### **Cách tạo Class trong Models**

### 

### 

### 

### 

### 

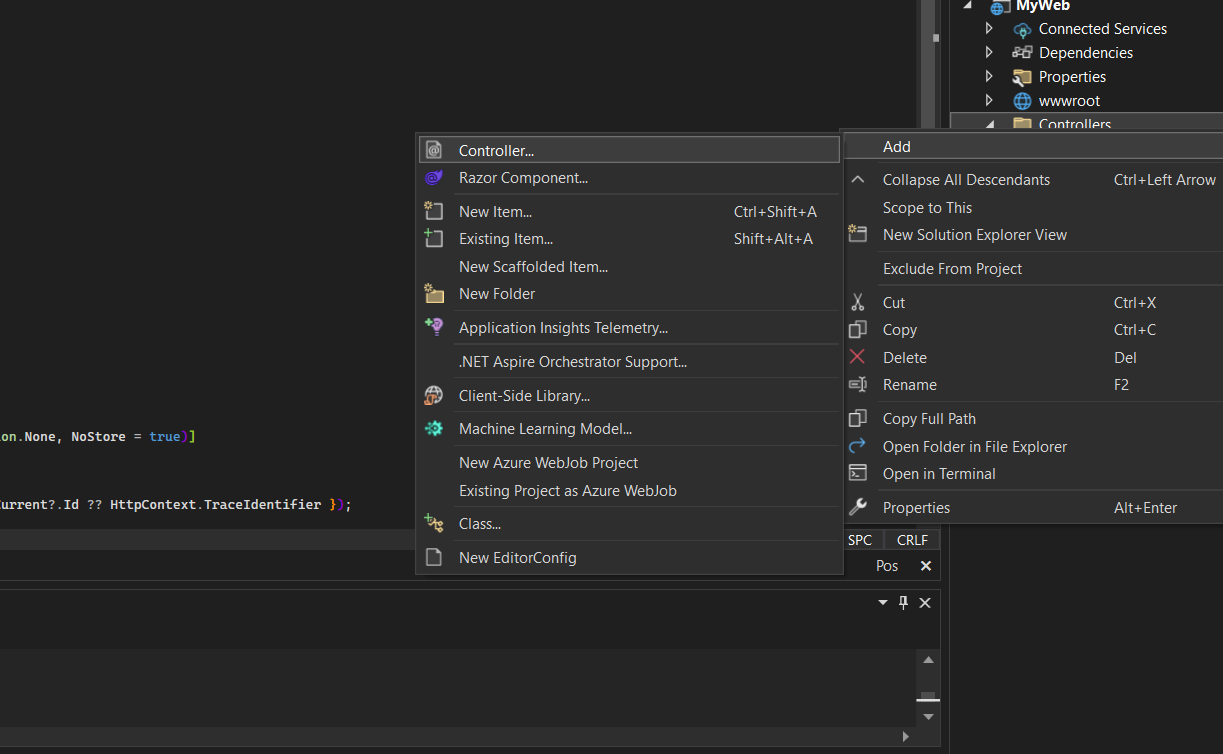
### 

### 

### 

### 

### **cách tạo Controller**

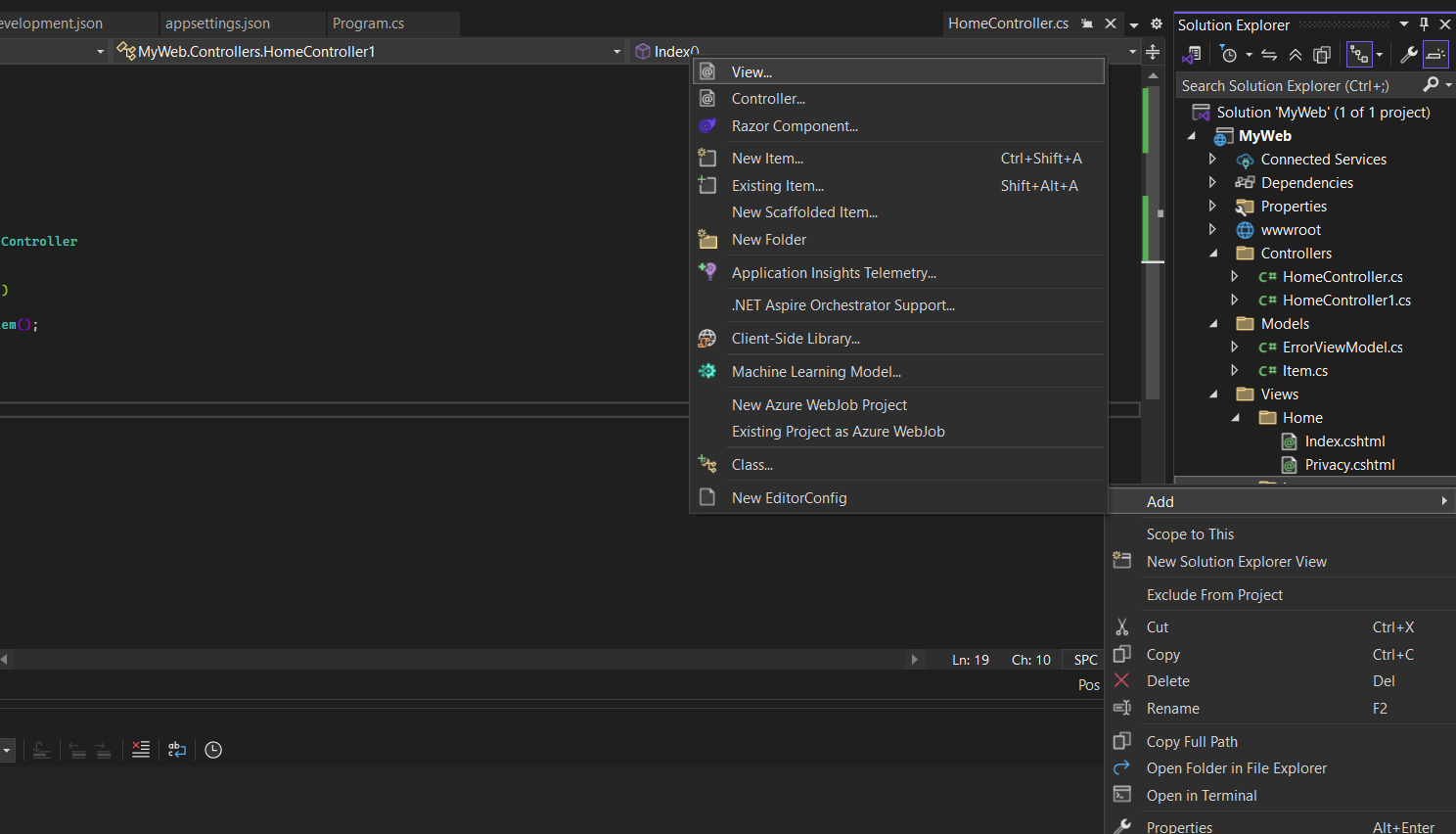


### 

### 

### 

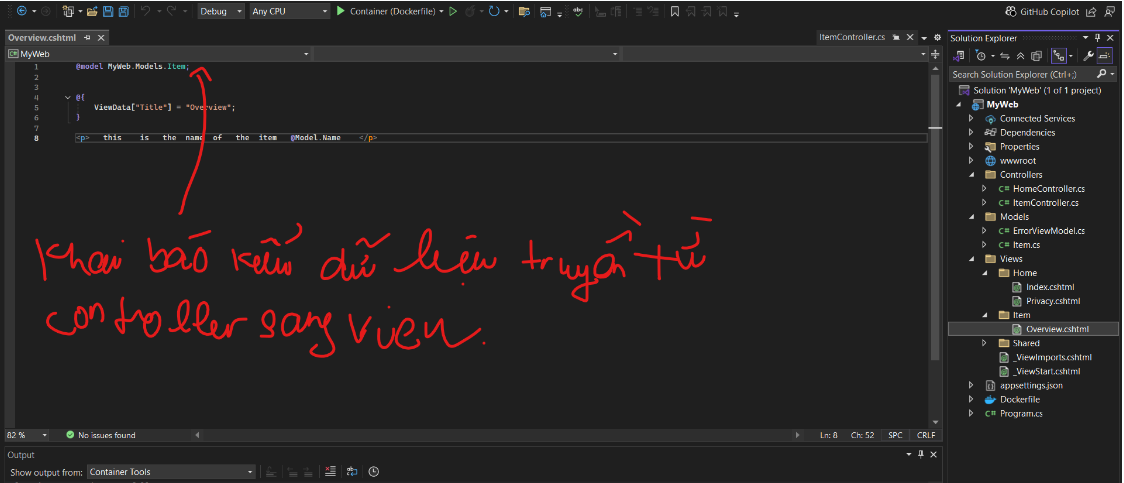
### **Cách tạo View**



### 

Trong ASP.NET MVC, một View chỉ có thể khai báo một kiểu dữ liệu với @model duy nhất.

Trong ASP.NET MVC Một action chỉ cho ra 1 view

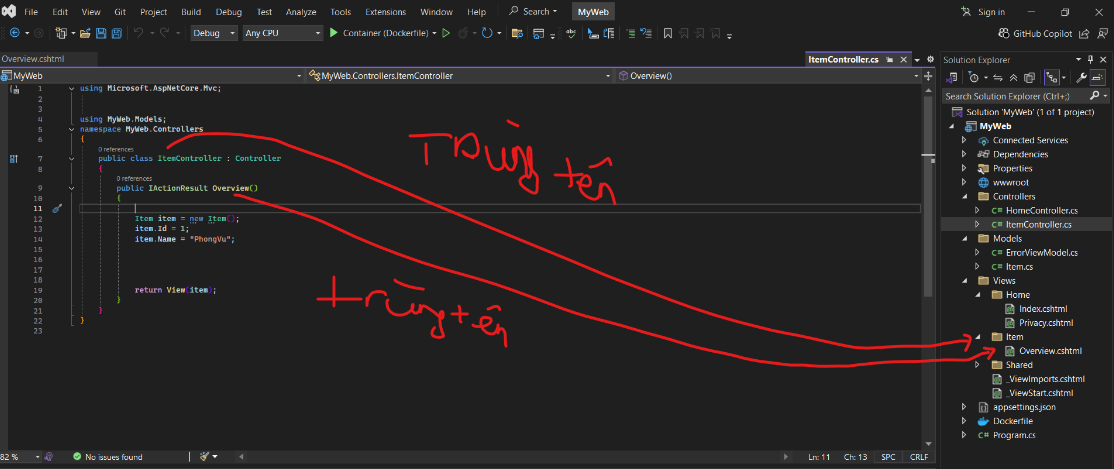


### 

* {controller=Home}/{action=Index}/{id?}") thứ tự đường link truy cập vào trang view 👍

**một route (đường dẫn)** trong ASP.NET MVC, dùng để xác định **cách URL được ánh xạ tới Controller và Action** sau đó trả kết quả cho view

### 



Phân biết return View vs return Content

### **1️⃣ return View()**

* **Cần phải có file View** (thường là .cshtml)  
  + Ví dụ: Action Overview() trong ItemController → View mặc định: /Views/Item/Overview.cshtml
* **Render bằng Razor**
  + Có thể dùng @Model để hiển thị dữ liệu từ Controller
* **Có thể dùng layout, partial view, helper…**
* **Dữ liệu** gửi từ Controller → Model / ViewModel

public IActionResult Overview()

{

var item = db.Items.Find(1);

return View(item); // Phải có Overview.cshtml

}

### **2️⃣ return Content()**

* **Không cần file View**
* Trả trực tiếp **chuỗi text hoặc HTML**
* Không có Razor, layout, hay binding
* Dùng **nhanh chóng** để debug hoặc trả text/HTML tạm

public IActionResult Test()

{

return Content("<h1>Hello World</h1>", "text/html"); // Không cần file View

}

Trong **ASP.NET MVC**, **Razor** là **một cú pháp (syntax) để viết code C# trực tiếp trong file View (.cshtml)**. Nó giúp **kết hợp HTML và C# một cách liền mạch**, rất tiện cho việc hiển thị dữ liệu từ Model lên giao diện

=> Cái gì là **HTML** 👉 viết bình thường.

Cái gì là **C# code** 👉 thêm **@** phía trước để Razor hiểu.

cú pháp : @{

string message = "hello world";

}

<p> @message </p>

### 

### **Ví dụ cụ thể:**

#### **🔹 1. Model**

namespace MyApp.Models

{

public class Item

{

public int Id { get; set; }

public string Name { get; set; }

}

}

#### **🔹 2. Controller**

public class HomeController : Controller

{

public IActionResult Index()

{

var item = new Item { Id = 1, Name = "Laptop" };

return View(item); // Truyền model sang view

}

}

#### **🔹 3. View (Index.cshtml)**

@model MyApp.Models.Item <!-- Khai báo kiểu dữ liệu -->

<h2>Item Info</h2>

<p>ID: @Model.Id</p>

<p>Name: @Model.Name</p>

### **✅ Chức năng:**

* @model giúp View biết kiểu dữ liệu mà nó sẽ nhận (Item).
* @Model (chữ **M** viết hoa) được dùng để truy cập đối tượng đó trong View.

👉 Nói ngắn gọn:  
 @model MyApp.Models.Item = “View này sẽ làm việc với **một đối tượng Item**”.