Course duration

- 4 days

Course Benefits

- Understand the goals and benefits of the .NET Core platform
- Learn to make good decisions about application architecture and the choice of data access technology
- Use ASP.NET Core's routing system to achieve a REST-style architecture
- Learn how to build a compelling and maintainable HTML user interface using the Razor view engine and client-side JavaScript
- Gain experience using Blazor, Razor Pages, SignalR, and gRPC
- Learn how to extend and modify ASP.NET Core by creating custom components and templates
- Understand the different cross-platform deployment options available including via Docker containers

Available Delivery Methods

Public Class
Public expert-led online training from the convenience of your home, office or anywhere with an internet connection. Guaranteed to run .

Private Class
Private classes are delivered for groups at your offices or a location of your choice.

Course Outline

1. Introduction
    1. Evolution of .NET and .NET Core
    2. .NET SDKs and Runtimes
    3. Visual Studio and Visual Studio Code
2. .NET SDK
    1. Installation
    2. Version Management
    3. Command-Line Interface (CLI)
3. What's New in C# 9
    1. Introduction

    2. Record Types
    3. Init Only Setters
    4. Nullable Reference Types
4. ASP.NET Core Application Architecture
    1. Introduction
    2. NuGet Packages
    3. Application Startup
    4. Hosting Environments
    5. Middleware and the Request Pipeline
    6. Services and Dependency Injection
    7. MVC vs. Razor Pages
5. Application Configuration
    1. Configure and ConfigureServices
    2. Configuration Providers and Sources
    3. Configuration API
    4. Options Pattern
    5. HTTPS and HTTP/2
6. Request Routing
    1. RESTful Services
    2. Endpoint Routing
    3. Route Templates
    4. Route Constraints
    5. Attribute-Based Routing
7. Models
    1. Introduction
    2. Persistence Ignorance
    3. Object-Relational Mapping
    4. Entity Framework (EF) Core
    5. Dapper ORM
8. Controllers
    1. Responsibilities
    2. Requirements and Conventions
    3. Dependencies
    4. Action Results
9. Views
    1. Responsibilities
    2. Conventions
    3. Razor Syntax
    4. Layouts
    5. ViewData and ViewBag
    6. Strongly-Typed Views
    7. Partial Views
    8. HTML and URL Helpers
    9. Tag Helpers
    10. View Components
    11. Client-Side Dependencies
    12. Razor Pages

    13. View Models
10. HTML Forms
    1. Form Tag Helper
    2. Input Tag Helper
    3. Select Tag Helper
    4. Form Submissions
    5. Model Binding
11. Data Validation
    1. Introduction
    2. Data Annotations
    3. Model Binding
    4. Input Tag Helpers
    5. Validation Tag Helpers
12. Application State
    1. Client-Side vs. Server-Side
    2. HttpContext.Items
    3. Session State
    4. TempData
13. Error Handling
    1. Best Practices
    2. HTTP Error Status Codes
    3. Status Code Pages
    4. Developer Exception Page
14. Logging
    1. Configuration
    2. ILogger
    3. Serilog and Seq
15. Testing
    1. Unit Testing
    2. xUnit
    3. Testing Controllers
    4. Integration Testing
16. Authentication
    1. ASP.NET Core Identity
    2. Cookie Middleware
    3. Authorization
    4. Claims-Based Authorization
17. Web APIs
    1. API Controllers
    2. Testing APIs
    3. CRUD Operations
    4. OpenAPI (Swagger)
    5. Cross-Origin Resource Sharing (CORS)
18. Remote Procedure Calls (gRPC)
    1. Introduction
    2. Protobuf
    3. Server

    4. Client
    5. Limitations
19. Blazor
    1. Razor Components
    2. Blazor Server
    3. Blazor WebAssembly
20. Deployment
    1. dotnet publish
    2. Kestrel
    3. IIS
    4. Docker
21. Conclusion

Class Materials

Each student will receive a comprehensive set of materials, including course notes and all the class examples.

Class Prerequisites

Experience in the following *is required* for this ASP.NET class:

- Previous experience developing web-based applications with C#.
- Some familiarity with HTML, CSS, and JavaScript.