



Web API Development with ASP.NET Core 6

Course ISI-1567 4 Days Instructor-led, Hands-on

Introduction

This ASP.NET Core 6 training course teaches attendees how to design and build high-performance, secure, well-architected REST APIs that can be consumed by a variety of clients. This course focuses specifically on APIs and does not include coverage of web UI topics such as Views and Razor Pages. An introduction to Blazor is included but it is not covered in-depth.

At Course Completion

After completing this course, students will be able to:

- Understand the goals and benefits of ASP.NET Core 6.0
- Learn to make good decisions about application architecture and data access technology
- Use ASP.NET's routing system to achieve a REST-style architecture
- Gain experience building a service that makes data available via a modern web API
- Learn best practices for employing unit testing, logging, and error handling
- Understand different authentication choices for securing a web API
- Get an introduction to Blazor and gRPC
- Understand the different cross-platform deployment options available including via Docker containers

Prerequisites

Before attending this course, students should be comfortable with the following:

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

Course Materials

The student kit includes a workbook and other necessary materials for this class.

Course Outline

Module 1: Introduction

- Evolution of .NET and .NET Core

Contact ISInc for more information at 916.920.1700 or by visiting our website at <http://www.isinc.com>



- .NET SKDs and Runtimes
- Visual Studio and Visual Studio Code

Module 2: .NET 6.0 SDK

- Installation
- Version Management
- Command-Line Interface (CLI)

Module 3: What's New in C#

- Record Types
- Init Only Setters
- Nullable Reference Types
- Global Using Directives
- File-Scoped Namespace Declarations
- Top-Level Statements

Module 4: ASP.NET Core Application Architecture

- NuGet Packages
- Application Startup
- Hosting Environments
- Middleware and the Request Pipeline
- Services and Dependency Injection

Module 5: Application Configuration

- Configuration Providers and Sources
- Configuration API
- Options Pattern
- HTTPS and HTTP/2

Module 6: Request Routing

- RESTful Services
- Endpoint Routing
- Route Templates
- Route Constraints
- Route Template Precedence
- Attribute-Based Routing

Module 7: Models

- Persistence Ignorance
- Dependency Inversion
- Asynchronous Data Access
- Object-Relational Mapping

Contact ISINC for more information at 916.920.1700 or by visiting our website at <http://www.isinc.com>



- Entity Framework Core
- Dapper ORM

Module 8: Controllers

- Responsibilities
- Requirements and Conventions
- Dependencies
- Action Results
- ApiController Attribute

Module 9: Web APIs

- API Controllers
- Minimal APIs
- OpenAPI / Swagger
- Testing APIs
- Content Negotiation
- CRUD Operations
- Patch Requests
- Microservice Architecture
- API Gateway Pattern
- Cross-Origin Resource Sharing (CORS)

Module 10: Data Validation

- Introduction
- Data Annotations
- Model Binding

Module 11: Error Handling

- Best Practices
- HTTP Error Status Codes
- Exception Handling Middleware

Module 12: Logging

- Configuration
- ILogger
- Serilog and Seq

Module 13: Testing

- Unit Testing
- xUnit
- Testing Controllers
- Integration Testing

Contact ISINC for more information at 916.920.1700 or by visiting our website at <http://www.isinc.com>



Module 14: Security

- Authentication
- ASP.NET Identity
- Authorization
- Web API Authentication
- JSON Web Tokens (JWT)
- OAuth 2.0 and OpenID Connect
- Secrets Management

Module 15: Remote Procedure Calls (gRPC)

- Introduction
- Protobuf
- Server
- Client
- Limitations

Module 16: Blazor

- Razor Components
- Blazor Server
- Blazor WebAssembly

Module 17: Deployment

- dotnet publish
- Kestrel
- IIS
- Docker