

## Models in ASP.NET Core MVC

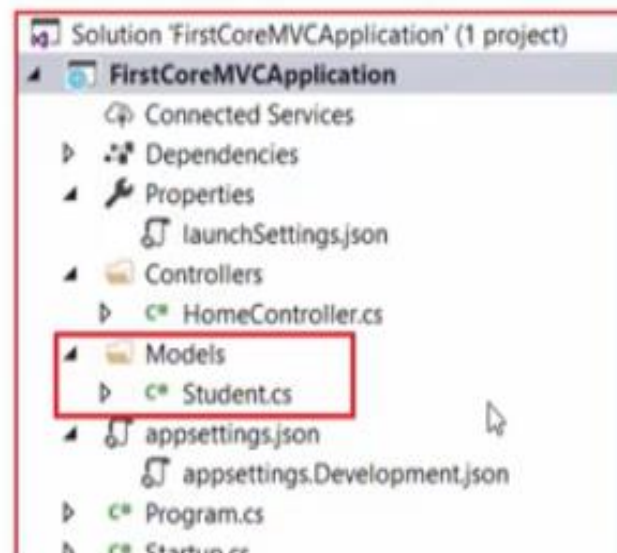
-

# What is Model?

- ▶ Model in ASP.NET Core MVC contains a set of classes which are used to represent the domain data as well as it also contains logic to manage the domain data.
- ▶ So in the simple word we can say that the model in MVC is used to manage the data i.e. the state of the application in memory.
- ▶ It is not mandatory, but it is a good programming practice to store all model classes within the Models folder.

# Adding Models folder:

- ▶ Right-click on your project, then select **add => new folder** option which will add a new folder. Then rename the folder name as **Models**. Here we want to create a model for displaying the student detail. So, create a class file with the name **Student.cs** within the Models folder. Once you create the Student model then the folder structure of your application should look as shown below.





Visual Studio interface showing the development of an ASP.NET Core project.

**Menu Bar:** File, Edit, View, Project, Build, Debug, Test, Analyze, Tools, Extensions, Window, Help. Search (Ctrl+Q).

**Toolbar:** Includes icons for file operations, debugging, and running. Status: Debug, Any CPU, IIS Express.

**Solution Explorer:** Displays the project structure for 'AspDotnetCore\_Project1' (1 of 1 project). The structure includes:

- AspDotnetCore\_Project1
  - Connected Services
  - Dependencies
  - Properties
  - wwwroot
  - Controllers
    - C# HomeController.cs
  - Models
    - C# Employee.cs (selected)
    - C# Program.cs
    - C# Startup.cs

**Code Editor:** Shows the code for `Employee.cs` in the `AspDotnetCore_Project1.Models` namespace. The code defines the `Employee` class with properties `Id`, `Name`, `City`, and `Gender`.

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Threading.Tasks;
5
6 namespace AspDotnetCore_Project1.Models
7 {
8     public class Employee
9     {
10         public int Id { get; set; }
11         public string Name { get; set; }
12         public string City { get; set; }
13         public string Gender { get; set; }
14     }
15 }
16
```

**Status Bar:** 120% zoom, No issues found, Ln: 13, Ch: 41, SPC, CRLF.

**Taskbar:** Windows taskbar with search bar and various application icons. System clock shows 10:21 on 18-01-2021.

EmployeeController.csEmployeeRepository.csIEmployeeRepository.csEmployee.csObject Browser

AspDotnetCore\_Project1AspDotnetCore\_Project1.Models.IEmployeeGetEmployee(int empid)

```
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Threading.Tasks;
5
6 namespace AspDotnetCore_Project1.Models
7 {
8     public interface IEmployeeRepository
9     {
10         Employee GetEmployee(int empid);
11     }
12 }
13
```

120 % No issues found Ln: 13 Ch: 1 SPC CRLF

Output

t1, Configuration: Debug Any CPU  
-----  
1>AspDotnetCore\_Project1 -> C:\Users\Lenovo\Desktop\AspNetCore\_Examples\AspNetCore\_Project1\bin\Debug\netcoreapp2.1\AspNetCore\_Project1.dll  
-----  
Build: 1 succeeded, 0 failed, 0 up-to-date, 0 skipped  
-----

Solution Explorer

Search Solution Explorer (Ctrl+;)  
Solution 'AspDotnetCore\_Project1' (1 of 1 projects)  
AspDotnetCore\_Project1  
    Connected Services  
    Dependencies  
    Properties  
    wwwroot  
    Controllers  
        C# EmployeeController.cs  
        C# HomeController.cs  
    Models  
        C# Employee.cs  
        C# EmployeeRepository.cs  
        C# IEmployeeRepository.cs  
    C# Program.cs  
    C# Startup.cs

FileEditViewProjectBuildDebugTestAnalyzeToolsExtensionsWindowHelp

Search (Ctrl+Q)

AspDotnet...e\_Project1Sign in

Live Share

EmployeeController.csEmployeeRepository.csIEmployeeRepository.csEmployee.csObject Browser

AspDotnetCore\_Project1AspDotnetCore\_Project1.Models.Employee\_emList

```
4  using System.Threading.Tasks;
5
6  namespace AspDotnetCore_Project1.Models
7  {
8      public class EmployeeRepository : IEmployeeRepository
9      {
10         private List<Employee> _emList ;
11         public EmployeeRepository()
12         {
13             _emList = new List<Employee> {
14                 new Employee{Id=1,Name="Raj",City="Delhi",Gender="Male"},
15                 new Employee{Id=2,Name="Rajee",City="Agra",Gender="Female"},
16                 new Employee{Id=3,Name="Raj kuamr",City="Noida",Gender="Male"}
17             };
18         }
19
20
21
22         public Employee GetEmployee(int empid)
23         {
24             return this._emList.SingleOrDefault(x => x.Id == empid);
25         }
26     }
27 }
28
```

t1, Configuration: Debug Any CPU

1>AspDotnetCore\_Project1 -> C:\Users\Lenovo\Desktop\Asp.net\_Core\Examples\AspDotnetCore\_Project1\bin\Debug\netcoreapp2.1\AspDotnetCore\_Project1.dll

Build: 1 succeeded, 0 failed, 0 up-to-date, 0 skipped

Search Solution Explorer (Ctrl+;)

Solution 'AspDotnetCore\_Project1' (1 of 1 projects)

AspDotnetCore\_Project1

Connected Services

Dependencies

Properties

wwwroot

Controllers

EmployeeController.cs

HomeController.cs

Models

Employee.cs

EmployeeRepository.cs

IEmployeeRepository.cs

Program.cs

Startup.cs

120 %

No issues found

Ln: 8 Ch: 58 SPC CRLF

Ready

Type here to search

18-01-2021 10:32

Visual Studio interface showing the development of an ASP.NET Core project.

**File Explorer:** File, Edit, View, Project, Build, Debug, Test, Analyze, Tools, Extensions, Window, Help. Search (Ctrl+Q). AspDotnet...e\_Project1. Sign in. Live Share.

**Toolbox:** Server Explorer, Toolbox.

**Code Editor:** EmployeeController.cs, EmployeeRepository.cs, IEmployeeRepository.cs, Employee.cs, Object Browser. AspDotnetCore\_Project1. AspDotnetCore\_Project1.Controllers.Emplo. GerEmployeeId(int id).

```
1 using AspDotnetCore_Project1.Models;
2 using Microsoft.AspNetCore.Mvc;
3 using System;
4 using System.Collections.Generic;
5 using System.Linq;
6 using System.Threading.Tasks;
7
8 namespace AspDotnetCore_Project1.Controllers
9 {
10     public class EmployeeController : Controller
11     {
12         public JsonResult GerEmployeeId(int id)
13         {
14             IEmployeeRepository ob = new EmployeeRepository();
15             Employee emp= ob.GetEmployee(id);
16             return Json(emp);
17         }
18     }
19 }
20
```

**Output:** t1, Configuration: Debug, Any CPU. 1>AspDotnetCore\_Project1 -> C:\Users\Lenovo\Desktop\Asp.net\_Core\_Examples\AspDotnetCore\_Project1\bin\Debug\netcoreapp2.1\AspDotnetCore\_Project1.dll. Build: 1 succeeded, 0 failed, 0 up-to-date, 0 skipped.

**Solution Explorer:** Search Solution Explorer (Ctrl+;). Solution 'AspDotnetCore\_Project1' (1 of 1 project). AspDotnetCore\_Project1. Connected Services, Dependencies, Properties, wwwroot, Controllers, C# EmployeeController.cs, C# HomeController.cs, Models, C# Employee.cs, C# EmployeeRepository.cs, IEmployeeRepository.cs, C# Program.cs, C# Startup.cs.

**Status Bar:** 120 %, No issues found, Ln: 17 Ch: 10 SPC CRLF.

**Taskbar:** Ready, Type here to search, Add to Source Control, 10:33 18-01-2021.

Visual Studio interface showing the development of an ASP.NET Core project. The main editor displays the `Startup.cs` file, which includes the `ConfigureServices` and `Configure` methods. The `ConfigureServices` method adds MVC services and JSON formatters. The `Configure` method sets up the HTTP request pipeline, including development exception pages and the default MVC route.

```
// For more information on how to configure your application, visit https://go.microsoft.com/fwlink/?Li
public void ConfigureServices(IServiceCollection services)
{
    //create mvc project add this services first
    // services.AddMvc();
    //Create Core MVC project add this service
    //services.AddMvcCore();//Error
    var builder = services.AddMvcCore();
    builder.AddJsonFormatters();
}

// This method gets called by the runtime. Use this method to configure the HTTP request pipeline.
public void Configure(IApplicationBuilder app, IHostingEnvironment env)
{
    if (env.IsDevelopment())
    {
        app.UseDeveloperExceptionPage();
    }
    //Add Default root to configure the Index page from Home Controller
    app.UseMvcWithDefaultRoute();
    //app.Run(async (context) =>
    //{
    //    await context.Response.WriteAsync("Hello World!");
    //});
```

The right sidebar shows the Output window with the following content:

```
roject1,
Configuration:
Debug Any CPU
-----
1>AspDotnetCore_P
roject1 -> C:
\Users\Lenovo
\Desktop
\Asp.net_Core
\Examples
\AspDotnetCore_
Project1\bin
\Debug
\netcoreapp2.1
\AspDotnetCore_
Project1.dll
===== Build:
1 succeeded, 0
failed, 0 up-
to-date, 0
skipped
=====
```

The bottom status bar indicates "Ready" and "No issues found". The taskbar at the bottom shows the Windows search bar and various application icons.



19- Models in ASP.NET Core x | localhost:54284 x | Internal Server Error x | localhost:54284 x | localhost:54284/Employee/ x + -

localhost:54284/Employee/GetEmployeeId/1

Apps https://mail.google... Google localhost:1031/Web... Linux File Permissio... 011 39885050 Redirection in Linux... Gmail YouTube Maps New Tab »

`{"id":1,"name":"Raj","city":"Delhi","gender":"Male"}`

Windows taskbar with search bar and icons for Chrome, Mail, Edge, File Explorer, Firefox, Task View, Paint, Word, PowerPoint, and system tray showing time (10:33) and date (18-01-2021).

Thanks