

LESSON 1.1

10754 Microsoft .NET Fundamentals

Understand Basic Application Settings

Lesson Overview

What are application settings?

In this lesson, you will :

- Learn about application settings
- Create an application setting using the Designer

Guiding Questions

1. Why do developers use application settings?
2. How can developers develop basic application settings in Visual Studio®?

Anticipatory Set

Why would it be a good idea to store a database connection string or network storage path as an application setting rather putting it directly in the application's source code?

Application Settings

- Make it easy to create, store, and maintain custom application and user preferences on the client computer.
- Useful for important data that you don't want to include in the application's code. Examples include:
 - Database connection strings
 - User preferences, such as a color scheme, font sizes, etc.
- Individual settings can be “scoped” as **Application** settings or **User** settings.

.config Files

- Generally, Application Settings are stored in .config files:
 - App.config for Windows® Forms, WPF, and other desktop applications
 - Web.config for ASP.NET web applications
- Settings that are specific to individual users (“user-scoped”) are stored in a file named user.config, where ‘user’ is the user name of the person currently running the application.
- Within the configuration file, settings are stored as XML. These files are not encrypted, so never store security-related data in clear text in an application setting!

Create Application Settings in the Project Designer

1. In **Solution Explorer**, right-click on the project name and select **Properties**; then select the **Settings** pane.
2. Enter “Version” in the **Name** column. The name cannot contain spaces.
3. Select “double” in the **Type** drop-down list.
4. Select “Application” from the **Scope** drop-down list.
5. Enter a default value for the setting in the Value column. The value must be appropriate for the selected data type.

Create settings in the Properties Window

You can also create settings through the (**Application Settings**) property for a form or control.

1. From the (PropertyBinding) option, choose the property for which you wish to bind the new setting and select (New...) from the drop-down list.

For this example, choose “**BackColor**”

2. Select the Name property and enter “**UserColor**” for the name of the setting. The name cannot contain spaces.
3. Set **DefaultValue** property to “**Maroon**” (from the Web tab), and “**User**” for the scope of the setting.

Remove Application Settings

1. Select a project in **Solution Explorer**; on the **Project** menu, click **Properties**. Or, right-click the project name and select **Properties**.
2. Select the **Settings** pane.
3. Click on the row in the **Settings** grid for the setting you wish to remove.
4. Press the **Delete** key, or right-click and select **Remove Setting**.

Assignment

1. In Visual Studio, create the following Application Settings. Give each an appropriate value of your choice.
 - o **UserName**, a user-scoped setting for storing the name of the user.
 - o **BackgroundColor**, a user-scoped setting for saving the user's choice of background color in the application
 - o **BackupLocation**, an application-scoped setting for storing the location of a folder for backing up the application's data.
2. In **Solution Explorer**, open the app.config file and examine the XML code that the Designer has created.
 - o In this XML, change the value of **UserName** to a different name.
 - o Save and close app.config then check the Settings pane to see if your change was incorporated.

LESSON 1.1

10754 Microsoft .NET Fundamentals

Lesson Review

List 3 reasons why developers use application settings?