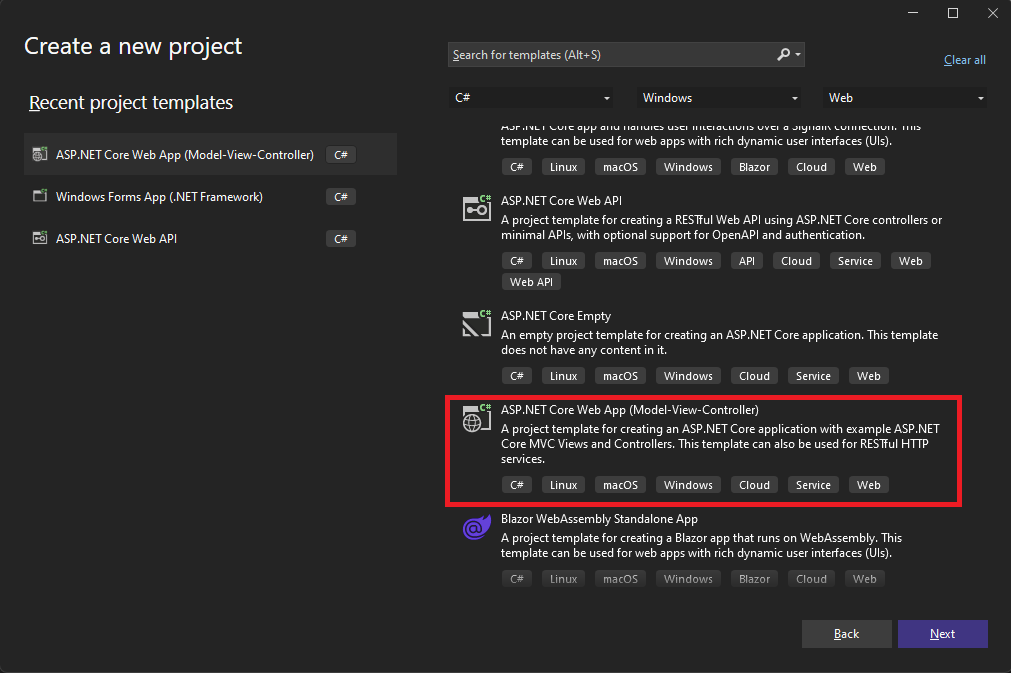
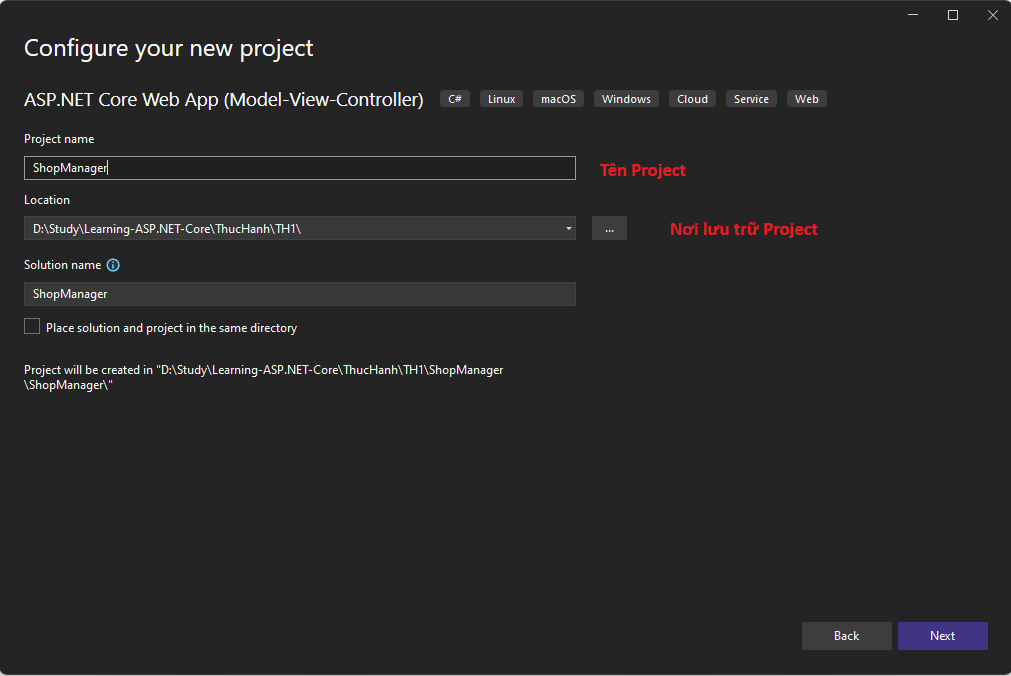


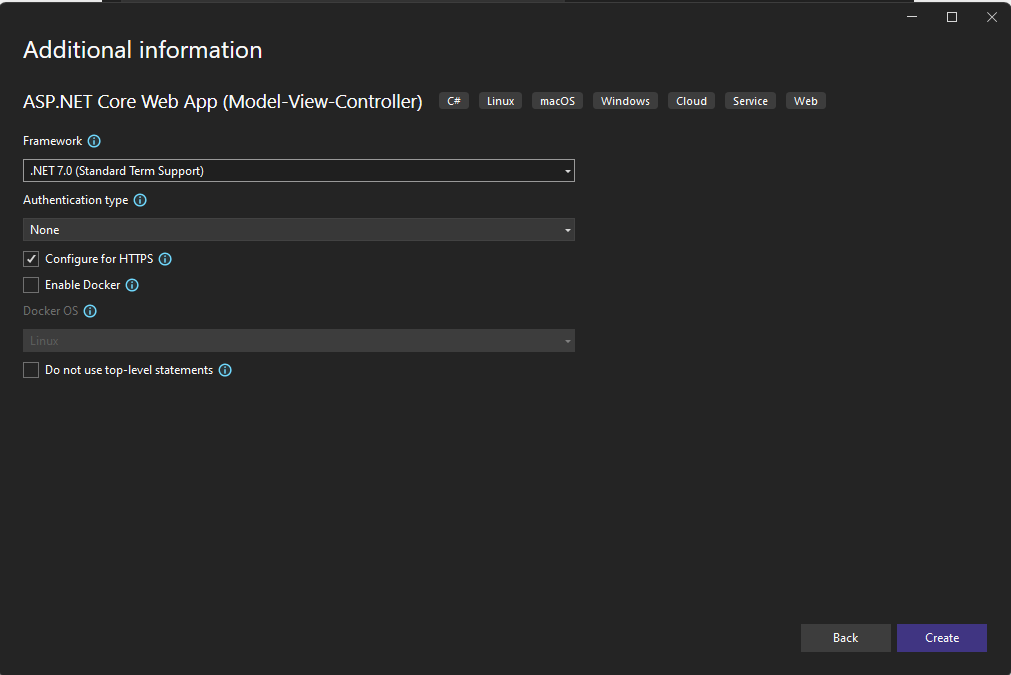
Sau khi khởi động Visual Studio 2022, để tạo mới Project ta chọn Create a new project



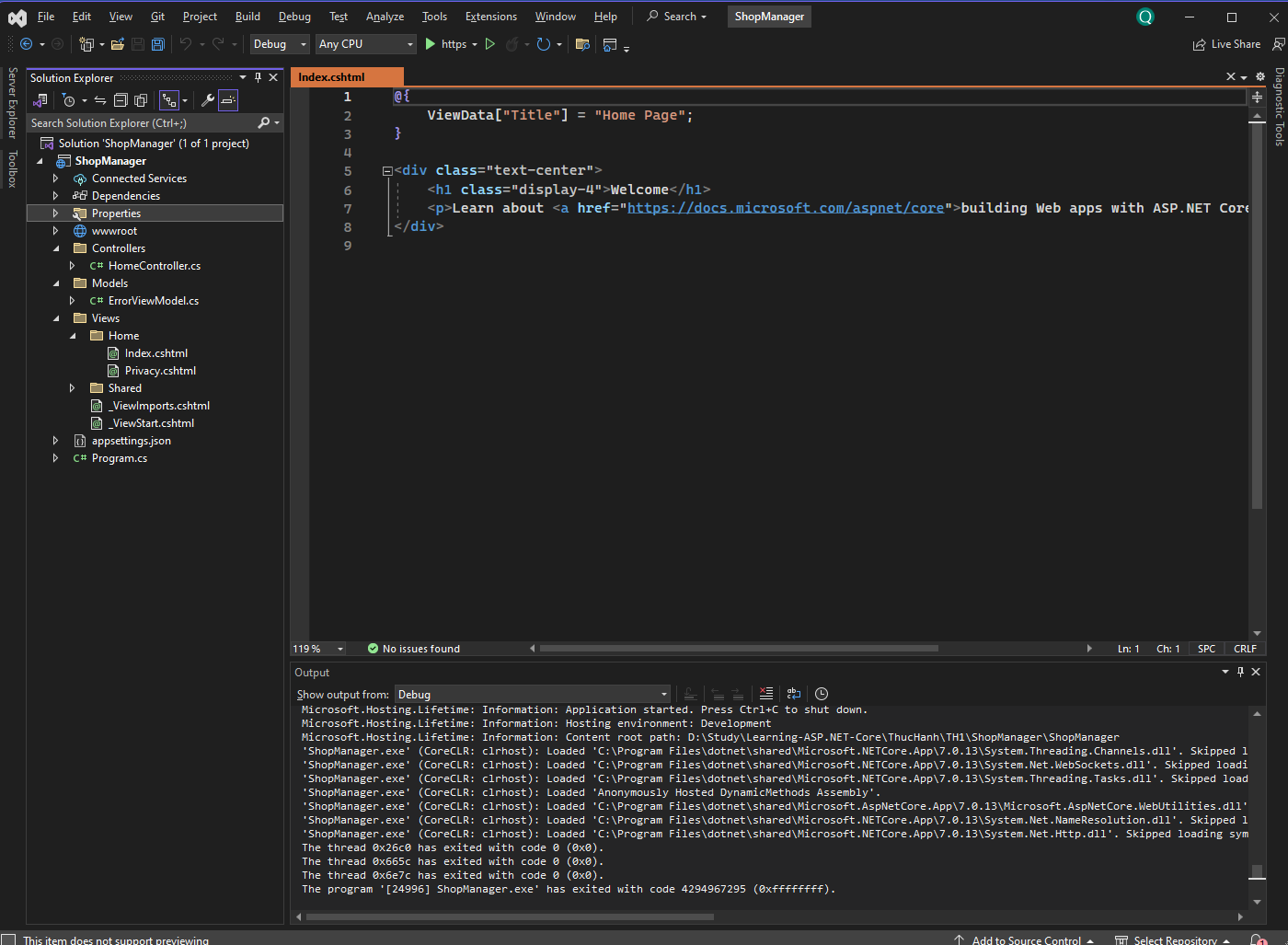
Chọn Tempate cho project: ASP.NET Core Web App (Model-View-Controller)



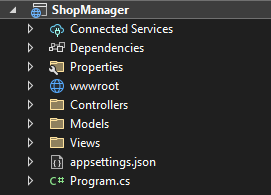
Điền tên và đường dẫn thư mục lưu trữ Project



Chọn phiên bản Framework (tùy thuộc phiên bản dotnet đã cài đặt).

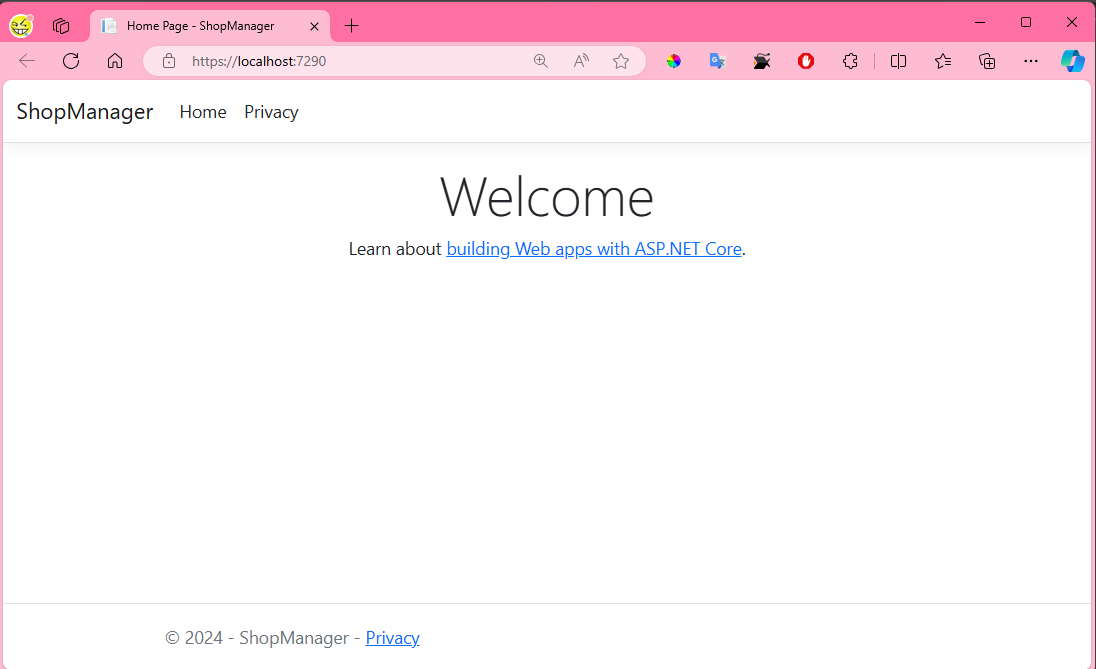


Kết quả sau khi khởi tạo Project thành công

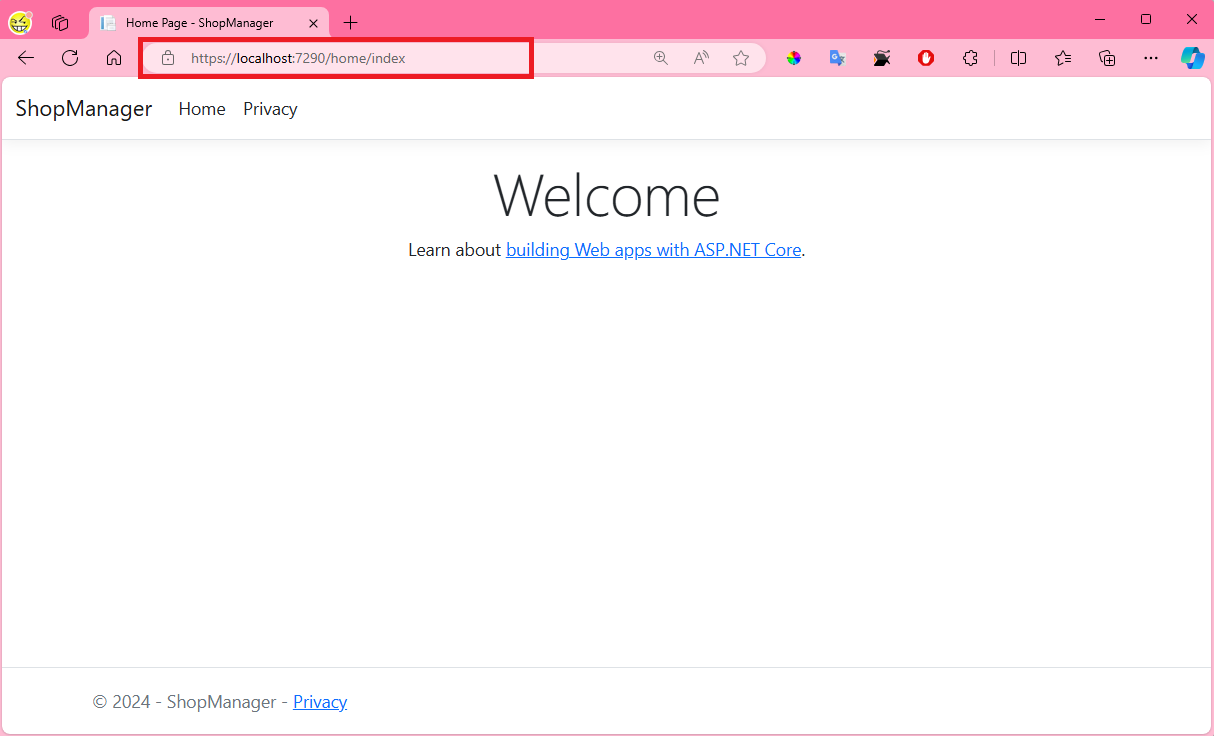


Cấu trúc thư mục của project , bao gồm:

* Properties: Chứa các tệp tin cấu hình dự án, bao gồm launchSettings.json để cấu hình cách dự án sẽ được chạy.
* wwwroot: Chứa tất cả các tài nguyên tĩnh của ứng dụng, chẳng hạn như các file CSS, JavaScript, hình ảnh và các tài liệu tĩnh khác.
* **Controllers**: Chứa các file điều khiển (controllers) của ứng dụng.
* **Views**: Chứa các file chế độ xem (views) của ứng dụng, thường được sử dụng để tạo các giao diện người dùng.
* **Models**: Chứa các models của ứng dụng, đại diện cho dữ liệu và logic trong ứng dụng.
* appsettings.json: Chứa cấu hình ứng dụng, bao gồm chuỗi kết nối cơ sở dữ liệu và các cài đặt khác.
* Program.cs: Là file chứa nơi khởi đầu ứng dụng và cấu hình máy chủ web. Chứa cả logic cấu hình và khởi động.

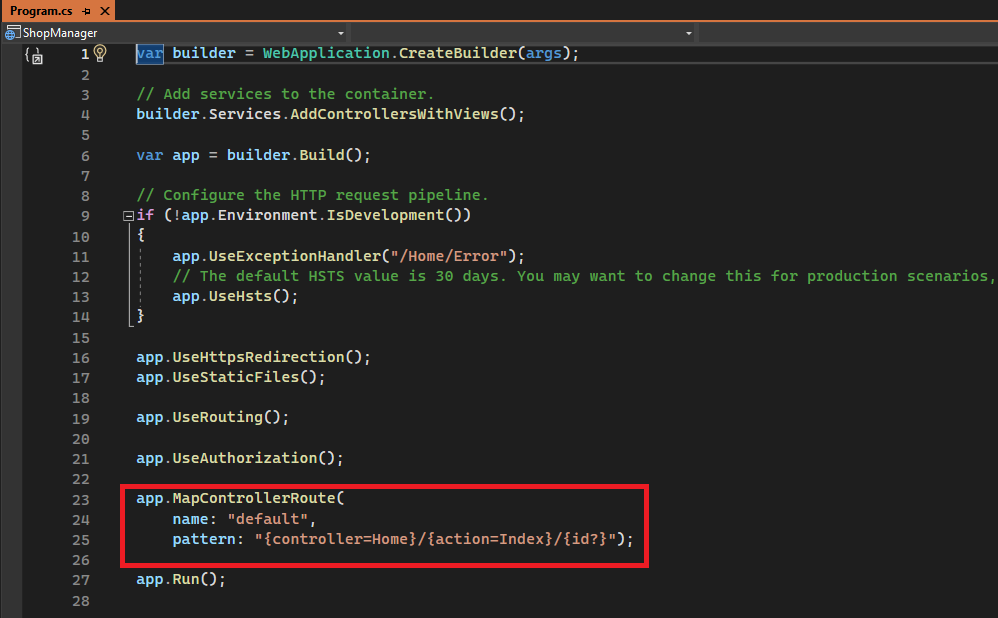


Kết quả sau khi build project thành công

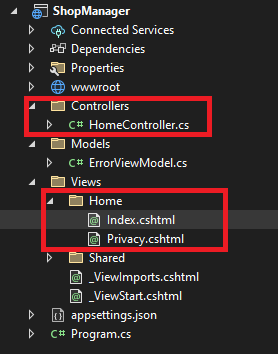


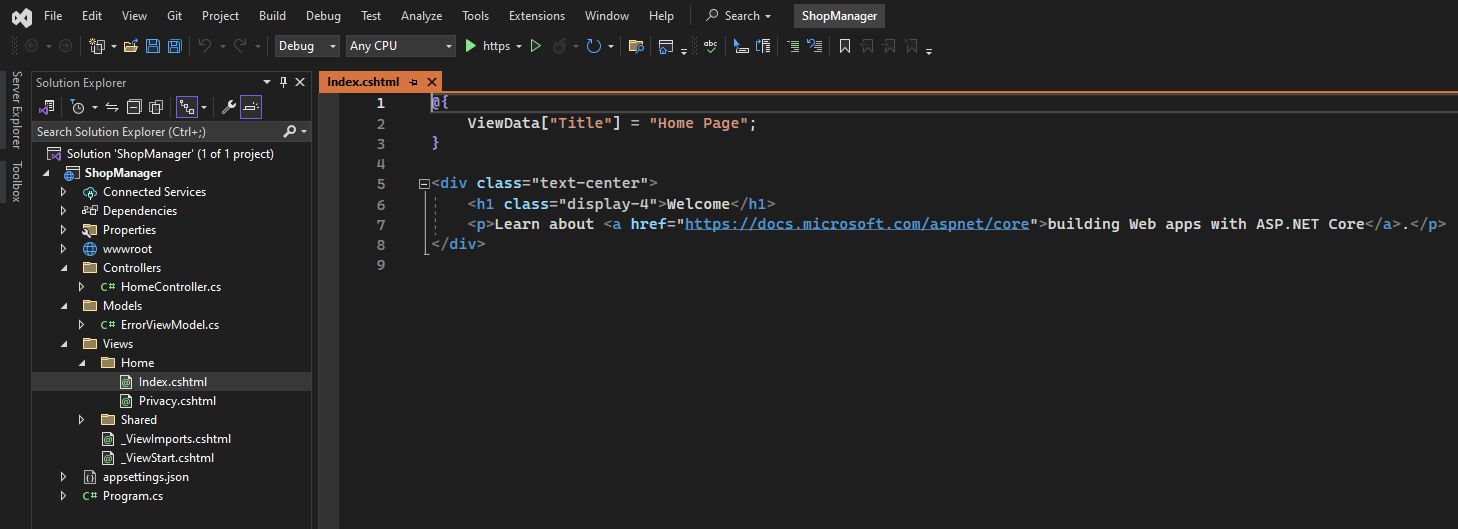
Không có sự khác biệt khi thay đổi đường dẫn từ <https://localhost:7290/>

Thành <https://localhost:7290/Home/Index>



Do đã định nghĩa đường dẫn mặc định khi build project





Đây là đoạn code của trang Home/Index

@{

ViewData["Title"] = "Hello World";

}

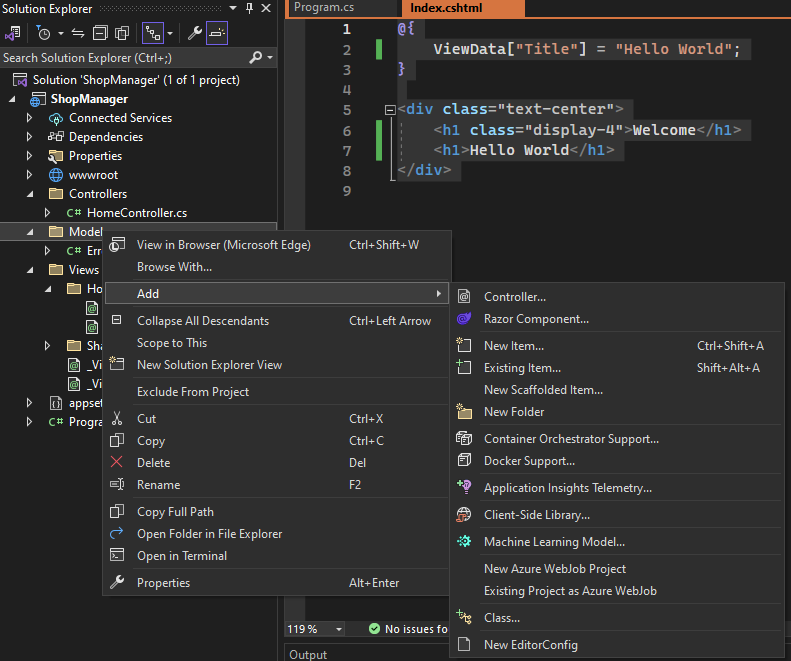
<div class="text-center">

<h1 class="display-4">Welcome</h1>

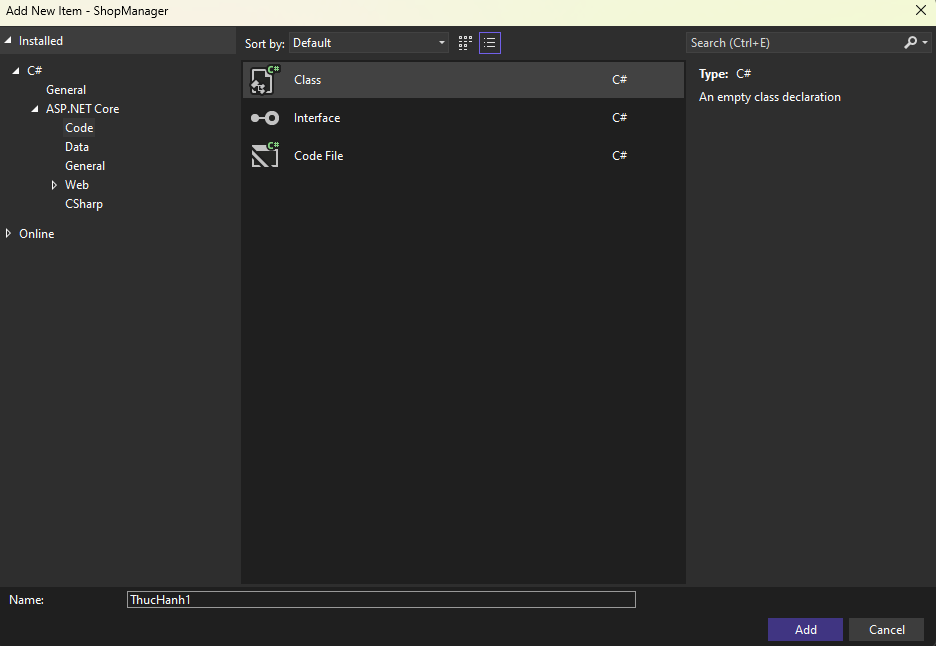
<h1>Hello World</h1>

</div>

Tạo một Model để hiển thị ở trang Home/Index



Chọn Thư mục Model -> chuột phải -> Add -> Class -> Đặt tên Class -> Add



Vd tên Class : ThucHanh1

namespace ShopManager.Models

{

public class ThucHanh1

{

public int Id { get; set; }

public string Name { get; set; }

public string Description { get; set; }

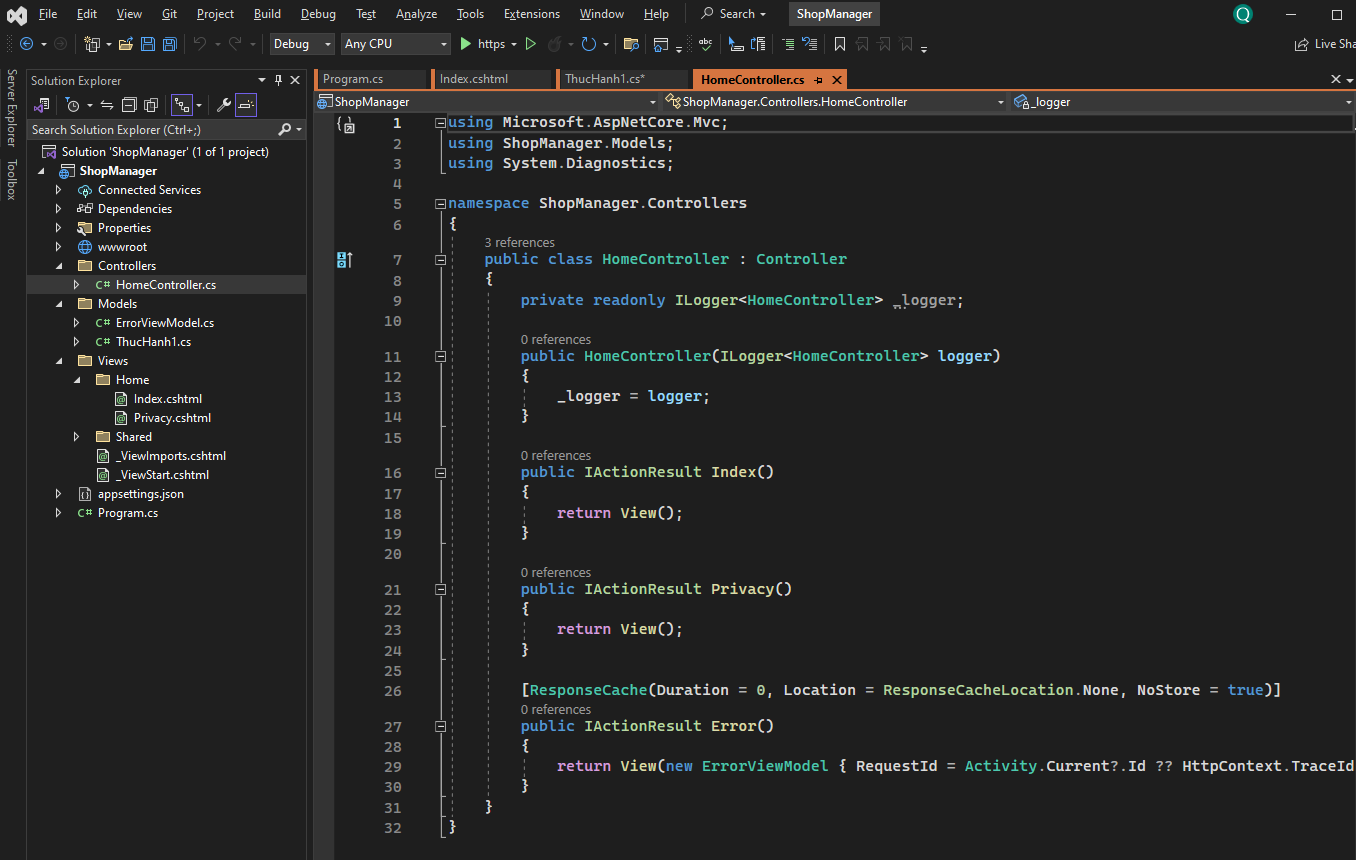
}

}

Định nghĩa cho Class vừa tạo 3 thuộc tính

Để truyền dữ liệu cho View ở trang Home/Index:

Chọn Controllers -> HomeController.cs



Bao gồm 2 View là Index và Privacy tương ứng với 2 file Index.cshtml và Privacy.cshtml trong thư mục Views/Home

**Controller/HomeController.js**

public IActionResult Index()

{

ThucHanh1 thucHanh1 = new ThucHanh1()

{

Id = 1,

Name ="Thực hành buổi 1",

Description="Cài đặt Visual Studio, Build dự án Hello World"

};

return View(thucHanh1);

}

Mở File **View/Home/Index.cshtml**

Khai báo Model được sử dụng ở đầu đoạn code, sau đó hiển thị Model với các thuộc tính tương ứng bằng cú pháp **@Model.**

@model ShopManager.Models.ThucHanh1

@{

ViewData["Title"] = "Hello World";

}

<div>

<h1 class="display-4">Welcome</h1>

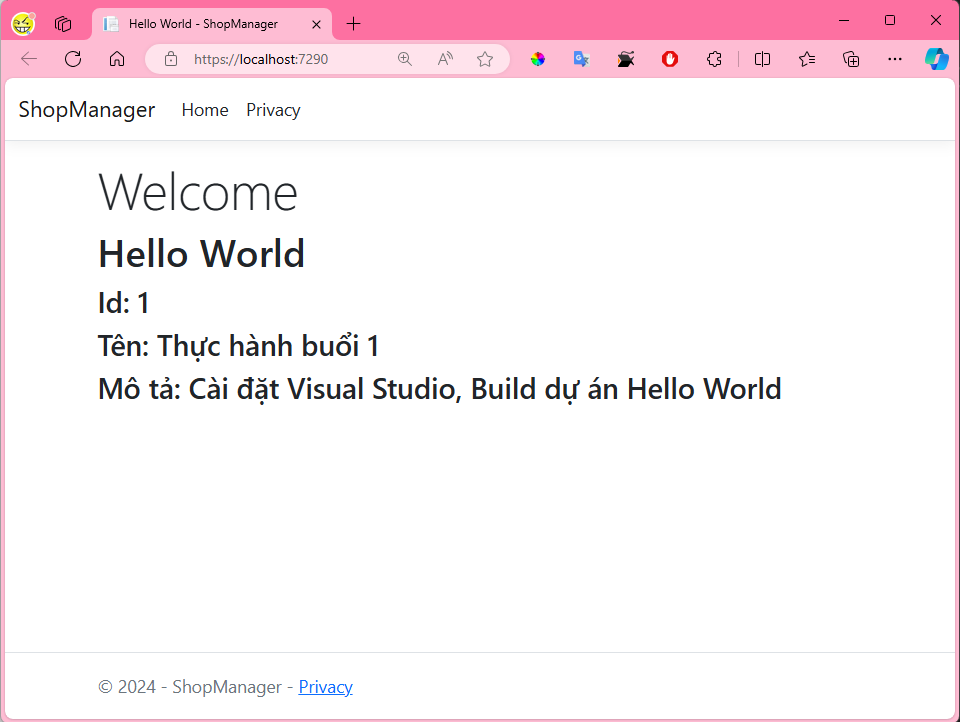
<h1>Hello World</h1>

<h3>Id: @Model.Id</h3>

<h3>Tên: @Model.Name</h3>

<h3>Mô tả: @Model.Description</h3>

</div>



Hiển thị List Model thay vì hiển thị 1 Model duy nhất

**Controller/HomeController.cs**

public IActionResult Index()

{

List<ThucHanh1> thucHanh1s = new List<ThucHanh1> {

new ThucHanh1()

{

Id = 1,

Name = "Thực hành buổi 1",

Description = "Cài đặt Visual Studio, Build dự án Hello World"

},

new ThucHanh1()

{

Id = 2,

Name ="Thực hành buổi 2",

Description="Cài đặt Visual Studio, Build dự án Hello World"

},

new ThucHanh1()

{

Id = 3,

Name ="Thực hành buổi 3",

Description="Cài đặt Visual Studio, Build dự án Hello World"

},

};

return View(thucHanh1s);

}

View/Home/Index.cshtml

Thay đổi Khai báo Model thành List Model: @model IEnumerable<ShopManager.Models.ThucHanh1>

Sử dụng **@foreach(var item in @Model)** để truy vấn toàn bộ List Model

@model IEnumerable<ShopManager.Models.ThucHanh1>

@{

ViewData["Title"] = "Hello World";

}

<div >

<h1 class="display-4">Welcome</h1>

<h1>Hello World</h1>

@foreach(var item in @Model)

{

<div >

<h3>Id: @item.Id</h3>

<h3>Tên: @item.Name</h3>

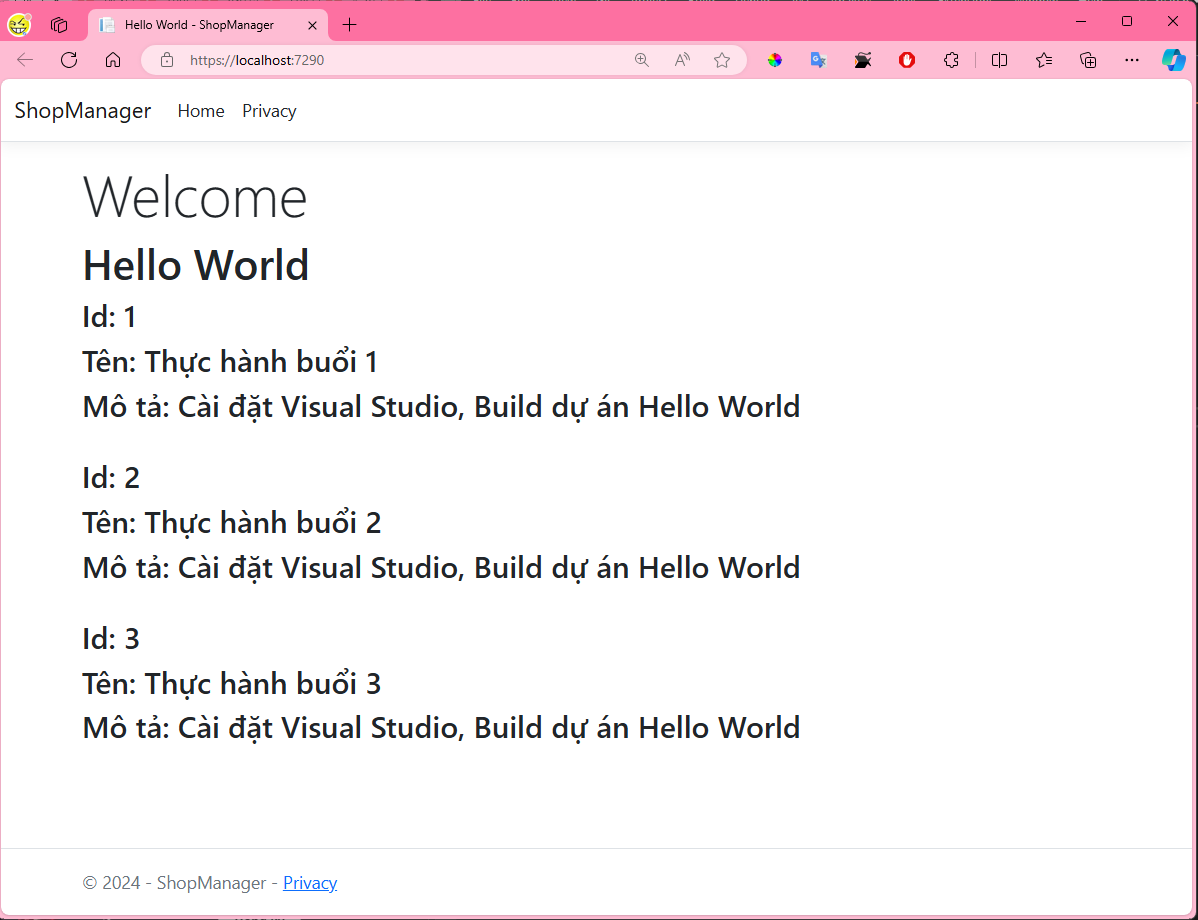
<h3>Mô tả: @item.Description</h3>

<br/>

</div>

}

</div>



Bài tập: Tạo List Model với 5 Model, với Model có Id là lẻ thì hiển thị Id và Name, ngược lại thì hiển thị Id và Description

Gợi ý: dùng @if()

@model IEnumerable<ShopManager.Models.ThucHanh1>

@{

ViewData["Title"] = "Hello World";

}

<div>

<h1 class="display-4">Welcome</h1>

<h1>Hello World</h1>

@foreach (var item in @Model)

{

<div style="border-style: solid">

<h3>Id: @item.Id</h3>

@if (item.Id % 2 == 1)

{

<h3>Tên: @item.Name</h3>

}

else

{

<h3>Mô tả: @item.Description</h3>

}

</div>

<br />

}

</div>

===The End===