

LESSON 2.4

10754 Microsoft .NET Fundamentals

Understand and Create Class Libraries

Lesson Overview

Understand and create class libraries.

In this lesson, you will:

- Learn about the purpose of class libraries.
- Create a class library project in Microsoft® Visual Studio®.

Guiding Questions

1. Why are class libraries useful?
2. How can developers create class libraries in Visual Studio?
3. How are class libraries used?

Anticipatory Set

Imagine that you created a game with several well-designed classes representing enemy characters that the player is trying to defeat. You realize that these classes could be helpful in another game that you are just starting.

How can you use the classes in your second game without having to make changes in both projects if you choose to adjust how the enemies behave?

Reusing Classes

- Often, a class developed for one application may be useful in another application.
 - For example, if a developer creates a class with methods for calculating shipping costs for an internal inventory system, that class may be helpful when developing a retail website where the customer may want to know shipping costs.
- If you rewrite the code—or copy it to the second project—you have *maintainability* concerns.
 - If you fix a bug or make any improvement in one of the projects, you have to make the same change in the other.
- To avoid this problem, developers often use *class libraries*.

Class Library

- A class library is a collection of classes (and related code) that is packaged for reuse.
- Unlike an application, typically packaged as an executable (.exe) file, a class library project creates a *dynamic-link library* (.dll) file.
- There are several key advantages of class libraries:
 - DLLs are easy to distribute and easy to replace when making software revisions.
 - A DLL can be used by developers in many different programming languages regardless of the language used for creating the class library.
 - During the development process, you can add a reference to a DLL (rather than the code itself) so that you always use the latest build from the latest updates to the DLL.

Class Library Templates

- Visual Studio includes project templates for creating class libraries with the .NET language of your choice.
 1. Create a new project in Visual Studio and select the language of your choice.
 2. In the Windows category, select Class Library.
 3. In the Name text box, enter **SimpleLibrary** and select a location.
 4. Click OK.
- Notice that Visual Studio has created a `Class1` within the `SimpleLibrary` namespace.

Implementing a Simple Class Library

1. Rewrite Class1 as shown:

```
namespace SimpleLibrary
{
    public static class Greeting
    {
        public static void Greet()
        {
            System.Console.WriteLine("Hello World");
        }
    }
}
```

2. Build the DLL file by selecting Build Solution from the Build menu.
3. Browse to the location Bin\debug folder in your application folder to see the SimpleLibrary.dll file.

Using a DLL or Class Library

To use a DLL file—whether it is a class library that you’ve created or one that comes from some other source—you need to add a reference in your application project. To do this, perform the following steps:

1. In Visual Studio, create a new Windows Forms application.
2. From the Project menu, select Add Reference.
3. In the Add Reference dialog box, choose the Browse tab and locate the desired DLL file.
4. In this case, navigate to the application folder for SimpleLibrary, then navigate to the Bin\debug folder. The file is SimpleLibrary.DLL.

Using a DLL or Class Library (continued)

- To see the class in use, add a `Button` to the form and double-click the `Button` to create an event handler. Add this code:

```
SimpleLibrary.Greeting.Greet();
```

- Build and run the application. When you click the button, you should see

```
Hello World
```

written to the output window.

LESSON 2.4

10754 Microsoft .NET Fundamentals

Lesson Review

Complete the Class Library project in the Student Activity.