Hands-On Lab

# Introduction to Test Planning with Microsoft Test Manager 2015

Lab version: 14.0.25123.0

Last updated: 5/27/2016



1. **TABLE OF CONTENT**

[Introduction to Test Planning with Microsoft Test Manager 2015 1](#_Toc452125539)

[Overview 3](#_Toc452125540)

[Prerequisites 3](#_Toc452125541)

[Exercises 3](#_Toc452125542)

[Exercise 1: Creating and Configuring a Test Plan 4](#_Toc452125543)

[Exercise 2: Adding Test Suites and Test Cases to a Test Plan 11](#_Toc452125544)

[Exercise 3: Creating and Adding Requirements to a Test Plan 16](#_Toc452125545)

[Exercise 4: Managing Tests Using the Test Hub 21](#_Toc452125546)

## Overview

* 1. Microsoft Test Manager 2015 can be used to organize your test plans, author and manage your test cases, and run manual tests. In this lab, you will be introduced to test planning in Microsoft Test Manager.
  2. This hands-on-lab is one out of a number of labs that deal with Microsoft Test Manager 2015. The recommended order to complete the labs in is as follows:
  + Introduction to Test Planning with Microsoft Test Manager 2015
  + Introduction to Test Case Management with Microsoft Test Manager 2015
  + Authoring and Running Manual Tests using Microsoft Test Manager 2015
  + Introduction to Platform Testing with Microsoft Test Manager 2013

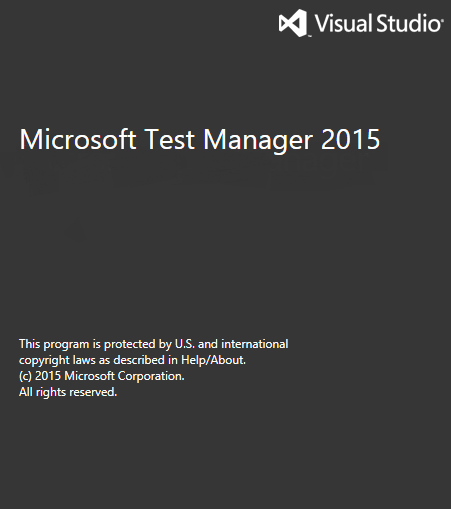
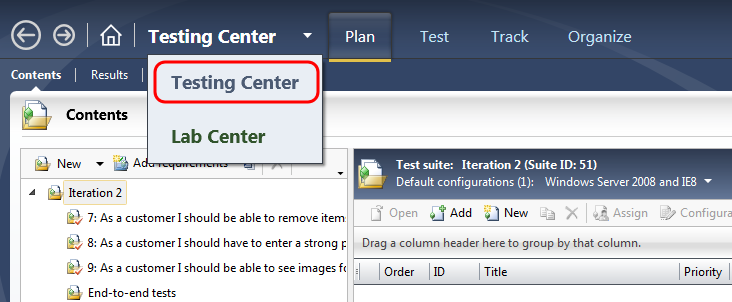
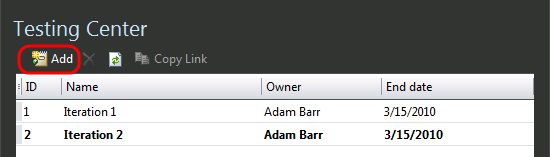
### Prerequisites

In order to complete this lab you will need the Visual Studio 2015 virtual machine provided by Microsoft. For more information on acquiring and using this virtual machine, please see [this blog post](http://aka.ms/almvm).

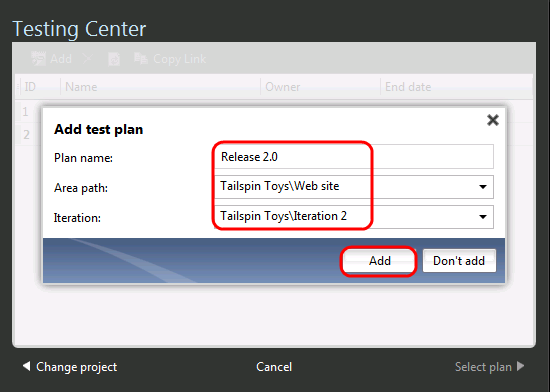
### Exercises

* 1. This hands-on lab includes the following exercises:
  2. Creating and Configuring a Test Plan
  3. Adding Test Suites and Test Cases to a Test Plan
  4. Creating and Adding Requirements to a Test Plan
  5. Estimated time to complete this lab: **60 minutes**.

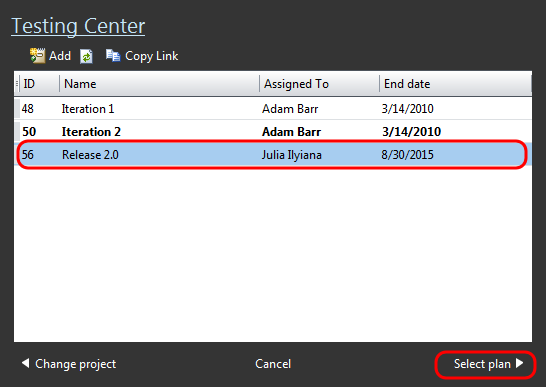
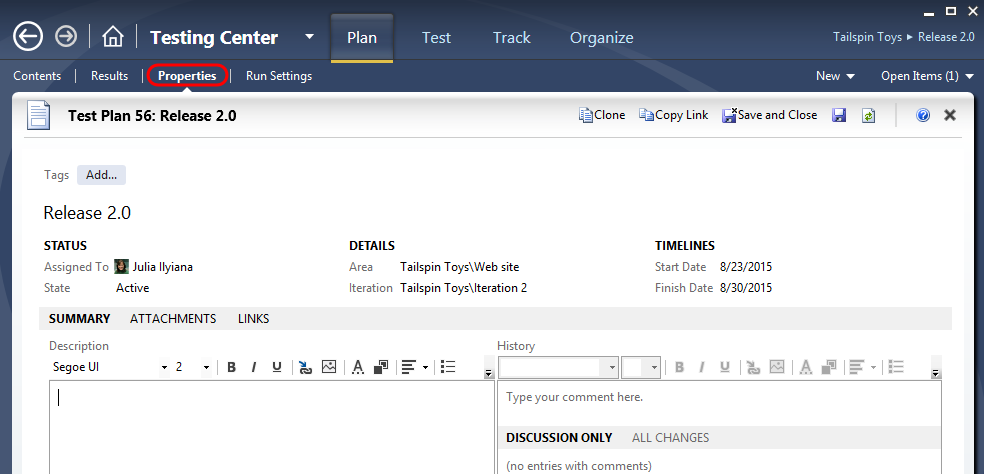
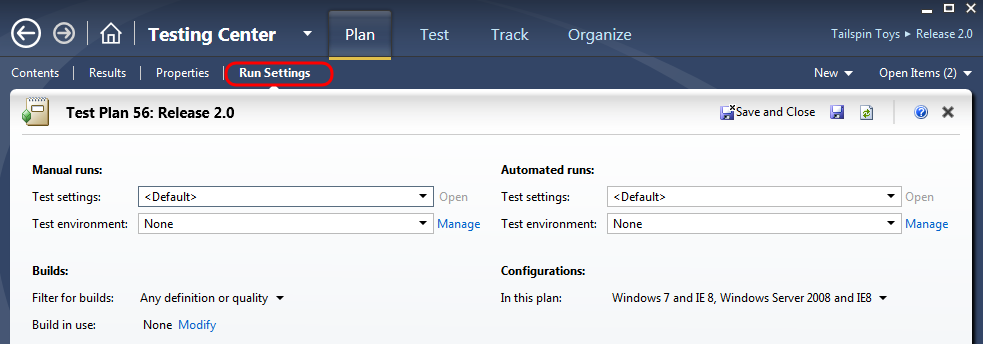
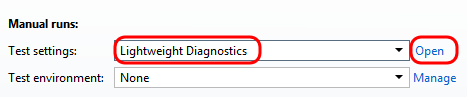
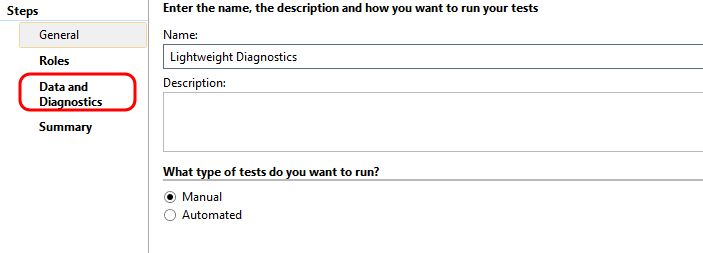
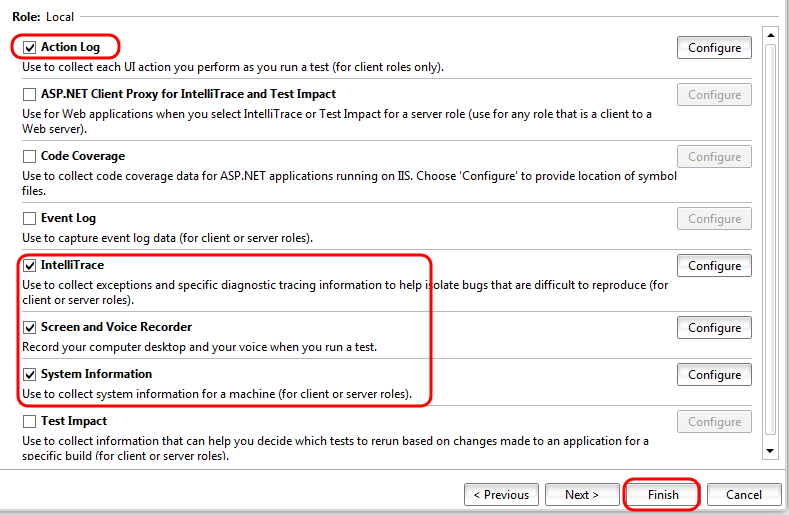
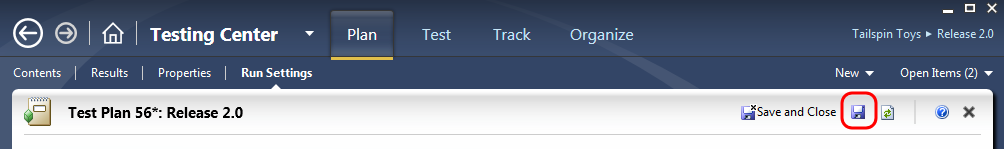
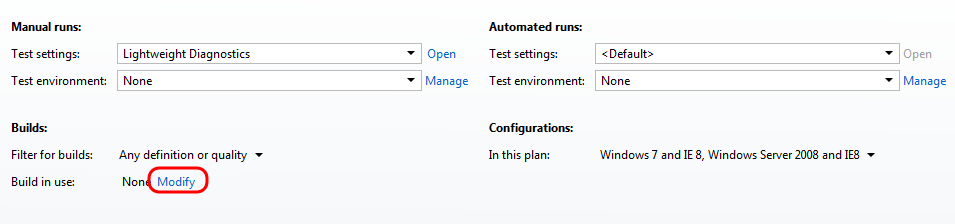
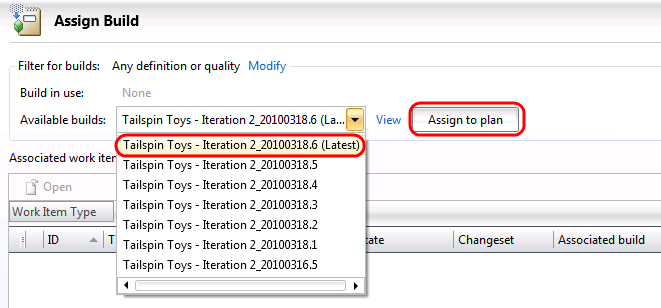
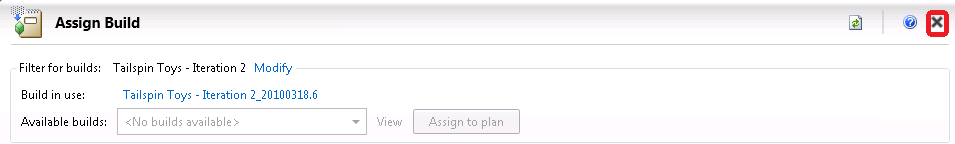
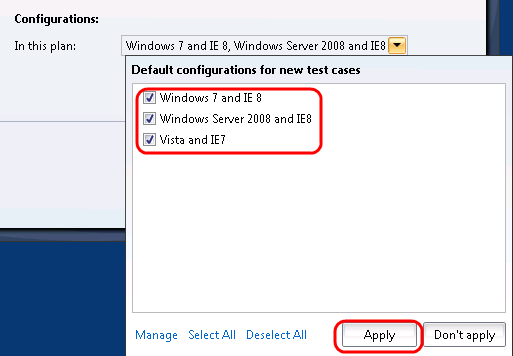
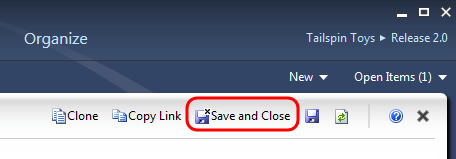
## Exercise 1: Creating and Configuring a Test Plan

* 1. In this exercise, you will learn how to create and configure a new test plan in Microsoft Test Manager. This test plan can be used, for instance, to test the new release of your software.
  2. Log in as **Julia Ilyiana** (VSALM\Julia). All user passwords are **P2ssw0rd.**
  3. Launch **Microsoft Test Manager** from the taskbar. Microsoft Test Manager allows testers to work with test plans, author and organize manual test cases, execute test cases, file bugs, and post results back to Team Foundation Server.
     1. 
     2. Figure
     3. Microsoft Test Manager splash screen
  4. Select the activity center drop down from the top menu that currently shows Testing Center. The **Testing Center** is used to work with manual test cases and other general test case management. **Lab Center** is used to work with physical or virtual testing labs. For this lab, we will focus on the Testing Center.
     1. 
     2. Figure
     3. Testing Center showing the Plan activity
  5. Click the **Home** button next to the activity center dropdown to open the **Testing Center** window.
     1. 
     2. Figure
     3. Open the Testing Center window using the Home button
  6. In the **Testing Center** window, click **Add** to add a new test plan.
     1. 
     2. Figure

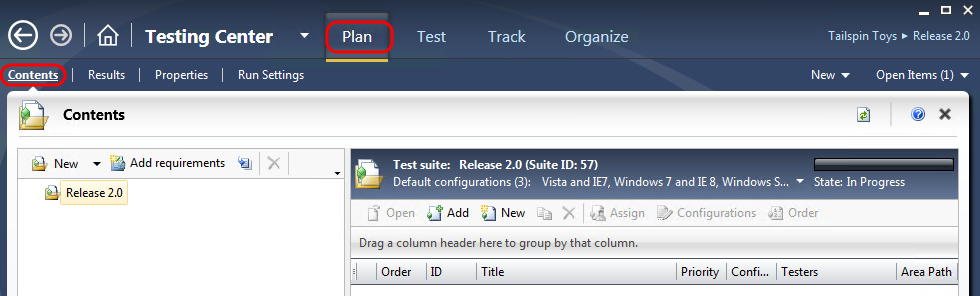
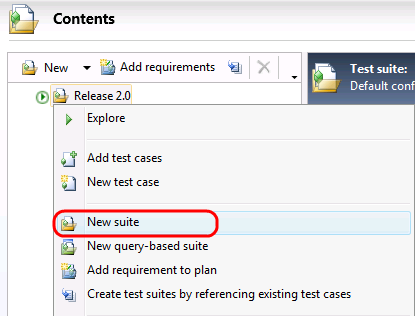
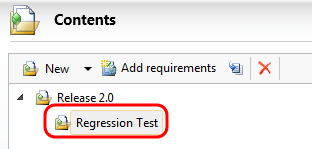
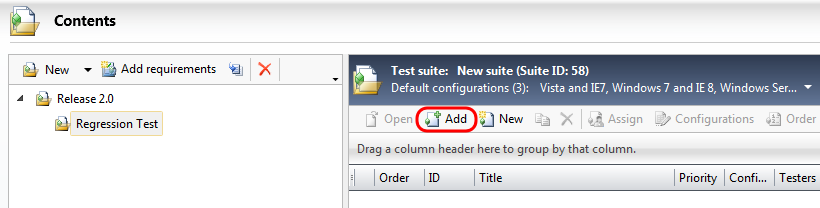
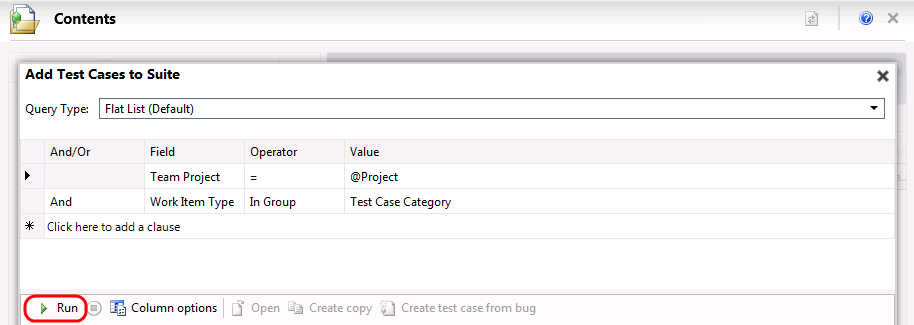
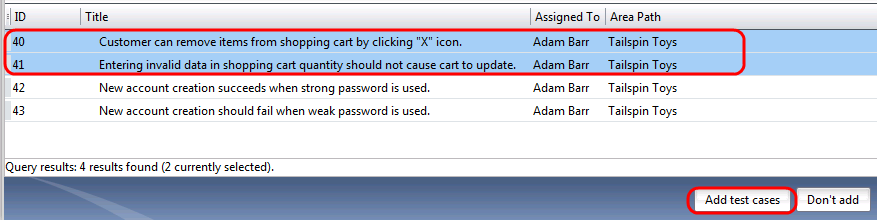
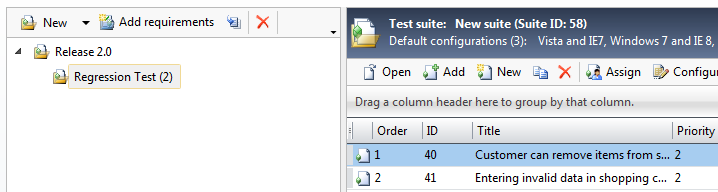
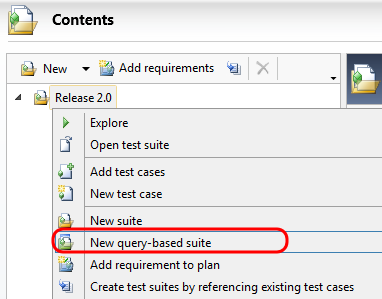
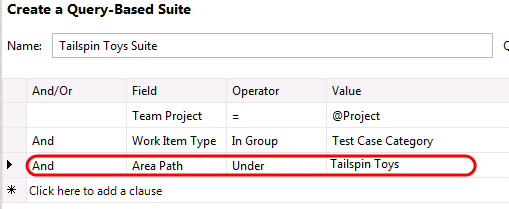
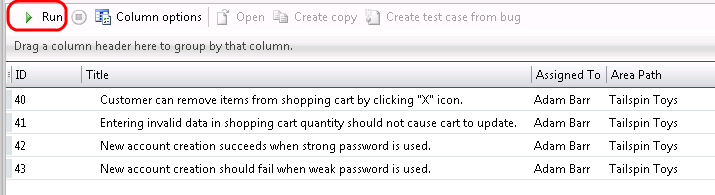
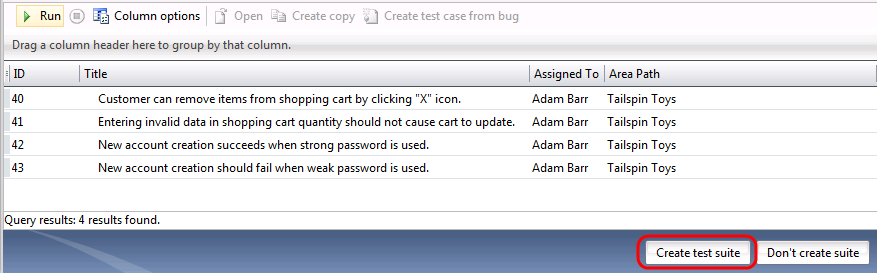
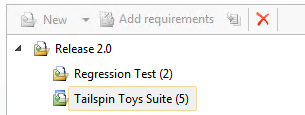
Adding a new test plan

* 1. For **Plan name**, enter “**Release 2.0**”.
  2. For **Area path**, choose **Tailspin Toys\Web site.**
  3. For **Iteration**, choose **Tailspin Toys\Iteration 2**.
  4. Click **Add** to continue.
     1. 
     2. Figure

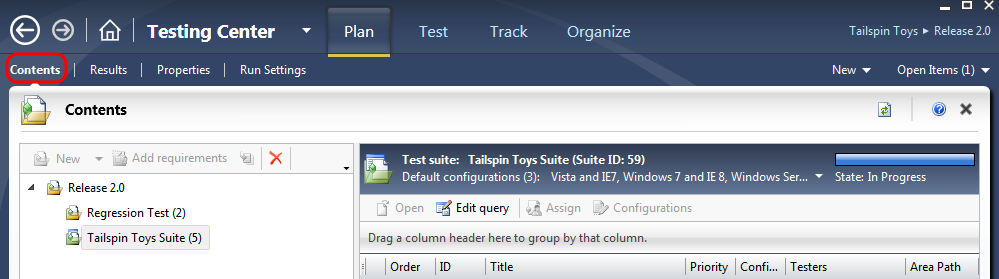
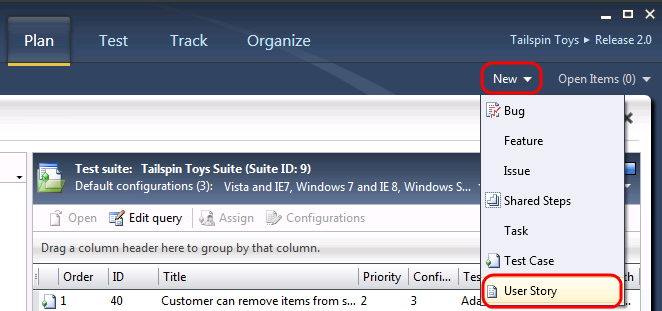
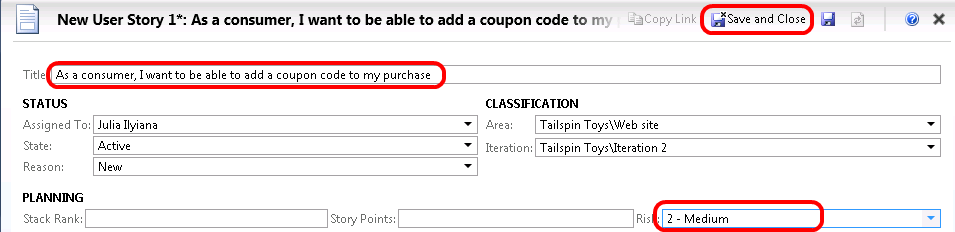
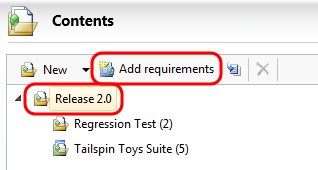
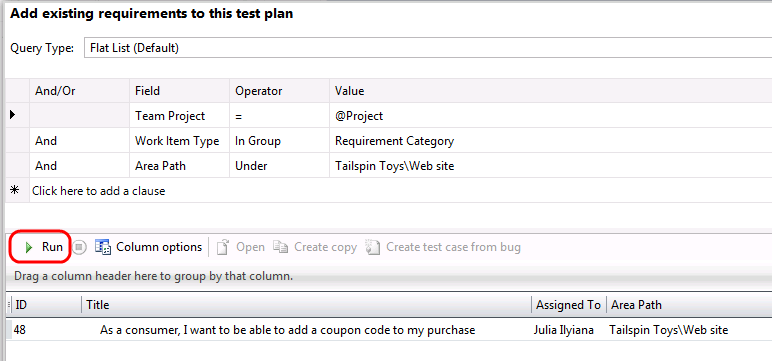
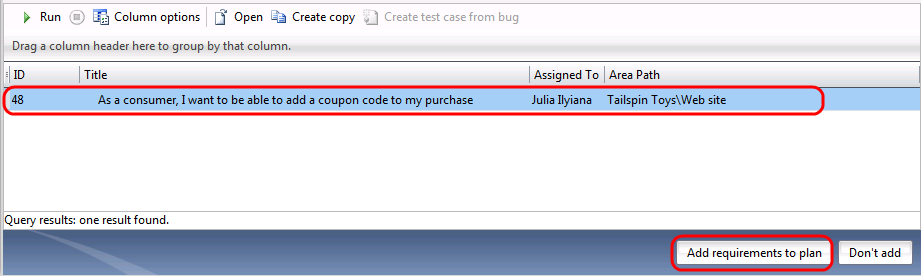
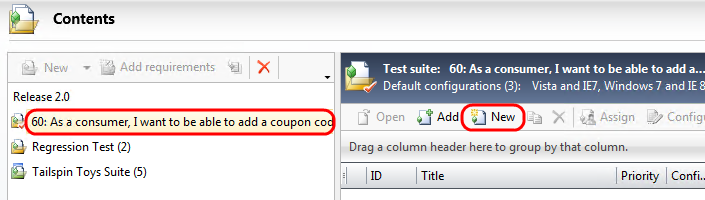
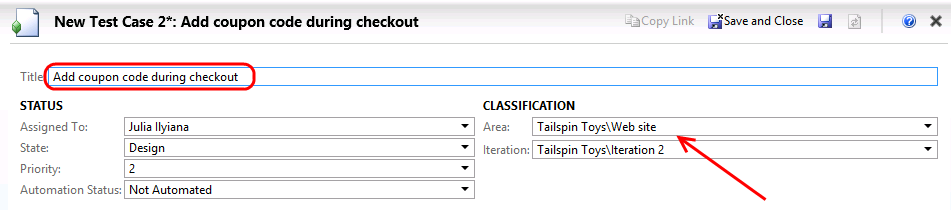
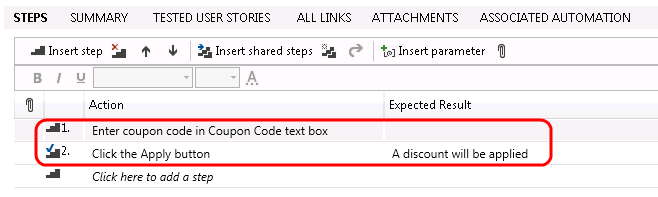
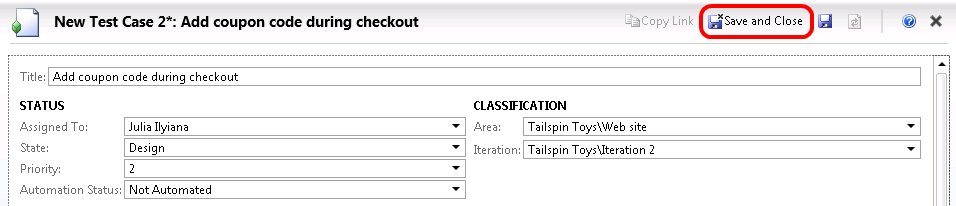
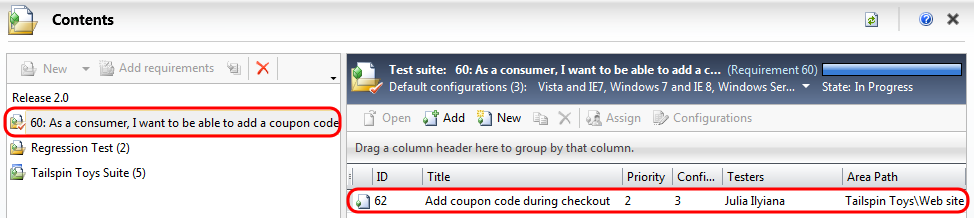
Adding a new test plan

* 1. In the Testing Center window, select the newly created Release 2.0 test plan and then click **Select Plan** to open it.
     1. 
     2. Figure
     3. Selecting a test plan
  2. Select the **Properties** tab to view the plan’s properties. The test plan properties window allows you to edit the basic properties like name, area path, iteration, state, and so on.
     1. 
     2. Figure
     3. Test plan properties window
  3. Select the **Run Settings** tab to view the plan’s run settings. The **Run Settings** section allows you to specify how manual and automated test runs should be setup, what test environment to use, and even the specific build to test against.
     1. 
     2. Figure
     3. Test plan run settings window
  4. In the **Manual Runs** section, choose **Lightweight Diagnostics** for the **Test settings** option and click **Open** to open the Test Settings window.
     1. 
     2. Figure
     3. Adjusting manual test run settings
  5. In the test settings window, select the **Data and Diagnostics** step to configure the Data Diagnostic Adapters to use during testing.
     1. 
     2. Figure
     3. Test settings window
  6. Select the **Action Log,** **IntelliTrace, System Information,** and **Screen and Voice Recorder** diagnostic data adapters (some may already be selected), then click **Finish** to close the test settings window and return to the test plan properties window.
     1. 
     2. Figure
     3. Selecting data diagnostic adapters
  7. Click **Save** to save the test plan.
     1. 
     2. Figure
     3. Location of Save button
     4. **Note:** The test plan needs to be saved before selecting a build.
  8. In the **Builds** section, click **Modify** next to the **Build in use** label. The selected build will be used when recording test results or filing bugs.
     1. 
     2. Figure
     3. Selecting the build to use during test runs
  9. In the **Assign Build** window, select the first available build labeled **Tailspin Toys – Iteration 2\_20100318.6 (Latest)** and click **Assign to Plan**.
     1. 
     2. Figure
     3. Assigning a build to the test plan
  10. Click the **Close** button () to exit the **Assign Build** window. Be careful not to close the main Test Manager window. This will bring you back the Test Plan properties window.
      1. 
      2. Figure
      3. Location of Close button
  11. Test configurations describe which platforms to test against during test runs. In the **Configurations** section, select the drop down that currently shows “Windows 7 and IE 8, Windows Server 2008 and IE8”, and select all available configurations.
  12. Click **Apply** to continue.
      1. 
      2. Figure
      3. Location of Apply button
  13. Click **Save and Close** to save and close the new test plan.
      1. 
      2. Figure
      3. Location of Save and Close button

## Exercise 2: Adding Test Suites and Test Cases to a Test Plan

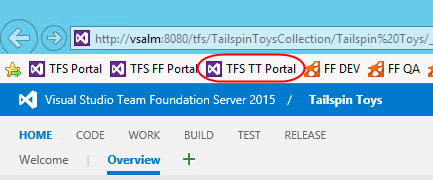
* 1. In this exercise, you will learn how to add test suites and existing test cases to a test plan.
  2. In **Microsoft Test Manager**, select the **Plan** tab and then select its **Contents** child tab to view the new **Release 2.0** test plan if it is not already open.
     1. 
     2. Figure
     3. Test plan contents window for Release 2.0
  3. Create a manual test suite by right-clicking the **Release 2.0** node and selecting the **New suite** option from the context menu.
     1. 
     2. Figure
     3. Creating a new test suite
  4. Enter **Regression Test** as the suite name and press the **Enter** key to accept.
     1. 
     2. Figure
     3. Providing a descriptive name for the new test suite
  5. Select the **Regression Test** suite in the left pane and click **Add** in the right pane to add existing test cases.
     1. 
     2. Figure
     3. Location of the Add button
  6. In the **Add Test Cases to Suite** window, click **Run** to see all of the test cases for the current team project.
     1. 
     2. Figure
     3. Run query to find test cases to add to new test suite
  7. Select test cases with **ID** = **40** and **41** and then click **Add test cases** to add them to the new **Regression Test** suite.
     1. 
     2. Figure
     3. Adding test cases
     4. 
     5. Figure
     6. Test cases added to new test suite
  8. Now let’s add another type of test suite to the test plan. Right-click the **Release 2.0** node and select **New query-based suite** from the context menu. Query-based test suites allow work item queries to be constructed in order to gather test cases, as you will see shortly.
     1. 
     2. Figure
     3. Adding a query-based test suite
  9. In the **Create a Query-Based Suite** window, enter “**Tailspin Toys Suite”** as the **Name**.
     1. 
     2. Figure
     3. Entering a name for the new test suite
  10. Select the grid row labeled **Click here to add a clause** (the row showing \* at the beginning) to add a new query clause that restricts work items to those under the Tailspin Toys area path. Use the values shown in the following screenshot.
      1. 
      2. Figure
      3. Creating query to select test cases
  11. Click **Run** to see the results of this query.
      1. 
      2. Figure
      3. Location of Run button
  12. Click **Create test suite** to create this query-based test suite and return to the test plan contents window.
      1. 
      2. Figure
      3. Location of Create test suite button
  13. You should now see that the **Release 2.0** test plan now has two test suites in it. The icon for the query-based test suite named **Tailspin Toys Suite** differs from the manual test suite named **Regression Test**. Query-based test suites dynamically determine test cases that are listed based on the defined query.
      1. 
      2. Figure
      3. Test plan showing the two new test suites
      4. **Note:** In this lab environment, you will see a limited number of test cases. This may result in the re-use of test cases across multiple test suites.

## Exercise 3: Creating and Adding Requirements to a Test Plan

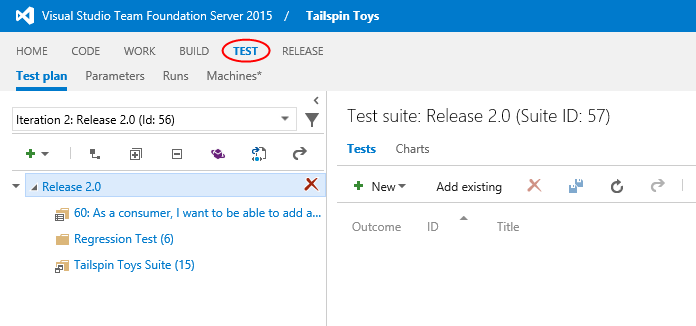
* 1. In this exercise, you will learn how to create a new requirement and add that requirement to your test plan. You will also learn how to create a new test case for the new requirement.
  2. In **Microsoft Test Manager**, select the **Plan** tab and then select its **Contents** child tab to view the new **Release 2.0** test plan if it is not already open.
     1. 
     2. Figure
     3. Test plan contents window for Release 2.0
  3. Create a new user story work item by selecting the **New** drop down box and selecting **User Story**.
     1. 
     2. Figure
     3. Creating new user story work item
  4. For the **Title** of the user story, enter “**As a consumer, I want to be able to add a coupon code to my purchase**”.
  5. For the **Risk** of the user story, select the “**2-Medium**” option from the dropdown box.
  6. Click **Save and Close** to continue.
     1. 
     2. Figure
     3. Define and save a new user story
  7. In the test plan contents window, select the **Release 2.0** node and then click **Add Requirements** to add a new requirement to the plan.
     1. 
     2. Figure
     3. Adding a new requirement to the test plan
  8. In the window titled “**Add existing requirements to this test plan**”, click **Run** to see the results of this query. You should see the new user story that we just created.
     1. 
     2. Figure
     3. Adjusting the query to help find the recently created user story
  9. Select the user story and then click **Add requirements to plan**. This will close the window and return to the test plan contents window.
     1. 
     2. Figure
     3. Selecting and adding the new user story as a requirement
  10. In the test plan contents window, you should now see a new requirements-based test suite listed. In the right pane, there are options to either link existing test cases or create new test cases.
  11. Click **New** in the right pane to create a new test case.
      1. 
      2. Figure
      3. Creating a new test case for the user story requirement
  12. For the **Title** of the new test case, enter “**Add coupon code during checkout**”. Note that the **Area** and **Iteration** values in the **Classification** section have already been correctly pre-populated.
      1. 
      2. Figure
      3. Creating new test case and assigning properties
  13. Enter the **Steps** for the new test case as shown in the screenshot below.
      1. 
      2. Figure
      3. Creating steps for new test case
  14. Click **Save and Close** to save the new test case and return to the test plan contents window.
      1. 
      2. Figure
      3. Location of Save and Close button
  15. In the test plan contents window, you should now see the new test case that was added to the requirement-based test suite.
      1. 
      2. Figure
      3. Requirement-based test suite showing the new test case

## Exercise 4: Managing Tests Using the Test Hub

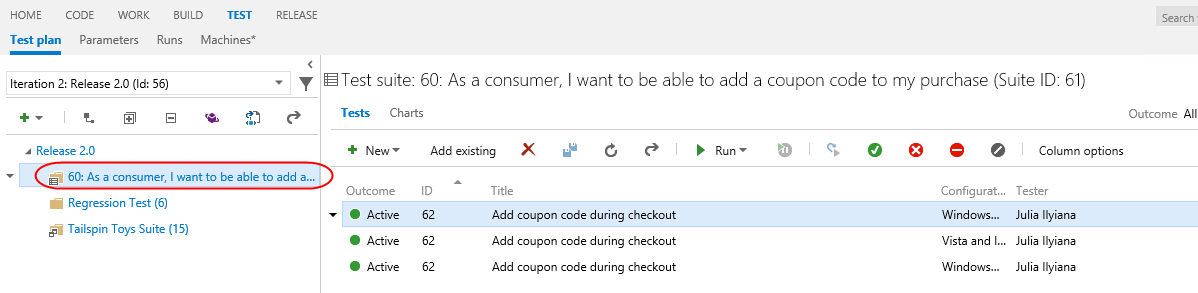
* 1. In this exercise, you will learn how to create a new requirement and add that requirement to your test plan. You will also learn how to create a new test case for the new requirement.
  2. Open **Internet Explorer** and click the **TFS TT Portal** bookmark at the top of the window.



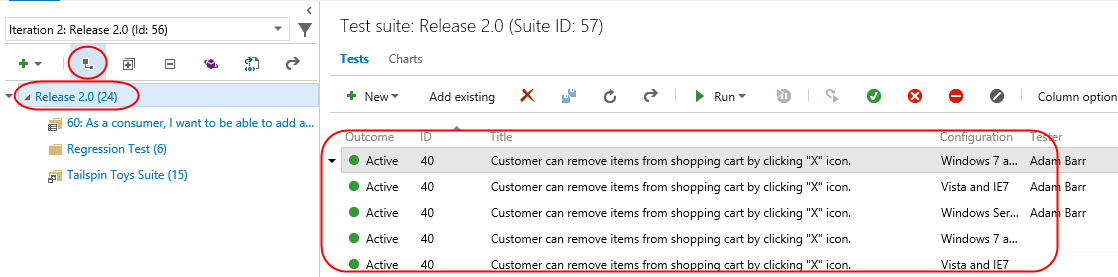
* 1. Select the **Test** tab to open the Test hub.



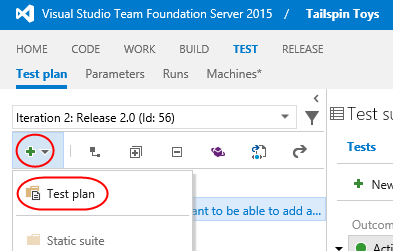
* 1. Select test suite **60** from the left panel to view all of the child test cases.



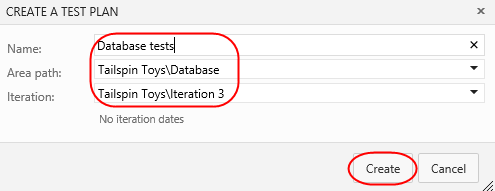
* 1. You can also view all of the test cases in the child test suites within a plan. Select the **Release 2.0** plan and click the **Show tests from child suites** button to view all of the test cases under this node.



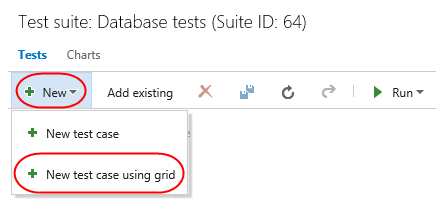
* 1. You can also create new test plans from the portal. Click the **Add** drop down and select **Test plan**.



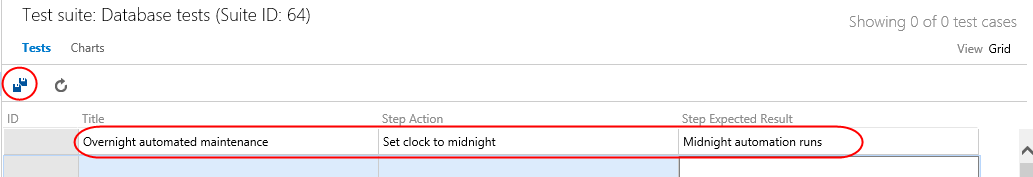
* 1. Enter the name **“Database tests”** and set the **Area Path** to **“Tailspin Toys\Database”**. Set **Iteration** to **“Tailspin Toys\Iteration 3”** and click **Create**.



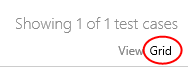
* 1. Now that the new test case has been created, you can begin to add test cases. If you have a lot of test cases to add, the grid approach might be the easiest process. Click **New | New test case using grid**.



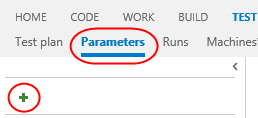
* 1. Enter a single test case with one step and result. This test case will be designed to confirm that the expected database automation runs at midnight, as scheduled. Set the **Title** to **“Overnight automated maintenance”**, **Step Action** to **“Set click to midnight**”, and **“Step Expected Result”** to **“Midnight automation runs”**. You can add multiple test cases if you like. If you have multiple steps for a given case, just add those to the following rows, leaving the **Title** field blank, and they’ll be rolled up to the current case. Click **Save** to save the new test case.



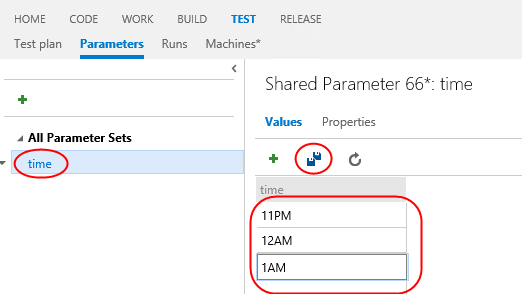
* 1. Click **Grid** to toggle back to the **List** view.



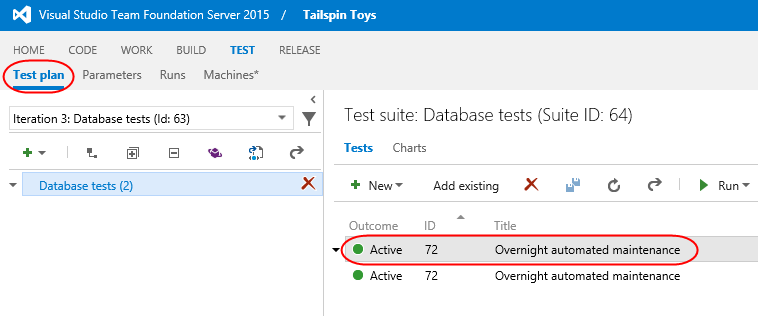
* 1. While the database automation case covers an important feature, it’s pretty incomplete. For example, it’s designed to confirm something happens at midnight, but doesn’t include other times. Rather than adding in additional tests to cover those scenarios, we can create a shared parameter that includes the time to test. Then we’ll go back and update the test to accept this parameter for each iteration. Select the **Parameters** tab and click the **New shared parameter** button.



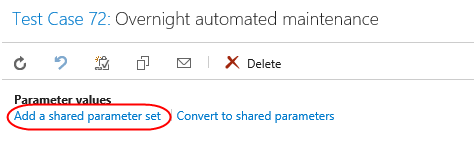
* 1. Set the new parameter’s name to **time** and add three values: **11PM**, **12AM**, and **1AM**. Click the **Save** button to confirm.



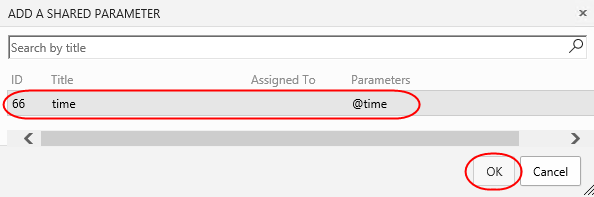
* 1. Return to the test plan view by selecting the **Test plan** tab. Double-click the first test case to open it. Note that there are two test cases shown for the one test created earlier because each represents the same test on a different configuration.



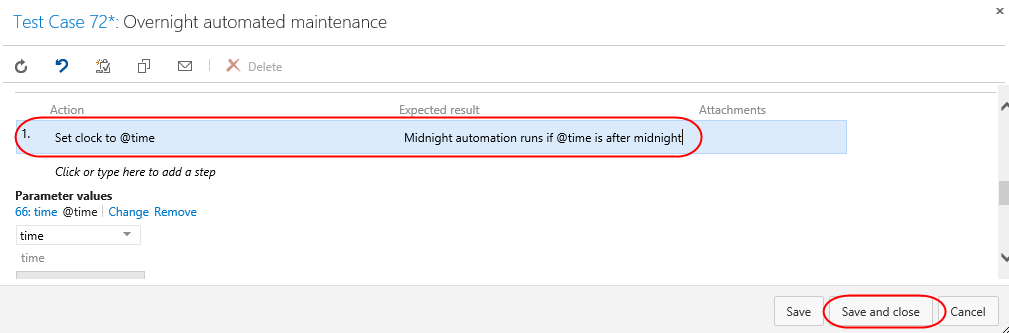
* 1. Scroll down in the test case editor and click **Add a shared parameter set**.



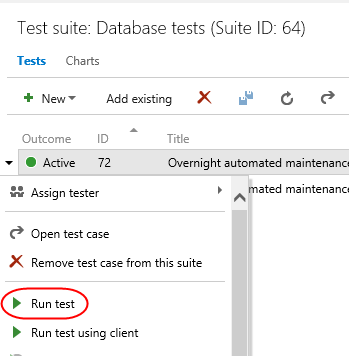
* 1. Locate the shared parameter created earlier and click **OK** to add it.



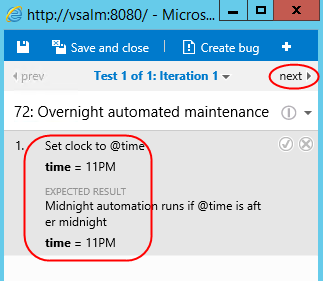
* 1. Now that the parameter is known to the test, update the **Action** and **Expected** result to include **“@time”** such that the tests make sense when the time is replaced with one of the values added earlier. Click **Save and close**.



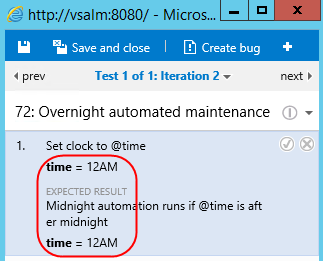
* 1. Right-click the first test case and select **Run test**.



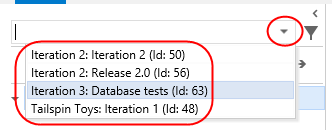
* 1. On the first iteration, the test window uses a time of **11PM**. Click **next** to move to the next iteration.



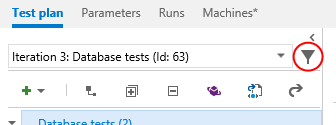
* 1. Note that the second time of **12AM** is provided for this iteration. Close the window to cancel the rest of the process.



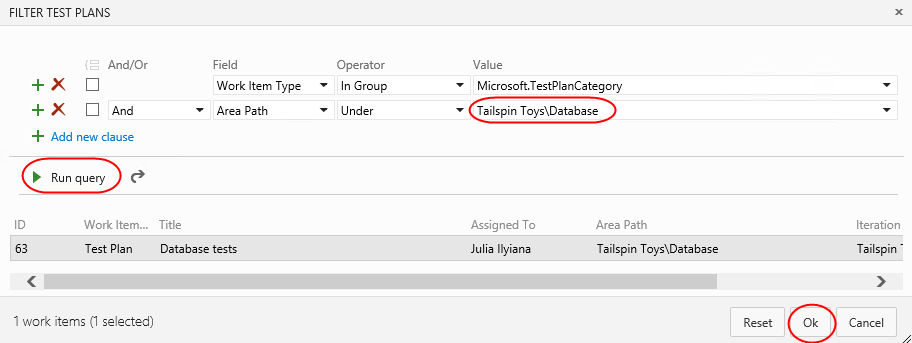
* 1. To make test management easier, you can filter which test plans are made available in the navigation. Expand the dropdown to see which plans are currently included.



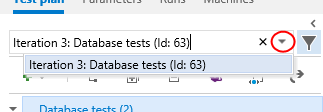
* 1. To edit how these plans are filtered, click the **Filter test plans** button.



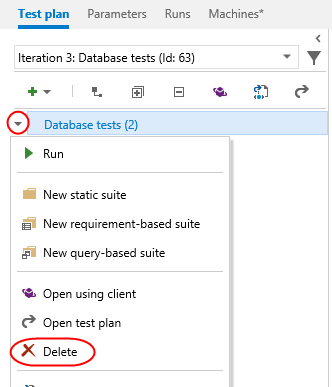
* 1. Update the second clause’s **Value** to **Tailspin Toys\Database** and click **Run query**. This new query should only return the single plan fitting the parameters, which is the one created in this exercise. Click **Ok**.



* 1. Expand the test plan dropdown and note that only the single expected plan is shown now.



* 1. In addition to creating and managing test plans, you can also easily delete test plans. Click the dropdown button next to the only test plan and select **Delete**. Confirm when asked.



To give feedback please write to [VSKitFdbk@Microsoft.com](mailto:VSKitFdbk@Microsoft.com)

Copyright © 2016 by Microsoft Corporation. All rights reserved.