**[Day 1 : Introduction To .NET Core 3.x and ASP.NET Core 3.x](https://www.abhishekluv.in/online-one-on-one-asp-net-core-mvc-training-and-coaching" \l "collapseTwo1)**

* .NET Core overview
* .NET Core features
* Discussing about .NET Core 3.x and .NET 4.x
* Development tools: Visual Studio Community and Visual Studio Code
* ASP.NET Core overview
* ASP.NET Core 3.x features
* Benefits of ASP.NET Core 3.x
* Understanding .NET CLI
* Using CLI Commands
* Creating a new Project using .NET CLI
* Understanding ASP.NET Core Project templates and Folder Structure

**[Day 2 : ASP.NET Core Fundamentals, Startup Class, Kestrel, Middleware, Razor Pages](https://www.abhishekluv.in/online-one-on-one-asp-net-core-mvc-training-and-coaching" \l "collapseTwo21)**

* Application startup in ASP.NET Core
* Kestrel: Web Server for ASP.NET Core
* Middleware and Request Pipeline
* ASP.NET Core Configuration
* Serving Static Files in ASP.NET Core
* Understanding Razor Pages
* Creating Simple ASP.NET Core Application

**[Day 3 : ASP.NET Core Routing, Controller Basics, Action Results, Selectors, and Verbs](https://www.abhishekluv.in/online-one-on-one-asp-net-core-mvc-training-and-coaching" \l "collapseThree31)**

* Understanding MVC Design pattern
* Building the first ASP.NET Core MVC Application
* Routing in ASP.NET Core
* Routing Constraints and Attribute Routing
* Working with Controllers
* Understanding Action Results, Action Selectors and Action Verbs

**[Day 4 : Razor View Engine, Strongly Typed View, Tag Helpers, Layouts, Sections](https://www.abhishekluv.in/online-one-on-one-asp-net-core-mvc-training-and-coaching" \l "day4)**

* View Basics
* Razor View Engine
* Layouts and Sections
* Strongly-Typed Views
* Tag Helpers in ASP.NET Core
* \_ViewStart
* \_ViewImports
* View Components
* Creating Custom Tag Helpers

**[Day 5 : ViewBag, ViewData, Model and ViewModel Basics, Forms and Model Binding](https://www.abhishekluv.in/online-one-on-one-asp-net-core-mvc-training-and-coaching" \l "day5)**

* Data Passing from Controller to View using ViewBag
* Data Passing from Controller to View using ViewData
* Data Passing from Controller to View using TempData
* Data Passing from Controller to View using Model and ViewModel
* Understanding HttpGet and HttpPost attribute verbs for Action Methods
* Understanding Model Binding feature
* Benefits of Model Binding
* Accessing Form data in Controller using Model Binding
* Accessing Form data in Controller using Model Binding with ViewModel
* Accessing Form data in Controller using Form input fields as parameters in action method
* Sessions, QueryString and Cookies

**[Day 6 : Model and Form Validation, Validation Tag Helpers, Client-Side Validation in ASP.NET Core](https://www.abhishekluv.in/online-one-on-one-asp-net-core-mvc-training-and-coaching" \l "day6)**

* Understanding Model and Form Validation in ASP.NET Core MVC
* Server-Side Validation
* using Validation attributes from System.ComponentModel.DataAnnotations namespace
* Validation Tag Helpers
* Client-Side Validation

**[Day 7 : Mastering LINQ before using Entity Framework Core](https://www.abhishekluv.in/online-one-on-one-asp-net-core-mvc-training-and-coaching" \l "day7)**

* Anonymous Methods
* Delegates and Lambda Expressions
* IEnumerable
* Extension methods
* Implicitly-typed local variables and Anonymous types
* LINQ Query operators
* LINQ keywords and Syntax
* Select Many, Grouping, Joining
* Aggregation operator, Conversion and Caching

**[Day 8 : Mastering Entity Framework Core #1](https://www.abhishekluv.in/online-one-on-one-asp-net-core-mvc-training-and-coaching" \l "day8)**

* Understanding what is an ORM
* Introduction To Entity Framework Core
* Benefits of Entity Framework Core
* Entity Framework Core Version History and Feature History
* Understanding Different approaches of EF Core
* Getting started with EF Core Code-First approach
* Downloading and Installing EF Core using Nuget Package Manager Console
* Understanding DbContextOptions
* Writing CRUD Data access code using EF Core

**[Day 9 : Mastering Entity Framework Core #2](https://www.abhishekluv.in/online-one-on-one-asp-net-core-mvc-training-and-coaching" \l "day9)**

* Using Data Annotation attributes for Data Modeling
* Using Data Annotation attributes for Validation
* Using EF Core Fluent API for Data Modeling
* Overriding OnModelCreating method
* Overriding OnConfiguring method
* Creating 1-to-Many relationship using Annotations and Fluent API
* Creating Many-to-Many relationship using Annotations and Fluent API
* Understanding the need of EF Core Code-First Migration
* Seeding the Database
* Calling Stored Procedures and Functions using EF Core

**[Day 10 : Mastering Entity Framework Core #3 with Repository, Unit Of Work & ASP.NET Core Dependency Injection](https://www.abhishekluv.in/online-one-on-one-asp-net-core-mvc-training-and-coaching" \l "day10)**

* Understanding Repository & Unit Of Work pattern, DI & IoC
* Creating Repository Layer using Interfaces and Concrete classes
* Managing dependencies using Built-In Dependency Injection in ASP.NET Core
* ConfigureServices -> IServiceCollection -> Transient, Singleton, Scoped
* Injecting dependencies using Dependency Injection (Constructor Injection)
* Using Unit Of Work pattern to commit to the database

**[Day 11: ASP.NET Core Identity Authentication and Authorization #1](https://www.abhishekluv.in/online-one-on-one-asp-net-core-mvc-training-and-coaching" \l "day11)**

* Introduction To ASP.NET Core Identity
* Using the Authorize attribute
* Creating a Custom User using IdentityUser
* Configuring Identity for ASP.NET Core Web Application
* Implementing the Register action method
* Implementing the Login and Logout action method

**[Day 12 : Creating RESTful API with ASP.NET Core](https://www.abhishekluv.in/online-one-on-one-asp-net-core-mvc-training-and-coaching" \l "day12)**

* Understanding HTTP Basics
* API Basics
* JSON basics
* Creating Action Methods with GET,POST,PUT, and DELETE
* Writing CRUD operation methods
* Using PostMan/Fiddler to Test the API's

**[Day 13: ASP.NET Core Identity Authentication and Authorization #2](https://www.abhishekluv.in/online-one-on-one-asp-net-core-mvc-training-and-coaching" \l "day13)**

* Implementing Login feature for API's
* Using JWT Tokens for API

**[Day 14: Gird View using jQuery DataTable and RESTful API in ASP.NET Core](https://www.abhishekluv.in/online-one-on-one-asp-net-core-mvc-training-and-coaching" \l "day14)**

* Consuming JSON data using jQuery DataTable
* Using Bootstrap Model popup for CRUD operations
* Using AJAX calls for CRUD operations

**[Day 15-20 : Real-Time Project](https://www.abhishekluv.in/online-one-on-one-asp-net-core-mvc-training-and-coaching" \l "day15)**

We will be creating a Content Management System(CMS) using ASP.NET Core, EF Core, and many other technologies. All the topics that are to be covered in project work are listed below:

* Creating a Multi-Layered Application
* Creating separate projects for Data, Model, Web
* Managing package dependencies using Nuget
* Referencing projects to other projects
* Using Models and ViewModels
* Using EF Core Fluent-API for Data Modeling
* Using EF Core Code-First Migration
* Generating Repository interfaces and concrete class
* Creating interface & concrete class for Unit Of Work
* Managing dependencies using ASP.NET Core: ConfigureServices methods
* Understanding Lazy & Eager loading in EF Core
* Disabling Lazy loading and ProxyCreation
* Using Include method to fetch related data using JOIN Sql Query via Entity Framework Core
* User and Role Management using ASP.NET Core Identity
* CMS features: Pages, Sidebars, Menu
* Using RichTextEditor for Pages,Sidebar
* Creating Admin area
* Using Bootstrap to style the Layout file
* Using LINQ with EF Core
* Calling Stored Procedure with EF Core
* Securing ASP.NET Core MVC application: Antiforgery tokens, CSRF, and more..
* Applying Caching within ASP.NET Core MVC application for performance
* Creating Partial views
* Displaying notifications using TempData and Bootstrap alerts
* Creating Custom routes with Custom URL patterns
* Using attribute routing
* Using Client-Side validation
* Creating ViewComponents to display Navigation and Sidebar area in Layout file
* Restricting permissions & functionality to admin role users only using Authroize attribute