we set the Continent as the Argument and the Sum as the Value. Once again, we filter the results according to the KPI Name set as a dimension.

6.3.7 Completing the Dashboard

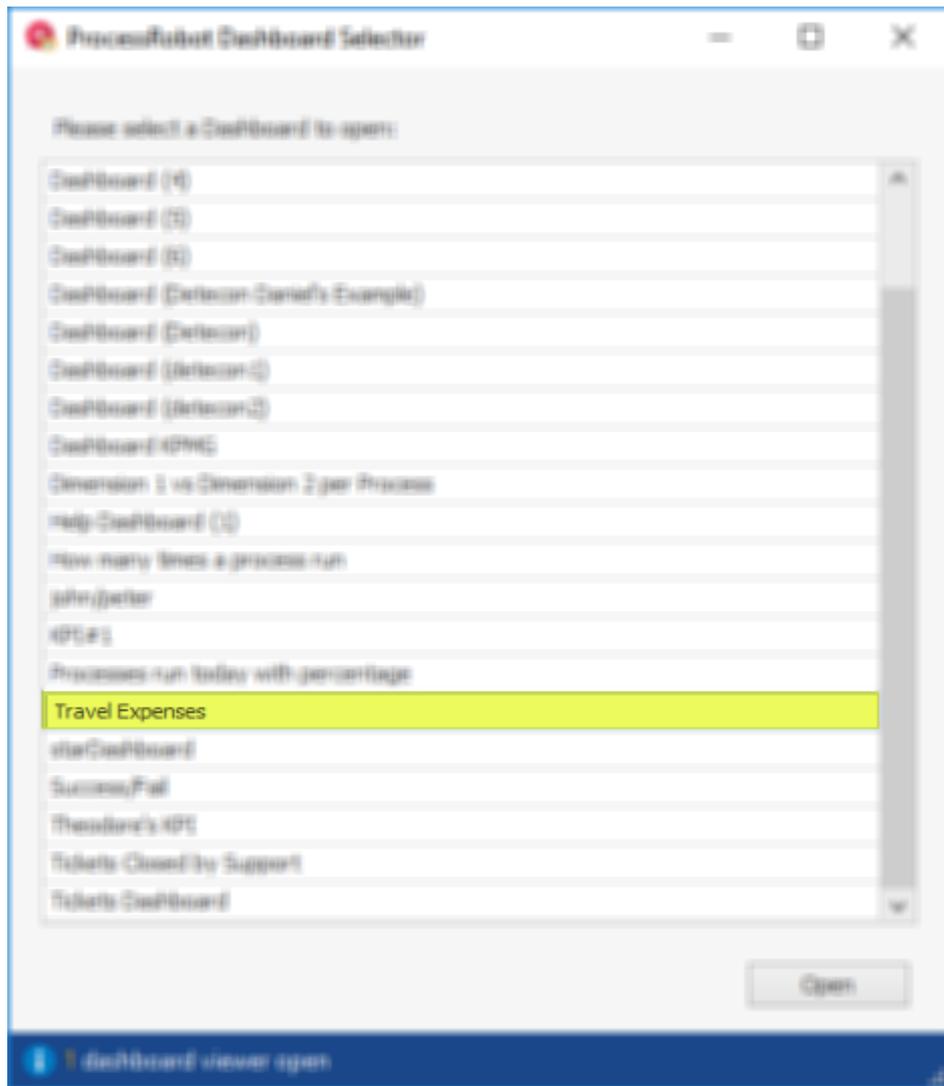
Now that the dashboard is complete, we can click on the save as button to save it as the name of preference.

The screenshot shows the ProcessRobot Dashboard Designer interface. The main workspace displays a dashboard titled 'Travel Expenses' with the following components:

- Expenses Grid:** A table listing employee travel expenses.

Full Name	City	Continent	Date	Date (Day of Week)	Date (Month)	Sum	Category
Anna Allen	Berlin	Europe	14-Apr-16	Thursday	April	1865.00	Food & Dining
Anna Allen	Mumbai	Asia	07-May-16	Saturday	May	1312.00	Hotel
Anna Allen	Mumbai	Asia	17-May-16	Tuesday	May	1723.00	Travel Expenses
Anna Allen	Mumbai	Asia	29-May-16	Sunday	May	1653.00	Food & Dining
Anna Allen	Mumbai	Asia	15-Jul-16	Friday	July	1886.00	Food & Dining
Anna Allen	Mumbai	Asia	10-Nov-16	Thursday	November	1056.00	Fees & Charges
Anna Allen	Mumbai	Asia	17-Nov-16	Thursday	November	1646.00	Travel Expenses
Anna Allen	Mumbai	Asia	20-Nov-16	Sunday	November	1307.00	Hotel
Anna Allen	Mumbai	Asia	26-Dec-16	Monday	December	1703.00	Fees & Charges
Anna Allen	Singapore	Asia	15-Jan-16	Wednesday	January	1776.00	Fees & Charges
Anna Allen	Madrid	Europe	18-Feb-16	Thursday	February	1945.00	Hotel
Anna Allen	Madrid	Europe	15-May-16	Sunday	May	1692.00	Travel Expenses
- Save Dashboard As... Dialog:** A modal dialog box is open, prompting the user to enter a dashboard name. The text 'Save Dashboard As...' is visible, and the 'OK' button is highlighted.
- Expenses Pie per City:** A collection of pie charts showing the breakdown of expenses by city (Athens, Beijing, Berlin, Casablanca, Mumbai, New Delhi, New York, Sao Paulo, Singapore).
- Expenses Pie per Continent:** A pie chart showing the distribution of expenses across continents: Asia (64.3K, 42.61%), Africa (2.31K, 4.87%), South America (21.0K, 14.08%), Europe (42.3K, 28.32%), and North America (14.3K, 9.80%).
- Expenses Bar Chart:** A bar chart showing monthly expense trends from January to December.

Our dashboard is now in the Dashboard Viewer's list.



Index

- % -

% 501
 %ButtonPressed% 304
 %EventTriggerEventID% 164
 %EventTriggerEventLogName% 164
 %EventTriggerEventMessage% 164
 %EventTriggerEventSource% 164
 %EventTriggerEventType% 164
 %FileTriggerEventType% 165
 %FileTriggerFileName% 165
 %FileTriggerFilePath% 165
 %FileTriggerOldFileName% 165
 %LoopIndex% 452, 473
 %name% 501
 %PingTriggerHostName% 167
 %ProcessTriggerEventType% 169
 %ProcessTriggerProcessName% 169
 %ServiceTriggerServiceName% 168
 %ServiceTriggerServiceState% 168
 %TriggerName% 152, 153, 154
 %VariableName[ItemNumber]% 438
 %VariableName[RowNumber][ColumnNumber]%
 439

- . -

.exe 638
 .wav 546

- \ -

\% 501

- = -

= 474

- > -

> 474

>= 474

- 1 -

1000 Separator 1034

- A -

ABBYY 937
 About Enate 477
 About Enate Actions 477
 access 138
 access the file 664
 Action 304, 309, 312, 318, 321, 529, 536, 540,
 541, 542, 543, 545, 546, 547, 548, 550, 552, 553,
 554, 556, 557, 559, 560, 561, 562, 571, 573, 576,
 577, 584, 586, 587, 588, 589, 590, 593, 594, 595,
 598, 605, 606, 608, 611, 614, 615, 617, 618, 619,
 621, 629, 630, 632, 634, 636, 638, 642, 647, 649,
 651, 652, 653, 654, 657, 668, 669, 670, 672, 673,
 675, 676, 678, 679, 680, 682, 683, 686, 688, 692,
 694, 701, 704, 708, 710, 711, 712, 713, 714, 715,
 717, 719, 721, 722, 723, 746, 751, 753, 755, 757,
 758, 762, 765, 766, 767, 769, 772, 961, 962, 1000,
 1001, 1036, 1049, 1064, 1065, 1067
 Action Icons 312
 Action Input 304
 Action Output 304
 Action Properties 435
 Action Properties Box 502, 504
 Action Property Window 294
 action reference 536, 683, 686, 688
 Action to Perform 543
 Action's Inputs 435
 actions 508, 529, 536

- Actions Reference 529, 531, 533, 535, 537, 540, 541, 542, 543, 545, 546, 547, 548, 550, 554, 556, 557, 559, 560, 561, 562, 566, 571, 573, 576, 577, 584, 586, 587, 588, 589, 590, 593, 594, 595, 598, 605, 606, 608, 611, 614, 615, 617, 618, 619, 621, 629, 630, 632, 633, 634, 636, 638, 640, 642, 645, 647, 649, 651, 652, 653, 654, 657, 659, 661, 663, 664, 665, 667, 668, 669, 670, 672, 673, 674, 675, 676, 678, 679, 680, 682, 692, 694, 698, 701, 704, 708, 710, 711, 712, 713, 714, 715, 717, 719, 721, 722, 723, 746, 751, 753, 755, 757, 758, 762, 765, 766, 767, 769, 772, 776, 778, 780, 781, 782, 783, 785, 786, 787, 788, 789, 790, 791, 792, 796, 797, 806, 808, 875, 876, 878, 880, 884, 886, 887, 888, 889, 890, 892, 893, 894, 896, 898, 900, 903, 904, 905, 906, 907, 908, 985, 986, 988, 989, 991, 992, 995, 996, 997, 998, 999, 1000, 1001, 1006, 1007, 1026, 1027, 1028, 1030, 1031, 1032, 1033, 1034, 1036, 1037, 1038, 1040, 1041, 1042, 1044, 1046, 1047, 1048, 1049, 1062, 1063, 1064, 1065, 1067, 1068, 1069, 1070, 1071, 1076, 1077, 1078
- activate 894
- Activate cell in excel Worksheet 894
- Activate cell in excel Worksheet Action 894
- Activate License 250
- activates 692, 710, 711, 712, 713, 714, 715, 751, 753, 755, 757, 758, 887, 892, 893, 894
- active 630
- active cell 894
- active directory 195, 203
- Active Mode 776, 778
- Active Worksheet 887, 892, 893, 894
- activities 321
- Actual Internet Explorer 726, 741, 743, 744, 760
- Add 311, 654, 769, 772, 904, 905, 906, 907, 964, 1048
- add an Action 303
- add breakpoints 317
- Add control 338
- Add credential 241
- Add Date or Time 654
- Add DateTime 654
- Add group 217
- Add Item 438, 1010, 1069
- Add Item to List 1069
- Add Item to List Action 1069
- Add New Trigger 155
- Add New Variable 311
- Add New Worksheet 888
- Add New Worksheet Action 888
- Add Number 654
- Add robot 201, 203, 217
- add sidebot 201
- add solobot 203
- Add Text 654
- Add to DateTime 1048
- Add to DateTime Action 1048
- add working item in queue 151, 224, 1010
- add working item in queue action 1010
- Adding an Action 303
- Adding Breakpoints 317
- Additional Data 179
- Administrator 193
- Advanced 452, 473, 501, 502, 504, 506, 507
- Advanced Topics 501, 502, 504, 506, 507
- Agent 769, 772
- all 893
- Allow 642
- allow deny 193
- Allow Empty Selection 642
- Allow Multiple Selection 642, 645
- Alt 630, 632
- amount 589
- amount of seconds 589
- Analyze Entities 842
- Analyze Image 858
- Analyze Sentiment 840
- Analyze Syntax 844
- Analyze Tone 852
- API 514
- append 659, 661
- Append Line 1027
- Append Line to Text 1027
- Append Line to Text Action 1027
- Application 453, 529, 531, 533, 535, 537, 539, 630
- Application Path 529
- App-specific password, Email, Email account, Email Action, Email error, Email Monitor Trigger, extra layer of security, generate password, secondary email address, security, Send Email Action, SMS, trusted device, trusted phone number, Two Step, Two-Step Verification, Verification code 489
- archive 675, 676
- Archive Comment 675
- Archive Path 675, 676

Arguments 529, 1000
 artificial delay 312
 ascending 649
 ASCII 783, 786
 assign permissions 201, 203
 assign roles 201, 203
 assign sidebot 201
 assign solobot 203
 assign to user 203
 assing environment 200
 assing robot to user 201, 203
 Assing to Environment 237
 assing to machine 203
 associated 529
 Attach 155, 726, 741, 880
 attach a Trigger 155
 Attach Excel 880
 Attach to Excel Action 880
 Attachment 797, 806, 808, 985, 986, 988, 989, 991, 992, 995
 attribute 1056, 1057, 1059, 1060, 1061
 Audit 164, 178, 179, 189
 Audit Failure 164
 Audit Success 164
 authentication 478, 797, 806, 808, 985, 986, 988, 989, 991, 992, 995
 Autologin 203
 Automatic distribution 224
 automatically 152, 153, 154, 452, 472, 474, 645
 Automation Browser 726, 741, 743, 744, 760
 Available 952
 avoid 630

- B -

backslash 501
 Basic Logs 178
 BCC 797, 806, 808, 985, 986, 988, 989, 991, 992, 995
 beginning 560, 561, 562, 1031
 between 630
 binary 783, 786
 blank 562, 571, 573, 595
 block 504, 557
 Block Input 617
 Block Input Action 617

block of actions 452, 473, 474, 475, 576
 blocks of actions 474
 Body 797, 806, 808, 985, 986, 988, 989, 991, 992, 995
 Boolean 436
 bottom 321
 Box 634, 636, 638, 640, 642, 645, 647
 braces 630
 brackets 501
 breakpoints 286
 browser 769, 772
 browser identity 769, 772
 Browser Tab 739
 bug 178
 Building 321
 button 309, 318, 712, 713, 714, 715, 755, 757, 758, 760
 Button Pressed 634, 636, 638, 640, 645, 647
 ButtonPressed 304

- C -

Cancel 645, 647
 capitalized 630
 CAPS 633
 CAPS LOCK 633
 Capture 548, 554, 556, 598, 767
 Capture with Delay 598
 Case 953, 955, 956, 1032
 casing 1032
 category 436
 Cautions 878, 880, 884, 886, 887, 888, 889, 890, 892, 893, 894, 896, 898, 900, 903, 907, 908
 CC 797, 806, 808, 985, 986, 988, 989, 991, 992, 995
 cell 587, 894, 896, 898, 900, 903
 Cell Column 884
 Cell Row 884
 Cell Value 880
 certain condition 474
 certain file 589, 590
 Change Extension 654
 Change Text Case 1032
 Change variable dynamically 318
 Change Variable value 318
 Change Working Directory 782

- Change Working Directory Action 782
- changed 165
- changes 168, 435, 673
- changes the name 654
- Char 1030
- Character 1028
- Character Position 1028
- Characters to Use 1038
- check 714, 755
- checkbox 714, 755
- checkboxes 721
- checked 561, 562, 589, 590, 593
- Checker 136
- checking 506
- Checklist 963
- Checkout 462
- checks 474
- Choose 887, 892, 893, 894, 896, 898, 900, 903, 904, 905, 906, 907
- Class 562, 571, 573, 595, 682, 692, 694, 698, 701, 704, 708, 710, 711, 712, 713, 714, 715, 751, 753, 755, 757, 758
- Classify Image 846
- Clear 679, 769, 772
- clear all cookies 769, 772
- Clear Clipboard Contents 679
- Clear Cookies 769, 772
- clear list 1068
- Clear Logs 178
- click 317, 682, 712, 713, 714, 715, 755, 757, 758, 760
- Click Download Link 744
- click element 682
- Click Link 743, 760
- clipboard 435, 548, 767
- Clipboard Category 678, 679
- Clipboard Contents 679
- Clipboard Text 678, 679
- close 634, 636, 688, 708, 878, 886, 995
- close cmd 826
- close cmd session 826
- close cmd session action 826
- close command 826
- close command line session 826
- Close Excel 886
- Close Excel Action 886
- Close FTP 780
- Close FTP Connection Action 780
- close outlook 995
- Close SQL Statement 878
- Close SQL Statement Action 878
- Close Window 708
- Close Window Action 708
- Closing 880, 886
- CMD 824, 826, 828, 829
- codes 508
- Cognitive Action 858, 860, 861, 863, 865, 867, 869, 871, 872, 874
- Cognitive Services 830, 831, 833, 834, 836, 837, 839, 840, 842, 844, 846, 848, 850, 852, 854, 856
- collapse 688
- collections 438
- colored green 309
- column 439, 587, 894, 896, 898, 900, 903, 906, 907
- column names 880
- ColumnsCount 439
- combining 321
- Command 531, 533, 535, 537, 796, 824, 826, 828, 829, 1000
- command line 824, 826, 828, 829
- Command Line Arguments 529, 1000
- Comment 309
- Commit 455, 465
- Commit Details 462
- Commit Process 465
- Common 1076
- Common List Items 1076
- compare 557, 586
- compiled 1000, 1001
- Complete 999
- Completed 284
- complications 138
- Componenets 19
- Component Types 331
- compress 675
- Compression 675, 676
- Compression Category 675, 676
- Compression Level 675
- Computer 543
- concept 452, 472, 474
- concurrency 138

concurrency policy 201, 203
condition 474, 576
conditional 474, 475, 557, 559, 560, 562, 566, 571, 573
conditional block 559, 560, 561, 562, 566, 571, 573
conditional structure 474
Conditionals 453, 474, 475, 557, 559, 560, 561, 562, 566, 571, 573, 576, 577
Conditionals Category 557, 559, 560, 561, 562, 566, 571, 573, 576, 577
configurable 679
Configuring 304
Configuring an Action 304
connect 875, 876, 878
Connection 436, 780, 782, 783, 787, 788, 789, 790, 791, 796, 875, 876, 878
Connection String 875, 876, 878
connects 875, 876, 878
considered 317
Console 638
console application 531
Console Options 502, 504
contains 880
contents 668, 669, 1078
continue 502, 504
Continue Anyway 589, 590, 594, 999
Control 550, 630, 632
Control Desk 19, 120, 121, 143, 190
Control Desk Scheduler 173
Control Desk tabs 124
Control Repository 296, 338, 343
Control Screen Saver 550
Control Screen Saver Action 550
controls 485
convert 1032, 1083, 1084
Convert DateTime to Text 436, 1037
Convert DateTime to Text Action 1037
Convert Document 850
Convert Number 1034
Convert Number to Text 1034
Convert Number to Text Action 1034
Convert Text to DateTime 1036
Convert Text to DateTime Action 1036
Convert Text to Number 1033
Convert to 1032
converted 638
converts 321, 1033, 1078
Cookies 769, 772
coordinates 618, 619, 629, 701
copied cell 900, 903
copies 651, 679
copies a folder 670
copy 286, 670, 900, 903
copy cells in excel Worksheet 900
copy cells in excel Worksheet Action 900
Copy File 651
Copy File(s) 651
Copy File(s) Action 651
Copy Folder 670
Copy Folder Action 670
copy your actions 321
copy/paste 900, 903
Count 439, 1067
cpcolumn 907
Create 667, 790, 953, 957, 1068
Create Folder 667
Create Folder Action 667
Create FTP Directory 790
Create FTP Directory Action 790
Create Function 302
create global variable 1006, 1007
Create New Folder 281
Create New Folder into 667
Create New List 438, 1068
Create New Process 281
Create new tab 739
Create Random Text 1038
Create Random Text Action 1038
Create UI 329
created 165, 589, 590
created on the fly 435
created variable 1062
created/deleted 589, 590
creates 797, 806, 808, 985, 986, 988, 989, 991, 992, 995
creates a new folder 667
Creating 435
Creating Cases 481
Creating Variables 435
CreationTime 436, 439
credential 241, 1015, 1021
Credential Manager 241

Credential Manager tab 241
 criteria 649
 crosshair 562, 566, 571, 573
 crosshairs 562, 566, 571, 573
 CSS 485, 573, 636
 CSV 661
 ctrl + alt + delete 617
 Ctrl-Alt-T 126, 191
 Current 584, 1047
 Current Index 584
 Current Item 587
 current iteration 452, 473
 current position 618, 619
 Current Value 288
 cursor 605, 618
 Custom 654, 1037, 1040, 1041, 1080
 Custom Delimiter 1040, 1041
 Custom Designer 331
 Custom Dialog 329, 636
 Custom Format 1037
 custom messages 284
 Custom Objects 1080, 1083, 1084
 Cut 286
 Cybele zScope 514

- D -

Daily 173, 174
 Dashboards 19
 data 435, 587
 data extraction 485
 data row 452, 472, 587
 data table 439, 452, 472, 587
 data table's scheme 439
 Data Tables 436, 439
 data type 474
 Data type Properties 435, 439
 Data Types 436
 database 178, 191
 Database Category 875, 876, 878
 DataRow 439
 datatable 1067
 DataTable column 1078
 DataTable variable 665
 date 435
 Date and Time 1047
 Date format 444
 Date Time 436, 474
 Date Time type 474
 Dates 173, 1049
 DateTime 288, 439, 654, 1037, 1048
 DateTime Actions 1047, 1048, 1049
 DateTime Actions Category 1047, 1048, 1049
 DateTime Format 654
 DateTime to Add 654
 DateTime to Convert 1037
 DateTime to Text 1037
 Day 439
 DayOfWeek 439
 DayOfYear 439
 days 1048, 1049
 Debug 315, 318
 Debugger 312
 Debugger Options 312
 debugging 286
 Decimal 1034
 Decimal Places 1034
 declare 311
 Decrease 1064
 Decrease Variable 1064
 Decrease Variable Action 1064
 decreases 1064
 default 502, 504
 default policies 139
 Default Printer 540, 541
 default response 502, 504
 Default Value 304, 638, 640
 Define a new variable 435
 Defined in Action 311
 delay 312, 502, 504, 629
 Delay Between Keystrokes 630
 delete 553, 653, 668, 769, 772, 788, 791, 889, 905, 907
 delete column in excel action 907
 delete column in excel worksheet 907
 delete contents 669
 Delete Credential 241
 Delete Environment 235
 Delete File 653
 Delete File(s) 653
 Delete File(s) Action 653
 Delete Folder 668

Delete Folder Action 668
Delete FTP Directory 791
Delete FTP Directory Action 791
Delete FTP File 788
Delete FTP File Action 788
Delete Function 302
delete items from list 1068
Delete Queue 220
delete robot 200
delete row in excel action 905
delete row in excel worksheet 905
Delete User 195
Delete Worksheet 889
Delete Worksheet Action 889
deleted 165, 589, 590
deletes 653, 788, 791
delimiter 1041
Delimiter 1040, 1041
denote 501
Deploy 1091
deployment data 131
Descending 649, 665
Describe Image 860
Description 130, 311, 647
Design Time 294
Design Time Errors 294
Designer 312, 314
Designer Components 331
Designer Elements 331
Designer Options 312
Desktop 138, 544, 680
destination 651, 652, 670
destination folder 651, 670, 672
Destination Folder: 676
details 179
Detect Face 865, 867
Detect Faces 848
Detect Language 869
Development 455
Dialog 126, 636, 638, 640, 642, 645, 647
dialog box 642
Dialog Message 304
Dialog Title 304, 642
differ 598, 621, 746
difference 1049, 1077
different 502, 504
different response 502, 504
different Robot 502, 504
different ways 502, 504
differentiate 309
directory 663
Directory to Delete 791
disable 284
Disable Action 306
Disable Process 128
Disable Robot 152, 153, 154
disabled 128, 152, 153, 154, 286
disables 550
Discard 455
Display 634, 636, 638, 640, 642, 645, 647
Display custom dialog 636
Display dialog Action 636
Display Input Dialog 638
Display Input Dialog Action 638
Display Message 634
Display Message Action 634
Display Notification 284
Display Notification Action 284, 638
Display Select Date 640
Display Select Date Dialog 640
Display Select File Dialog Action 645
Display Select Folder Dialog 647
Display Select Folder Dialog Action 647
Display Select From List Dialog Action 642
distinction 435
DNS 550
Do not wait more than 529
document 529
Document Path 878, 880, 886
Document to Print 540
documentation 130
domain 550
DOS 531
DOS Command 531
double backslashes 501
down 621, 746
Download 744
Download File 783
Download File(s) 783
Download File(s) from FTP 436, 783
Download Folder 785
Download Folder(s) 785

Download Folder(s) from FTP Action 785
Download from Web 769
Download from Web Action 769
Download into Folder 783, 785
Downloaded File 769
downloads 769
dpi 554, 556
drag 686
drag and drop 686
drag and drop element in window 686
drop 686
dropdown 715
Dynamic change 318

- E -

Each 454, 587
each Action 312
each number 654
easier interface 321
Edit 769, 772
Edit Credential 241
Edit Environment 235
Edit Folder 281
Edit Process 281
Edit Process Properties 128
Edit Queue 220
edit robot 200
Edit Triggers 155
Edit User 195
edited 321
effects 304
efficient 506
element 571, 573, 608, 682, 686, 712, 713, 714, 715, 719, 721, 722, 723, 755, 757, 758, 762, 766, 1053, 1055, 1056, 1057, 1059, 1060, 1061
elements 1041
Else 474, 576
Else Action 576
Else If 576
Else If Action 576
email 139, 159, 985, 995
email address 159
Email Category 797, 806, 808, 985, 986, 988, 989, 991, 992, 995
email message 797, 806, 808, 985, 986, 988, 989, 991, 992, 995
Email Trigger 159
empty 669, 1068
Empty Folder 669
Empty Folder Action 669
empty list 1068
Empty Recycle Bin 547
Empty Recycle Bin Action 547
empty temporary file 664
empty text 153, 154
Enable 128, 306
Enable Action 306
Enable Process 128
Enable/Disable Actions 306
enables 550
Enate 477, 478, 480, 481, 483, 951, 952, 953, 955, 956, 957, 958, 960, 961, 962, 963, 964, 965, 966
encrypted 241
end 1031
End If 474, 577
End If Action 577
End Loop 588
End Loop Action 588
end of a conditional 577
End Robot 615, 617
End Robot Action 615, 617
End To 584
EndIf 474
endless loop 453
Engine 931, 934, 937
enter 1007
Enter License 250
enter locked region 1007
enter locked region action 1007
Enter Password 776
entire screen 548, 554, 556
Environment 235
environmental 552, 553
environmental variable 552, 553
Environments Tab 235
equal 474
Erroneous 248
Erroneous State 136
Error 178, 179, 320, 510
Error Handling 139, 248, 502, 504

- error icons 315
 - Error Message 614, 615, 617
 - Errors 248
 - Errors Pane 288, 294
 - Escape variable 501
 - event 165, 629
 - Event coded 189
 - Event Details 179
 - Event Id 179
 - Event ids 189
 - Event Log 164
 - Event Log Monitor 164
 - Event Log Monitor Trigger 164
 - Event Source 164
 - Events 189
 - Events list 179
 - EventTriggerEventID 164
 - EventTriggerEventLogName 164
 - EventTriggerEventMessage 164
 - EventTriggerEventSource 164
 - EventTriggerEventType 164
 - every item 454
 - exact number 452, 473
 - Excel 436, 878, 880, 884, 886, 887, 888, 889, 890, 892, 893, 894, 896, 898, 900, 903, 904, 905, 906, 907, 908
 - Excel Category 878, 880, 884, 886, 887, 888, 889, 890, 892, 893, 894, 896, 898, 900, 903, 904, 905, 906, 907, 908
 - Excel document 878
 - excel file 436
 - Excel Instance 288, 436, 878, 884, 887, 892, 893, 894, 896, 898, 900, 903, 904, 905, 906, 907, 908
 - Exception 502, 504, 595
 - Exception Block 504
 - Exception Handling 502, 504, 510, 598
 - exceptions 294, 502, 504, 1003
 - Exclude Mask 676
 - executable 153, 154, 1001
 - execute 318, 612, 747, 876, 1052
 - Execute Next 317, 318
 - Execute Next Action 317, 318
 - Execute SQL Statement 876
 - Execute SQL Statement Action 876
 - Execute step by step 318
 - executed 474, 587
 - executes 529, 999
 - executing 529
 - execution 312, 614, 615, 617
 - existing folder 668, 669, 672
 - existing text 1031
 - Exists 439, 559, 560
 - exit 587, 1009
 - Exit Code 529, 531
 - Exit FUnction 613
 - exit locked region 1009
 - exit locked region action 1009
 - Exit Loop 587
 - Exit Loop Action 587
 - Exit ProcessRobot 153, 154
 - expand 688
 - Export 283
 - Export Process 283, 462
 - ext: 444
 - extension 663
 - External 444
 - External Variables 444, 501
 - extract 485, 717, 719, 721, 722, 723, 762, 765, 766, 911, 914, 917, 921, 940
 - Extract data from web 762
 - extract data from web page 762
 - Extract data from web page action 762
 - Extract data from window 723
 - Extract data from window action 723
 - extract from PDF 917, 921
 - extract images from PDF 917, 921
 - extract images from PDF action 917, 921
 - extract text from PDF 911
 - extract text from PDF action 911
 - extract text from PDF with OCR 914
 - extract text from PDF with OCR action 914
 - extract text with OCR 940
 - extract text with OCR action 940
 - extraction 723, 762
- F -**
- Face 865, 867
 - Face - Detect 865
 - Face Detection 830
 - Fail 598
 - fail that Robot 502, 504

- failure 167
- false 453
- feature 598, 621, 746
- field 710, 711, 751, 753
- File 435, 548, 767, 769, 772, 964, 965
- File Category 649, 651, 652, 653, 654, 657, 659, 661, 663, 664
- File Dialog 645
- File Exists 559, 645, 651, 654
- File Filter 645, 649
- file from the web 769, 772
- File Monitor 165
- File Monitor Trigger 165
- File Name 663
- File Name without Extension 663
- file object 664
- File Path 559, 589, 657, 659, 661, 663
- File to Copy 651
- File to Delete 788
- File to Download 785
- File to Move 652
- File to Play 546
- File to Rename 654, 789
- File to Upload 786
- File(s) to Delete 653
- File(s) to Download 783
- File(s) to Zip 675
- file's properties 436
- filename 663
- files 668, 669
- Files Category 649, 651, 652, 653, 654, 657, 659, 661, 663, 664
- Files in Folder 649
- Filestream 108
- FileTriggerEventType 165
- FileTriggerFileName 165
- FileTriggerFilePath 165
- FileTriggerOldFileName 165
- fill in 711, 753
- find 598, 1044, 1046
- Find Common List Items 1076
- Find Common List Items Action 1076
- Find Window 595, 692
- fired 164, 165
- fires 165, 167
- First 586
- First Free 907, 908
- First Free Column 907
- First Free on Column 908
- First Free Row 907, 908
- First line 880
- First Occurrence 1042
- First Operand 557, 586
- fixed 1030
- fixed length 1030
- FlexiCapture 937
- Flow Control 611, 614, 615, 617
- Flow Control Category 611, 614, 615, 617
- Focus 692, 710, 751
- Focus text field 710, 751
- Focus text field Action 710, 751
- Focus Window 692
- Focus Window Action 692
- Folder 165, 288, 436, 439, 560, 647, 649, 665, 667, 668, 669, 670, 672, 673, 674, 787
- Folder Category 665, 667, 668, 669, 670, 672, 673, 674
- Folder Dialog 647
- Folder Exists 560
- Folder Path 560
- Folder Security 143
- Folder to Copy 670
- Folder to Delete 668
- Folder to Empty 669
- Folder to Move 672
- Folder to Rename 673
- Folders Category 665, 667, 668, 669, 670, 672, 673, 674
- Follow Redirection 769, 772
- For 454, 587
- For each 452, 472
- For Each Action 587
- Force 542, 543
- Force Log Off 542
- Force Selected Action 543
- foreground 598, 692
- Foreground Window 548, 554, 556, 621, 690
- foremost 618, 619, 621, 629
- format 769, 772, 1034, 1037
- Format to Use 1037
- four responses 502, 504
- Free 907, 908

Free Column 907, 908
 frequency 176
 From 797, 806, 808, 985, 986, 988, 989, 991, 992, 995
 FTP 436, 776, 778, 780, 781, 782, 783, 785, 786, 787, 788, 789, 790, 791, 792, 796
 FTP Category 776, 778, 780, 781, 782, 783, 785, 786, 787, 788, 789, 790, 791, 792, 796
 FTP Command 796
 Ftp Connection 789
 FTP Connection to close 780
 FTP Directory 790
 FTP File 288, 788, 789
 FTP Folder 288
 Ftp Server 776, 778
 FTPFile 439
 FTPFolder 439
 Full Error View 320
 full path 589
 FullName 439
 Function 153, 154, 301, 302
 Functions 301, 507, 612, 613

- G -

gear 435
 General 190, 191, 201, 203
 General Options 191
 General tab 126
 generate 438
 generating 1038
 Get 540, 552, 554, 618, 678, 911, 914, 917, 921, 940, 955, 958, 961, 1000, 1001, 1007, 1015, 1026, 1028, 1067
 Get Active Worksheet 892
 Get Active Worksheet Action 892
 Get Active Worksheets 893
 get all 893
 Get all Active Worksheets Action 893
 Get Clipboard Text 678
 Get Clipboard Text Action 678
 Get Command Line Arguments 1000
 get credential 1015
 get credential action 1015
 Get Current Date and Time 1047
 Get Current Date and Time Action 1047
 Get Data 723, 762
 get decimal part 1064
 Get Default Printer 540
 Get Default Printer Action 540
 get details of element on web 766
 Get Details of element on web page 766
 Get Details of element on web page action 766
 get details of web 765
 Get Details of web page 765
 Get Details of web page action 765
 Get Details of Window 717
 Get Details of Window action 717
 Get Details of Window element 719
 Get Details of Window element action 719
 Get Difference in 1049
 get exception 1003
 Get FilePath 663
 Get FilePath Part 663
 Get FilePath Part Action 663
 Get Files 649
 Get Files in Folder Action 649
 Get First Free Column/Row 907
 Get First Free Column/Row Action 907
 Get First Free Row on Column/Row 908
 Get First Free Row on Column/Row Action 908
 get global Variable 1007
 get global Variable Action 1007
 get integer part 1064
 get item 1012
 Get Items Count 1067
 Get Items Count Action 1067
 Get Mouse Action 618
 Get Mouse Position 618
 Get Process Path 1001
 Get Process Path Action 1001
 get selected 898
 get selected cell 898
 get selected cells in excel Worksheet 898
 get selected cells in excel Worksheet Action 898
 Get selected checkboxes in Window 721
 Get selected checkboxes in Window action 721
 Get selected radiobuttons in Window 722
 Get selected radiobuttons in Window action 722
 Get Special Folder 674
 Get Special Folder Action 674
 Get Subfolders 665

Get Subfolders in Folder 665
 Get Subfolders in Folder Action 665
 Get Subtext 1028
 Get Subtext Action 1028
 Get Temporary File 664
 Get Temporary File Action 664
 Get Text 769, 772
 Get Text Length 1026
 Get Text Length Action 1026
 get Value 1007
 Get Window 690
 Get Working Item 151
 get working item in queue 224, 1012
 get working item in queue action 1012
 given dates 1049
 global 237, 552, 553
 global variable 552, 553
 Global Variables Tab 237
 go to 299, 502, 504, 741
 Go To Action 611
 Go to line 299
 go to web 741
 Google 830, 831, 833, 834, 836, 837, 839, 840, 842, 844, 931
 Google API 830, 831, 833, 834, 836, 837, 839, 840, 842, 844
 Google Cloud Natural Language service 840, 842, 844
 Google Cloud Storage 830, 831, 833, 834, 836, 837, 839, 840, 842, 844
 grab 618, 619
 gray column 317
 grayed-out 128
 greater 474
 Group of Robots 217
 Group policy 199
 Group Robots 199
 Group tab 199
 Groups 199
 GUI Components 331

- H -

Handle 529, 595, 682, 692, 694, 698, 701, 704, 708, 710, 711, 712, 713, 714, 715, 757, 758
 handle an exception 502, 504

Handling 502, 504, 595
 hidden 682, 698
 Hide 544
 hides 682, 698
 highlighted 286, 321
 highlighting 303
 History 124, 462
 HLLAPI 514
 hold 618, 619
 holidays 174
 Home 121
 homonym 453
 host 167
 HotKeys 286
 Hour 439
 hourglass 605
 hours 1048, 1049
 hover 746
 hover Mouse 746
 hover Mouse to element 746
 HTML 573, 636, 797, 806, 808, 985, 986, 988, 989, 991, 992, 995
 http 769, 772
 HTTP Authentication 769, 772
 https 769, 772

- I -

IBM 846, 848, 850, 852, 854, 856
 IBM API 846, 848, 850, 852, 854, 856
 IBM Document Conversion 850
 IBM Language Translator 854, 856
 IBM Personal Communications 514
 IBM Tone Analyzer 852
 IBM Visual Recognition Service 846, 848
 icon 435
 Id 189
 ideal 452, 473
 identify 436
 Identify Language 856
 identity 769, 772
 Idle 167
 Idle Monitor 167
 Idle Monitor Trigger 167
 IE Enhanced Security Configuration 391
 IE security settings 383

- IE settings Server 391
- If 474, 475, 557, 559, 560, 561, 562, 566, 571, 573, 576, 577
- If Action 557
- If File 559
- If File Exists 475, 559, 652, 659, 661, 783, 786
- If File Exists Action 559
- If Folder 560
- If Folder Exists 475, 560, 670
- If Folder Exists Action 560
- If Image 475
- If Image Action 566
- If Process 475, 562
- If Process Action 562
- If Service 475, 561
- If Service Action 561
- If Variations 475
- If Web page Contains 573
- If web page Contains Action 573
- If Webpage contains 475
- If Window 475, 562
- If Window Action 562
- If Window Contains 571
- If Window Contains Action 571
- If/Else/EndIf 474
- image 297, 548, 566, 767, 830, 831, 833, 834, 836, 837, 839, 840, 842, 844, 846, 848, 850, 852, 854, 858, 860, 861, 917, 921
- image does not appear 598
- Image File 548, 767
- Image Format 548, 767
- Image Properties Detection 837
- image recognition 506, 566, 598, 621, 917, 921
- Image repository 297
- IMAP 159, 797, 806
- implements 452, 473
- Import 283
- Import Process 283
- Include 649
- Include Mask 676
- Include Subfolders 649, 665
- Increase 1063
- Increase Variable 1063
- Increase Variable Action 1063
- Increment 584, 654
- indent 452, 472, 474
- index 1070
- Indicator 1005
- information 435
- Initial 647
- Initial Folder 645, 647
- Initial Value 288, 311
- input 138, 167, 435, 507
- Input Dialog 638
- Input Type 304, 638
- insert 904, 906, 1060, 1061
- insert column 906
- insert column in excel action 906
- insert column in excel worksheet 906
- insert row in excel action 904
- Insert row in excel worksheet 904
- installed 561
- Instance 878, 880, 884, 887, 888, 889, 890, 892, 893, 894, 896, 898, 900, 903, 904, 905, 906, 907, 908
- instances 138
- interacting 545
- interactions 138
- interactive 138
- Interactive Robot 321
- Interactive/Non Interactive Robots 507
- interacts 507
- Internal Exception 178
- internet 550
- Internet Cache 674
- Internet Explorer 351, 383, 391, 393, 726, 741, 743, 744, 760
- internet protocol 550
- interval 167, 176
- into global Variable 1006, 1007
- into Variable 1062
- Intoduction 19
- Introduction 276
- Invoke 612, 772, 796
- Invoke FTP Command 796
- Invoke FTP Command Action 796
- IP 550
- IP address 167
- Is Regular Expression 1041
- IsArchive 439
- IsHidden 439
- IsReadOnly 439
- IsSystem 439

item 587
ItemNumber 438
items 438
Items Count 1067
iterate 454
iterates 584, 586
iterating 438
iteration 452, 472

- J -

Java 535
Javascript 535, 747
Join 1040
Join Text 1040
Join Text Action 1040
Jquery 535
Json 1083, 1084
jumps 611

- K -

Keep Extension 654
Key 632, 633, 1005
key combination 152, 153, 154, 191
Key Phrases 871
keyboard 507, 545
Keys 508, 590, 630
keystrokes 507, 590, 630
kill 536, 539
KPI 1005
KPI update 1005

- L -

label 611
Label Detection 831
Label Name 611
Landmark Detection 833
last 1003
last Working Day of Month 174
LastAccessed 439
LastModified 439
Late 174
launch 321, 529, 726, 741, 878, 985
Launch browser 726, 741

Launch Excel 436, 878
Launch Excel Action 878
launch new internet explorer 726
launch outlook 985
Launched 152, 153, 154, 169
Layout 314
left 1030
length 1030
less time 506
less-common color 506
Levels 510
License 250
Lifecycle 124, 455
Lifecycle stages 455
Limit 642
Limit to List 642
Line 299, 1000, 1027
Link 743, 744, 760
list 288, 454, 587, 649, 758, 1040, 1067
List 1 1076
List 2 1076
List Dialog 642
List FTP 781
List FTP Directory 781
List FTP Directory Action 781
list notation 439
list of files 649, 651
list of subfolders 665
List to choose from 642
List to Join 1040
lists 1076
literal 557, 586, 796
location 611, 717, 765, 766
Lock 545
Lock Workstation 545
Lock Workstation Action 545
locked 617, 1007, 1009
locked region 1007, 1009
Log 164, 542
log database 178
log file 178
Log Off 542
Log Off User 542
Log Off User Action 542
logged 178, 507
logged in 167

logged on 321
 logged out 507
 Logging 178, 189
 login 951, 1015, 1021
 Logo Detection 836
 Logout 966
 Logs 179, 189, 191
 logs off 542
 Logs Window 179
 Loop 317, 452, 453, 472, 584, 586, 587, 588
 Loop Action 584
 Loop Condition 452, 453, 472, 586
 Loop Condition Action 586
 LoopIndex 452, 473
 Loops 452, 472, 473, 584, 586, 587, 588
 Loops Category 584, 586, 587, 588
 lowercase 1032

- M -

Macro Recorder 281, 321
 Main Window Handle 529, 694
 Make Instance Visible 878
 Make Sequential 654
 Maker 136
 Manage Functions 302
 Manage Process Versions 455
 Manage Repository 343
 manage robots 199
 Manager 311
 manually 152, 153, 154
 Massive Rename 319
 match 506
 matches 506
 maximizes 694
 Maximum 1038
 Maximum Length 1038
 menu 621, 682, 683, 686, 688, 746
 Menu Option 683
 Merge list 1073
 message 797, 806, 808, 985, 986, 988, 989, 991, 992, 995
 Message Boxes 634, 638, 640, 642, 645, 647
 Message Boxes Category 634, 636, 638, 640, 642, 645, 647
 met 474
 Method 769, 772
 Micro Focus Reflection 514
 MicroFocus Reflection 514
 MicroFocus RUMBA 514
 Microsoft 934
 Microsoft Cognitive Service 858, 860, 861, 863, 865, 867, 869, 871, 872, 874
 Minimize all Windows 544
 minimizes 694
 Minimum 1038
 Minimum Length 1038
 Minute 439
 minutes 1048, 1049
 mode 138
 MODI 934
 modified 165
 modifier 632
 Modules 537
 Monitor 121, 164, 165, 167, 168, 169, 544, 554, 556
 monitor email 159
 monitor email accounts 159
 Month 439
 Monthly 173, 174
 More 953
 More Information 674
 Mouse 605, 617, 618, 619, 621, 629, 630, 632, 633, 746
 Mouse and Keyboard 321, 617, 618, 619, 621, 629, 630, 632, 633
 Mouse and Keyboard Category 617, 618, 619, 621, 629, 630, 632, 633
 Mouse Click 629
 mouse events 507
 mouse movement 138
 Move 619, 621, 629, 652, 672, 701, 746
 Move File 652
 Move File(s) 652
 Move File(s) Action 652
 Move Folder 672
 Move Folder Action 672
 Move Mouse 619, 621, 746
 Move Mouse Action 619
 Move Mouse to element 746
 Move Mouse to Image 506, 621
 Move Mouse to Image Action 621

Move Process 455
Move to Development 124
Move to Production 124, 455
Move to Review 124, 455
Move Window 701
Move Window Action 701
moved 652
MS outlook 985, 986, 988, 989, 991, 992, 995
multiple 452, 472
Multiple Instances 138
Multiple Selection 645
My Pictures 674

- N -

Name 288, 311, 439, 561, 611, 717, 765, 766, 887, 888, 889, 890, 892, 893, 894, 896, 898, 900, 903, 904, 905, 906, 907, 1078
name of a folder 673
NameWithoutExtension 439
navigation 351, 393
network 550
New 888, 1068
New Credential 241
new data 907, 908
New Directory 790
New Environment 235
New Extension 654
New File Name 654, 789
New Folder Name 667, 673
New Instance 726, 741
new line 1031
New List 1068
New Name 654
new Process 281
New Queue 220
New robot 200
New Row 904
new size 704
New Tab 739
New User 195
New Variable 311
New Window 704
New Window Position 701
New Window Size 704
New Worksheet 888
Next Action 318
Next Scheduled Run 174
no affect 309
no input 167
no Logs 315
no properties 303
no yellow and white fields 435
node 688
Non Interactive 507
non-interactive 138, 507
not 562, 571, 573
not met 576
not using 167
notation 438, 439
notes 130, 309
notes to yourself 309
Notification 284, 638
notification popup 638
Notifications tab 284
NUM 633
NUM LOCK 633
number 439, 584
Number of Chars 1028
number of times 584
Number to Convert 1034
Number to Text 1034
Numerical 288

- O -

Object 1080
objects 649
Occurrence 1042
occurrences 1031, 1042, 1044, 1046
ocr 861, 914, 931, 934, 940
Off 633
Offset 621, 746
Offset X 621, 746
Offset Y 621, 746
OK 647
On 633
one or more actions 452, 472
one or more files 652
Open 645, 688, 776, 778, 875
open cmd 824
open cmd session 824

open cmd session action 824
 open command 824
 open command line session 824
 open connection 780
 Open FTP Connection 436, 776
 Open Secure FTP Connection 778
 Open Secure FTP Connection Action 778
 Open SQL Statement 875
 Open SQL Statement Action 875
 opens 595
 operand 474
 operands 474
 operating 632
 operator 474
 optical character recognition 914, 940
 option 683, 713, 714, 715, 755, 757, 758
 Options 191, 312
 Organize 307
 Original 1027, 1028
 originally 621, 746
 outlook 985, 986, 988, 989, 991, 992, 995
 output 138, 438, 507, 531
 Output property 311
 overwrite 1006, 1007, 1062

- P -

Pad 1030
 Pad Text 1030
 Pad Text Action 1030
 Padding 1030
 Pane 288, 294, 297, 318
 panic button 126
 Parameters 769, 772
 Parent 439
 Parse 911, 914, 917, 921, 940, 1042
 Parse Text 1042
 Parse Text Action 1042
 parses 1042
 passing data 444
 password 241, 675, 676, 776, 778, 797, 806, 808, 985, 986, 988, 989, 991, 992, 995, 1015, 1021
 passwords 1038
 paste 286, 900, 903
 paste cells in excel Worksheet 903
 paste cells in excel Worksheet Action 903
 Path 878, 880, 886, 1001
 pattern 1042
 Pause 997
 Pause Service 997
 Pause Service Action 997
 paused 561, 594, 998
 pauses 168, 589, 590, 594, 595, 997
 pdf 911, 914, 917, 921, 940
 percent 501
 percent sign 501
 percentage 435
 percentage characters 435
 Perform 543
 permissions 143, 193
 ping 167, 550
 Ping Trigger 167
 PingTriggerHostName 167
 pixel 506, 629
 pixel coordinates 629
 pixels 621, 701, 704, 746
 placed 474
 Play 546
 Play Sound 546
 Play Sound Action 546
 PcessRobot Actions 999
 Pools 217
 Pools settings 217
 populate 711, 753
 Populate text field 711, 753
 Populate text field Action 711, 753
 Popup 638
 Port 776, 778, 797, 806, 808, 985, 986, 988, 989, 991, 992, 995
 position 618, 619, 629, 701
 POST 769, 772
 Post Parameters 769, 772
 Post's Parameter 769, 772
 Power 536
 PowerShell 536
 preceding 576
 predefined Trigger 126
 press 632, 712, 760
 press button 712, 760
 press button in window Action 712
 Press Key 632
 Press/Release Key 632

Press/Release Key Action 632
pressed 632, 645
prevent 545
previously 1006, 1007, 1062
Print 540
Print Document 540
Print Document Action 540
printer 540
Printer Name 540, 541
Process 169, 536, 539, 552, 553, 562, 593, 806, 989, 991, 992, 1001, 1005
Process Category 1001
Process Designer 152, 153, 154
Process Email 806, 989, 991, 992
Process Email Action 806
Process Id 529
Process Lifecycle 124, 455
Process Monitor 169
Process Monitor Trigger 169
Process Name 562, 593
Process' Path 1001
Process Properties 126, 128
Process Properties Dialog 128
process robot 1006, 1007
Process Robot Actions 1001
Process Robot Requirements 22
Process Studio 19
Process to Run 999
Processes 143
Processes Tab 124, 143
ProcessRobot Actions Category 999
ProcessRobot Log 1002
ProcessRobot Service 153, 154
ProcessStudio 276
ProcessTriggerEventType 169
ProcessTriggerProcessName 169
Production 124, 455
programming 439, 452, 472
Prompt 640
Properties 303, 439, 529, 531, 543, 548, 554, 556, 767
Properties Box 304
Protocol 550, 778
Python 537

- Q -

quarter 629
queing 1007, 1009
Query 875, 876, 878
Query Result 875, 876, 878
queue 224, 1010, 1012
Queue Name 220
Queues 151, 220, 224
Queues Tab 151
queuing 1010, 1012
quicker 506

- R -

radio 713, 722, 757, 758
radiobutton 722
Random 1038, 1065
Random List 1065
Random Number 1065
Random Text 1038
range 880
rdesktop 514
read 880, 911, 914, 917, 921, 940, 1052, 1053, 1055, 1056, 1057, 1059, 1060, 1061
read cmd 828
read cmd session 828
read cmd session action 828
read command 828
read command line session 828
Read from Excel 436, 880
Read from Excel Action 880
read from PDF 911, 914
Read Text 657
Read Text from File Action 657
read xml 1050
read xml from file 1050
read xml from file action 1050
reboots 543
Recognition 506
recognize 506
record 321, 502, 504, 619
record a mouse movement 619
record of the exception 502, 504
recorded 315

- Recording 321
- Recording vs Building a Robot 321
- records 321
- Recurring 173, 176
- Recurring schedule 176
- Recycle Bin 547
- red dot 286, 317
- redirect 769, 772
- Redirection 769, 772
- Reference 529, 531, 540, 541, 542, 543, 545, 546, 547, 548, 550, 554, 556, 557, 559, 560, 561, 562, 566, 571, 573, 576, 577, 584, 586, 587, 588, 589, 590, 593, 594, 595, 598, 605, 611, 614, 615, 617, 618, 619, 621, 629, 630, 632, 633, 634, 636, 638, 640, 642, 645, 647, 649, 651, 652, 653, 654, 657, 659, 661, 663, 664, 665, 667, 668, 669, 670, 672, 673, 674, 675, 676, 678, 679, 682, 692, 694, 698, 701, 704, 708, 710, 711, 712, 713, 714, 715, 746, 751, 753, 755, 757, 758, 767, 769, 772, 776, 778, 780, 781, 782, 783, 785, 786, 787, 788, 789, 790, 791, 792, 796, 797, 806, 808, 875, 876, 878, 880, 884, 886, 887, 888, 889, 890, 892, 893, 894, 896, 898, 900, 903, 904, 905, 906, 907, 908, 986, 988, 989, 991, 992, 995, 996, 997, 998, 999, 1000, 1001, 1006, 1007, 1026, 1027, 1028, 1030, 1031, 1032, 1033, 1034, 1036, 1037, 1038, 1040, 1041, 1042, 1044, 1046, 1047, 1048, 1049, 1062, 1063, 1064, 1065, 1067, 1068, 1069, 1070, 1071, 1076, 1077, 1078
- regain 617
- regain control 617
- regional settings 436
- Regions 307
- regular expression 1041, 1042
- Regular Expressions 1044, 1046
- regular list notation 439
- Reject 963
- relationship 586
- relative 618, 619, 621, 746
- Release 632
- Release Key 632
- released 632
- remember 130
- remote computer 550
- remove 654, 1056, 1057, 1059, 1061
- remove column 907
- Remove Item 1070
- Remove Item from List 1070
- Remove Item from List Action 1070
- remove items from list 1068
- remove Row 905
- Remove Text 654
- Rename 319, 654, 673, 789, 890
- Rename File 654
- Rename File(s) 654
- Rename File(s) Action 654
- Rename Folder 673
- Rename Folder Action 673
- Rename FTP File Action 789
- Rename Scheme 654
- Rename Worksheet 890
- Rename Worksheet Action 890
- Renamed 165
- repeat 452, 472
- repeated 452, 473
- replace 1044, 1046
- Replace Text 654, 1044, 1046
- Replace Text Action 1044, 1046
- Replace with 654
- Reply Code 796
- Reply Text 796
- Repository 297, 338
- Resize 704
- Resize Window 704
- Resize Window Action 704
- resolution 554, 556
- resolve 550
- resource-intensive 167
- respond 167, 992
- response 502, 504
- Restart 543
- Result 1030
- results 649
- Resume 998
- Resume Service 998
- Resume Service Action 998
- resumes 998
- Retrieve 438, 439, 665, 797, 986, 988
- Retrieve DataTable Column 1078
- Retrieve DataTable Column into List 1078
- Retrieve Email 797, 986, 988
- Retrieve Email Action 797
- Retrieved Files 649
- retrieves 674, 1067
- Retrieving Work 480

retry 502, 504
Return 613, 769, 772
Return Data 769, 772
Return to Console 281
Reveal Desktop 544
reverse list 1074
Review 455
right 621, 746, 1030
right-click 306
right-clicking 306
Robot Designer 312, 315, 638
Robot Designer Options 286, 312
Robot is Late 174
Robot manually 317
Robot Pools 217
Robot Properties 201, 203, 507
Robot Trigger 165
Robots 191, 200, 507
Robots Tab 147, 200, 277
RocketSoftware BlueZone 514
Roles 143, 193
Roles and Permissions 143
Root Path 663
RootPath 439
Rotate Excel Data 452, 473
round number 1064
Row 880, 884, 894, 896, 898, 900, 903, 904, 905, 907, 908
rows 439
RowsCount 439
Run 529, 999
run a different Robot 502, 504
Run another Robot 139
Run Application 529
Run Application Action 529
Run DOS Command 531
Run DOS Command Action 531
Run Function 612
run immediately 174
run on schedule 173
Run Process 126
Run Properties 126, 138, 507
Run Time Errors 294
running 288, 561, 562, 593, 997
Run-Time Error 502, 504

- S -

Safe Search Detection 839
same type 438
SAP, GUI, SAP GUI, GUI Scripting, SAP GUI Scripting, S_SCR, SAP Server, RZ11, Transaction RZ11 517
save 965, 991
Save and Commit 455, 465
Save as New Process 462
Save File 769
Save File to 769
Save Screen Shot 548, 554, 556, 767
Save to Disk 769
saves 548, 554, 556, 767
Schedule 173, 174, 176
Schedule Expire 174
Schedule Trigger 174
Schedule Trigger Properties 174
Scheduler 173
Scheme 131, 654
screen 506, 544, 548, 554, 556, 598, 621, 767
Screen Saver 550
screensaver 550
ScreenShot 548, 554, 556, 767
Script 533, 535, 537
SCROLL 633
SCROLL LOCK 633
Search 598, 621
searched 621
second 629
Second Operand 557, 586
seconds 589, 1048, 1049
secure connection 778
Secure FTP 436, 778
Secure FTP Connection 436
Secure FTP Protocol 778
security 143, 201, 203
select 640, 642, 647, 683, 686, 688, 713, 743, 744, 757, 760, 896
select any variable 435
select cells in excel Worksheet 896
select cells in excel Worksheet Action 896
Select Date 640
Select Desktop 680

Select File 645
Select File Dialog 645
Select Folder 647
Select Folder Dialog 647
Select From List 642
select machine 203
Select menu option action 683
select option 683
select radio 713, 757
select radio button 713, 757
select radio button in window Action 713, 757
select user 203
select web element 353, 393
selected 640, 642, 647, 721, 722
Selected Action 543
selected cell 896
Selected Date 640
Selected Folder 647
Selected Index 642
Selected Item 642
selected Worksheet 896, 898, 900, 903
selection 642, 683, 896, 898, 900, 903
selectors 485
send 629, 630, 806, 808, 986, 988
Send an email 139
Send Email 808
Send Email Action 808
Send Keys 508, 590, 630
Send Keys Action 630
Send Mouse Click 629
Send Mouse Click Action 629
Sender 797, 806, 808, 986, 988, 989, 991, 992
sends 797, 806, 808, 986, 988, 989, 991, 992
sentence case 1032
Sentiment 872
separate list items 1040
separates 1041
Separator 654, 1034
Service 168, 561, 594, 772, 995, 996, 997, 998
Service Category 995, 996, 997, 998
Service Monitor 168
Service Monitor Trigger 168
Service Name 561, 594
Service to Pause 997
Service to Resume 998
Service to Start 995
Service to Stop 996
Services Category 995, 996, 997, 998
ServiceTriggerServiceName 168
ServiceTriggerServiceState 168
session 824, 826, 828, 829
Set 541, 552, 556, 633, 679, 682, 694, 698, 714, 715, 755, 758, 1006, 1059, 1062
set a Variable 502, 504
Set Active Worksheet 887
Set Active Worksheet Action 887
set checkbox state on web page 755
set checkbox state on web page Action 755
set checkbox state in window Action 714
Set Clipboard Text 679
Set Clipboard Text Action 679
Set Credential 241
Set Default Printer 541
Set Default Printer Action 541
set drop down list 758
set drop down list value in web page 758
set drop down list value in web page Action 758
set dropdown state in window Action 715
Set global Variable 1006
Set global Variable Action 1006
Set Key 633
Set Key State 633
Set New Name 654
set number of seconds 589, 590, 594
set number of times 452, 473
set of Actions 321
set screen 556
set state 714, 715, 755
Set Value 1006, 1062
Set Variable 1062
Set Variable Action 1062
Set Window 682, 698
Set Window State 694
Set Window State Action 694
Set Window Visibility 682, 698
Set Working Directory to 782
sets 704
Settings 190, 235
Settings Tab 235, 237
shedule 173
Shift 630, 632
Shutdown 543

- Shutdown Computer 543
- Shutdown Computer Action 543
- shuts 543
- shuts down 543
- Sidebot 147, 199, 200, 201, 1091
- Sidebot Interface Window 1090
- Sidebot Introduction 1089
- sign 501
- similar actions 769, 772
- Simple Loops 452, 472, 473
- simulate 321
- Simulating 321
- site 573
- situation 502, 504
- size 312, 554, 556, 704, 717, 765, 766
- skip 174, 474
- skipped 474
- smaller 474
- SMTP 159, 808, 986, 988, 989, 991, 992
- SMTP Server 139
- Solobot 147, 199, 200, 203
- Sort 649, 1071
- Sort by 649
- Sort List 1071
- Sort List Action 1071
- Sound 546
- Sound to Play 546
- source 179
- space 1031
- special 435, 501
- special "gear" icon 435
- Special Folder 674
- Special Folder Name 674
- special kind 309
- specific 439, 562, 584, 692, 710, 711, 712, 713, 714, 715, 751, 753, 755, 757, 758
- specific dates 173
- specific time 152, 153, 154, 174
- specified 589
- Spell Check 874
- Split 1041
- Split Text 1041
- Split Text Action 1041
- SQL 108, 875, 876, 878
- SQL Category 875, 876, 878
- SQL Server 108
- SQL Statement 875, 876, 878
- SSL 797, 806, 808, 986, 988, 989, 991, 992
- standalone 152, 153, 154
- Standard 1037, 1040, 1041
- Standard Delimiter 1040, 1041
- Standard Format 1037
- Start 315, 584, 880, 995
- Start button 315
- Start From 584
- Start Numbering 654
- Start Parsing 1042
- Start Process Action 999
- Start process on scheme 220
- Start Service 995
- Start Service Action 995
- started 594, 1000
- starts 168, 169, 550, 593, 594
- state 560, 561, 562, 571, 573, 714, 715, 755
- Statement 875, 876, 878
- stats 1005
- status 284
- Status Code 830, 831, 833, 834, 836, 837, 839, 840, 842, 844, 846, 848, 850, 852, 854, 856, 858, 860, 861, 863, 865, 867, 869, 871, 872, 874
- step by step 318
- sticky-pad 309
- stop all running Robots 191
- Stop Process 126, 536, 539, 614
- Stop Process Action 614
- Stop Robot 284
- Stop Service 996
- Stop Service Action 996
- stopped 561, 594
- stops 550, 593, 594, 614, 615, 617, 996
- storage 435
- storage bin 435
- Store 584, 587, 638, 640, 642, 645, 678, 781, 796, 875, 876, 878, 880, 907, 908, 1000, 1028, 1042, 1044, 1046, 1047, 1067
- Store Button Pressed 634, 636, 638, 647
- Store Clipboard Text Into 678
- Store Command Line Arguments 1000
- Store Command Line Arguments into 1000
- Store Common Items into List 1076
- Store Connection 778
- Store Connection into 776

- Store Content 657
 - Store Copied Files 651
 - Store Copied Folder into 670
 - Store Current Index 584
 - Store Downloaded File 769, 772
 - Store Excel Instance into 878
 - Store Files into 781
 - Store Length 1026
 - Store Main Window Handle 595
 - Store Match 1042
 - Store Match(es) into 1042
 - Store Moved Files 652
 - Store Moved Files into 652
 - Store Moved Folder into 672
 - Store Newly Created Folder into 667
 - Store Output into 531, 533, 535, 537
 - Store Position 1042
 - Store Position(s) of found text into 1042
 - Store Process Path into 1001
 - Store Random Text 1038
 - Store Renamed Folder into 673
 - Store Reply Code 796
 - Store Reply Code into 796
 - Store Reply Text into 796
 - Store Result 1027, 1030, 1033, 1034, 1036, 1037, 1040, 1041, 1044, 1046, 1049
 - Store Result into 1030, 1033, 1034, 1036, 1037, 1041
 - Store Result Text 1044, 1046
 - Store Retrieved DateTime 1047
 - Store Retrieved Subfolders into 665
 - Store Robot's Path 1001
 - Store Selected File 645
 - Store Selected Folder 647
 - Store Special Folder Path into 674
 - Store Subdirectories into 781
 - Store Subtext 1028
 - Store Temporary File 664
 - Store Temporary File into 664
 - Store Text 1032
 - Store Text with New Case 1032
 - Store Text with New Case into 1032
 - Store the result into 1048
 - Store Trimmed Text into 1031
 - Store User Input 638
 - Store Web Page Text 769, 772
 - Store Zip File into 675
 - stored 288
 - Stored Renamed Files 654
 - String 875, 876, 878
 - Subdirectories 781
 - subfolders 668, 669
 - Subject 797, 806, 808, 986, 988, 989, 991, 992
 - subnet 550
 - substrings 1041
 - subtext 1042, 1044, 1046
 - Subtract 1049, 1077
 - Subtract Dates 1049
 - Subtract Dates Action 1049
 - Subtract Lists 1077
 - Subtract Lists Action 1077
 - subtracts 1048
 - success 167
 - suspends 543, 589, 590, 593, 594, 595, 605, 606, 608
 - sync 792
 - synchronization 792
 - synchronize 792
 - System 529, 531, 540, 541, 542, 543, 545, 546, 547, 548, 550, 554, 556
 - System Category 529, 531, 540, 541, 542, 543, 545, 546, 547, 548, 550, 554, 556
- T -**
- tab 739, 1031
 - tables 439
 - tabs 294
 - tabular 439
 - Tag Image 863
 - Take 548, 767
 - Take ScreenShot 548, 767
 - Take ScreenShot Action 548
 - Take ScreenShot of web page 767
 - Take ScreenShot of web page Action 767
 - technical 179
 - Temporary File 664
 - terminate 536, 539
 - Terminated 152, 153, 154, 169
 - terminates 168, 169
 - Tesseract 931
 - test 306

- Text 288, 436, 439, 606, 608, 630, 659, 661, 678, 679, 710, 711, 712, 713, 751, 753, 757, 758, 1026, 1028, 1030, 1031, 1036, 1037, 1040, 1041, 1042
- Text Actions 1026, 1027, 1028, 1030, 1031, 1032, 1033, 1034, 1036, 1037, 1038, 1040, 1041, 1042, 1044, 1046
- Text Actions Category 1026, 1027, 1028, 1030, 1031, 1032, 1033, 1034, 1036, 1037, 1038, 1040, 1041, 1042, 1044, 1046
- Text Analytics 869, 871, 872, 874
- Text Case 1032
- Text Detection 834
- text elements 1041
- text file 139
- Text Length 1026
- Text to Add 654
- Text to Convert 1033, 1036
- Text to Convert: 1032
- Text to DateTime 1036
- Text to Find 1042
- Text to Measure 1026
- Text to Number 1033
- Text to Pad 1030
- Text to Parse 1042, 1044, 1046
- Text to Remove 654
- Text to Replace 654
- Text to store into Clipboard 679
- Text to Trim 1031
- Text to Write 659, 661
- text value 1026, 1028, 1040
- than 474
- this is not a Variable 501
- Ticket 957, 958, 960
- Time 1047
- time intervals 173
- Timeout 634, 636, 875, 876, 878
- times 584
- title 562, 571, 573, 595, 634, 636, 638, 640, 642, 645, 682, 692, 694, 698, 701, 704, 708, 710, 711, 712, 713, 714, 715, 717, 751, 753, 755, 757, 758, 765, 766
- To 797, 806, 808, 986, 988, 989, 991, 992
- To CSV 661
- To Variable 1048
- Tolerance 598, 621
- toolbar 315
- Tools 190
- Topic 501, 502, 504, 506, 507
- Total 1030
- Total Length 1030
- Translate 854, 856
- tree 688
- Trigger 126, 155, 159, 164, 165, 167, 168, 169, 173
- Trigger Types 155
- Trigger Variables 152, 153, 154, 311
- TriggerName 152, 153, 154
- Triggers 152, 153, 154, 164
- Triggers tab 152, 153, 154
- Triggers Window 155
- Trim 1031
- Trim Text 1031
- Trim Text Action 1031
- true 586
- truncate number 1064
- two kinds of errors 294
- two-dimensional arrays 439
- Type 288, 311, 638, 640
- ## - U -
- UI 329, 338, 343, 571, 680, 682, 683, 686, 688, 712, 713, 714, 715
- UI actions 338
- UI Automation Window 690
- UI selectors 343
- unauthorized 545
- unblock 617
- Uncommitted Changes 462
- uncompress 676
- unexpected 502, 504
- unexpected situation 502, 504
- Unzip 675, 676
- Unzip Category 675, 676
- Unzip Files 676
- Unzip Files Action 676
- update 956, 960, 962, 963, 1005, 1021
- update credential 241, 1021
- update credential action 1021
- update KPI action 1005
- updated 1044, 1046
- Upload 786, 787
- Upload File 786
- Upload File(s) 786

Upload File(s) to FTP Action 786
 Upload Folder 787
 Upload Folder(s) 787
 Upload Folder(s) to FTP Action 787
 uppercase 1032
 URL 167, 769, 772
 Usage 454, 474
 Usage Examples 454, 474
 Use 1037
 Use 1000 Separator 1034
 Use Desktop 680
 useful information 435
 User 542, 769, 772
 User Agent 769, 772
 User Defined Variables 311
 User Input 638
 User Name 776, 778, 808, 986, 988, 989, 991, 992
 User roles 193
 username 241, 1015, 1021
 username password 203
 Users 195
 Using 435
 Using Enate Actions 483
 Using Variables 435

- V -

v3 435
 Valid 796
 Valid Reply Code 796
 Valid Reply Code(s) 796
 valuable information 435
 value 638, 884, 1057
 Variable 237, 288, 319, 501, 502, 504, 552, 553, 587, 1062, 1063, 1064, 1065
 variable property 435
 Variable Types 304
 VariableName[ItemNumber] 438
 VariableName[RowNumber][ColumnNumber] 439
 Variables 152, 153, 154, 164, 165, 169, 288, 304, 311, 318, 435, 436, 438, 439, 444
 Variables Actions 1062, 1063, 1064, 1065, 1067, 1068, 1069, 1070, 1071, 1076, 1077, 1078
 Variables Actions Category 1062, 1063, 1064, 1065, 1067, 1068, 1069, 1070, 1071, 1076, 1077, 1078

Variables Manager 152, 153, 154, 288, 311
 Variables Manager Window 311, 435
 Variables pane 294, 318
 Variables Visualizer 290
 VBS 533
 VBscript 533
 Version 455, 465
 Video 188
 Video Log 188
 View Logs 178
 Visibility 682, 698
 visible 598, 621, 682, 698
 visually 312
 vs 321

- W -

Wait 589, 590, 593, 594, 595, 598, 605, 606, 608, 629
 Wait Action 589
 Wait Category 589, 590, 593, 594, 595, 598, 605, 606, 608
 wait cmd 829
 wait cmd session 829
 wait cmd session action 829
 wait command 829
 wait command line session 829
 Wait For Content 606, 608
 Wait For File 589
 Wait For File Action 589
 Wait for file to be 589
 Wait For Hotkey 590
 Wait For Hotkey Action 590
 Wait for Image 506
 Wait For Image Action 598
 Wait for key 590
 Wait For Mouse 605
 Wait For Mouse Action 605
 Wait for Process 593, 999
 Wait For Process Action 593
 Wait for Process to Complete 999
 Wait For Service 594
 Wait For Service Action 594
 Wait for service to 594
 Wait For text 606, 608
 wait for text on cmd 829

- wait for text on cmd session 829
 - Wait For Window 595
 - Wallpaper 544
 - Warning 164
 - Warnings 248
 - waste 547
 - Web 485, 608, 726, 741, 743, 744, 751, 753, 755, 757, 758, 760, 769, 772
 - web address 769, 772
 - web applications 351, 393
 - web automation 351, 353, 360, 393
 - Web Category 769, 772
 - web data extraction 360, 373, 378, 382
 - web forms 351, 393
 - web harvesting 373, 378, 382
 - web page 573, 751, 753, 755, 757, 758, 769, 772
 - Web Page Text 769, 772
 - web recorder 360
 - web scrapping 373, 378, 382
 - web server 391, 769, 772
 - web sites 351, 393
 - Webpage 608, 726, 741, 743, 744, 760
 - WebService 772
 - Website 726, 741, 743, 744, 751, 753, 755, 760
 - weekday 174
 - Weekly 174
 - What to Trim 1031
 - What's new 435
 - What's new about variables 435
 - white space 1031
 - wildcards 562, 571, 573, 595, 682, 692, 694, 698, 701, 704, 708, 710, 711, 712, 713, 714, 715
 - Win 632
 - Window 311, 562, 571, 595, 606, 618, 619, 621, 638, 680, 682, 683, 686, 688, 692, 694, 698, 701, 704, 708, 710, 711, 712, 713, 714, 715
 - window application 682
 - Window Class 562, 571, 573, 595, 682, 692, 694, 698, 701, 704, 708, 710, 711, 712, 713, 714, 715
 - Window Handle 595, 682, 692, 694, 698, 701, 704, 708, 710, 711, 712, 713, 714, 715
 - Window Position 701
 - Window Size 704
 - Window State 694
 - Window Style 529
 - Window Title 562, 571, 573, 595, 682, 692, 694, 698, 701, 704, 708, 710, 711, 712, 713, 714, 715
 - Windows 547, 674, 682, 692, 694, 698, 701, 704, 708, 710, 711, 712, 713, 995, 996, 997, 998
 - Windows Applications Event Log 139
 - Windows Category 682, 692, 694, 698, 701, 704, 708, 710, 711, 712, 713
 - Windows Event Log 164
 - Windows Recycle Bin 547
 - Windows Service 995, 996, 997, 998
 - Windows' special folder 674
 - wish 633
 - Work 952, 953, 963
 - Working 174
 - Working Directory 782
 - Working Folder 529, 531
 - Working item 151, 220
 - Working items count 224
 - worksheet 907, 908
 - Worksheet Index 887, 892, 893
 - Worksheet Name 887, 892, 893
 - Workspace 286, 288, 294, 303
 - Workstation 545
 - Write 659, 661, 884
 - write cmd 826
 - write cmd session 826
 - write cmd session action 826
 - write command 826
 - write command line session 826
 - Write Text 659, 661
 - Write Text to CSV 661
 - Write Text to CSV Action 661
 - Write Text to File 659
 - Write Text to File Action 659
 - Write to Excel Action 884
 - write xml 1051
 - write xml to file 1051
 - write xml to file action 1051
- X -**
- xml 1050, 1051, 1052, 1053, 1055, 1056, 1057, 1059, 1060, 1061
 - xpath 1050, 1051, 1052, 1053, 1055, 1056, 1057, 1059, 1060, 1061

- Y -

Year 439
yellow 309

- Z -

Zip 675, 676
Zip archive 675, 676
Zip Category 675, 676
Zip Files 675
Zip Files Action 675