# ASP.NET Core MVC 6.0

**Education and Training Solutions 2023**

**The Learning Hub**

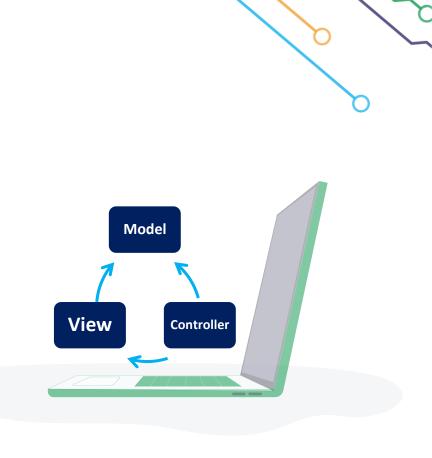1. Overview of ASP.NET Core MVC

2. Differences between .NET Core and .NET Framework

3. Create ASP.NET Core MVC Project

4. ASP.NET Core Project Structure

5. Packages Installation

Model

View

Controller

# Overview of ASP.NET Core MVC

## Overview of ASP.NET Core MVC

**.NET** is a software framework that is designed and developed by Microsoft. The first version of the .NET framework was 1.0 which came in the year 2002. In easy words, it is a virtual machine for compiling and executing programs written in different languages like C#, VB.NET, etc.

## Overview of ASP.NET Core MVC

The **MVC** is an architectural design pattern used in developing software systems, especially web applications. There are three main components in the MVC: Model, View, and Controller.  It follows the principle of separating different aspects of an application to make it more manageable, testable, and maintainable.

**Overview of ASP.NET Core MVC**

**MVC consist of three main components:** Models, Views, and Controllers,  this architectural pattern helps to achieve separation of concerns.

**Overview of ASP.NET Core MVC**

**Model:** is an MVC component that represents the classes of database tables.

The model is the central component of the MVC pattern application.

**Model represents the data.**

## Overview of ASP.NET Core MVC

**View:** The View in an MVC application represents the content through the user interface.
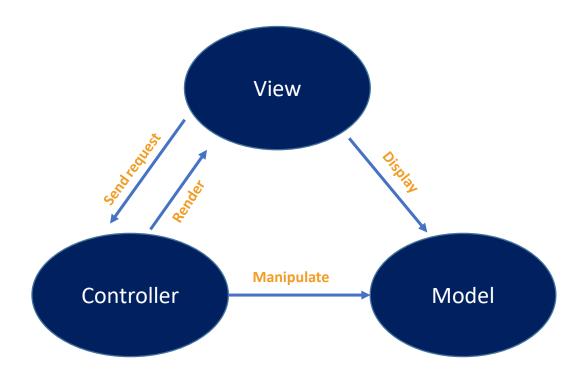
Razor view is used to embed .NET code in HTML markup.
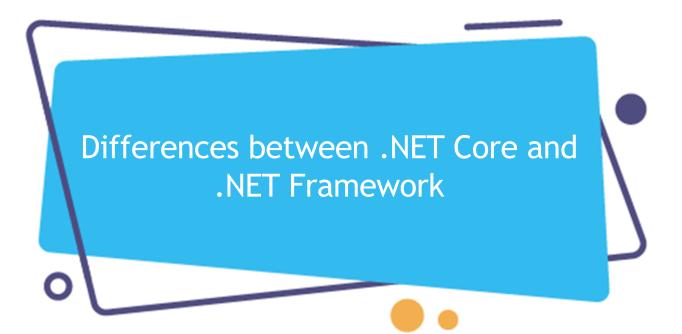
**View is the User Interface.**

**Overview of ASP.NET Core MVC**

**Controllers:** are the components that process user interaction, work with the model, and finally select a view to render.

**Controller is the request handler.**

# Overview of ASP.NET Core MVC

# Differences between .NET Core and .NET Framework

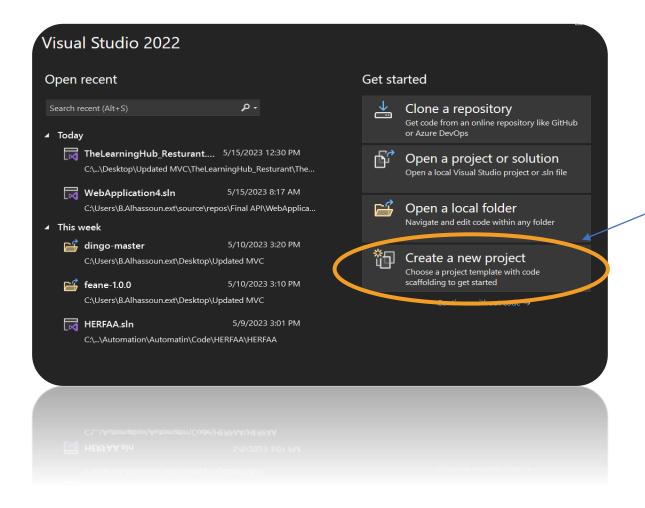| .NET Framework | .NET Core |
|---|---|
| Is the first implementation of .NET which works on Windows platform only. | is the latest implementation of .NET which runs on Windows, Linux, and macOS(**cross-platform**). |
| The .Net Framework includes certain open-source components. | open-source and Microsoft accepts third-party contributions to .NET Core. |
| has a very rich desktop top development framework for windows application which include Windows Forms and WPF. | supports desktop frameworks like Windows Forms and WPF from version 3.0. |

| .NET Framework | .NET Core |
|---|---|
| .NET Framework is less scalable and provides low performance in comparison to .NET Core. | .NET core provides high scalability and performance in comparison to .NET Framework. |
| A huge third-party packages library is available for .NET Framework. | Support a large number of third-party packages, it doesn't compete with .NET Framework in this area because it's much newer than the .NET framework. |

# Create ASP.NET Core MVC Project

➢ Start, Visual Studio 2019 and select **Create a new project**.
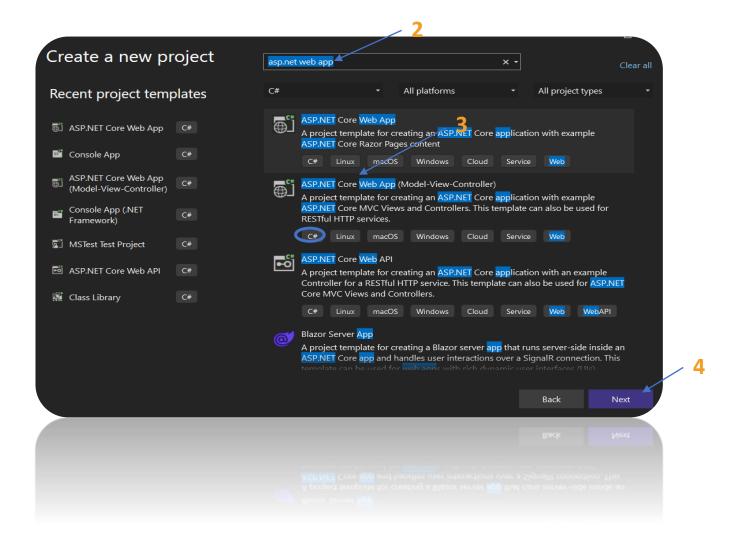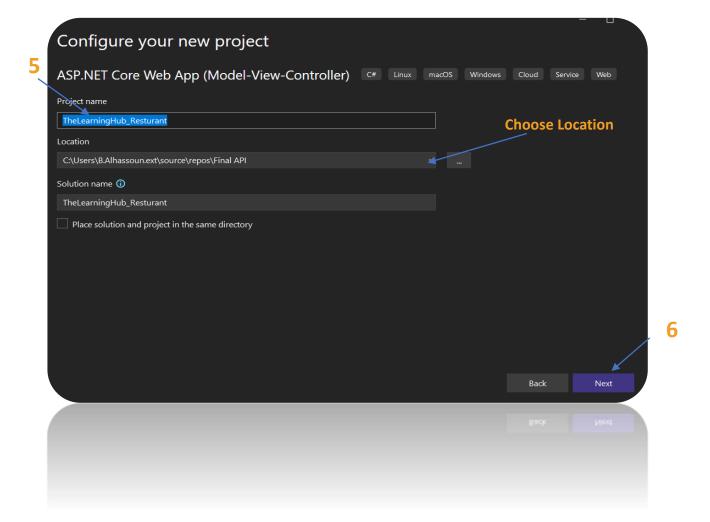
➢ choose **Asp .net core web app(Model-view-controller)** C#.
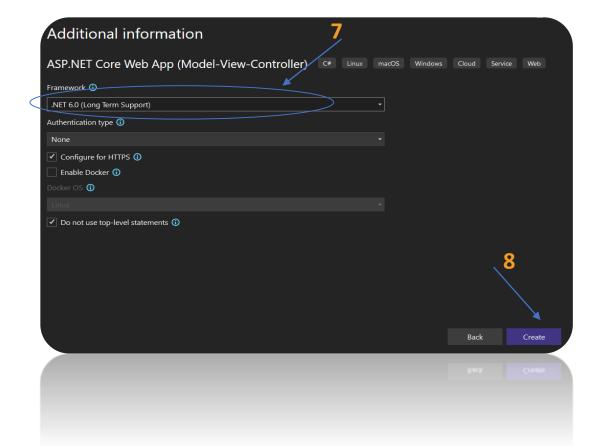
➤ Set Project Name {**Restaurant_2023**}.

**5**

Configure your new project

ASP.NET Core Web App (Model-View-Controller)   C#   Linux   macOS   Windows   Cloud   Service   Web

Project name

TheLearningHub_Resturant

**Choose Location**

Location

C:\Users\B.Alhassoun.ext\source\repos\Final API                    ...

Solution name ⓘ

TheLearningHub_Resturant

☐ Place solution and project in the same directory
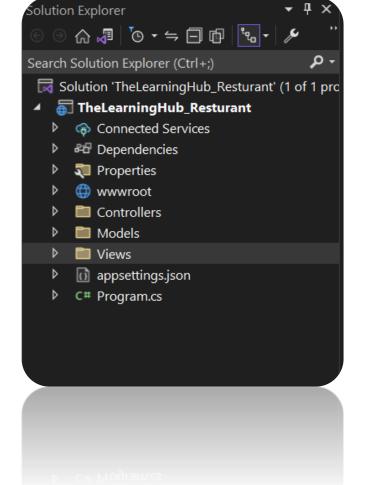
**6**

Back          Next

➢ Choose Target Framework 6.0

- **Note:** Check the box: Do not use top-level statements

- The project is created.

# ASP.NET Core Project Structure

- **wwwroot Folder:** is considered the web root folder.
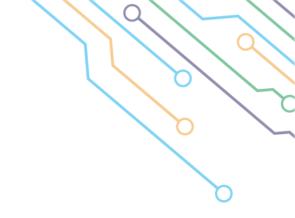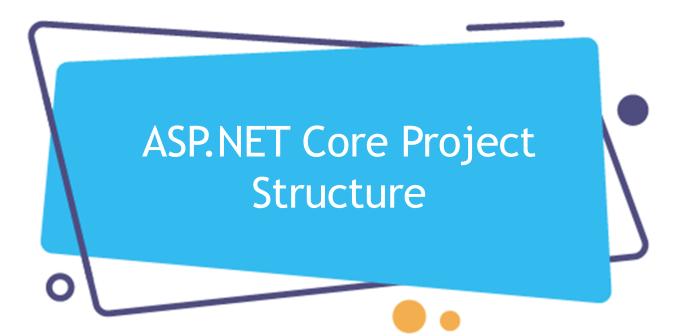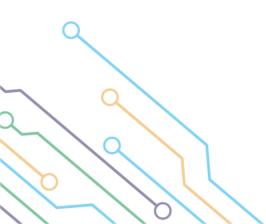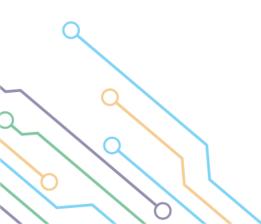
- In the ASP.NET Core project, the **wwwroot folder** using to stored Static files, all design files (css,images,js,bootstrap) store in wwwroot .

- **Program.cs:** The Program class is responsible for configuring the web host, setting up dependency injection, and registering any middleware that the application may need.

# Packages Installation

**<u>How to Install packages:</u>**

Tools => NuGet Package Manager => Manage NuGet Packages
for Solution:

ODP.NetCore  V(2.0.12)
Microsoft.EntityFrameworkCore V(7.0.5).
Microsoft.EntityFrameworkCore.SqlServer V(7.0.5)
Oracle.EntityFrameworkCore  V(7.21.9).
Microsoft.EntityFrameworkCore.Tools V(7.0.5).
Microsoft.VisualStudio.Web.CodeGeneration.Design (6.0.9)

**<u>Another way to install packages:</u>**

Tools => NuGet Package Manager => Package Manager Console

Install-Package ODP.NetCore –Version 2.0.12
Install-Package  Microsoft.EntityFrameworkCore –Version 7.0.5
Install-Package  Oracle.EntityFrameworkCore –Version 7.21.9
Install-Package  Microsoft.EntityFrameworkCore.SqlServer 7.0.5
Install-Package  Microsoft.EntityFrameworkCore.Tools  –Version 7.0.5
Install-Package Microsoft.VisualStudio.Web.CodeGeneration.Design –Version 6.0.9

**Packages:**

➢ **ODP.NetCore**

ODP.NET Core is an ADO.NET driver that provides fast data access from Microsoft .NET Core clients to Oracle databases. It runs on both Windows and Linux.

➢ **Microsoft.EntityFrameworkCore**

Entity Framework Core is a modern object-database mapper for .NET. It supports LINQ queries, change tracking, updates, and schema migrations. EF Core works with SQL Server, Azure SQL Database, SQLite, Azure Cosmos DB, MySQL, PostgreSQL, and other databases through a provider plugin API.

➢ **Oracle.EntityFrameworkCore**

Entity Framework Core is a cross-platform Microsoft object-relational mapper that enables .NET developers to work with relational databases using .NET objects.

➢ **Microsoft.EntityFrameworkCore.Tools**

Entity Framework Core Tools enables these commonly used commands:
- Add-Migration
- Bundle-Migration
- Drop-Database
- Get-DbContext
- Get-Migration
- Optimize-DbContext
- Remove-Migration
- Scaffold-DbContext
- Script-Migration
- Update-Database

## Exercise

Create a class diagram for the **Restaurant** database.