

ASP.NET MVC Programming Using C# and .NET6

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 via [on-demand training](#).

Description

This .NET MVC class provides students with hands on experience using Visual Studio to create dynamic web applications using ASP.NET MVC, C#, and .NET6. The course 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.

During class, students learn how to leverage the power of the Model-View-Controller design pattern with the ASP.NET MVC design pattern to separate the layers of a web application. Students will then use the Razor view engine to design a user interface, and will learn how to build models to manage an application's data layer using both LINQ and the Entity Framework. Students also learn how to build controllers containing action methods to manage communication between views and models.

Other topics in the course include data scaffolding; URL routing; implementing security; unit testing; and deploying ASP.NET MVC applications. Comprehensive labs throughout the training provide students with experience creating, debugging, testing and deploying dynamic ASP.NET MVC applications.

Prerequisites

Knowledge of fundamental HTML syntax is helpful, but not a requirement. Prior experience with any 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
- Microsoft Intermediate Language (CIL)
- .NET Namespaces
- .NET Framework Class Library

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

Methods and Parameters

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

Collections

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

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
- Using the MSDN (Help)

Conditionals and Looping

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

Exception Handling

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

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

Overview of ASP.NET MVC

- Overview of Model-View-Controller Design Pattern
- ASP.NET MVC Application Architecture
- Understanding the MVC Execution Process
- Building an ASP.NET MVC Application Using Visual Studio
- Visual Studio MVC Project Templates
- Using a `web.config` File

Developing Views

- Creating Views
- Using the Razor View Engine
- Using ViewBag to Pass Controller Data
- Using HTML Helpers

Developing Controllers

- Creating Controllers
- Defining Action Methods
- Working with HTML Form Data
- Mapping URLs to Action Methods
- Understanding `ActionResult` Types
- Using Model Binding

MVC Unit Testing

- Test-Driven Development
- Designing Test Cases
- Creating Unit Tests
- Using xUnit for Testing
- Using Visual Studio Test Explorer

Deploying ASP.NET MVC Applications

Startup and Services

- Fundamentals of Dependency Injection
- Using the default DI container
- Using the Startup file
- Creating Services and Controlling Service Lifetime
- Building an ASP.NET MVC Application Using Visual Studio Project Templates
- Using Bootstrap CSS for the UI

Developing Models

- Creating Model Classes
- Working with Strongly-Typed Views
- Validating User Input
- Using Data Annotations for Display
- Working with the Entity Framework
- Working with LINQ to SQL
- Using Scaffolding to Generate Views

Routing Control

- Understanding Routing in ASP.NET MVC
- Defining URL Routes
- Registering Routes
- Adding Constraints to Routes
- Using Attribute Routing

Securing MVC Applications

- ASP.NET Security
- Securing the Application with Individual Accounts
- Securing the Application with Windows Accounts
- Using Attributes to Leverage Access
- Defending against Attacks
 - Cross-site Scripting
 - Session Hijacking
 - SQL Injection
 - Input Forgery

- Understanding the Kestrel Web Server
- Using Publish Profiles
- Choosing a Deployment Destination
- App Config using Environments
- Configuring IIS for ASP.NET MVC Deployment

Software Skills Training, Inc.
6 Hemlock Drive
Chelmsford, MA 01824
978.250.4983
www.software-skills-training.com

Copyright © 2021 Software Skills Training, Inc.