

ASP.NET Core Web API 10.0 Programming

Start learning how to build modern RESTful Web APIs using ASP.NET Core and become a professional Web API Developer from a beginner/novice level.

About the Course & Importance of ASP.NET Core Web API

ASP.NET Core Web API is Microsoft's modern framework for building high-performance, secure, scalable, and cross-platform RESTful services. It is widely used by organizations to develop enterprise applications, cloud-native solutions, mobile backends, microservices, e-commerce platforms, banking systems, healthcare applications, and Software-as-a-Service (SaaS) products. Whether you are a beginner or an experienced software professional, learning ASP.NET Core Web API is an essential skill for developing modern distributed applications that communicate efficiently across different platforms and devices.

This course is designed for students, fresh graduates, working professionals, software developers, and architects who want to build a strong foundation in ASP.NET Core Web API 10.0. It starts with Web API fundamentals and gradually progresses to advanced concepts such as REST Architecture, HTTP Methods, Routing, Model Binding, Model Validation, Dependency Injection, Entity Framework Core, Repository Pattern, CRUD Operations, Authentication & Authorization using JWT, Logging, Exception Handling, API Versioning, Caching, Rate Limiting, File Upload & Download, Background Services, Minimal APIs, GraphQL, OData, gRPC, Swagger/OpenAPI Documentation, API Security, Performance Optimization, and Deployment through practical real-world examples.

By the end of this course, you will have the confidence and practical knowledge needed to design, develop, secure, test, document, and deploy enterprise-grade Web APIs. You will also gain the skills required to build scalable backend services that integrate seamlessly with web applications, mobile applications, cloud platforms, and microservices-based architectures.

Learning ASP.NET Core Web API:

There are 4 phases to learn ASP.NET Core Web API.

1. Phase I –Introduction to APIs – Foundation of REST & Web API
2. Phase II –Building Real-World APIs (CRUD + EF + Services)
3. Phase III –Advanced API Security & Infrastructure
4. Phase IV –Modern API Approaches (Minimal APIs ,OData ,GraphQL & gRPC)

Pre-Requisites:

NA

Duration : 45 days

Course Materials & Textbooks : 1. Running Notes in class, 2. Pdfs, 3. HandOuts, 4. Certification Material in Pdfs will be provided for this course.

Reference Websites:www.dotnetgurukul.com

Instructor: Praveen Kumar M

Learning from Praveen Sir, is always different and you will experience it within few sessions – you attend and work regularly as per given guidance then you will be best in any team that you work in any company – assured.

Phase I - Introduction to APIs – Foundation of REST & Web API -

Available Free on YouTube

Topic	Key Concepts Covered
Introduction to APIs – Foundation of REST & Web API	Client–Server Architecture, Types of APIs (Public, Private, Partner), API Communication Styles (REST, SOAP, GraphQL, gRPC, OData), Requests & Responses (Path, Query, Body), JSON, HTTP Status Codes
REST Architecture, API Evolution & Core Constraints	What is REST, Why REST, REST Constraints (Client–Server, Resources, Representations, Statelessness, Uniform Interface, Cacheability, Layered System)
ASP.NET Core Web API – Project Structure & Pipeline	Web API Project Structure, Enterprise Folder Organization, Middleware Pipeline, Controllers, Controller vs ControllerBase, Benefits of [ApiController]
Web API Controllers & Routing Behavior	[ApiController] Features, Automatic Model Validation, Attribute Routing, Action Selection, Routing Conflicts
Routing Types & Advanced Routing	Attribute Routing vs Conventional Routing, Route Parameters, Optional & Default Values, Route Constraints (int, alpha, length, regex, guid, decimal)
HTTP Methods & Web API Operations	HTTP Methods (GET, POST, PUT, DELETE, PATCH, HEAD, OPTIONS), Safe vs Unsafe Methods, Idempotent vs Non-Idempotent Operations
Model Binding, DTOs & Validation	Model Binding Sources ([FromRoute], [FromBody], [FromQuery], [FromHeader]), DTOs vs Entities, Model Validation Pipeline
HTTP Status Codes & API Responses	HTTP Status Code Categories (2xx, 4xx, 5xx), Common Response Codes (200, 201, 204, 400, 404, 409, 500), Real-World API Response Design

Phase II – Building Real-World APIs (CRUD + EF + Services)- Premium

Phase

Topic	Key Concepts Covered
Dependency Injection in Web API (Services & Lifetimes)	Introduction to Dependency Injection (DI), Benefits of Decoupling & Testability, Service Lifetimes (Singleton, Scoped, Transient), Request-Based Lifetime Behavior
Repository Pattern for Web API	Repository Pattern, Clean Controller Design, IRepository & Repository Implementation, Optional Service Layer, Scoped DbContext, Dependency Injection Request Lifecycle
Applying DI Lifetimes in Real-World Applications	Choosing the Appropriate Service Lifetime, Transient Services, Singleton Services, Lifetime Rules, Best Practices
Entity Framework Core Integration & CRUD Operations	Entity Framework Core Overview, DbContext, DbSet, CRUD Operations (POST, PUT, DELETE), Clean Architecture Flow, Basic Exception Handling

Paging, Sorting & Filtering in Web APIs	Query Parameter Filtering, LINQ-Based Filtering, Sorting (Ascending & Descending), Pagination using Skip() & Take(), Edge Cases
AutoMapper for Web APIs (Entity $\rightleftharpoons$ DTO)	Importance of DTOs, Request & Response DTOs, AutoMapper Concepts, Manual vs Automatic Mapping, Performance Considerations
N-Tier Architecture for Web APIs	Layered Architecture, API Layer, Application Layer, Domain Layer, Infrastructure Layer, Transition from Repository Pattern to Enterprise Architecture
Asynchronous APIs & Performance Optimization	Importance of Asynchronous Programming, Thread Blocking, async/await, Task-Based Asynchronous Pattern (TAP), Scalability & Performance
File Upload & Download in Web APIs	IWebHostEnvironment, IFormFile, multipart/form-data, File Validation, File Upload & Download Strategies

Phase III –Advanced API Security & Infrastructure - Premium Phase

Topic	Key Concepts Covered
Rate Limiting in ASP.NET Core Web API	Introduction to Rate Limiting, Why APIs Need Rate Limits, Real-World Use Cases (Login, OTP, Banking APIs), IP-Based vs Client-Based Limiting, Time Windows & Thresholds, HTTP 429 (Too Many Requests), Best Practices for Secure APIs
JWT Authentication in ASP.NET Core Web APIs	Authentication vs Authorization, JWT Structure (Header, Payload, Signature), Real-World JWT Usage, JWT Configuration in appsettings.json, Issuer, Audience, Secret Key, Token Expiration, Claims
OAuth 2.0 Authentication in ASP.NET Core Web API	OAuth 2.0 Fundamentals, OpenID Connect (OIDC), Authorization Flow, Access Token vs ID Token, Google OAuth Endpoints, Scopes, Token Decoding
Role-Based & Policy-Based Authorization (JWT)	JWT Recap, Claims vs Roles vs Policies, Adding Role Claims to JWT, Creating Authorization Policies, Role-Based vs Policy-Based Authorization
CORS in ASP.NET Core Web APIs (Cross-Origin Resource Sharing)	Same-Origin Policy, Understanding CORS Errors, Configuring CORS Policies, AllowAnyOrigin, AllowAnyMethod, AllowCredentials, Security Considerations
Global Exception Middleware in ASP.NET Core	Exceptions vs Errors, Common API Error Scenarios (400, 409, 500), Limitations of try...catch in Controllers, Middleware-Based Global Exception Handling, RequestDelegate, Request Pipeline
Input Validation using FluentValidation	Why Validation Should Not Be Performed in Controllers, FluentValidation Fundamentals, FluentValidationAutoValidation, Request Pipeline Validation, Rule-Based Validation, Custom Validation Messages

Phase IV –Modern API Approaches (Minimal APIs ,OData ,GraphQL & gRPC) - Premium Phase

Topic	Key Concepts Covered
Introduction to Minimal APIs	Motivation & Evolution of Minimal APIs, Controllers vs Minimal APIs, Program.cs-Only API Model, Characteristics of Minimal APIs, Recommended Use Cases, Scenarios Where Minimal APIs Should Be Avoided
Parameter Binding & Request Handling in Minimal APIs	Lambda Handlers (MapGet, MapPost, MapPut, MapDelete), Request Lifecycle, Convention-Based Parameter Binding, Route, Query, Header & Body Parameters
Dependency Injection in Minimal APIs	Dependency Injection Fundamentals, Controllers vs Minimal APIs DI Model, Registering Services (Singleton, Scoped, Transient), Injecting Services into Lambda Handlers, Using Built-In Services (ILogger, IConfiguration)
OData in ASP.NET Core	Introduction to OData, Challenges with Traditional APIs, OData Query Options, Entity Data Model (EDM), OData Request Pipeline, Metadata, OData Convention Model Builder, IQueryable, Filtering, Sorting, Paging, Field Selection
Introduction to GraphQL	GraphQL Fundamentals, Over-Fetching & Under-Fetching Problems, Schema, Types, Queries, Mutations, GraphQL Setup in ASP.NET Core, Product & Order Schema Design, Single Endpoint for Multiple Entities, Error Handling, Performance Basics
gRPC in ASP.NET Core	Introduction to gRPC, Real-World Use Cases, Protocol Buffers (Proto Files), Unary & Streaming Calls, Creating gRPC Services, Message & Service Definitions, Client-Server Communication, Performance Comparison with REST
REST vs Minimal APIs vs OData vs GraphQL vs gRPC	Architectural Differences, Performance Comparison, Payload Size & Network Efficiency, Strengths & Limitations, Real-World Use Cases, API Style Selection, Hybrid API Architecture Design

Frequently Asked Questions (FAQ)

1. How do I purchase a course?

Purchasing a course is simple.

1. Contact us through ****WhatsApp:9000366774**** or ****Email : contactdotnetgurukul@gmail.com****
2. We'll share the payment details (UPI / Bank Transfer) based on your preferred payment method.
3. Once the payment is completed, please share the payment confirmation (screenshot or transaction ID).
4. After **verifying your payment, we'll create your account (if required) and provide access to your purchased course through our learning portal (Learnyst).**
5. You can then log in to Learnyst and start learning immediately.

Our team typically provides course access within a few hours after payment verification. In most cases, access is granted on the same day.

2. Will I get support if I have technical questions or doubts?

Yes. We provide support for course-related technical questions and concept clarifications.

If you have any doubts **while learning, you can reach out to us via ****WhatsApp****, ****Email****, or ****Google Meet / Microsoft Teams****. We'll do our best to help you understand the concepts, resolve technical issues, and guide you in the right direction.**

Our goal is to ensure you have a smooth learning experience and successfully complete the course.