

## **Microsoft 2310**

# **Developing MS Web Applications w/ Microsoft Visual Studio 2008**

### **Introduction**

Elements of this syllabus are subject to change.

This five-day instructor-led course will teach introductory-level Web developers the fundamentals of Web application development and best practices for Microsoft Web development technologies, including ASP.NET 3.5, ASP.NET AJAX Extensions and Silverlight. This course focuses on using the Microsoft Visual Studio 2008 development environment and the Microsoft .NET Framework 3.5 to create a Web application that delivers dynamic content to a Web site.

### **Audience**

This course is intended for introductory-level Web developers who have knowledge of Hypertext Markup Language (HTML) or Dynamic HTML (DHTML), along with some knowledge of a scripting language such as Visual Basic Scripting Edition or Microsoft JScript.

This course is also appropriate for Microsoft Visual Basic 6.0, Microsoft Visual Basic for Applications (VBA) or classic ASP developers who want to learn ASP.NET 3.5 and other Microsoft Web development technologies.

### **At Course Completion**

After completing this course, students will be able to:

- Describe the key features of the Microsoft .NET Framework and ASP.NET.
- Create a component by using Visual Basic or Visual C#.
- Create a Microsoft ASP.NET Web Form that contains server controls.
- Add functionality to server controls that are on an ASP.NET Web form.
- View runtime information about a Web application by using the Trace and Debug objects.
- Validate user input by using validation controls.
- Create and implement a user control.
- Access data by using Microsoft ADO.NET 3.5 and the built-in data access tools available in Visual Studio 2008.
- Accomplish complex data access tasks from an ASP.NET Web application.
- Access and manipulate data that was stored by using Extensible Markup Language (XML).
- Create and extend an ASP.NET AJAX application.
- Create a Microsoft Silverlight-based application that delivers dynamic content.
- Call a Web service from an ASP.NET Web application and incorporate the returned data into a Web application.
- Store ASP.NET Web application and session data by using a variety of methods.
- Configure and deploy an ASP.NET Web application.
- Secure an ASP.NET Web application by using a variety of technologies, including authentication and authorization.

## Prerequisites

Before attending this course, students must have:

- Knowledge of HTML or DHTML, including:
  - Tables
  - Images
  - Forms
- Programming experience using Microsoft Visual Basic or Microsoft Visual C#, including:
  - Declaring variables
  - Using loops
  - Using conditional statements

The completion of Course 4994, Introduction to Programming Microsoft .NET Framework Applications with Microsoft Visual Studio 2005, satisfies the preceding prerequisite skills requirements for Visual Basic and Visual C#.

## Course Outline

### Module 1: Overview of the Microsoft .NET Framework

This module describes the key features of the Microsoft .NET Framework and ASP.NET.

#### Lessons

- Introduction to the .NET Framework
- Overview of ASP.NET
- Overview of the Lab Application
- Resources

#### Lab: (There is no lab for this module)

After completing this module, students will be able to:

- Explain the advantages of using the .NET Framework.
- Describe the key functionality and purpose of ASP.NET in developing Web applications.
- Describe the basic functionality of the Web site that students will build in the labs in Course 2310C.
- List resources for Web application development with Visual Studio 2008.

### Module 2: Creating Web Applications by Using Microsoft Visual Studio 2008 and Microsoft .NET-Based Languages

This module explains how to create a Web application by using Visual Studio 2008. It also describes how to create a component by using Visual Basic or C#.

## Lessons

- Overview of Visual Studio 2008
- Creating an ASP.NET Web Application Project
- Overview of the Microsoft .NET-Based Languages
- Creating a Component by Using Visual Studio 2008

### **Lab: Creating Web Applications by Using Microsoft Visual Studio 2008 and Microsoft .NET-Based Languages**

- Exercise 1: Creating an ASP.NET Web Site
- Exercise 2: Creating a Class
- Exercise 3: Calling the Component

After completing this module, students will be able to:

- Navigate the Visual Studio 2008 integrated development environment (IDE).
- Create, build, and view an ASP.NET Web application project.
- Identify the languages that support the .NET Framework and choose an appropriate development language for your needs.
- Create a component by using Microsoft Visual Basic or Microsoft Visual C#.

#### Module 3: Creating a Microsoft ASP.NET Web Form

This module describes how to create a Microsoft ASP.NET Web Form that contains server controls and uses a master page.

## Lessons

- Creating Web Forms
- Adding Server Controls to a Web Form
- Creating Master Pages

### **Lab: Creating a Microsoft ASP.NET Web Form**

- Exercise 1: Creating the Default.aspx Web Form
- Exercise 2: Creating the benefitsMaster Master Page
- Exercise 3: Creating the Life.aspx Web Form

After completing this module, students will be able to:

- Add a Web Form to an ASP.NET Web application project.
- Add server controls to a Web Form by using the Microsoft Visual Studio 2008 toolbox.
- Create a Web Form that uses a master page.

#### Module 4: Adding Code to a Microsoft ASP.NET Web Form

This module explains how to add functionality to server controls that are on an ASP.NET Web form.

## Lessons

- Implementing Code-Behind Pages
- Adding Event Procedures to Web Server Controls
- Handling Page Events

### Lab: Adding Functionality to a Web Application

- Exercise 1: Creating a Page\_Load Event Procedure
- Exercise 2: Creating a Click Event Procedure
- Exercise 3: (If Time Permits): Implementing a Component in a User Control

After completing this module, students will be able to:

- Implement code-behind pages in a Web application.
- Create event procedures for Web server controls.
- Handle Page events in a Web application.

### Module 5: Tracing in Microsoft ASP.NET Web Applications

This module explains how to view runtime information about a Web application by using the Trace and Debug objects.

## Lessons

- Tracing and the Trace Object
- Remote Debugging

### Lab: Tracing in Microsoft ASP.NET Web Applications

- Exercise 1: Implementing Trace Statements
- Exercise 2: Tracing into a Component

After completing this module, students will be able to:

- View runtime information about a Web application by using the Trace object.
- View runtime information about a Web application by using the Debug object.

### Module 6: Validating User Input

This module explains how to validate user input by using validation controls.

## Lessons

- Overview of User Input Validation
- Validation Controls
- Page Validation

### **Lab: Validating User Input**

- Exercise 1: Implementing RequiredFieldValidator Controls
- Exercise 2: Implementing the ValidationSummary Control
- Exercise 3: Implementing the CompareValidator Control
- Exercise 4: Implementing the RegularExpressionValidator Control

After completing this module, students will be able to:

- Identify when input validation is appropriate in Web Forms.
- Verify user input on a Web Form by using input validation controls.
- Verify that all validation controls on a page are valid.

### **Module 7: Creating and Implementing User Controls**

This module explains how to create and implement a user control.

#### **Lessons**

- Adding User Controls to an ASP.NET Web Form
- Creating User Controls

### **Lab: Creating and Implementing User Controls**

- Exercise 1: Creating a User Control
- Exercise 2: Implementing a User Control

After completing this module, students will be able to:

- Add a user control to a Microsoft ASP.NET Web Form.
- Create a user control.

### **Module 8: Accessing Data with Microsoft ADO.NET and Visual Studio 2008**

This module explains how to access data by using Microsoft ADO.NET 3.5 and the built-in data access tools available in Visual Studio 2008.

#### **Lessons**

- Overview of ADO.NET
- Connecting to a Database
- Accessing Data
- Accessing Multiple Tables

### **Lab: Accessing Data with Microsoft ADO.NET and Visual Studio 2008**

- Exercise 1: Connecting to the Doctors Database
- Exercise 2: Paging and Selection in a GridView Control
- Exercise 3: Implementing a SqlDataReader
- Exercise 4: (If Time Permits) Viewing Doctors from All Cities

After completing this module, students will be able to:

- Describe the key features of ADO.NET.
- Create a connection to a database by using ADO.NET.
- Access data from a SQL Server database by using a DataSet and DataReader.
- Store multiple tables of data in a DataSet object and then display that data in GridView controls.

#### Module 9: Accomplishing Complex Data Access Tasks

This module explains how to call stored procedures from an ASP.NET Web application. It also covers how to access data from a database by using LINQ to SQL.

#### Lessons

- Overview of Stored Procedures
- Calling Stored Procedures
- Data Access with LINQ to SQL

#### Lab: Accomplishing Complex Data Access Tasks

- Exercise 1: Get Unique City Names
- Exercise 2: Get Doctor Specialties
- Exercise 3: Get Doctor Specialties by Using LINQ to SQL

After completing this module, students will be able to:

- Explain what a stored procedure is and the reasons for using stored procedures when accessing a database.
- Call stored procedures.
- Query and update data in a SQL Server database by using LINQ to SQL.

#### Module 10: Reading and Writing XML Data

This module explains how to access and manipulate data that was stored by using Extensible Markup Language (XML).

#### Lessons

- Overview of XML Architecture in ASP.NET
- XML and the DataSet Object
- Managing XML Data
- Accessing XML Data by Using the XML Web Server Control

#### Lab: Reading XML Data

- Exercise 1: Reading a List of Mutual Funds from an XML File
- Exercise 2: Reading, Transforming, and Displaying XML
- Exercise 3: (If Time Permits): Nested Data

After completing this module, students will be able to:

- Describe XML architecture in Microsoft ASP.NET.
- Read and write XML data into a DataSet object.
- Store, retrieve, and transform XML data by using XmlDocument and XsltTransform objects.
- Display, load, and save XML data by using the XML Web server control.

#### Module 11: Creating an ASP.NET AJAX Application

This module explains how to create and extend an ASP.NET AJAX application.

##### Lessons

- Introduction to ASP.NET AJAX
- Creating an ASP.NET AJAX Application by Using the ASP.NET AJAX Extensions
- Extending an Application by Using the ASP.NET AJAX Control Toolkit

##### Lab: Creating an ASP.NET AJAX Application

- Exercise 1: Implementing Partial Page Rendering with the UpdatePanel Control
- Exercise 2: Installing and Using the AJAX Control Toolkit

After completing this module, students will be able to:

- Explain the purpose of ASP.NET AJAX and list its key components.
- Create an ASP.NET AJAX application by using the ASP.NET AJAX extensions.
- Extend an ASP.NET AJAX application by using the ASP.NET AJAX control toolkit.

#### Module 12: Delivering Dynamic Content with Microsoft Silverlight

This module explains how to create a Microsoft Silverlight-based application that delivers dynamic content.

##### Lessons

- Overview of Microsoft Silverlight
- Creating Silverlight-Based Applications with Visual Studio 2008

##### Lab: Delivering Dynamic Content with Microsoft Silverlight

- Exercise 1: Creating a Microsoft Silverlight-Based Application
- Exercise 2: Adding Dynamic Content to a Microsoft Silverlight Application

After completing this module, students will be able to:

- Describe the purpose and features of Microsoft Silverlight.
- Create a Silverlight-based application by using Visual Studio 2008.

#### Module 13: Consuming and Creating XML Web Services

This module explains how to call a Web service from an ASP.NET Web application and incorporate the returned data into a Web application.

## Lessons

- Overview of Using XML Web Services
- Calling an XML Web Service
- Creating an XML Web Service

### Lab: Consuming and Creating XML Web Services

- Exercise 1: Creating the Dentist XML Web Service and the GetAllDentists XML Web Service Method
- Exercise 2: Creating the GetDentistsByPostalCode XML Web Service Method
- Exercise 3: Consuming the GetAllDentists XML Web Service Method
- Exercise 4: Consuming the GetDentistsByPostalCode XML Web Service Method

After completing this module, students will be able to:

- Describe the purpose and process behind calling an XML Web Service from a Web Form.
- Call an XML Web service directly from a browser by using HTTP and call a Web method from a Web Form.
- Create an XML Web service by using the templates in Visual Studio 2008.

## Module 14: Managing State

This module explains how to store ASP.NET Web application and session data by using a variety of methods.

## Lessons

- State Management
- Application and Session Variables
- Cookies and Cookieless Sessions

### Lab: Storing Application and Session Data

- Exercise 1: Implementing Session Variables
- Exercise 2: Implementing Cookies
- Exercise 3: Implementing Application Variables
- Exercise 4: Storing Session Variables in a Database

After completing this module, students will be able to:

- Describe state management and the options that are available to manage state in an ASP.NET Web application.
- Manage state in an ASP.NET Web application by using application and session variables.
- Manage state in an ASP.NET Web application by using cookies and cookieless sessions.

## Module 15: Configuring, Optimizing, and Deploying a Microsoft ASP.NET Web Application

This module explains how to configure and deploy an ASP.NET Web application.

## Lessons

- Implementing the Cache Object
- ASP.NET Output Caching
- Configuring an ASP.NET Web Application
- Deploying an ASP.NET Web Application

### **Lab: Configuring, Optimizing, and Deploying a Microsoft ASP.NET Web Application**

- Exercise 1: Caching a DataSet by Using the Cache Object
- Exercise 2: Reducing Response Times by Using the Page Output Cache
- Exercise 3: Partial Page Caching
- Exercise 4: Implementing Dynamic Properties
- Exercise 5: Deploying Your Site

After completing this module, students will be able to:

- Store information by using the Cache object.
- Store Web pages and Web page fragments by using ASP.NET output caching.
- Configure an ASP.NET Web application by using the Machine.config and Web.config files.
- Deploy an ASP.NET Web application.

### Module 16: Securing a Microsoft ASP.NET Web Application

This module explains how to secure an ASP.NET Web application by using a variety of technologies, including authentication and authorization.

## Lessons

- Web Application Security Overview
- Windows-Based Authentication
- Forms-Based Authentication

### **Lab: Securing a Microsoft ASP.NET Web Application**

- Exercise 1: Securing Your Web Site by Using Windows-Based Authentication
- Exercise 2: Securing Your Web Site by Using Forms-Based Authentication
- Exercise 3: (If Time Permits): Registering New Users
- Exercise 4: (If Time Permits): Permitting Users to Sign Out

After completing this module, students will be able to:

- Describe the ASP.NET and IIS authentication methods.
- Secure an ASP.NET Web application by using Windows-based authentication.
- Secure an ASP.NET Web application by using Forms-based authentication.