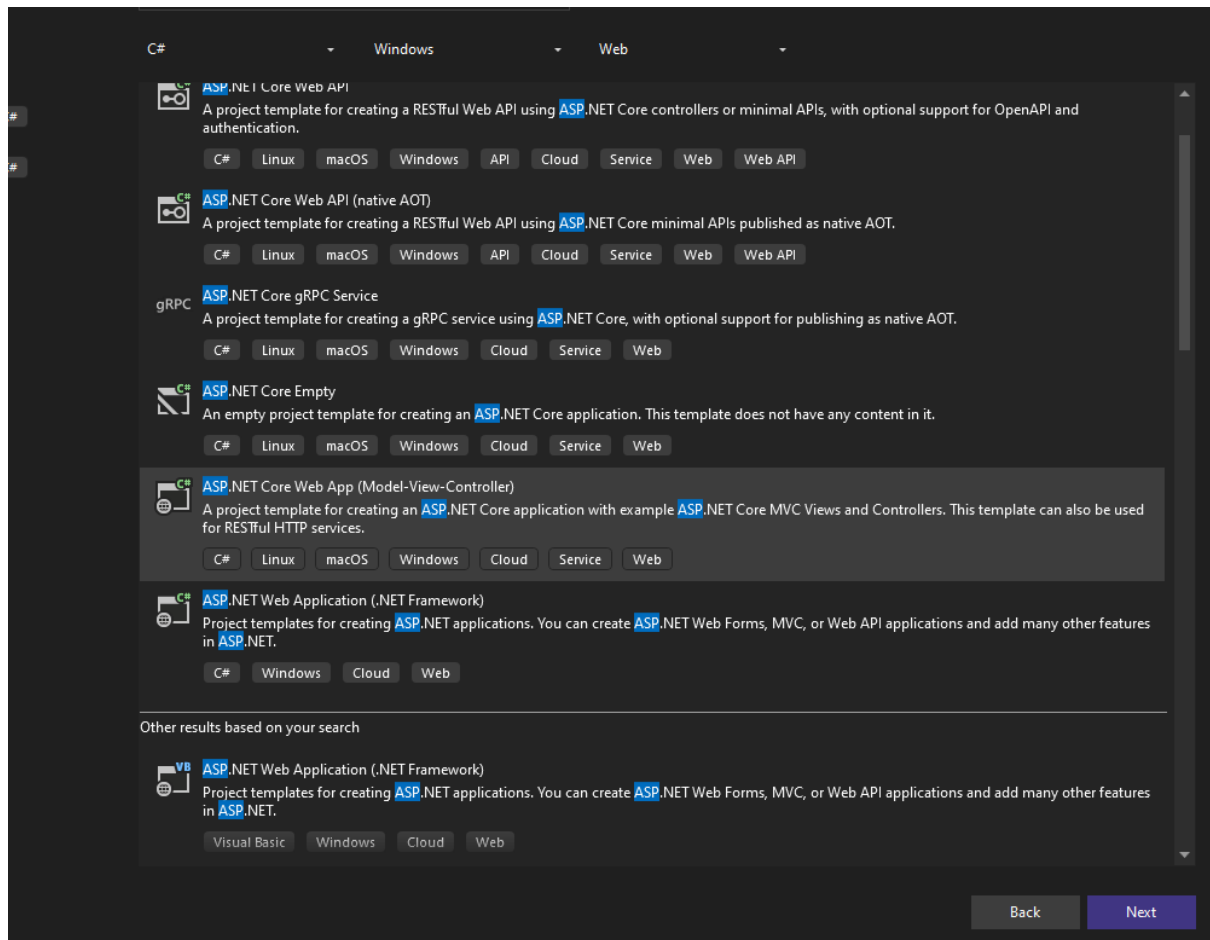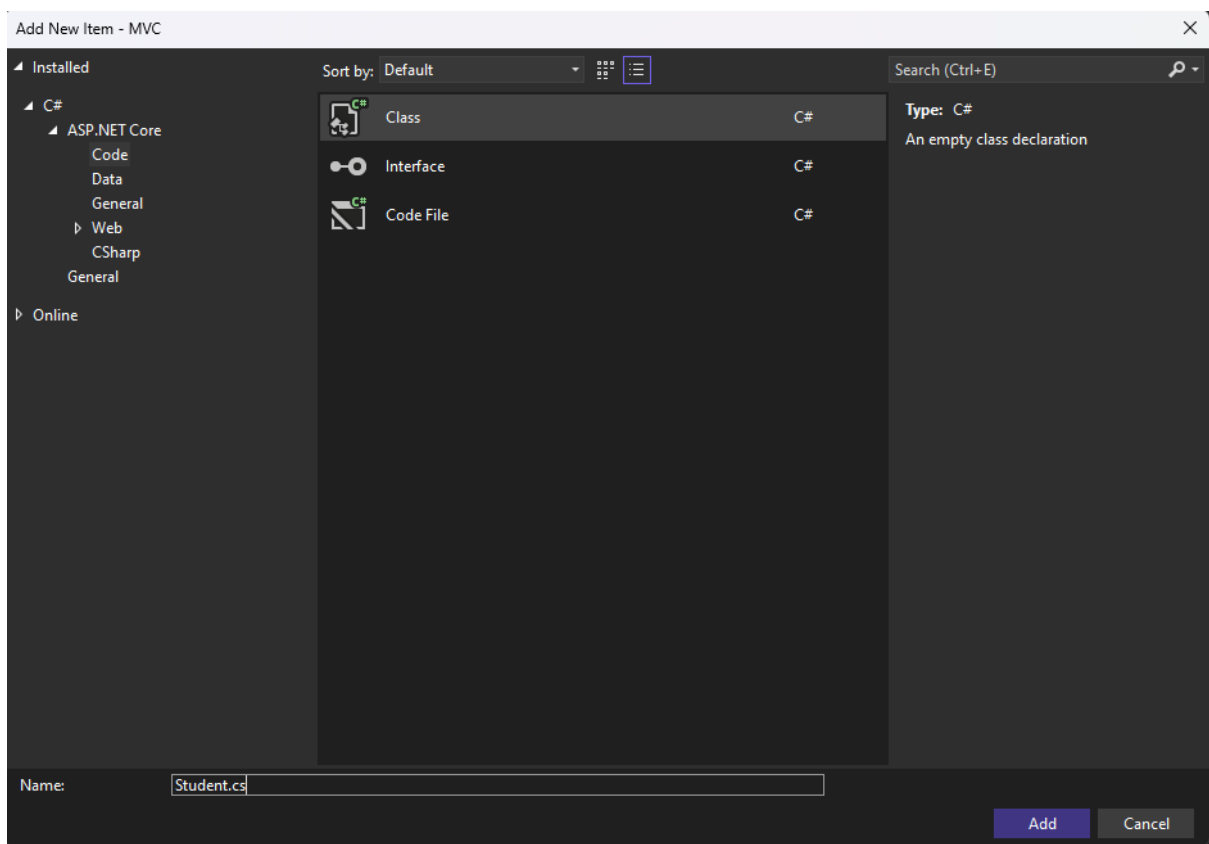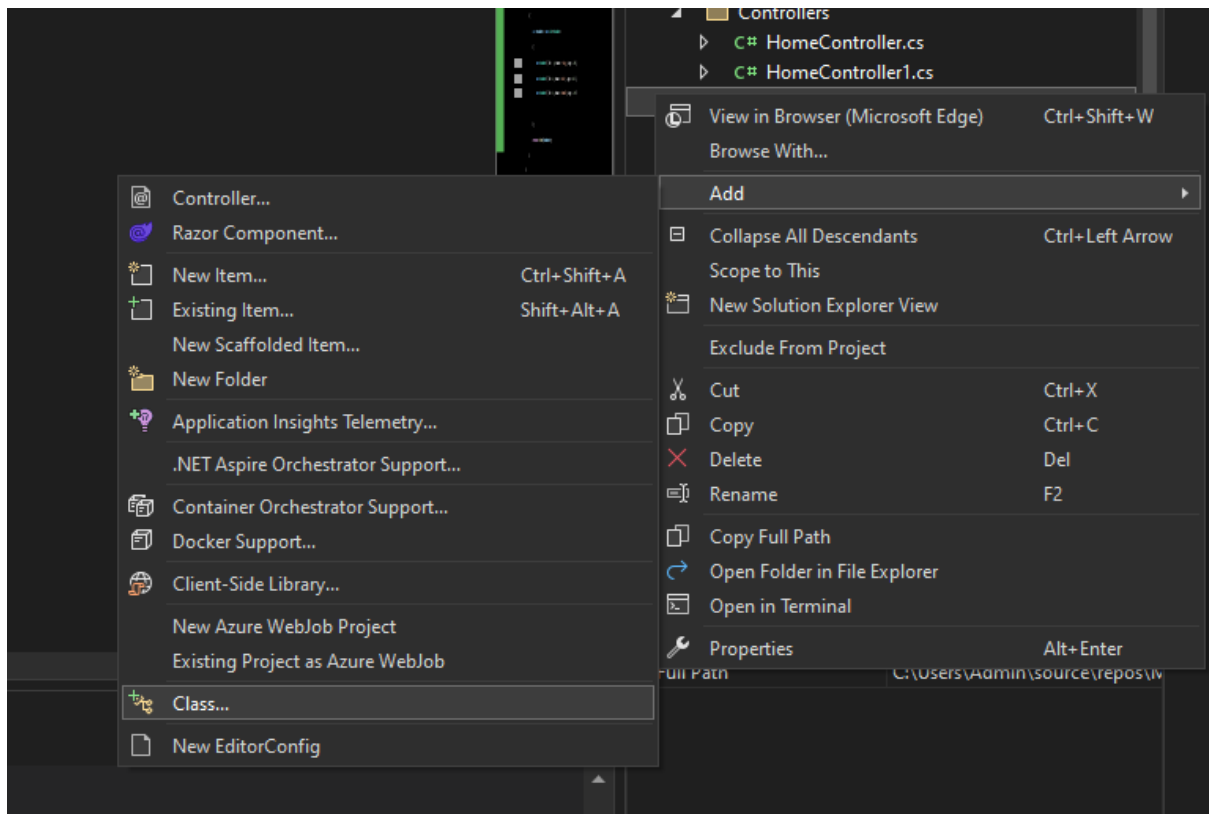**Name: Rajvardhan Ganpatrao Patil**
Roll No: 40   Div: A

1. MVC Application With Model view controller

Name: Rajvardhan Ganpatrao Patil
Roll No: 40   Div: A

Right click on model
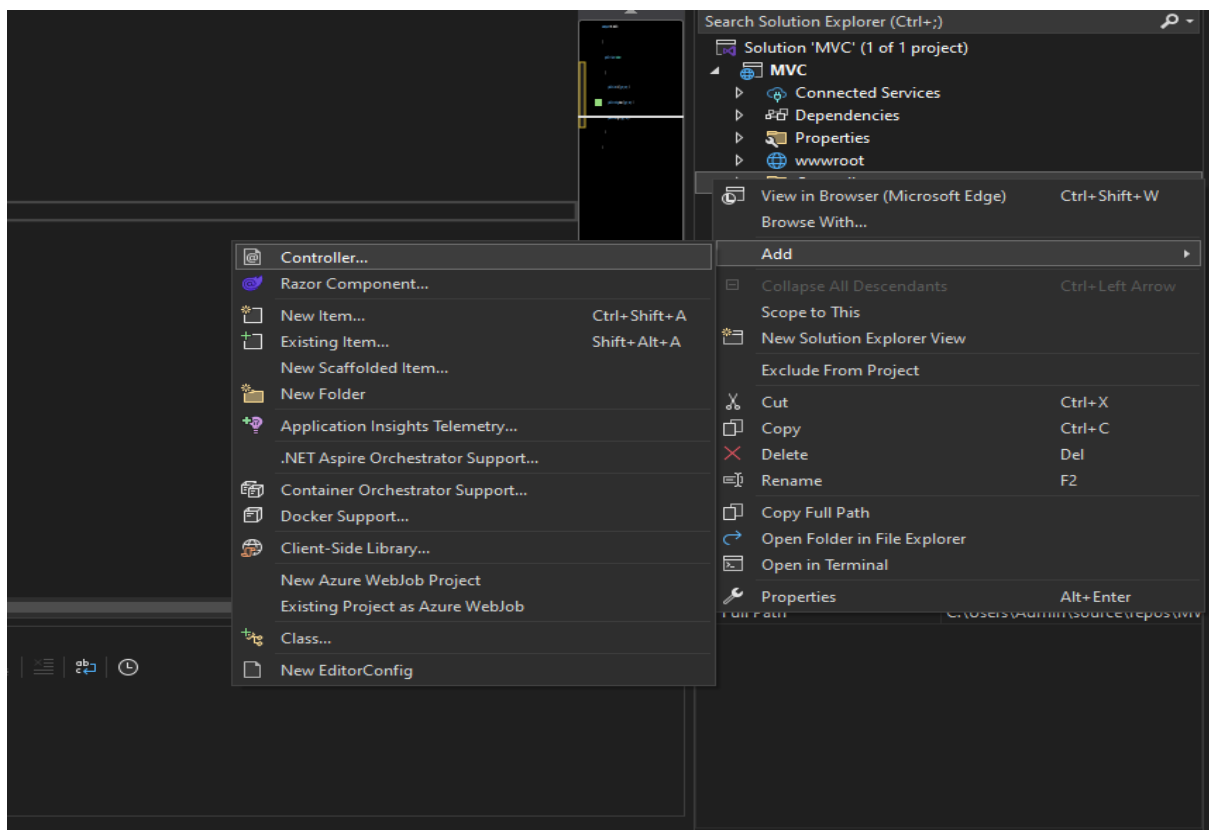
Name: Rajvardhan Ganpatrao Patil
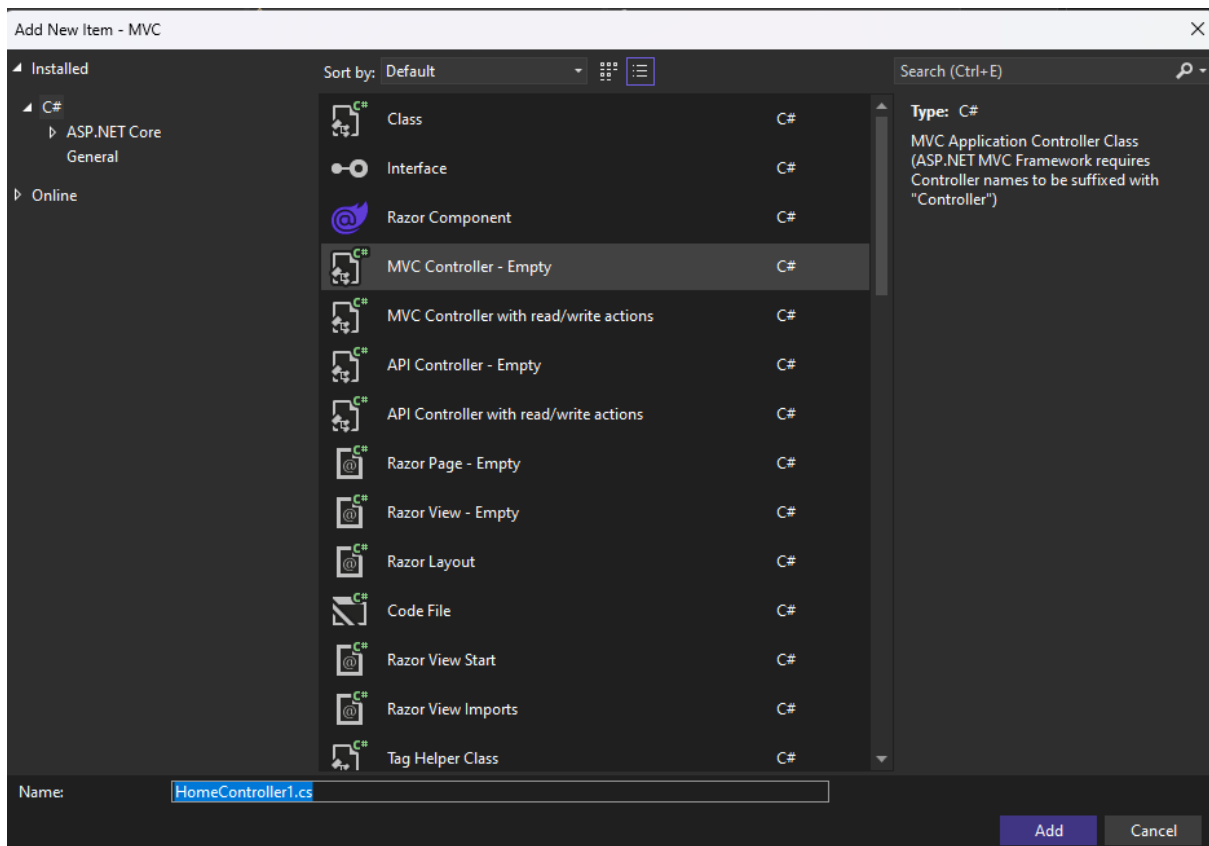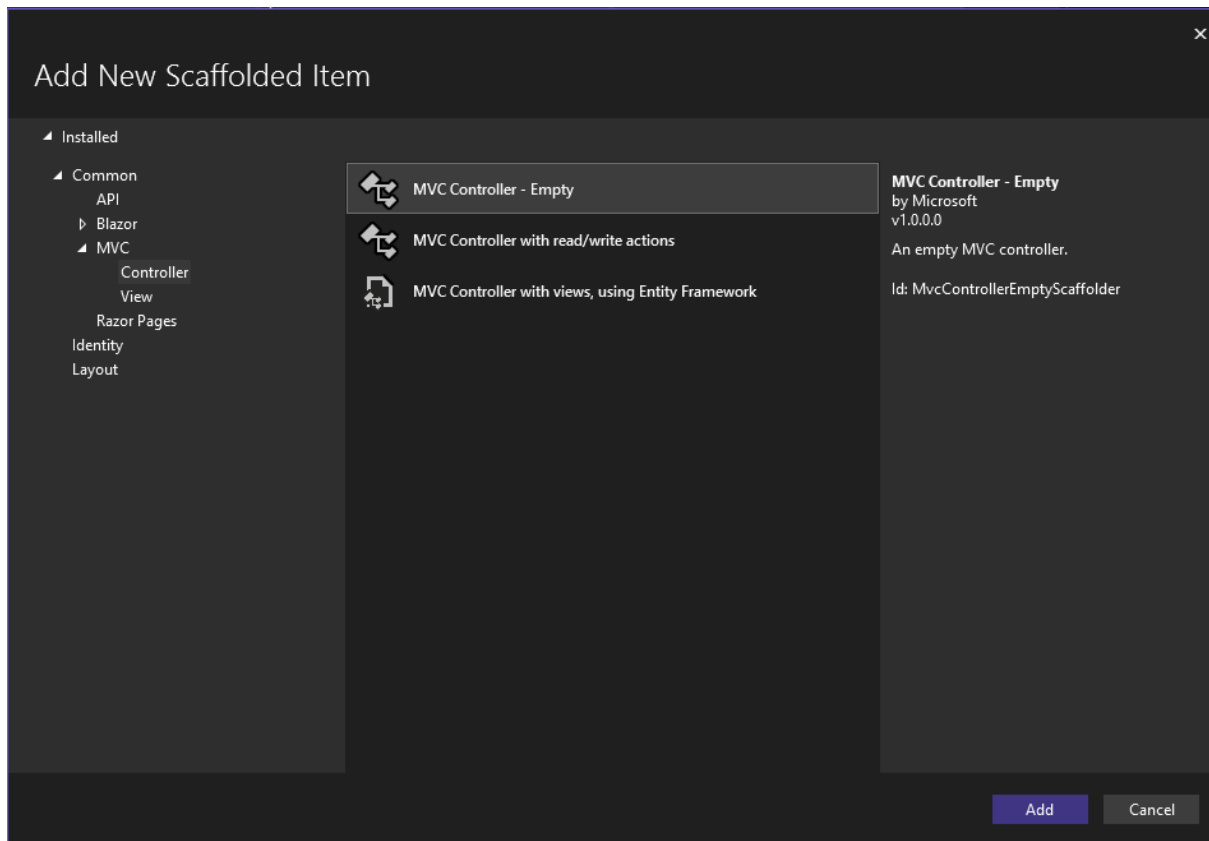**Roll No: 40   Div: A**

Student.cs

```csharp
namespace MVC.Models
{
    public class Student
    {
        public int Id { get; set; }
        public string Name { get; set; }
        public int Age { get; set; }
    }
}
```

Right click on Controller

Name: Rajvardhan Ganpatrao Patil
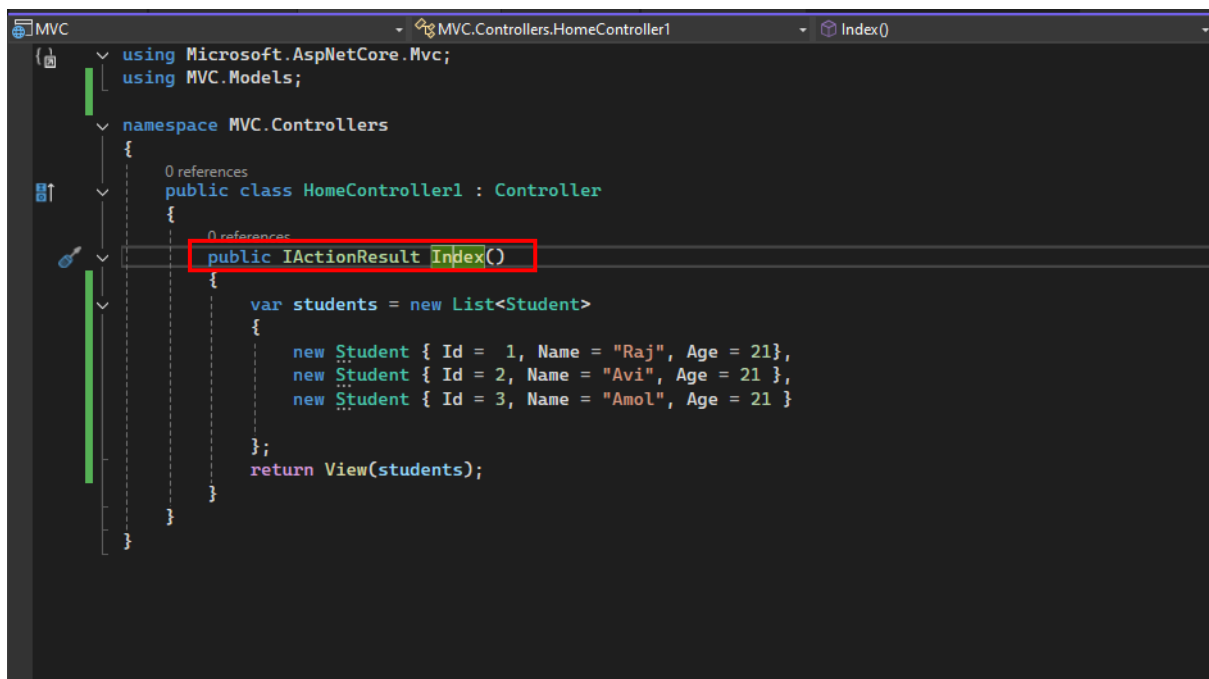**Roll No: 40   Div: A**

Add New Scaffolded Item

▲ Installed

▲ Common
    API
    ▷ Blazor
    ▲ MVC
        Controller
        View
    Razor Pages
  Identity
  Layout

MVC Controller - Empty

MVC Controller with read/write actions

MVC Controller with views, using Entity Framework

MVC Controller - Empty
by Microsoft
v1.0.0.0

An empty MVC controller.

Id: MvcControllerEmptyScaffolder

Add   Cancel

Add New Item - MVC

▲ Installed

▲ C#
    ▷ ASP.NET Core
    General
▷ Online

Sort by: Default

Search (Ctrl+E)

Class		C#
Interface		C#
Razor Component		C#
MVC Controller - Empty		C#
MVC Controller with read/write actions		C#
API Controller - Empty		C#
API Controller with read/write actions		C#
Razor Page - Empty		C#
Razor View - Empty		C#
Razor Layout		C#
Code File		C#
Razor View Start		C#
Razor View Imports		C#
Tag Helper Class		C#

Type: C#

MVC Application Controller Class
(ASP.NET MVC Framework requires
Controller names to be suffixed with
"Controller")

Name: HomeController1.cs

Add   Cancel

Name: Rajvardhan Ganpatrao Patil
Roll No: 40   Div: A
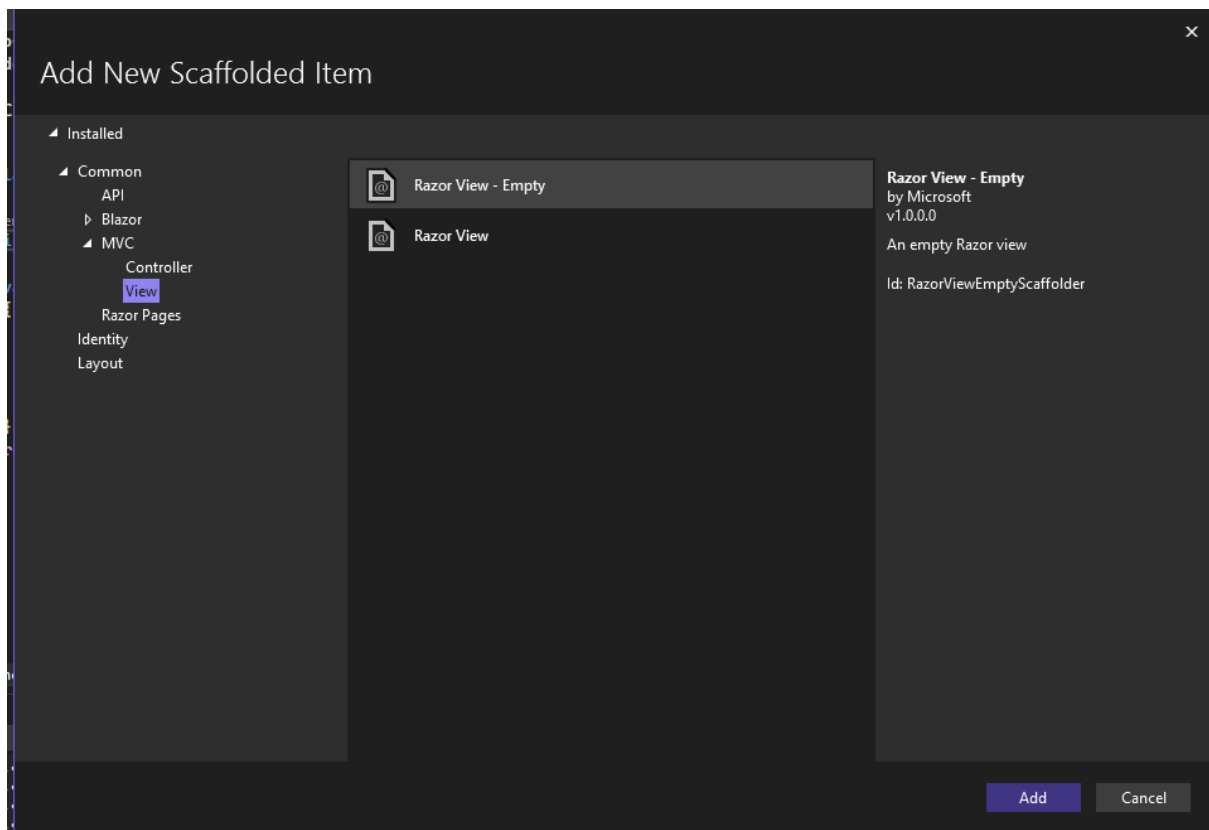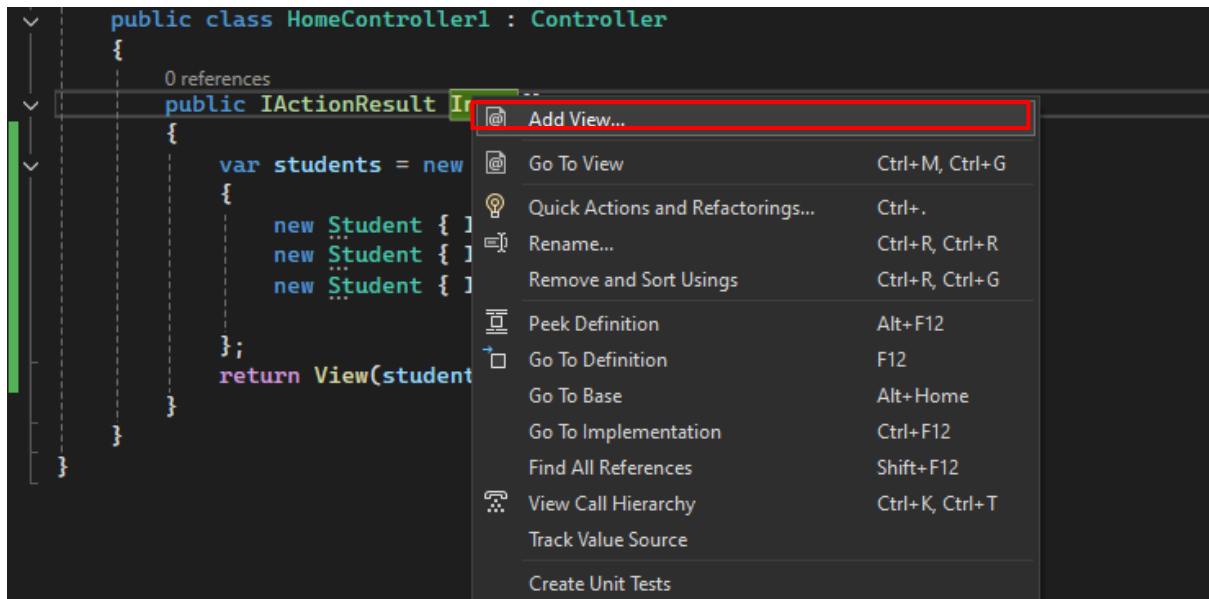
HomeController1.cs

```csharp
using Microsoft.AspNetCore.Mvc;
using MVC.Models;

namespace MVC.Controllers
{
    public class HomeController1 : Controller
    {
        public IActionResult Index()
        {
            var students = new List<Student>
            {
                new Student { Id =  1, Name = "Raj", Age = 21},
                new Student { Id = 2, Name = "Avi", Age = 21 },
                new Student { Id = 3, Name = "Amol", Age = 21 }

            };
            return View(students);
        }
    }
}
```
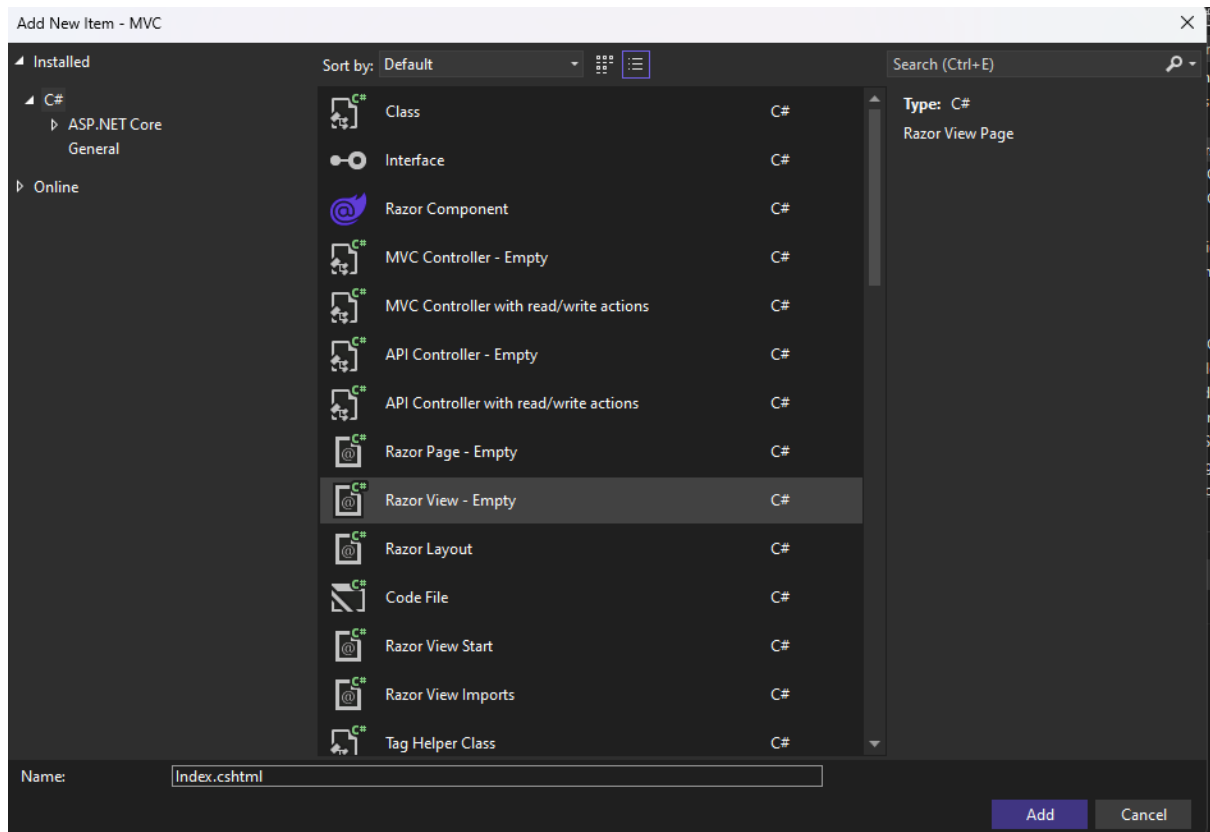
Right click on Index

Name: Rajvardhan Ganpatrao Patil
Roll No: 40   Div: A

```
public class HomeController1 : Controller
{
    0 references
    public IActionResult I⌐
    {
        var students = new
        {
            new Student { 1
            new Student { 1
            new Student { 1

        };
        return View(student
    }
}
}
```

	Add View...	
@	Go To View	Ctrl+M, Ctrl+G
💡	Quick Actions and Refactorings...	Ctrl+.
	Rename...	Ctrl+R, Ctrl+R
	Remove and Sort Usings	Ctrl+R, Ctrl+G
	Peek Definition	Alt+F12
	Go To Definition	F12
	Go To Base	Alt+Home
	Go To Implementation	Ctrl+F12
	Find All References	Shift+F12
	View Call Hierarchy	Ctrl+K, Ctrl+T
	Track Value Source	
	Create Unit Tests	

Add New Scaffolded Item

▲ Installed

▲ Common
  API
  ▷ Blazor
  ▲ MVC
    Controller
    View
  Razor Pages
Identity
Layout

@ Razor View - Empty

@ Razor View

Razor View - Empty
by Microsoft
v1.0.0.0

An empty Razor view

Id: RazorViewEmptyScaffolder

Add   Cancel

Index.cshtml

```
@*
    For more information on enabling MVC for empty projects, visit
https://go.microsoft.com/fwlink/?LinkID=397860
*@
@{
    @model List<MVC.Models.Student>
    <h2>Student List</h2>
    <table>
        <tr>
            <th>ID</th>
            <th>Name</th>
            <th>Age</th>
        </tr>
        @foreach(var student in Model)
        {
            <tr>
                <td>@student.Id</td>
                <td>@student.Name</td>
                <td>@student.Age</td>
            </tr>
        }
    </table>
}
```

Change controller path in program.cs

Name: Rajvardhan Ganpatrao Patil
Roll No: 40   Div: A

Write our controller name

```
app.MapControllerRoute(
    name: "default",
    pattern: "{controller=HomeController1}/{action=Index}/{id?}");

app.Run();
```

Output:

MVC   Home   Privacy

Student List

ID	Name	Age
1	Raj	21
2	Avi	21
3	Amol	21

2. MVC Application for customer data entry using data annotations for validation and display appropriate error messages.

Models\Customer.cs

```csharp
using System.ComponentModel.DataAnnotations;

namespace unit6.Models
{
    public class Customer
    {
        public int Id { get; set; }

        [Required(ErrorMessage = "Name is required")]
        [StringLength(50, ErrorMessage = "Name cannot exceed 50 charactors")]
        public required string Name { get; set; }

        [Required(ErrorMessage = "Email is required")]
        [EmailAddress(ErrorMessage = "Invalid email format")]
        public required string Email { get; set; }

        [Required(ErrorMessage = "Phone number is required")]
        [RegularExpression(@"^\d{10}$", ErrorMessage = "Phone number must be 10 digits")]
        public required string Phone { get; set; }

        [Required(ErrorMessage = "Address is required")]
        [StringLength(100, ErrorMessage = "Address cannot exceed 100 charactors")]
        public required string Address { get; set; }

        [Required(ErrorMessage = "Pincode is required")]
        [RegularExpression(@"^\d{6}$", ErrorMessage = "Pincode must be 6 digit")]
        public required string PinCode { get; set; }
    }
}
```

Controllers\CustomerController.cs

```csharp
using Microsoft.AspNetCore.Mvc;
using unit6.Models;

namespace unit6.Controllers
{
    public class CustomerController : Controller
    {
        private static readonly List<Customer> customer = new List<Customer>();
        public IActionResult Index()
        {
            return View(customer);
        }
        public IActionResult Create()
        {
            return View();
        }
        [HttpPost]
        public IActionResult Create(Customer cust)
        {
            if (ModelState.IsValid)
```

```
                {
                    cust.Id = customer.Count + 1;
                    customer.Add(cust);
                    return RedirectToAction("Index");
                }
                return View(customer);
            }
        }
}
```

Customer\Index.cshtml

```
@model List<Customer>

<h2>Customer List</h2>
<table border="1">
    <tr>
        <th>ID</th>
        <th>Name</th>
        <th>Email</th>
        <th>Phone</th>
        <th>Address</th>
    </tr>
    @foreach (var customer in Model)
    {
        <tr>
            <td>@customer.Id</td>
            <td>@customer.Name</td>
            <td>@customer.Email</td>
            <td>@customer.Phone</td>
            <td>@customer.Address</td>
        </tr>
    }
</table>

<a href="/Customer/Create">Add New Customer</a>
```

Customer\Create.cshtml

```
@model unit6.Models.Customer

@{
    ViewData["Title"] = "Create";
}

<h1>Create</h1>

<h4>Customer</h4>
<hr />
<div class="row">
    <div class="col-md-4">
        <form asp-action="Create">
            <div asp-validation-summary="ModelOnly" class="text-danger"></div>
            <div class="form-group">
                <label asp-for="Id" class="control-label"></label>
                <input asp-for="Id" class="form-control" />
                <span asp-validation-for="Id" class="text-danger"></span>
            </div>
            <div class="form-group">
                <label asp-for="Name" class="control-label"></label>
```
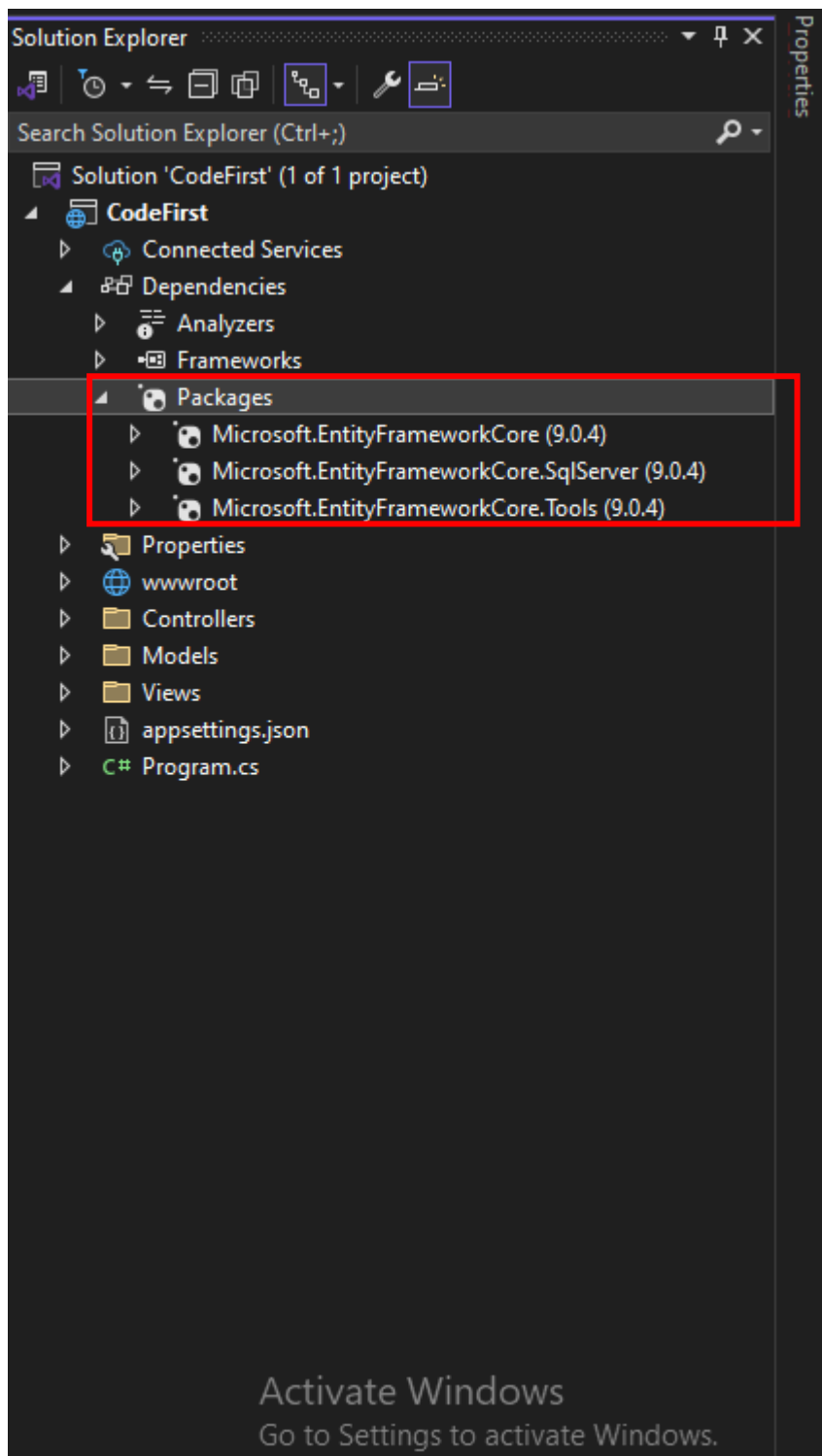
```html
                <input asp-for="Name" class="form-control" />
                <span asp-validation-for="Name" class="text-danger"></span>
            </div>
            <div class="form-group">
                <label asp-for="Email" class="control-label"></label>
                <input asp-for="Email" class="form-control" />
                <span asp-validation-for="Email" class="text-danger"></span>
            </div>
            <div class="form-group">
                <label asp-for="Phone" class="control-label"></label>
                <input asp-for="Phone" class="form-control" />
                <span asp-validation-for="Phone" class="text-danger"></span>
            </div>
            <div class="form-group">
                <label asp-for="Address" class="control-label"></label>
                <input asp-for="Address" class="form-control" />
                <span asp-validation-for="Address" class="text-danger"></span>
            </div>
            <div class="form-group">
                <label asp-for="PinCode" class="control-label"></label>
                <input asp-for="PinCode" class="form-control" />
                <span asp-validation-for="PinCode" class="text-danger"></span>
            </div>
            <div class="form-group">
                <input type="submit" value="Create" class="btn btn-primary" />
            </div>
        </form>
    </div>
</div>

<div>
    <a asp-action="Index">Back to List</a>
</div>

@section Scripts {
    @{await Html.RenderPartialAsync("_ValidationScriptsPartial");}

}
```

unit6\Program.cs

```csharp
var builder = WebApplication.CreateBuilder(args);

// Add services to the container.
builder.Services.AddControllersWithViews();

var app = builder.Build();

// Configure the HTTP request pipeline.
if (!app.Environment.IsDevelopment())
{
    app.UseExceptionHandler("/Home/Error");
    // The default HSTS value is 30 days. You may want to change this for
production scenarios, see https://aka.ms/aspnetcore-hsts.
    app.UseHsts();
}

app.UseHttpsRedirection();
app.UseStaticFiles();

app.UseRouting();

app.UseAuthorization();
```

```
app.MapControllerRoute(
    name: "default",
    pattern: "{controller=Customer}/{action=Index}/{id?}");

app.Run();
```

Output:

---

unit6   Home   Privacy

Create

Customer

Id

[⬍]

The Id field is required.

Name

[   ]

Name is required

Email

[fgdbdfvg   ]

Invalid email format

Phone

[fdf   ]

Phone number must be 10 digits

Address

[   ]

Address is required

PinCode

[   ]

[Create]

Back to List

unit6   Home   Privacy

Create

Customer

Id

```
1
```

Name

```
Rajvardhan
```

Email

```
rajpatil24@gmail.com
```

Phone

```
9876543211
```

Address

```
kolhapur
```

PinCode

```
543244
```

Create

Back to List

© 2025 - unit6 - Privacy

---

unit6   Home   Privacy

Customer List

ID	Name	Email	Phone	Address
1	Rajvardhan	rajpatil24@gmail.com	9876543211	kolhapur

Add New Customer

Name: Rajvardhan Ganpatrao Patil
Roll No: 40   Div: A

3. Implement CRUD operations using Entity Framework Code-First approach for emp data.

Create model class `AppDbContext`
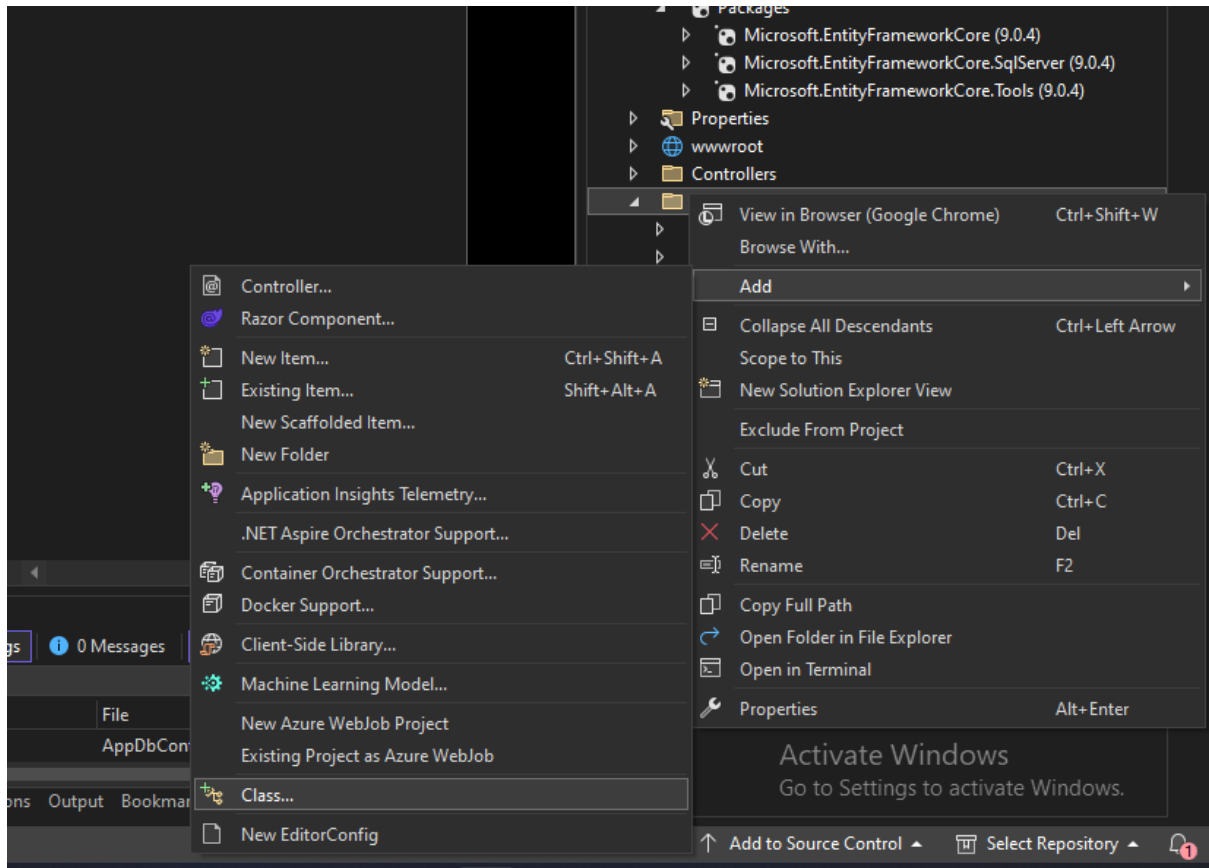

```csharp
using Microsoft.EntityFrameworkCore;
namespace CodeFirst.Models

{
    public class AppDbContext:DbContext
    {
        public AppDbContext(DbContextOptions<AppDbContext> options) :
base(options) { }
        public DbSet<Student> Students { get; set; }
    }
}
```

Name: Rajvardhan Ganpatrao Patil
Roll No: 40   Div: A

Appsettings.json

{

  "ConnectionStrings": {

    "DefaultConnection":
"Server=(localdb)\\MSSQLLocalDB;Database=StudentDb;Trusted_Connection=True;"

  }

}
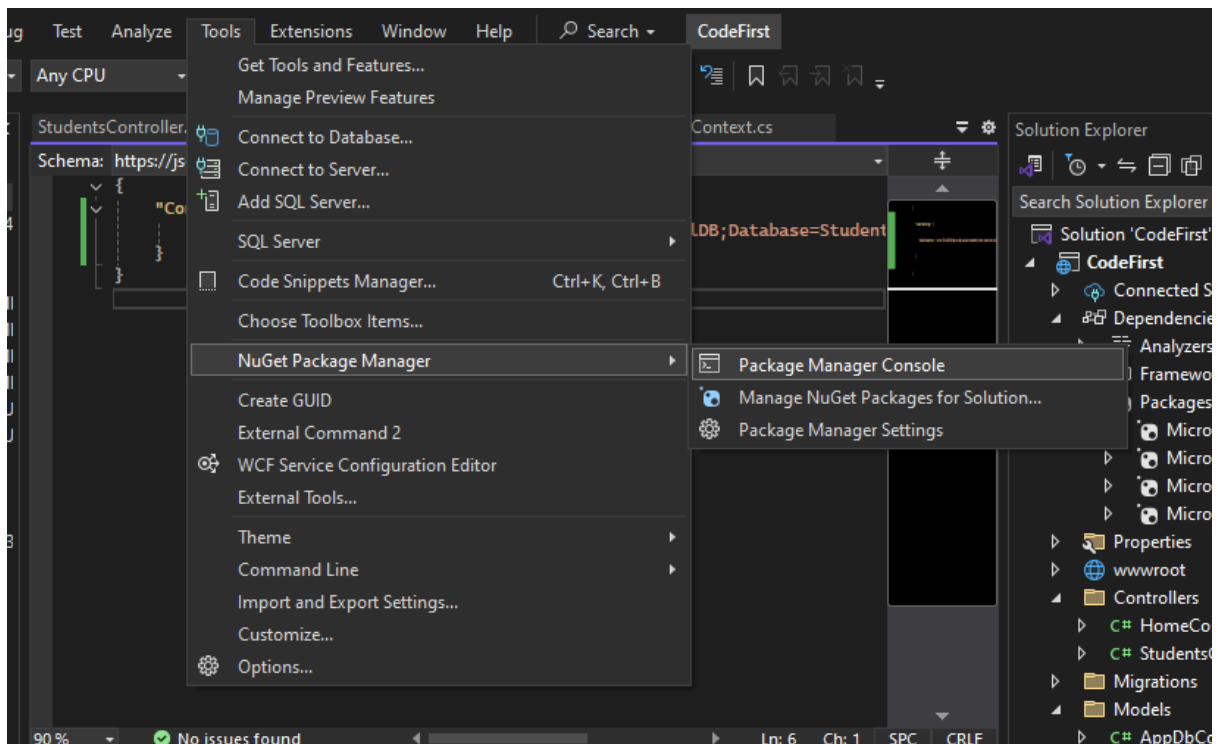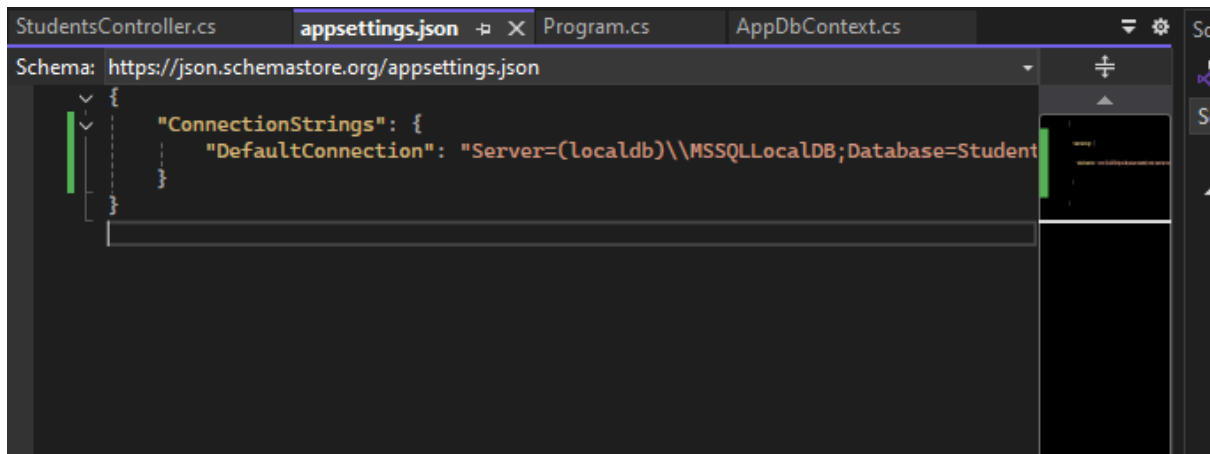
```
PM> Add-Migration InitialCreate
Build started...
Build succeeded.
To undo this action, use Remove-Migration.
PM> Update-Database
Build started...
Build succeeded.
Microsoft.EntityFrameworkCore.Database.Command[20101]
      Executed DbCommand (220ms) [Parameters=[], CommandType='Text', CommandTimeout='60']
      CREATE DATABASE [StudentDb];
Microsoft.EntityFrameworkCore.Database.Command[20101]
      Executed DbCommand (79ms) [Parameters=[], CommandType='Text', CommandTimeout='60']
      IF SERVERPROPERTY('EngineEdition') <> 5
      BEGIN
```

StudentDb is created
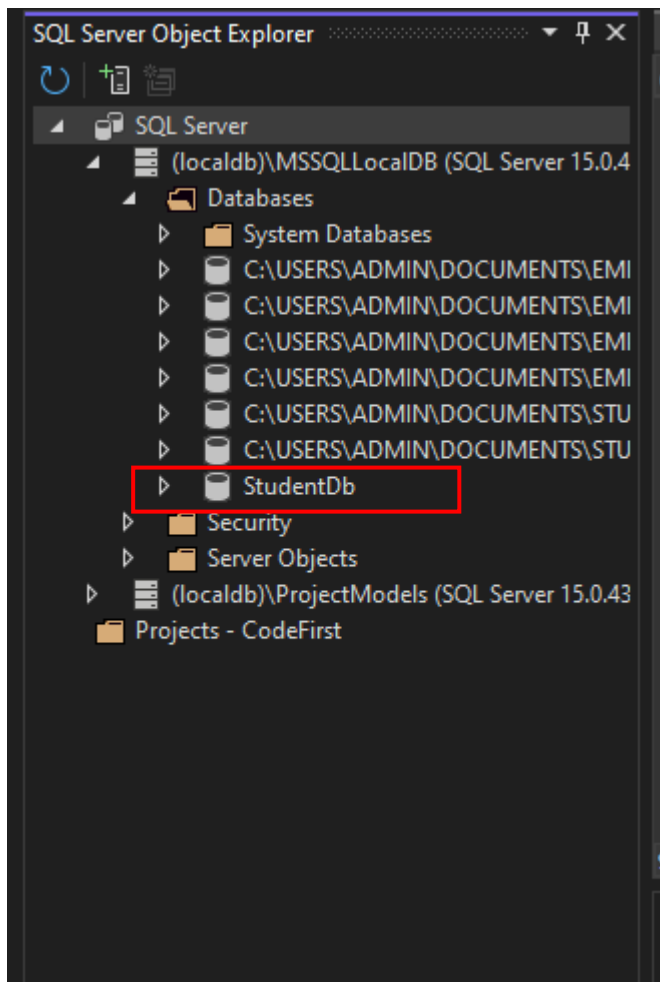
Name: Rajvardhan Ganpatrao Patil
Roll No: 40   Div: A

Create Controller
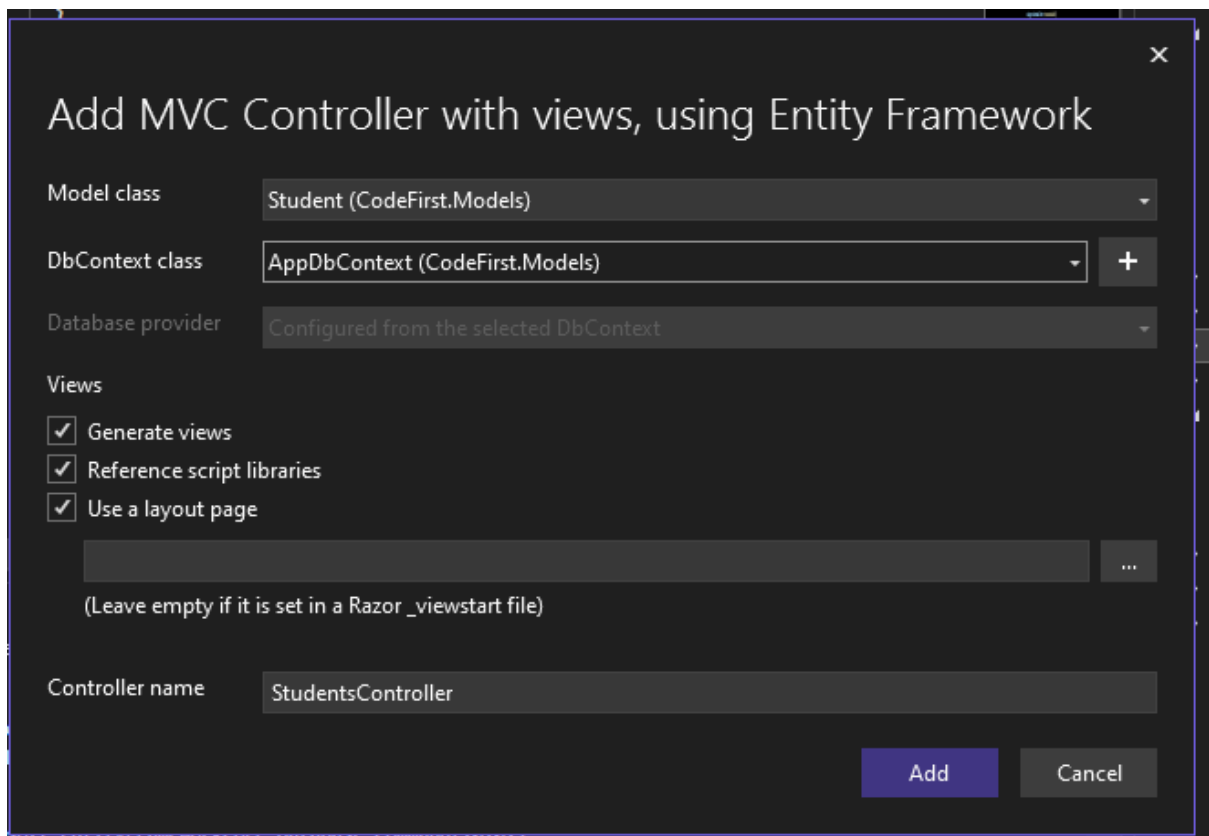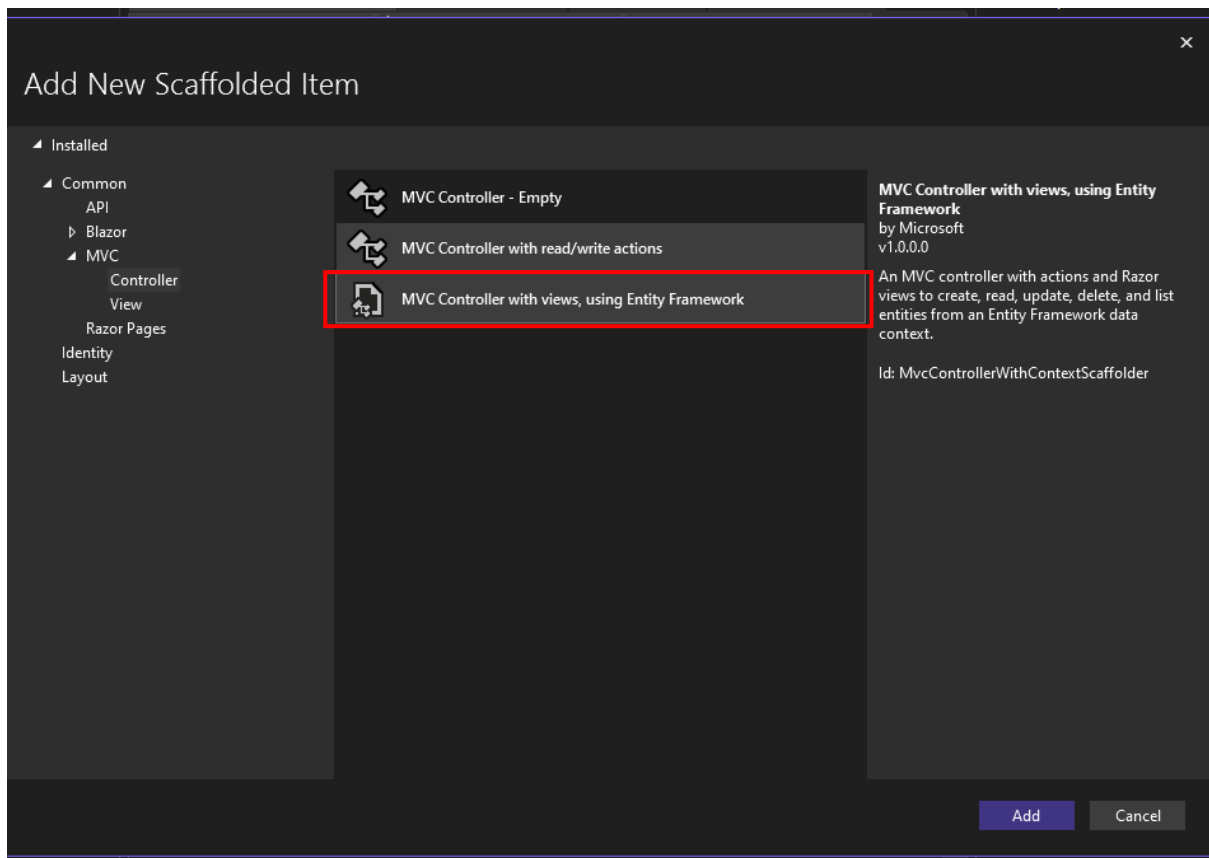
Name: Rajvardhan Ganpatrao Patil
Roll No: 40   Div: A

View and controller automatically created
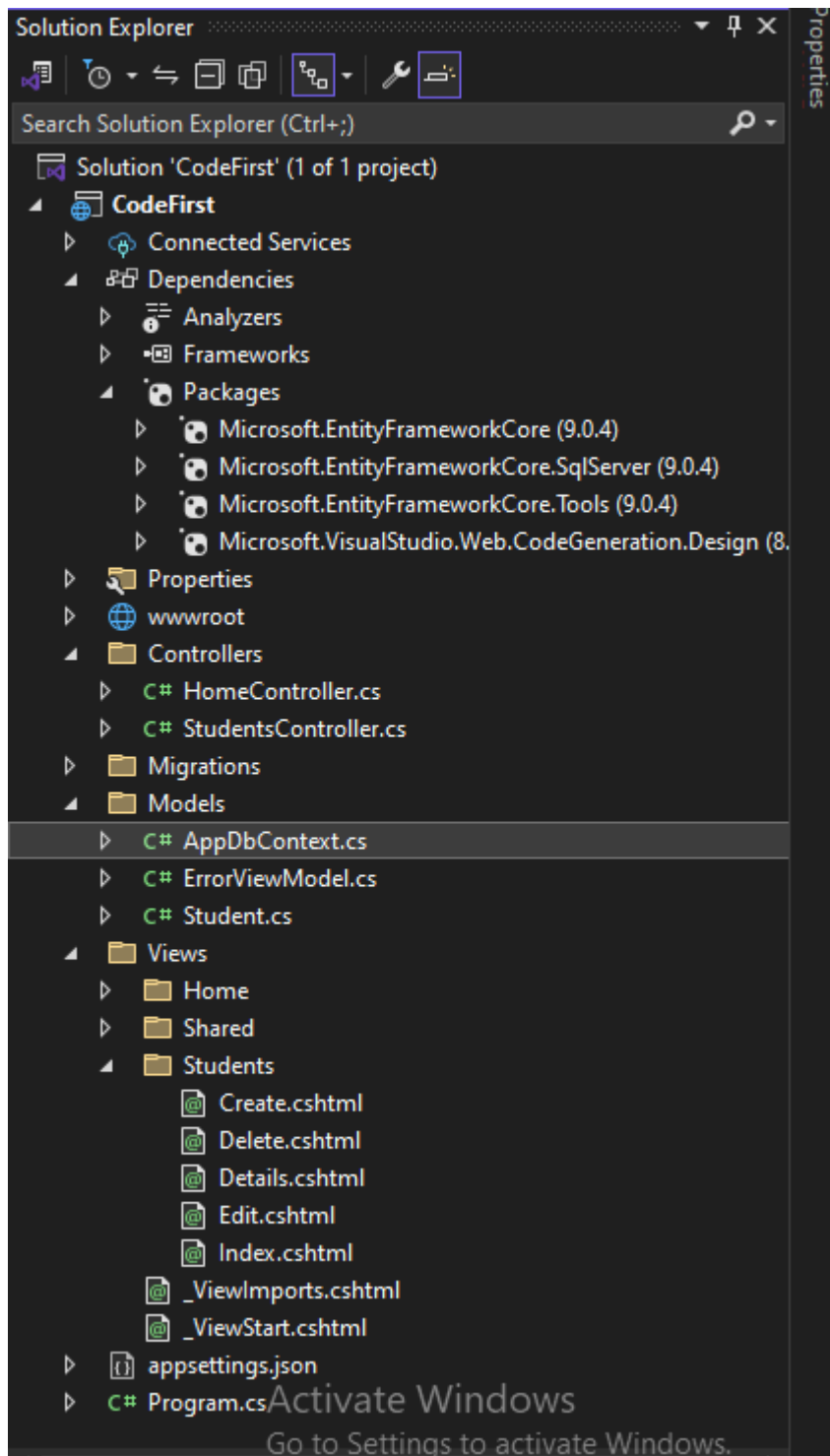
Name: Rajvardhan Ganpatrao Patil
Roll No: 40   Div: A

Change controller name in program.cs

```
app.MapControllerRoute(
    name: "default",
    pattern: "{controller=Students}/{action=Index}/{id?}");

app.Run();
```

Output:

Home page:

CodeFirst   Home   Privacy

Index

Create New

Name	Email	Contactno	
Rajvardhan	raj@gmail.com	987654321	Edit	Details	Delete

© 2025 - CodeFirst - Privacy

Create page:

CodeFirst   Home   Privacy

Create

Student

Name

Email

Contactno

Create

Back to List

© 2025 - CodeFirst - Privacy

Name: Rajvardhan Ganpatrao Patil
Roll No: 40   Div: A

Data inserted successfully:

Index

Create New

Name	Email	Contactno	
Rajvardhan	raj@gmail.com	987654321	Edit	Details	Delete
Amol	amol@gmail.com	58494968	Edit	Details	Delete

Edit page:

Edit

Student

Name

Rajvardhan

Email

raj24@gmail.com

Contactno

987654321

Save

Back to List

Name: Rajvardhan Ganpatrao Patil
Roll No: 40   Div: A

Details Page:

CodeFirst   Home   Privacy

Details

Student

Name	Rajvardhan
Email	raj24@gmail.com
Contactno	987654321

Edit | Back to List

© 2025 - CodeFirst - Privacy

Delete Page:

CodeFirst   Home   Privacy

Delete

Are you sure you want to delete this?

Student

Name	Amol
Email	amol@gmail.com
Contactno	58494968

Delete | Back to List

© 2025 - CodeFirst - Privacy

4. ASP.NET Core MVC with EF Core DB First

Install Pakages
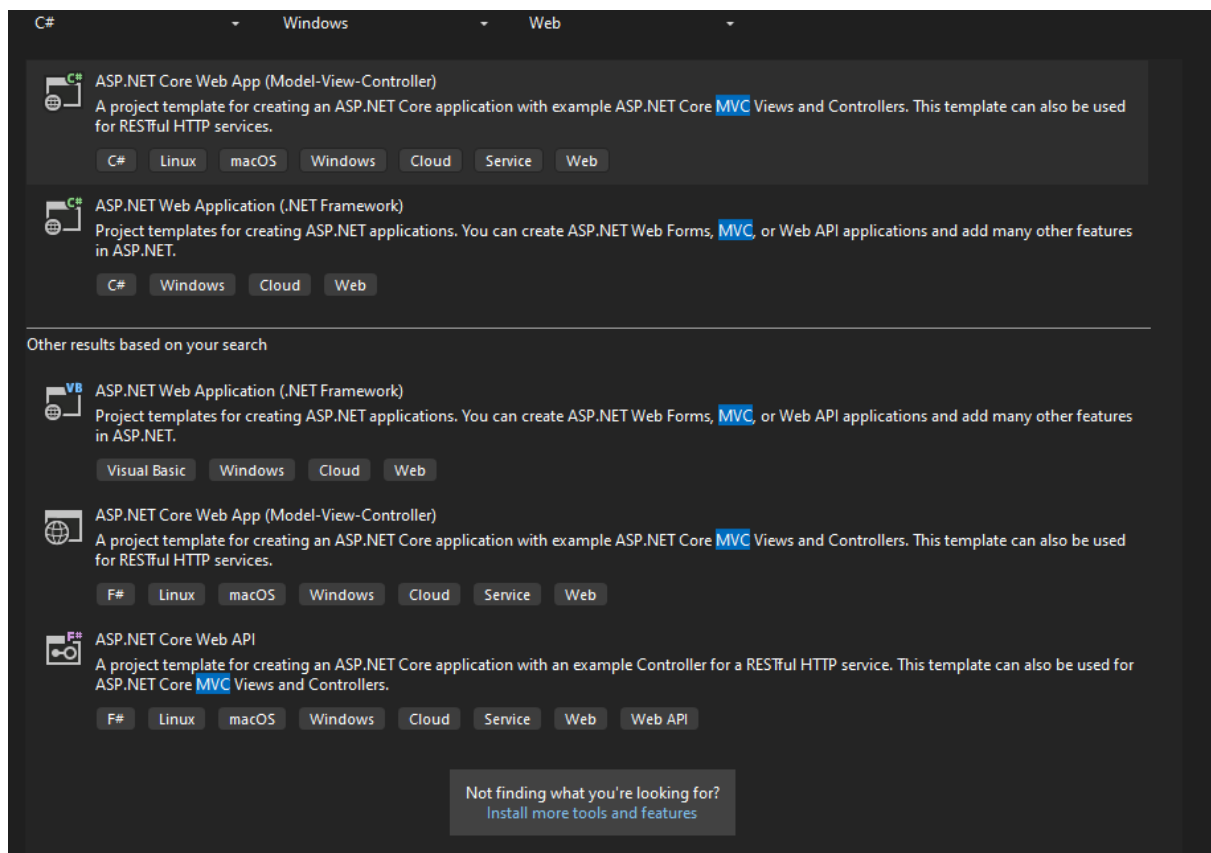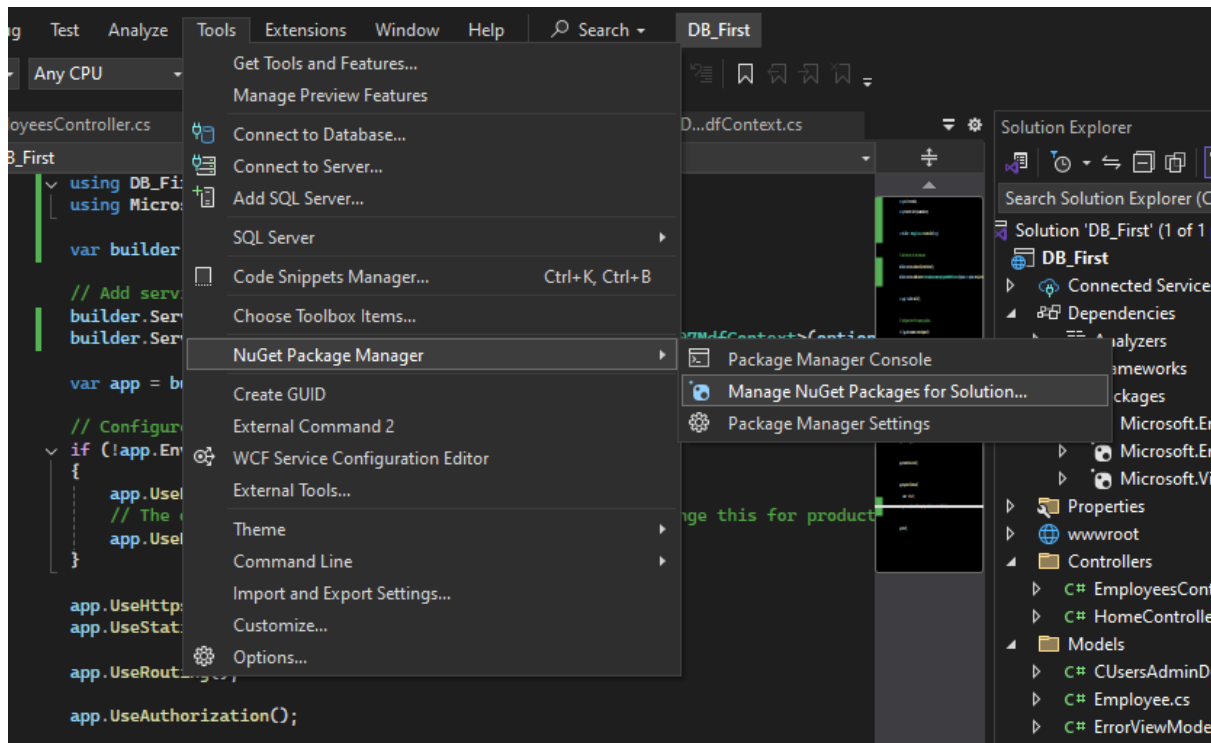
Name: Rajvardhan Ganpatrao Patil
Roll No: 40   Div: A
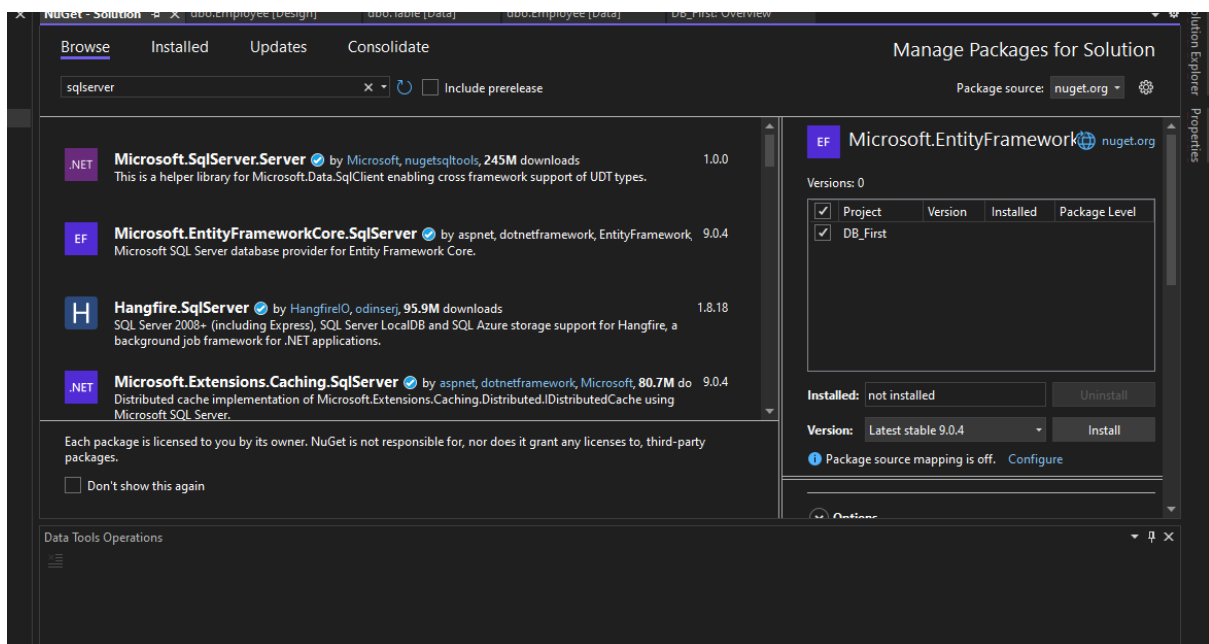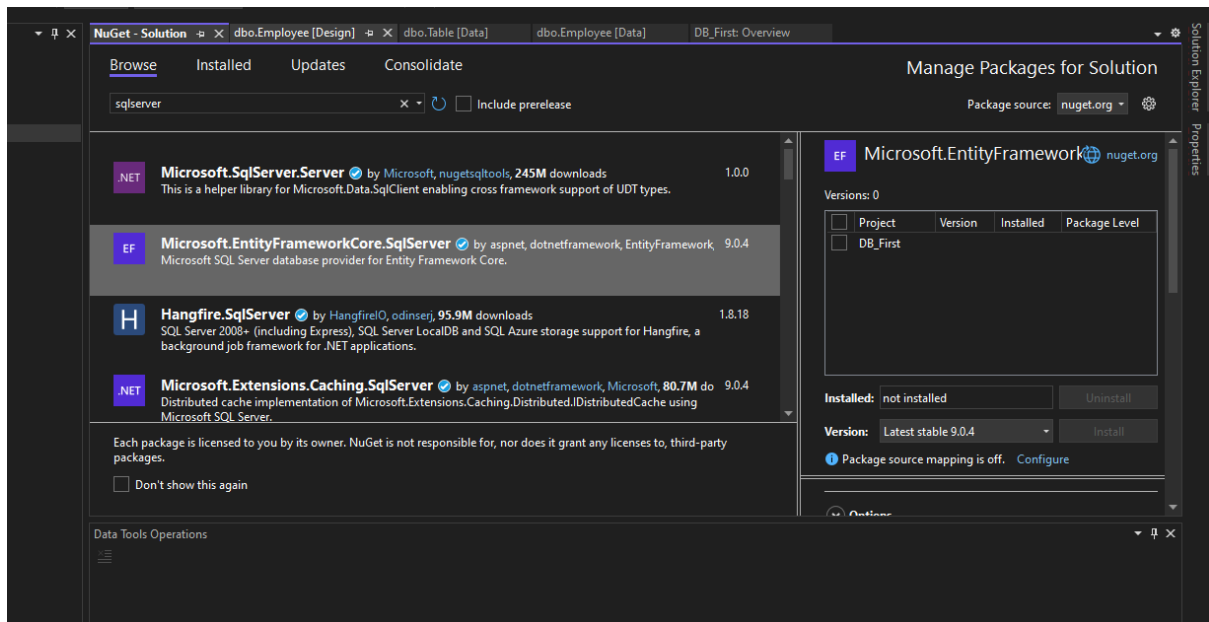
Name: Rajvardhan Ganpatrao Patil
Roll No: 40   Div: A
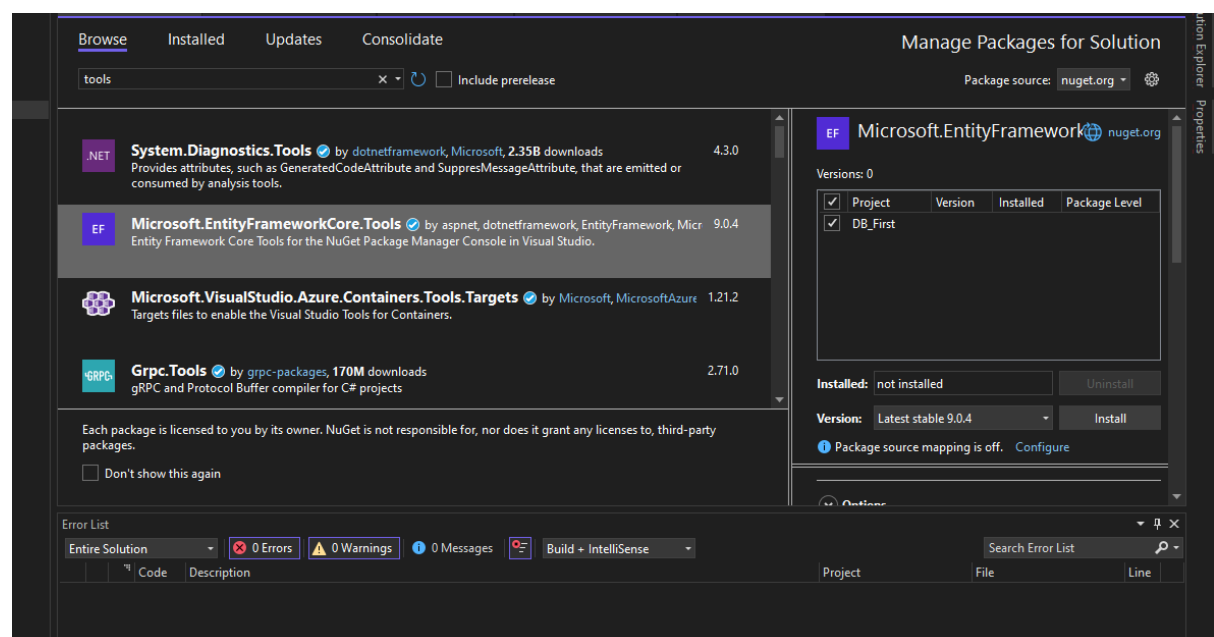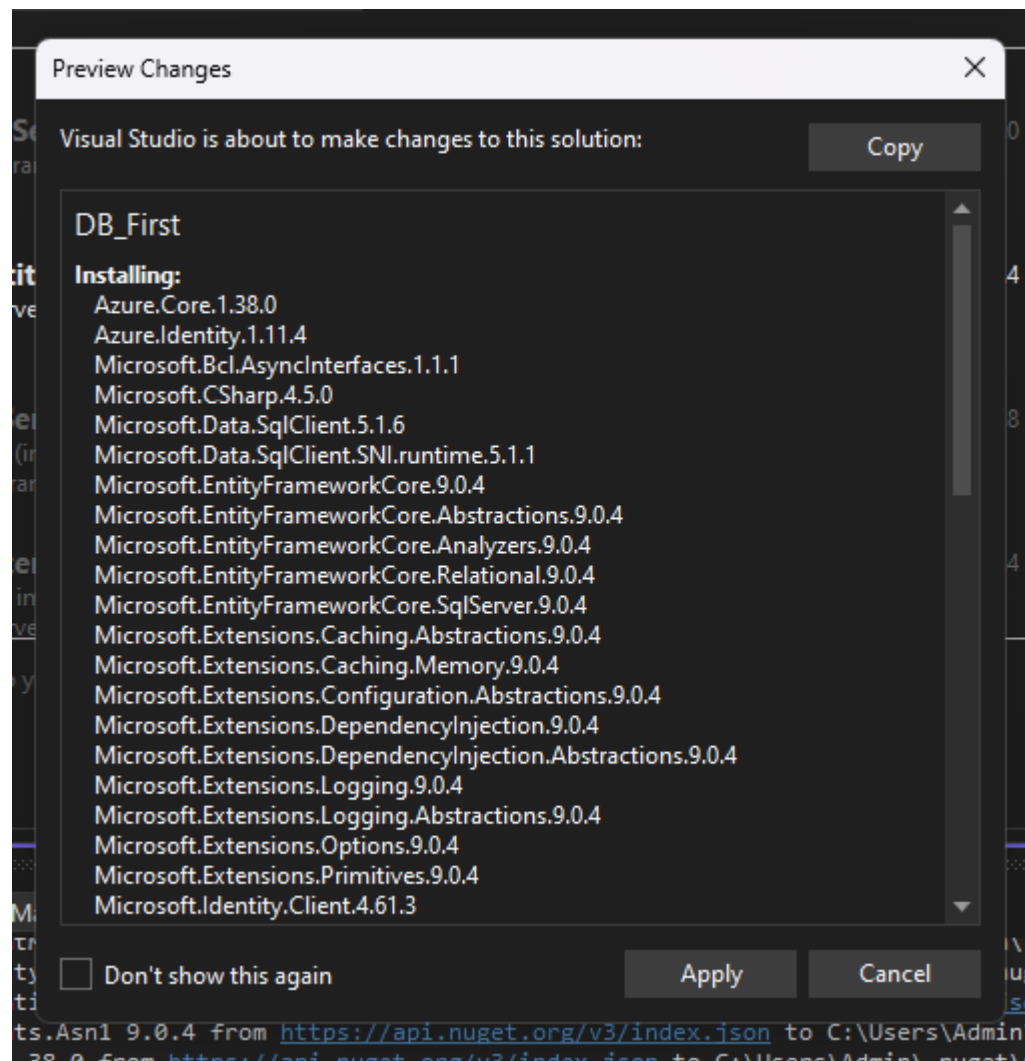
Name: Rajvardhan Ganpatrao Patil
Roll No: 40   Div: A

Verify Packages are installed
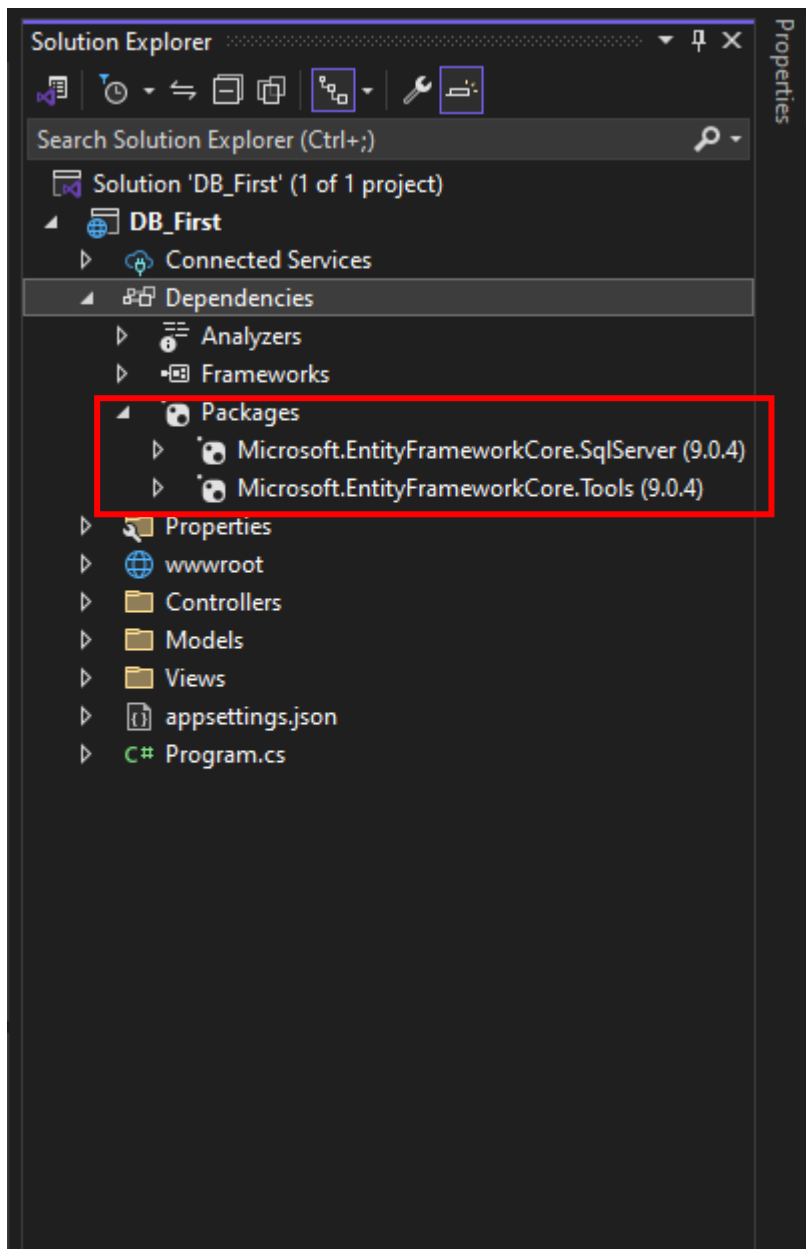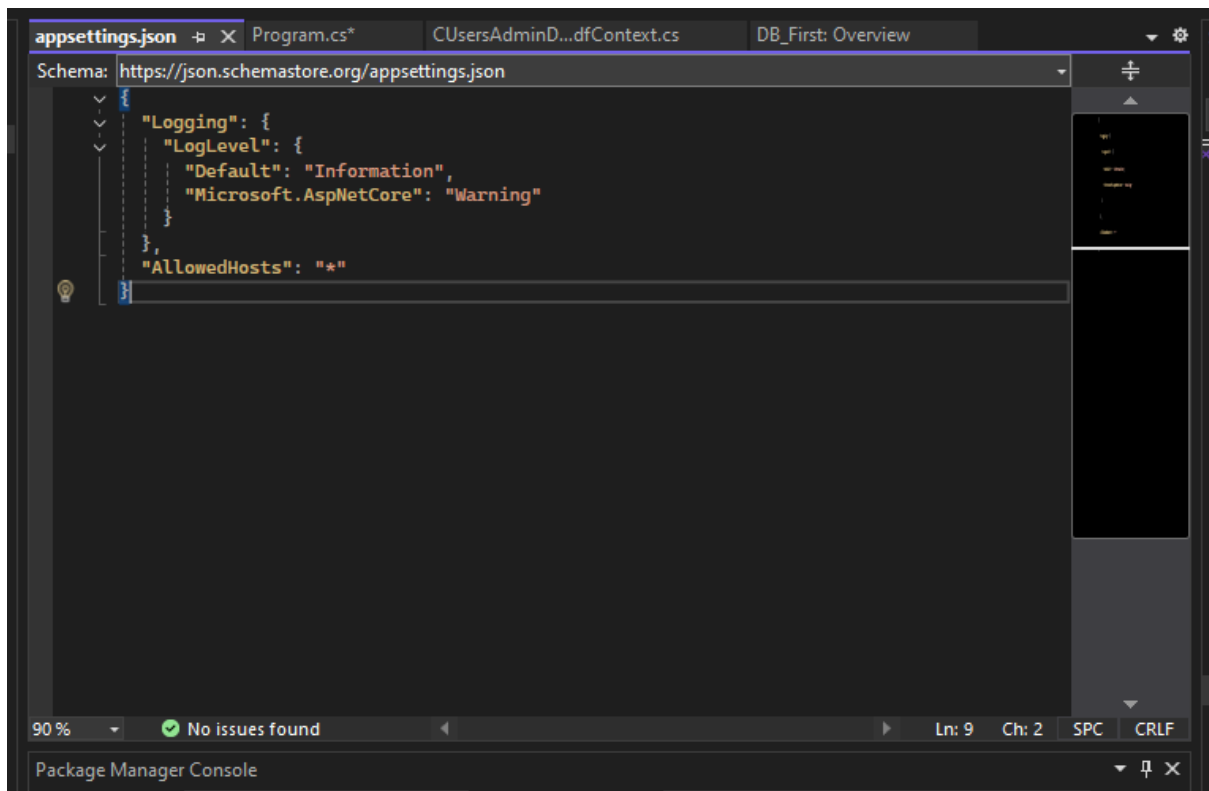
Name: Rajvardhan Ganpatrao Patil
Roll No: 40   Div: A

Scaffold-DbContext
"Server=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\Admin\Documents\Stud.mdf;Integrated Security=True;;TrustServerCertificate=True; "Microsoft.EntityFrameworkCore.SqlServer -OutputDir Models.

builder.Services.AddDbContext<YourDbContext>(options =>
options.UseSqlServer(builder.Configuration.GetConnectionString("DefaultConnection")));
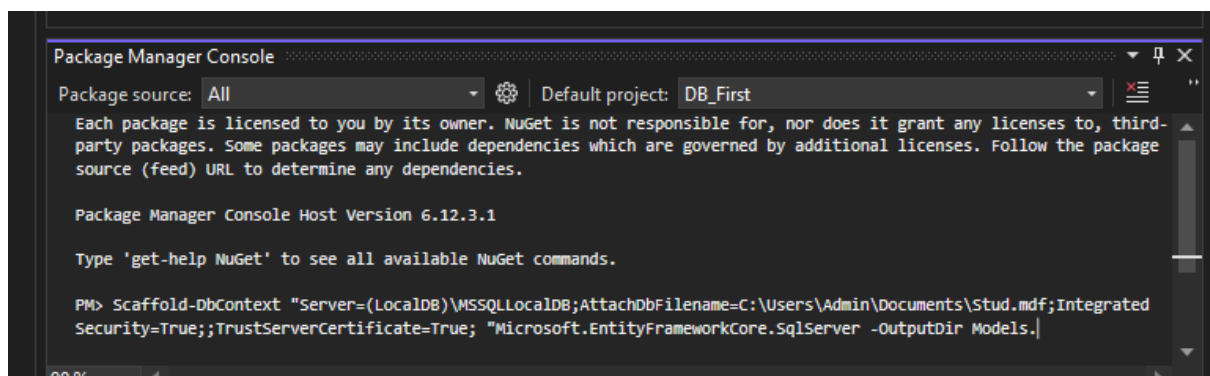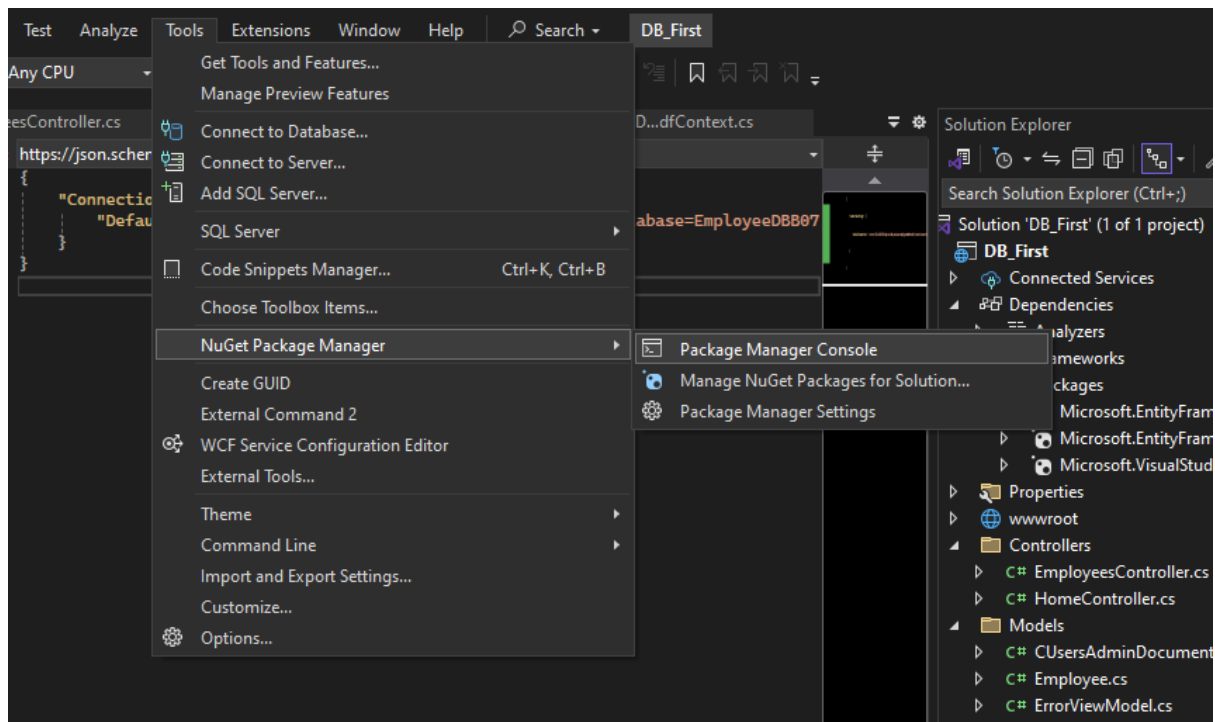
Name: Rajvardhan Ganpatrao Patil
Roll No: 40   Div: A

Change Server: Go into view-> sql server object explorer -> copy the sql server name (right click rename then copy)
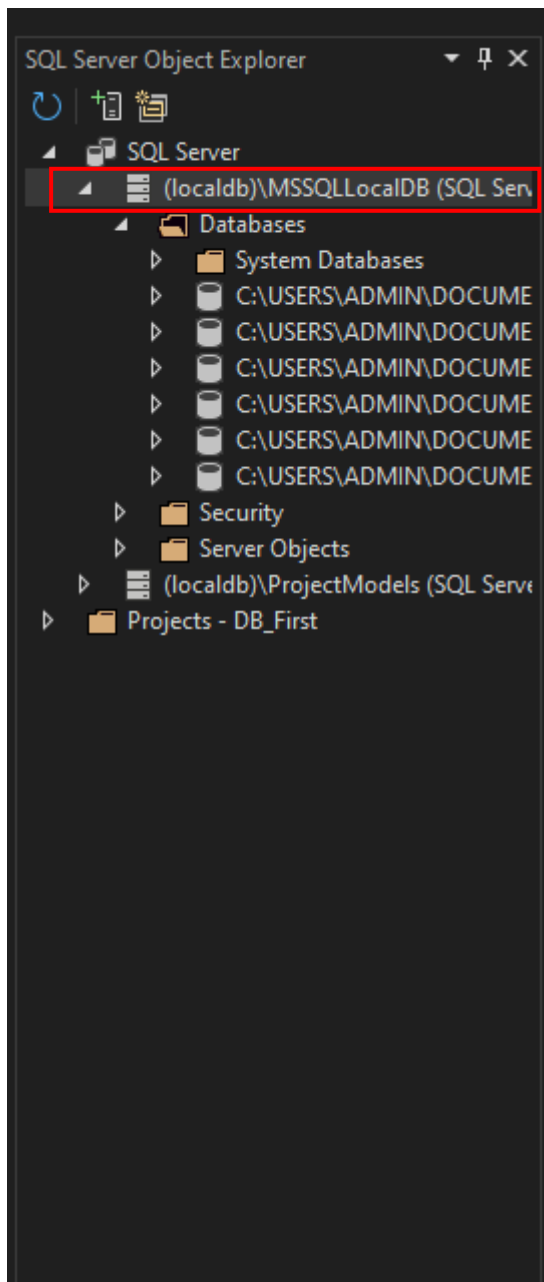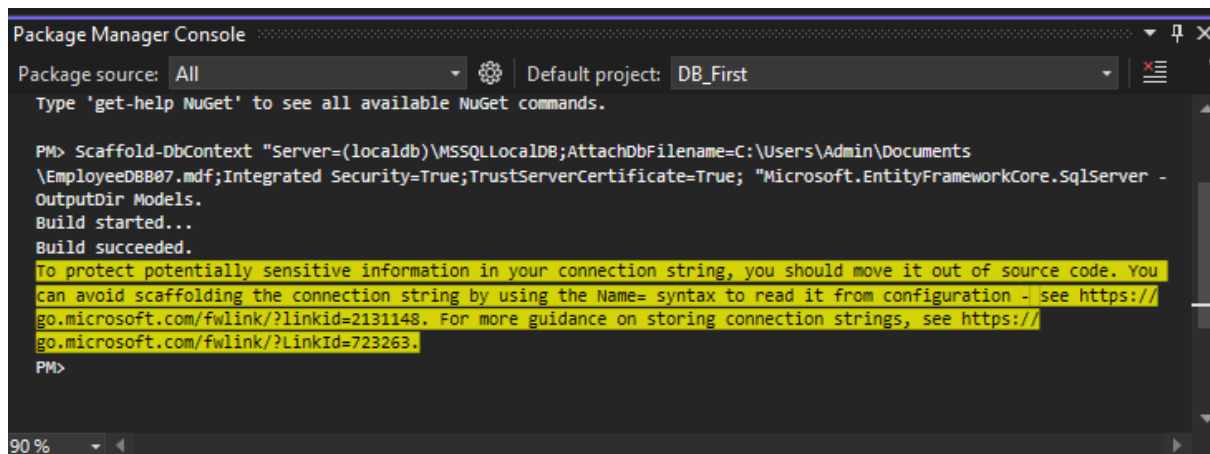
Change AttachDbFilename : copy the db connection string in that copy only path
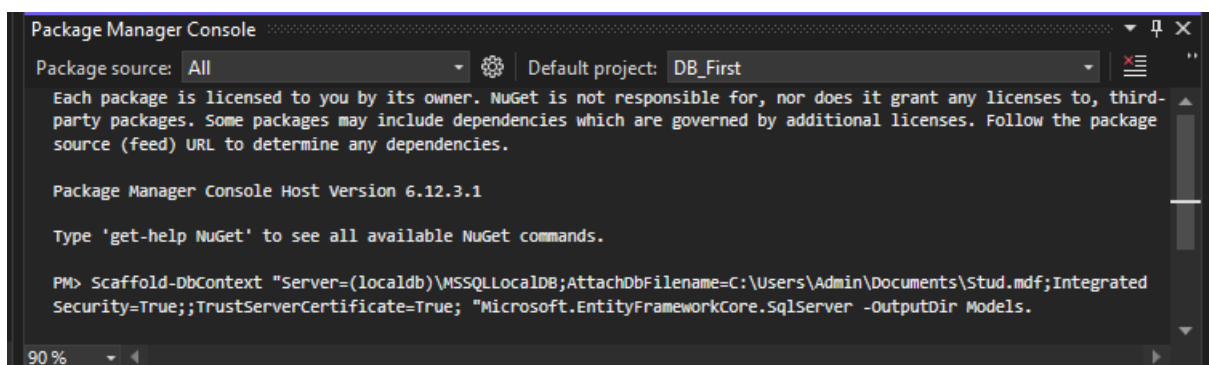
Name: Rajvardhan Ganpatrao Patil
Roll No: 40   Div: A

Press Enter

The classes is created automatically in model and view is also automatically created
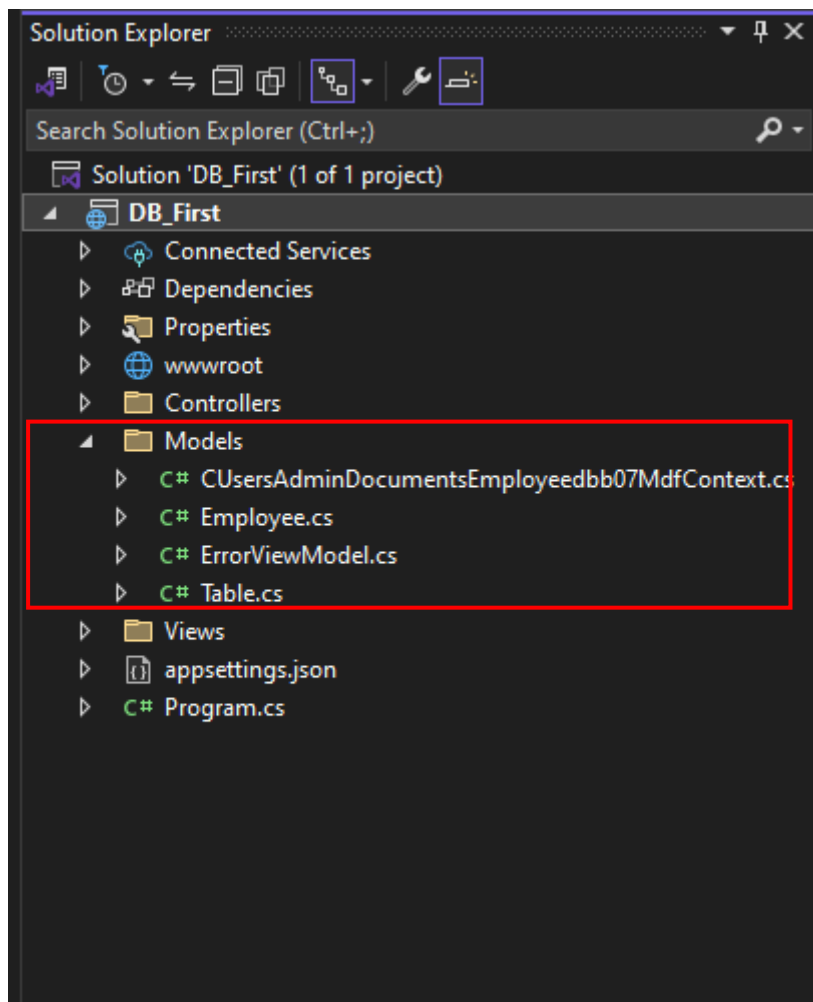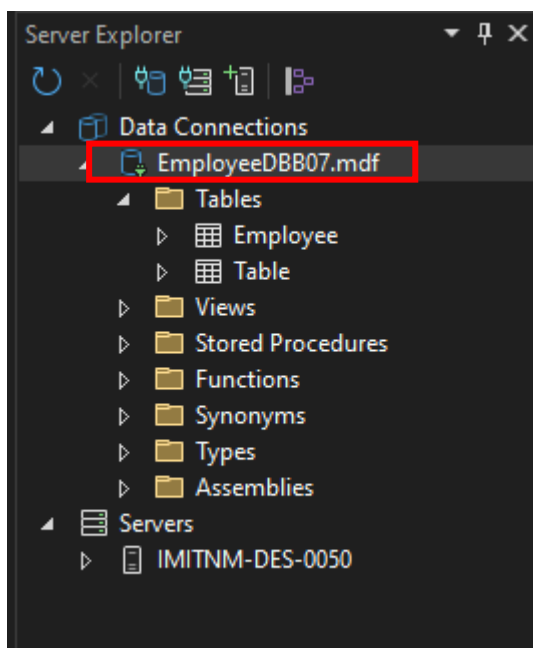
Name: Rajvardhan Ganpatrao Patil
Roll No: 40   Div: A

Appsettings.json

"ConnectionStrings": {

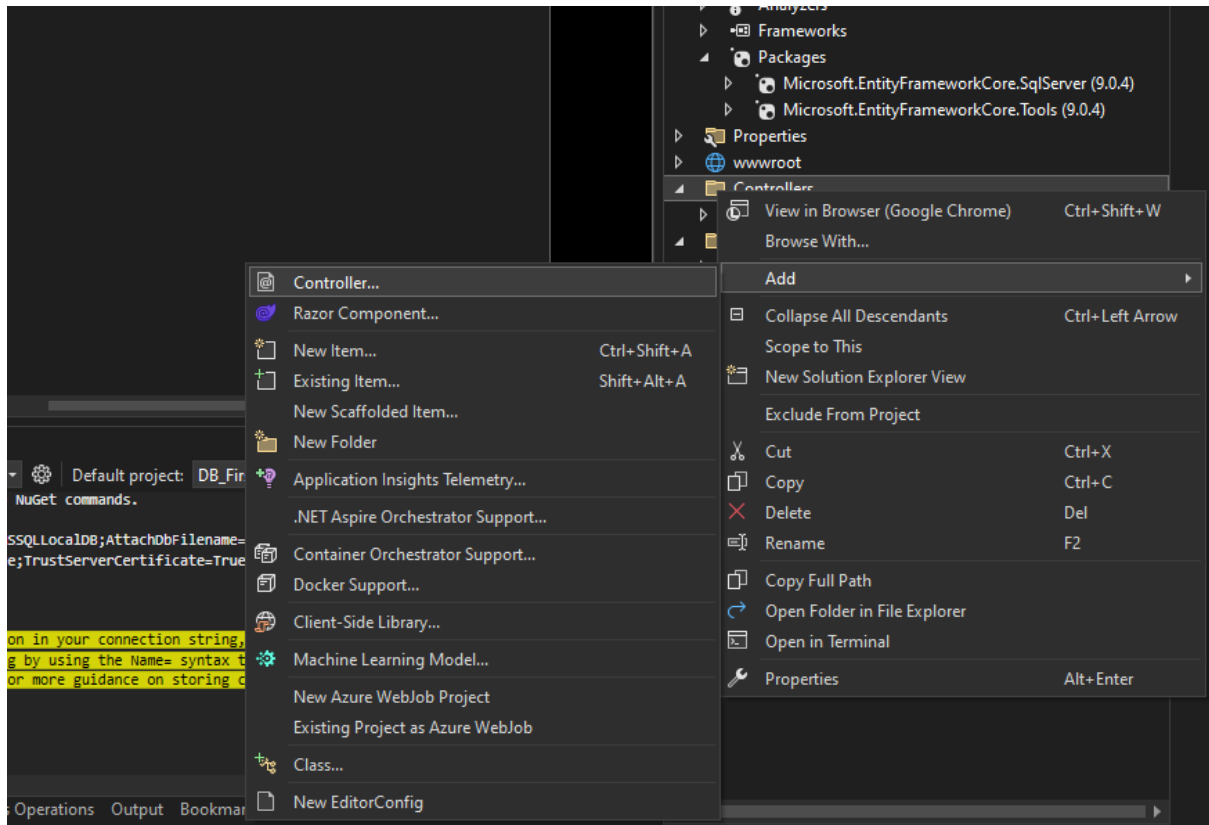 "DefaultConnection": "Server=.;Database=YourDbName;Trusted_Connection=True;"
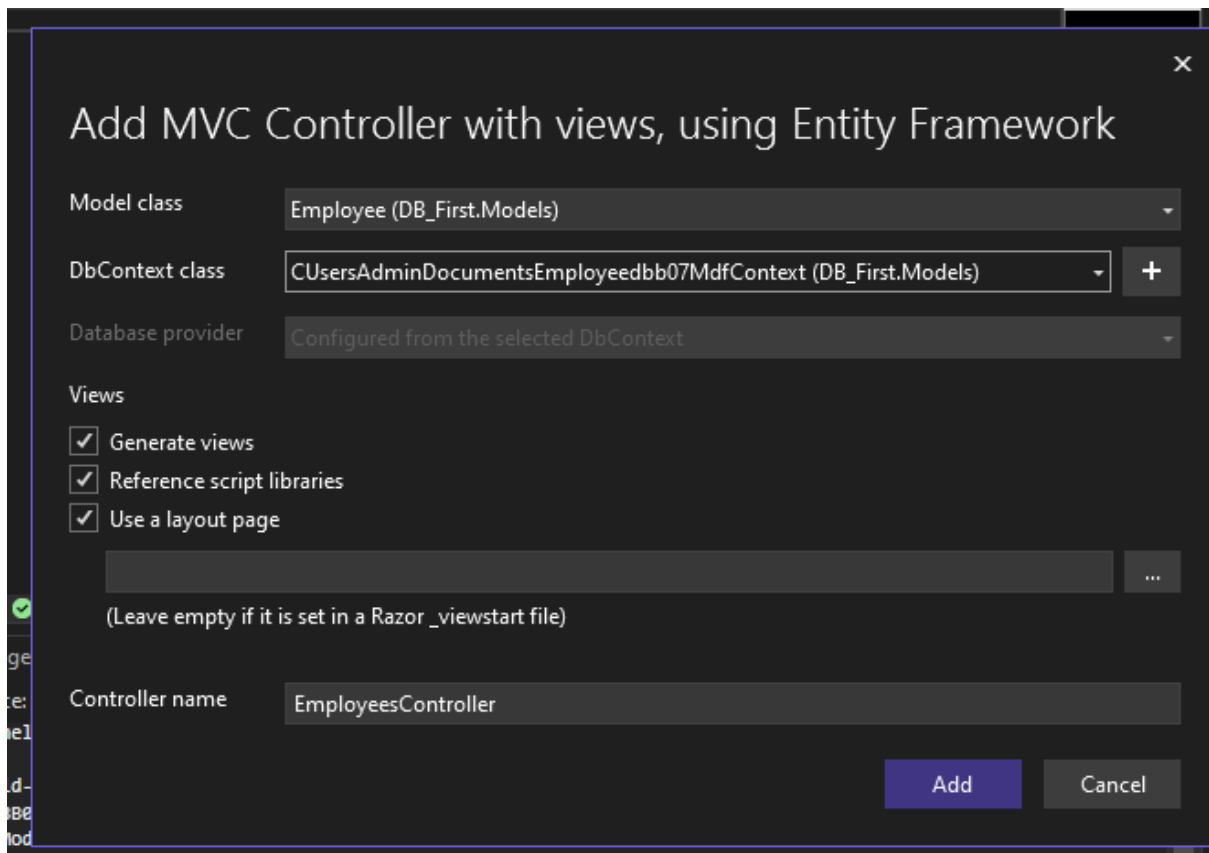
}

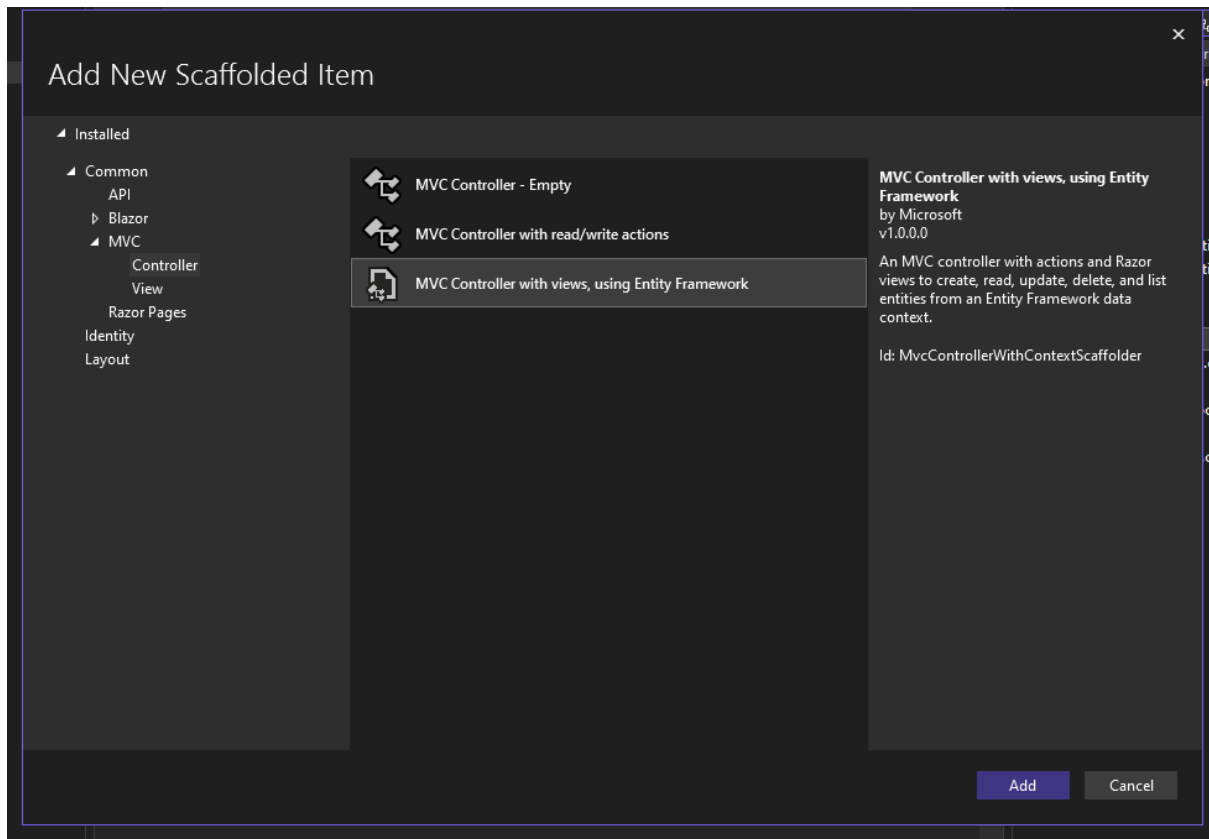Write the server name and write the database name

```
a: https://json.schemastore.org/appsettings.json
{
    "ConnectionStrings": {
        "DefaultConnection": "Server=(localdb)\\MSSQLLocalDB;Database=EmployeeDBB07
    }
}
```
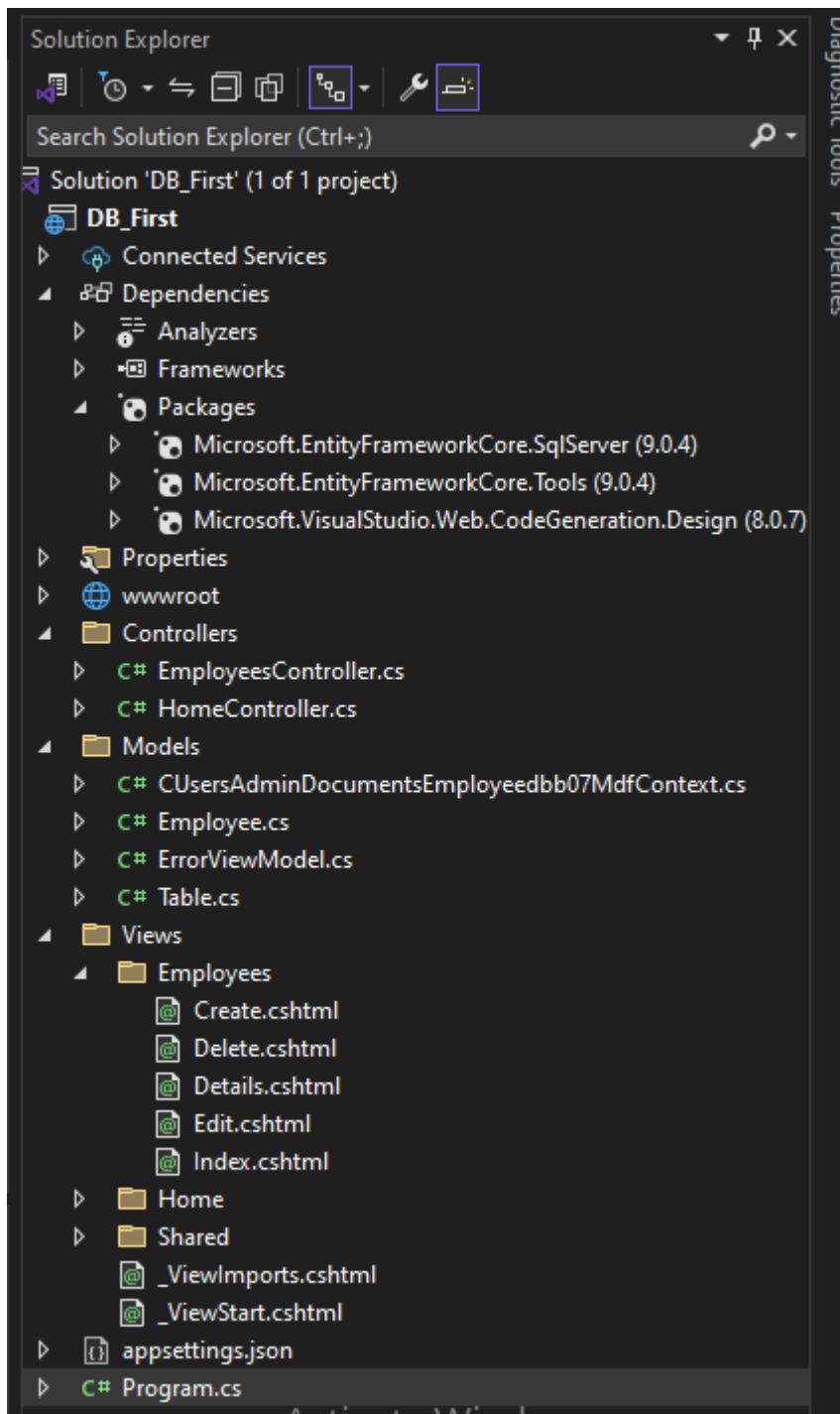
Create controller

Name: Rajvardhan Ganpatrao Patil
Roll No: 40   Div: A

Name: Rajvardhan Ganpatrao Patil
Roll No: 40   Div: A

Created controller and view automatically

```
    app.UseAuthorization();

    app.MapControllerRoute(
        name: "default",
        pattern: "{controller=Employees}/{action=Index}/{id?}");

    app.Run();
```

Output:

DB_First   Home   Privacy

Index

Create New

FirstName	LastName	Designation	Salary

Activate Windows
Go to Settings to activate Windows.

© 2025 - DB_First - Privacy

Create:

DB_First   Home   Privacy

Create

Employee

Id
1

FirstName
Rajvardhan

LastName
Patil

Designation
Student

Salary
0

Create

Back to List

Activate Windows
Go to Settings to activate Windows.

© 2025 - DB_First - Privacy

Name: Rajvardhan Ganpatrao Patil
Roll No: 40   Div: A

Record inserted successfully:

DB_First   Home   Privacy

Index

Create New

FirstName	LastName	Designation	Salary	
Rajvardhan	Patil	Student	0	Edit | Details | Delete

Details Page:

Details

Employee

FirstName	Rajvardhan
LastName	Patil
Designation	Student
Salary	0

Edit | Back to List

Delete Page:

Delete

Are you sure you want to delete this?

Employee

FirstName	Rajvardhan
LastName	Patil
Designation	Student
Salary	0

Delete | Back to List