

Full Stack Web Development with C# and ASP.NET Core

Duration: 15 days

Prerequisites: Knowledge of any OOP-based Programming and Web Development

DAY 1 – Programming in C# (Fundamentals)

Module 1: C# Syntax

- Writing Applications in C# and .NET
- Types of Data and Expressions
- C# Language Constructs

Lab: Build a basic class enrollment console app using core C# syntax.

Module 2: C# Language Concepts

- Methods
- Method Overloading
- Exception Handling
- Monitoring

Lab: Enhance the enrollment app with improved methods and error-handling.

DAY 2 – C# Data Structures, Classes, Events & OOP

Module 3: Structures, Collections & Events

- Structs
- Enums
- Built-in Collections
- Events

Lab: Create a grade-tracking prototype using collections and events.

Module 4: Classes, Interfaces & Generics

- Creating Classes
- Interfaces
- Generics in C#

Lab: Implement reusable classes and apply generics for data handling.

DAY 3 – Advanced C#, File I/O, LINQ, EF & Networking

Module 5: Inheritance & Polymorphism

- Hierarchies of Classes
- Polymorphism
- Extending Classes

Lab: Refactor existing code to use inheritance and polymorphism.

Module 6: Input and Output

- File I/O
- Serialization & Deserialization
- Streams

Lab: Create a grades report using file operations and serialization.

DAY 4 – Entity Framework and Database

Module 7: Database Access (EF + LINQ)

- Entity Framework
- LINQ

Lab: Update and retrieve grade data using LINQ and EF.

Module 8: Network Operations

- Web Services
- REST & OData
- ASP.NET Core MVC Introduction

Lab: Consume a remote web service and process response data.

DAY 5 – HTML Basics

Module 1: Overview of Web Development

- HTML, CSS, JavaScript overview
- Client-side vs Server-side
- Web technologies

Lab: None

Module 2: Introduction to HTML

- HTML Structure
- Tags, Attributes, Comments
- Special Characters

Lab: Build a simple HTML page with basic elements.

Module 3: Paragraphs, Headings & Text Formatting

- Paragraphs
- Headings
- Inline Semantic Elements
- Preformatted Text

Lab: Format a webpage using text and semantic tags.

Module 4: HTML Links

- Text Links
- Absolute/Relative Paths
- Email Links

- Internal Page Anchors
- Lab:** Create navigation and anchor-based links.
-

DAY 6 – Images, Lists, CSS Fundamentals

Module 5: HTML Images

- Adding Images
 - Image Links
 - Alt Text
- Lab:** Insert and format various types of images.

Module 6: HTML Lists

- Unordered Lists
 - Ordered Lists
 - Definition Lists
- Lab:** Build structured content using multiple list types.

Module 7: CSS Basics

- CSS Selectors & Combinators
 - CSS Cascade
 - External, Internal & Inline CSS
 - CSS Resets & Normalizers
- Lab:** Apply CSS styling using multiple stylesheet methods.
-

DAY 7 – CSS Fonts, Colors, Text & JavaScript Basics

Module 8: CSS Fonts

- font-family
 - font-size
 - font-weight
 - line-height
- Lab:** Apply font styling and typography enhancements.

Module 9: Colors & Opacity

- color
 - opacity
- Lab:** Add color and transparency styles to page elements.

Module 10: CSS Text

- text-align
- text-decoration
- text-transform
- white-space

- word-spacing
- Lab:** Apply text formatting styles.

Module 11: JavaScript Basics

- JavaScript Syntax
 - DOM Basics
 - Built-in Objects
- Lab:** Write simple scripts for UI interaction and DOM output.
-

DAY 8 – JavaScript Logic, Events, DOM & Forms

Module 12: Variables, Arrays & Operators

- Variables
 - Constants
 - Arrays
 - Operators
- Lab:** Implement small scripts using arrays and operators.

Module 13: Functions

- Global Functions
 - Custom Functions
 - Returning Values
- Lab:** Create custom functions to process user data.

Module 14: Built-in JavaScript Objects

- String
 - Math
 - Date
- Lab:** Use built-in objects to solve small logic tasks.
-

DAY 9 – Loops, Conditional and Event Handling

Module 15: Conditionals & Loops

- if/else
 - switch/case
 - for / while loops
- Lab:** Implement logic flows with conditions and loops.

Module 16: Event Handlers & Listeners

- on-event Handlers
 - addEventListener()
 - Timers
- Lab:** Add events to enable interactive webpage behavior.
-

DAY 10– DOM Manipulation, Forms & Validation

Module 17: The DOM

- Element Selection
- Node Creation/Removal
- Hierarchical Access

Lab: Build a dynamic item list using DOM methods.

Module 18: HTML Forms

- Inputs, Checkboxes, Radios
- Select Menus, Textareas
- Fieldsets

Lab: Create a complete user registration form.

Module 19: JavaScript Form Validation

- Accessing Form Data
- Validating Inputs
- Error Messages

Lab: Validate form fields with JavaScript.

Module 20: Regular Expressions

- RegEx Syntax
- Backreferences
- Form-cleaning patterns

Lab: Clean and validate inputs using Regular Expressions.

DAY 11 – ASP.NET Core MVC Essentials

Module 1: Exploring ASP.NET Core

- Web Technologies
- Razor Pages
- Intro to MVC

Lab: Explore project structure in a starter ASP.NET Core app.

Module 2: Designing MVC Applications

- Development Methodologies
- MVC vs Razor Pages
- Designing Models/Views/Controllers

Lab: Create a simple architecture design for an MVC app.

Module 3: Razor Pages & Middleware

- Razor Pages
- Middleware Pipeline
- Services & DI

Lab: Build a small Razor Page and configure middleware.

Day 12—Controllers and Views

Module 4: Developing Controllers

- Controllers/Actions
- Routing
- Action Filters

Lab: Create controllers with custom routing.

Module 5: Developing Views

- Razor Syntax
- Tag Helpers
- Partial Views

Lab: Build views using Razor markup and helpers.

DAY 13 – Advanced MVC: EF Core, Security

Module 6: Developing Models

- Model Classes
- Forms & Validation

Lab: Implement validated models in MVC.

Module 7: Entity Framework Core

- EF Core Fundamentals
- Database Providers

Lab: Perform CRUD operations using EF Core.

Module 8: Layouts, CSS & JavaScript Integration

- Layout Pages
- Applying CSS & JS

Lab: Create a shared layout and add client-side resources.

Day 14 – Testing and Client-Side Development

Module 9: Client-Side Development

- Bootstrap
- Sass
- Responsive Design

Lab: Apply responsive UI design to MVC pages.

Module 10: Testing & Troubleshooting

- Unit Testing
- Exception Handling

- Logging
- Lab:** Debug MVC app and configure error logging.
-

Day 15-APIs, Security and Performance

Module 11: Managing Security

- Authentication
 - Authorization
 - Preventing Common Attacks
- Lab:** Add login, roles, and secure page access.

Module 12: Performance & Communication

- Caching
 - State Management
 - SignalR
- Lab:** Implement caching and real-time updates.

Module 13: Implementing Web APIs

- REST API
 - Creating APIs
 - Consuming APIs
- Lab:** Build and consume a Web API inside MVC.