ASP.NET Web Forms Programming Using C#

Duration: 5 Days *(Face-to-Face & Remote-Live)*, or 35 Hours *(On-Demand)*

Price: $2495 *(Face-to-Face & Remote-Live)*, or $1495 *(On-Demand)*

Discounts: We offer multiple discount options. Click here for more information.

Delivery Options: Attend face-to-face in the classroom, remote-live or on-demand training.

Students Will Learn

- Using Visual Studio to create C# applications
- Working with .NET data types
- Creating variables with the proper scope and using operators to build complex expressions
- Designing and using classes
- Using control structures such as `if`, `while` and `for`
- Using procedures to build complex applications
- Throwing and trapping exceptions using the `try` and `catch` statements
- Using single and multi-dimensional arrays
- Working with .NET collections
- Using LINQ to make queries
- Defining and implementing interfaces
- Working with enumerations
- Creating ASP.NET Web Form applications
- Working with CSS in Web Forms
- Using Web server controls to interact with users and display data
- Creating event handlers to respond to control events
- Validating input using ASP.NET validation controls
- Reusing design with master pages and user controls
- Managing state in a Web Forms application
- Using ADO.NET to interact with databases
- Using DataSets to read/write XML data
- Using Data Binding to display complex data on Web Pages
- Using ASP.NET AJAX in Web Forms applications
- Deploying an ASP.NET Web application

Course Description

This course provides students with hands on experience using Visual Studio to create dynamic Web sites with ASP.NET Web Forms and the .NET Framework using C#. The class provides a thorough introduction to the C# programming language, including coverage of

the essentials of the C# programming language, built in data types, operators, control structures, classes and methods, collections and exception handling.

Students then learn how to leverage the power of the .NET Framework to build Web user interfaces. Students will learn use the power of ASP.NET Web server controls combined with HTML5 to design Web pages. They will use ASP.NET Validation controls to provide both client-side and server-side data validation for user input. Students will learn how to use master pages to enforce a consistent look and feel across a set of Web pages, as well as how to use user controls to reuse smaller pieces of HTML across multiple pages.

Students explore the complexities of state management in ASP.NET Web Forms and how to use session and view state objects to manage state between HTTP requests.

Students spend time exploring how to use ADO.NET to interact with databases by running SQL queries and executing stored procedures. They will also learn how to read/write XML files using DataSets and DataTables. Students learn how ASP.NET Web Forms data binding to easily display data in list controls, the GridView control and the DetailsView control.

Microsoft provides support for Ajax-enabled Web Forms applications using the ASP.NET AJAX Framework. Students will learn how to use some of the server-side features to make asynchronous postbacks from the browser, perform partial page updates using the UpdatePanel, use CSS to dim a page during postback and use the Timer fetch data from the server asynchronously.

Other topics include: using a Web.config file to control application configuration; working with the query string; working with cookies; reading and writing files; and deploying ASP.NET web applications. Comprehensive labs provide the students with extensive experience creating and deploying dynamic ASP.NET Web Form applications.

This course provides thorough coverage of the use of **Web Forms** for web sites. Students requiring additional coverage of **ASP.NET MVC**, **WCF**, **Windows Forms** or **Windows Presentation Foundation** should contact HOTT or refer to HOTT's complete course listing for additional training courses.

Students who are already familiar with C# language syntax may want to take the 3-day ASP.NET Web Forms Programming for Experienced C# Programmers course instead.

Course Prerequisites

Knowledge of fundamental HTML syntax is helpful, but not required. Prior experience with a scripting or programming language is required.

Course Overview

Introduction to .NET

- Overview of the .NET Framework
- How .NET is Different from Traditional Programming
- Common Language Runtime (CLR)
- Common Language Specification (CLS)
- Common Type System (CTS)
- .NET Assemblies

Introduction to Visual Studio

- Creating a Project
- Using the Code Editor
- Correcting Syntax Errors
- Setting Project Properties
- Adding References
- Compiling a Program
- Running a Program
- Debugging a Program

Microsoft Intermediate Language (CIL)
- .NET Namespaces
- .NET Framework Class Library

Using the MSDN (Help)

Language Fundamentals

- C# Program Structure
- Defining Namespaces
- Understanding C# Data Types
- Defining Variables and Constants
- Comparing Value Types vs. Reference Types
- Working with Operators and Expressions
- Performing Type Conversions
- Using Console I/O
- Formatting Numbers, Date and Times

Conditionals and Looping

- `if/else`
- `switch`
- `while` and `do/while`
- `for`
- `foreach`

Methods and Parameters

- Defining Static and Instance Methods
- Passing Parameters by value and by reference
- Overloading Methods
- Using Variable Length Parameter Lists

Exception Handling

- What are Exceptions?
- .NET Exception Hierarchy
- Catching Exceptions
- Throwing Exceptions
- Managing Resources with Finally

Collections

- Defining and Using Arrays
- Understanding `System.Array`
- .NET Collections vs Generic Collections
- Working with Lists
- Working with Dictionaries
- Using LINQ to Objects

Object-Oriented Programming

- Overview of Object-Oriented Programming
- Building Classes
- Defining Properties
- Using Auto-Implemented Properties
- Defining Methods
- Understanding Constructors
- Extending .NET Classes via Inheritance
- Defining and Implementing Interfaces
- Understanding the Role of Interfaces in .NET

Overview of ASP.NET Web Forms

- ASP.NET Technologies
- Web Forms vs. MVC
- Understanding the HTTP Request and Response Cycle
- Client-Side vs. Server-Side Code Execution
- Exploring an ASP.NET Web Form
- Understanding the Life-Cycle of an ASP.NET Page Object

Creating ASP.NET Web Forms Applications

- Building ASP.NET Web Forms Applications Using Visual Studio
- Understanding a Web Form Application Project Structure
- Designing Web Forms
- Incorporating CSS into the Web Application
- Understanding the Page Directive
- Working with the Code-Behind Model
- Understanding `Web.config`
- Transformational Config Files

Using ASP.NET Server Controls

- Working with Web Server Controls
- Using Hyperlinks and Button-Based Controls
- Using Label and Textbox Controls
- Using List Controls
- Using the Image Control
- Exploring Other Web Server Controls

Using ASP.NET Validation Controls

- Understanding ASP.NET Validation Controls
- Performing Client-Side Validation
- Performing Server-Side Validation
- Using the Required Field Validator
- Using the Comparison and Range Validators
- Using the Regular Expression Validator
- Customizing Validation
- Working with Validation Groups
- Unobtrusive Validation
- Using the NuGet Package Manager to Download `AspNet.ScriptManager.jQuery`

Maintaining State

- Maintaining State within ASP.NET Web Forms Applications
- Using the `Session` Object
- Using the `ViewState` Object
- Reading and Writing Cookies
- Using the Query String

Using XML

- Introducing XML and XML Schemas
- Writing XML Schemas with a DataSet
- Reading XML Schemas with a DataSet
- Writing XML Data with a DataSet
- Reading XML Data with a DataSet

Introduction to ASP.NET AJAX

Handling Events

- Working with Page-Based Event Handlers
- Writing Control Event Handlers
- Sharing Event Handlers
- Understanding PostBack vs. Non-PostBack Events
- Handling Application-Based Events in `Global.asax`

Reusing Design: User Controls and Master Pages

- Reusing HTML Snippets with User Controls
- Designing a User Control
- Embedding a User Control on a Page
- Understanding the Master Page Architecture
- Designing a Master Page
- Designing Content Pages

ADO.NET

- Understanding the ADO.NET Object Model
- Connected vs. Disconnected Access
- Using a Connection to Connect to a Data Source
- Using a Command to Execute Queries and Stored Procedures
- Using a DataReader to Work with Cursors
- Using a DataSet with Disconnected Data
- Using DataAdapters with DataSets

Using Data Binding

- Understanding ASP.NET Data Source Controls
- Configuring Data Sources
- Examining ASP.NET Data Bound Controls
- Using List Controls
- Using the `GridView` Control
- Using the `DetailsView` Control
- Exploring Other Data Bound Controls

Deploying ASP.NET Applications

- Understanding Ajax
- Enabling ASP.NET AJAX with the ScriptManager
- Configuring Partial Page Updates with the UpdatePanel
- Displaying a Progress Indicator Using UpdateProgress
- Making Periodic Postbacks Using the Timer
- Using the NuGet Package Manager to Download jQuery

- Configuring IIS for Deployment
- Configuring an ASP.NET Web Form Application for Deployment
- Using Publish to Deploy a ASP.NET Web Application