Introduction to ASP.NET Web Development
Instructor: Frank Stepanski

Overview

From this class, you will learn how to develop web applications using the Microsoft web technology ASP.NET (version 4.0) using the free web development tool Visual Studio 2010 Express (if you do not already own Visual Studio 2010 Professional).

Visual Studio 2010 Express will allow you to create web pages (.aspx) and web sites that use the .NET framework and the specific objects and controls of ASP.NET.

Free download: http://www.microsoft.com/express/Web/

The auto-install executable downloads everything you need to get started, but if you need additional components, like the .NET 4.0 Framework, you can get it here:

http://msdn.microsoft.com/en-us/netframework/aa569263

Note: If you already have the 2008 version of the free or commercial tool, that is fine too.

ASP.NET File Extensions

.aspx

When you go to a web browser and the web page has this extension, you know it is an ASP.NET page.

An additional extension would be **.aspx.cs**. This is the file extension where all your C# code will be stored ("code-behind"). You never see this extension on the web because this file gets compiled on the web server that contains the .NET Framework.

Note: The **.cs** stands for C# (# is "sharp").

If you use a different programming language for ASP.NET pages like VB.NET, it would have an **.aspx.vb** extension instead.

Six Important Facts about ASP.NET

1. ASP.NET is integrated within the .NET Framework:
The .NET Framework is divided into an almost painstaking collection of functional parts, with a staggering total of more than 8000 types (functions, classes, interfaces, etc.). The .NET framework allows for the development of Windows and Web applications. This means ASP.NET is only a part of the entire framework of .NET.

2. ASP.NET is Compiled, Not Interpreted:
ASP.NET applications actually go through two stages of compilation. In the first stage, the C# code you write (or whatever .NET language it's written in) is compiled into an intermediate language called Microsoft Intermediate Language (MSIL). The second level compilation happens just before the page is actually executed. At this point, the IL code is compiled into low-level native machine code. This stage is known as just-in-time (JIT) compilation and it takes place in the same way for all .NET applications (including Windows applications).

3. ASP.NET is Multi-language:
Though you will probably choose one language over another, you can use any .NET supported programming language to create your ASP.NET web pages. You can even use a different programming language for each new ASP.NET web page, although you probably don't want to. The default supported and most popular languages for ASP.NET are C# and VB.NET.

4. ASP.NET is Object-Oriented:
Not only does your code have full access to all objects in the .NET Framework, but you can also exploit all the conventions of OOP (object-oriented programming) environment, such as encapsulation and inheritance. For example, you can create reusable classes, standardize code with interfaces and bundle useful functionality in a distributable, compiled component.

5. ASP.NET is Easy to Deploy and Configure:

Every installation of the .NET Framework provides the same core classes. As a result, deploying an ASP.NET application is relatively simple. In most cases, you simply need to copy all the files to a directory on a web server (web host). There are configuration settings that can be made in a **web.config** file.

6. ASP.NET is Multidevice and Multibrowser:
One of the greatest challenges web developers face is the wide variety of browsers they need to support. Different browser brands, versions, and configurations differ in their support of XHTML. ASP.NET server controls render their HTML adaptively by taking the client's capabilities into account.

Web Server for ASP.NET

To view your ASP.NET web pages, it needs to be requested through a web browser running on a web server. The web server then dispatches the request to the ASP.NET engine installed on the web server. The ASP.NET engine processes the page and returns the resulting HTML markup to the browser.

ASP.NET web pages are normally developed and tested locally on your PC. You could develop them locally and test them on your webhost (FTP-ing the files over each time).

Visual Web Developer includes a built-in web server. So anytime you want to test your **.aspx** web pages that you've developed, all you have to do is launch a web browser from Visual Web Developer and it will create a running instance of a temporary web server.

Note: If you have XP Professional, Vista Home Premium, Windows 7 Ultimate or Professional then you have a Microsoft web server (IIS – Internet Information Services) already. You can use this web server instead if you want.

Brief Tour of Visual Studio 2010

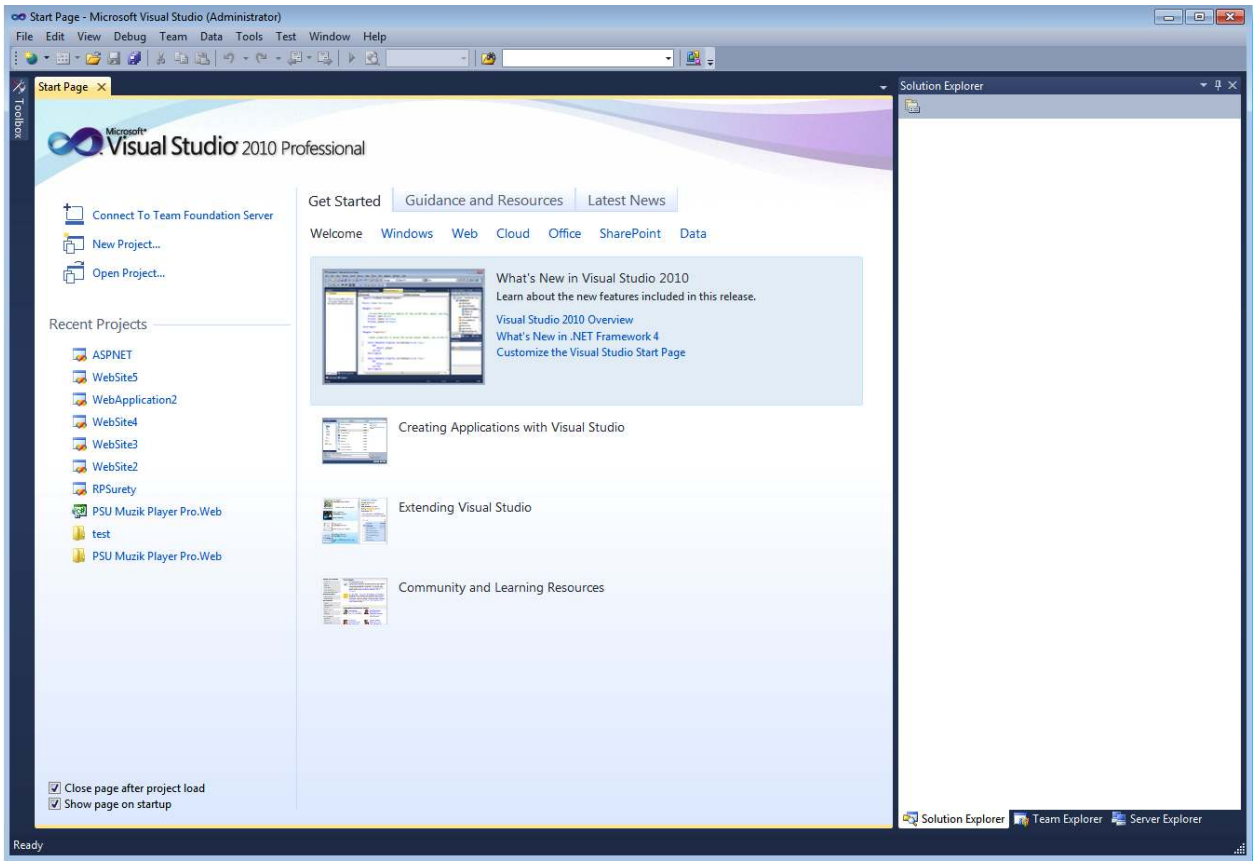

Figure 1 – Visual Studio 2010

When you open Visual Studio 2010, the Start Page is initially shown (Figure 1). This Start Page includes a list of Recent Projects in the upper-left corner, a Getting Started section with some links for accomplishing common tasks in the bottom left-corner, and a list of recent articles on Microsoft's MSDN site in the right column.

On the left you'll find the Toolbox. On the Start Page, the Toolbox is empty, but when you're working with an ASP.NET page, the Toolbox contains the plethora of ASP.NET Web controls that can be added to the page.

To the right of the screen, you'll find the Solution Explorer. Again, on the Start Page this is empty, but when you load or create an ASP.NET website, the Solution Explorer will list the website's files. These files include database files, XHTML pages, ASP.NET

pages, image files, CSS files, configuration files, and so on. In addition to the Solution Explorer, the right portion of the screen is also home to the Database Explorer.

Note: If you accidentally close the Solution Explorer, you can re-open it by going to View->Solution Explorer from the menu.

Creating a New ASP.NET Website

To create and work with an ASP.NET page, we must first create an Empty ASP.NET website.

Note: We are creating an 'Empty' ASP.NET website because in the 2010 version of VS, it adds files to a vase website that I don't you to worry about this early in the class.

You can go to the File menu and choose the New Web Site option and choose Empty Web ASP.NET Website option (Figure 2b).

Make sure you choose the C# language (on the left) as the language for all the web pages (Figure 2b). You can also specify the location of the folder where all your website files will be stored at the bottom (web location – file system).

Normally, you will create a new folder location for each new website.