

# Bài thực hành 01

## Tạo ứng dụng NetCore đầu tiên

### 1. Mục tiêu

- Tải và cài đặt Visual Studio 2017 ( Bản 2019 tương tự )
- Sinh viên có thể tạo ứng dụng NetCore
- Hiểu được cấu trúc file và thư mục của dự án
- Có thể chỉnh sửa được một số cấu hình cơ bản cho ứng dụng

### 2. Bài thực hành Step by Step

#### Bài 1: Hướng dẫn cài đặt phần mềm Visual Studio 2017

##### Bước 1: Tải bản cài đặt

Cách 1: Tải từ microsoft (Nếu cách này khó quá thì xem cách 2)

Lên google tìm visual studio community 2017 15.0

visual studio community 2017

About 117,000,000 results (0.47 seconds)

visualstudio.microsoft.com › vs › older-downloads ▾

**Visual Studio Older Downloads - 2017, 2015 & Previous**

link này

Expand All Collapse All

2017

Visual Studio 2017 and other Products

To download any product from the following list, click the download button and log in with your Visual Studio Subscription account when prompted. If you don't have a Visual Studio Subscription, you can create one for free by clicking on "Create a new Microsoft account" on the login page.

Download

Màn hình tiếp theo, bạn cần đăng nhập vào tài khoản của Microsoft để tải, nếu chưa có thì nên đăng ký một tài khoản



Sau khi có tài khoản và đăng nhập vào, im đến phiên bản Visual Studio Community 2017 (version 15.0)

Visual Studio Community 2017 (version 15.0)

Click tải

No key required Info

Release date: 11/Dec/2019

mul

English

EXE

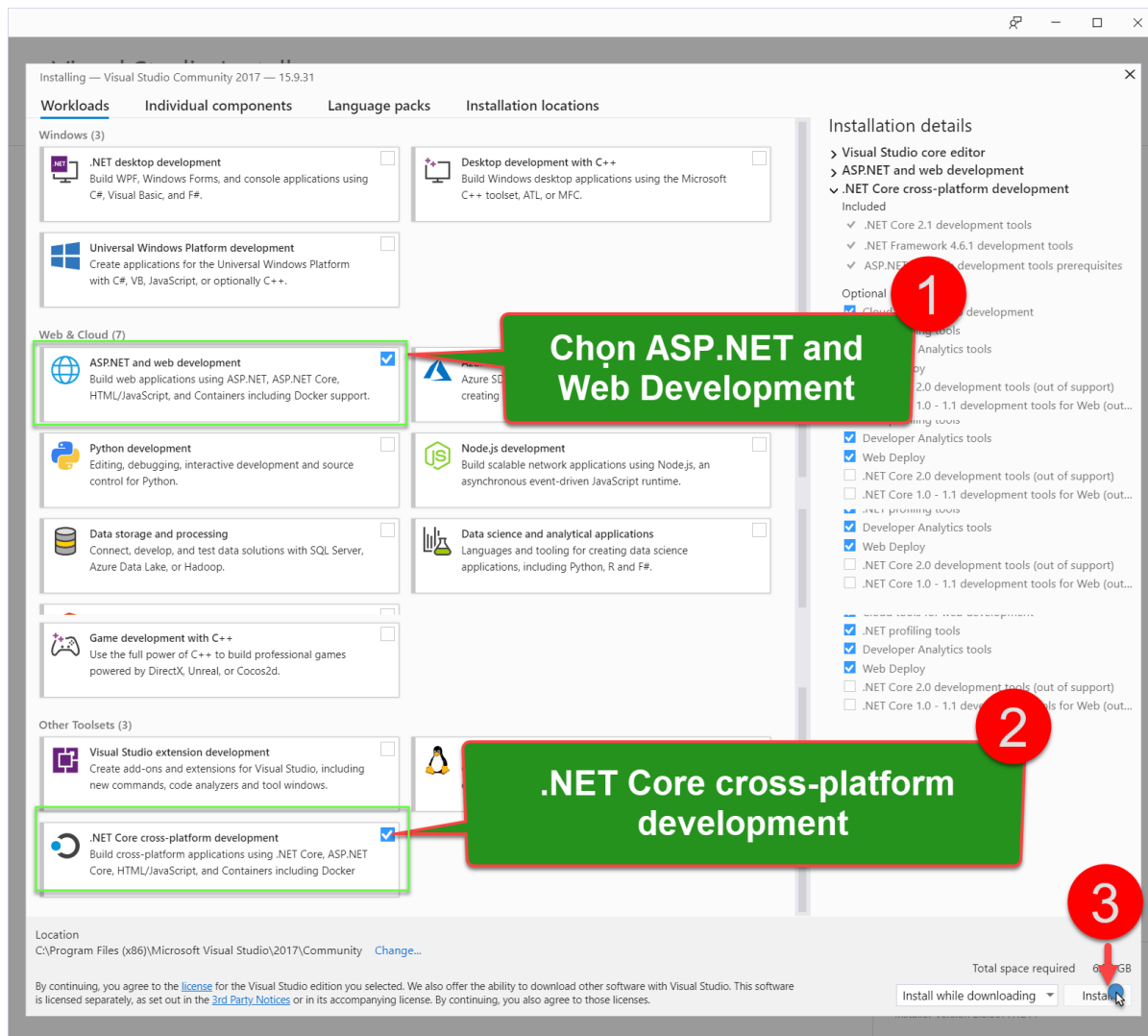
Download

**Cách 2: Tải bản đã tải sẵn trên google drive theo link sau**

<https://drive.google.com/drive/folders/1AZb9ytECGsFSiEUnyJ4aKNxtNOp2NLYs>

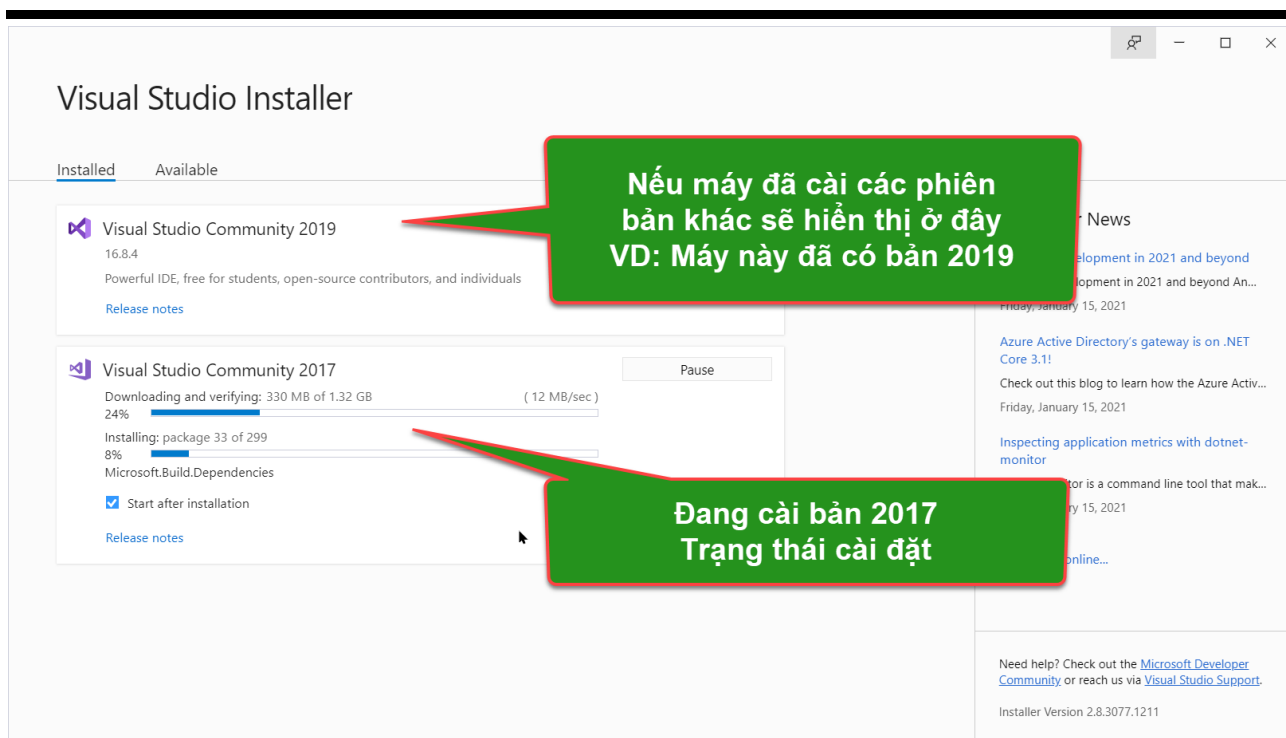
**Bước 2: Tiến hành cài đặt theo các bước sau,**

Mở file vs\_community.exe lên, màn hình load lên có dạng như hình sau đây, hãy lựa chọn như chỉ dẫn trong hình

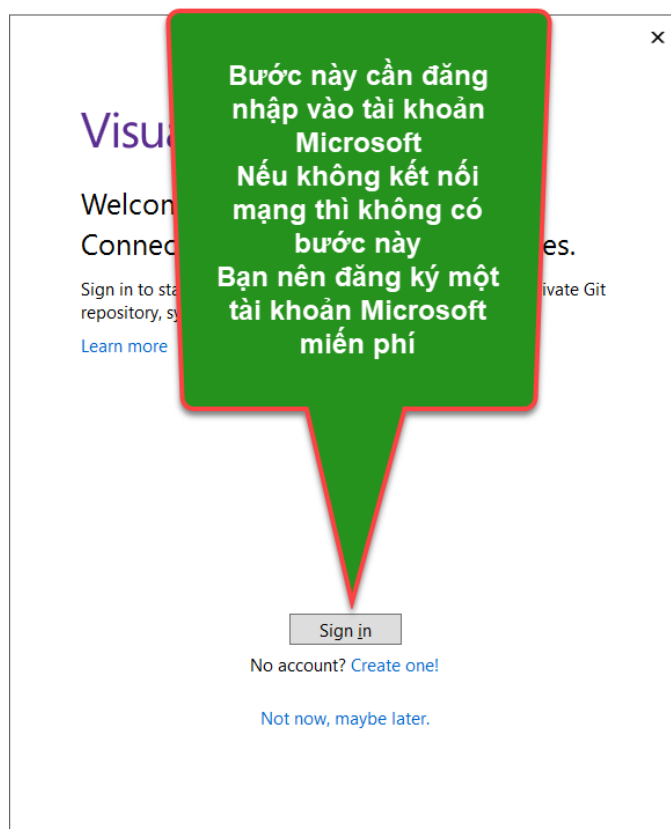


Màn hình tiếp theo sau khi click install sẽ là trạng thái cài đặt

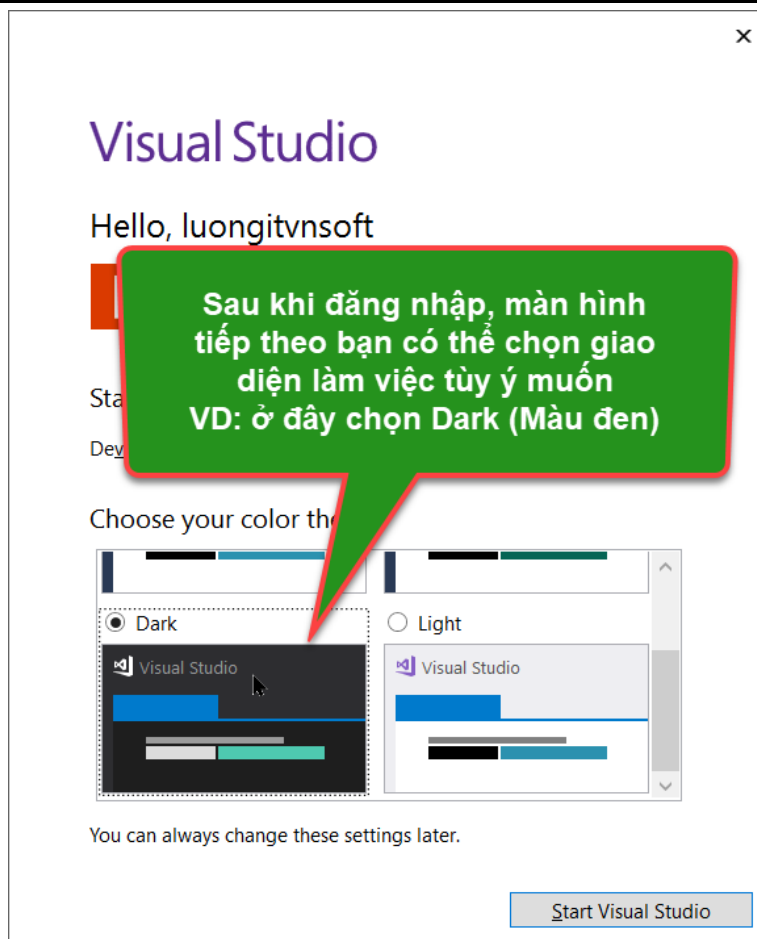




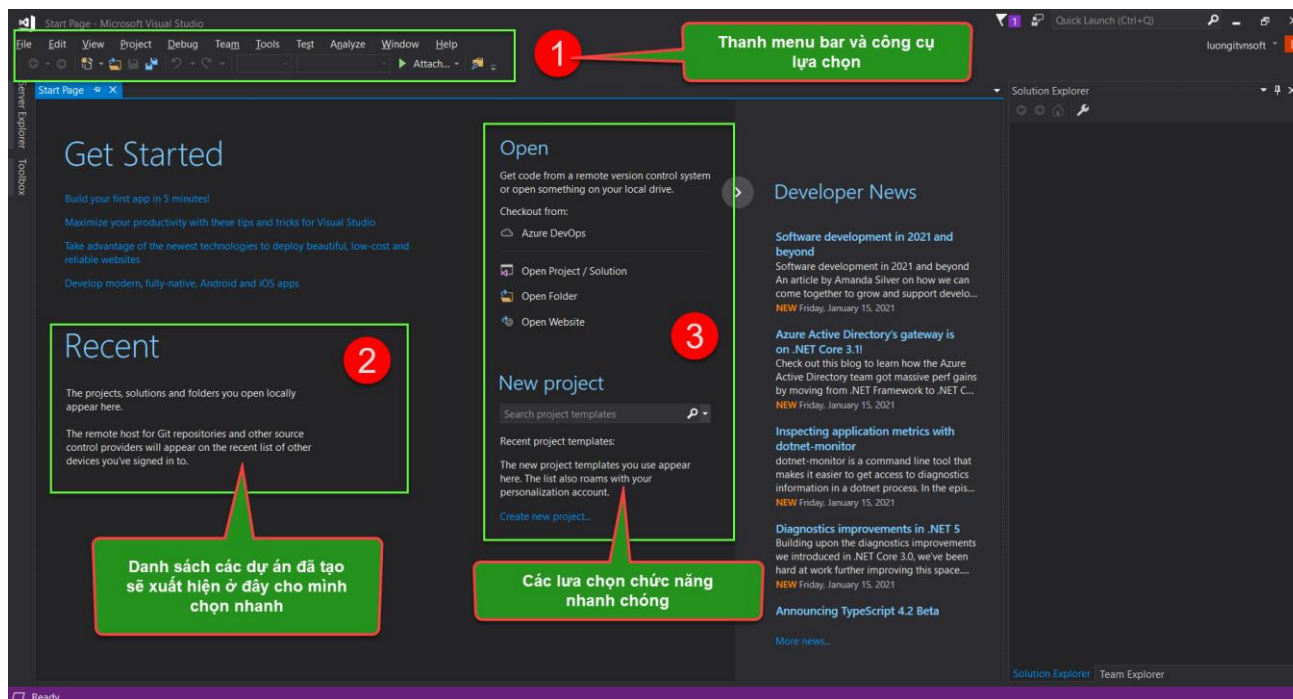
Màn hình tiếp theo sẽ là màn hình đăng nhập vào tài khoản Microsoft để bắt đầu cấu hình và sử dụng (Nếu không kết nối mạng thì không có bước này )



Qua màn tiếp theo, cấu hình giao diện lập trình ban đầu

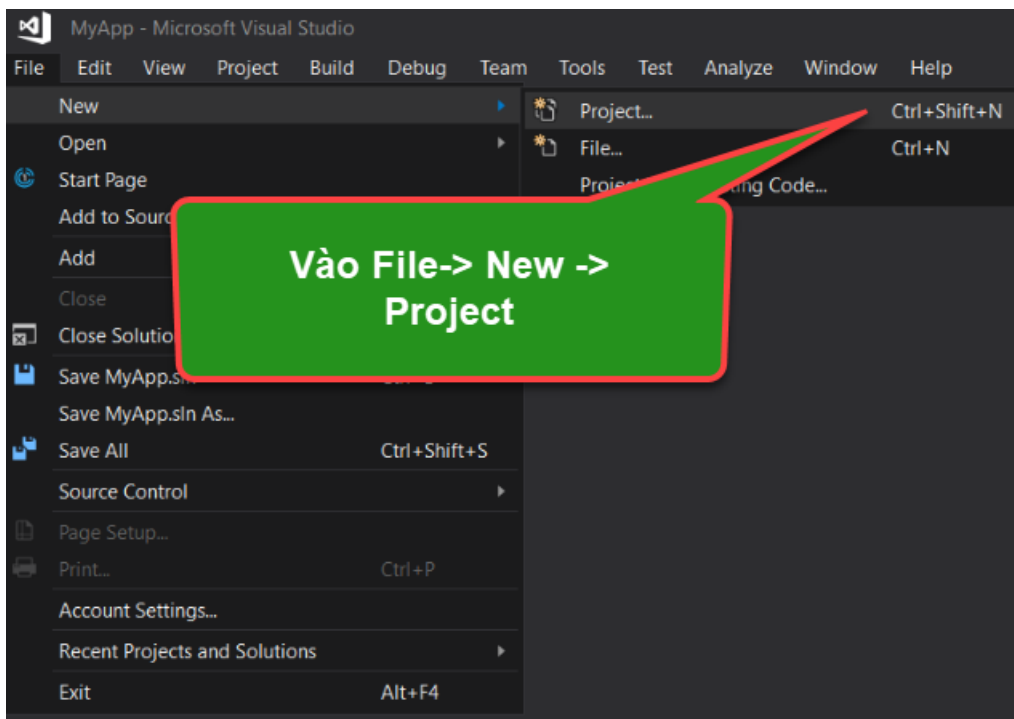


Sau khi vào màn hình làm việc đầu tiên như sau

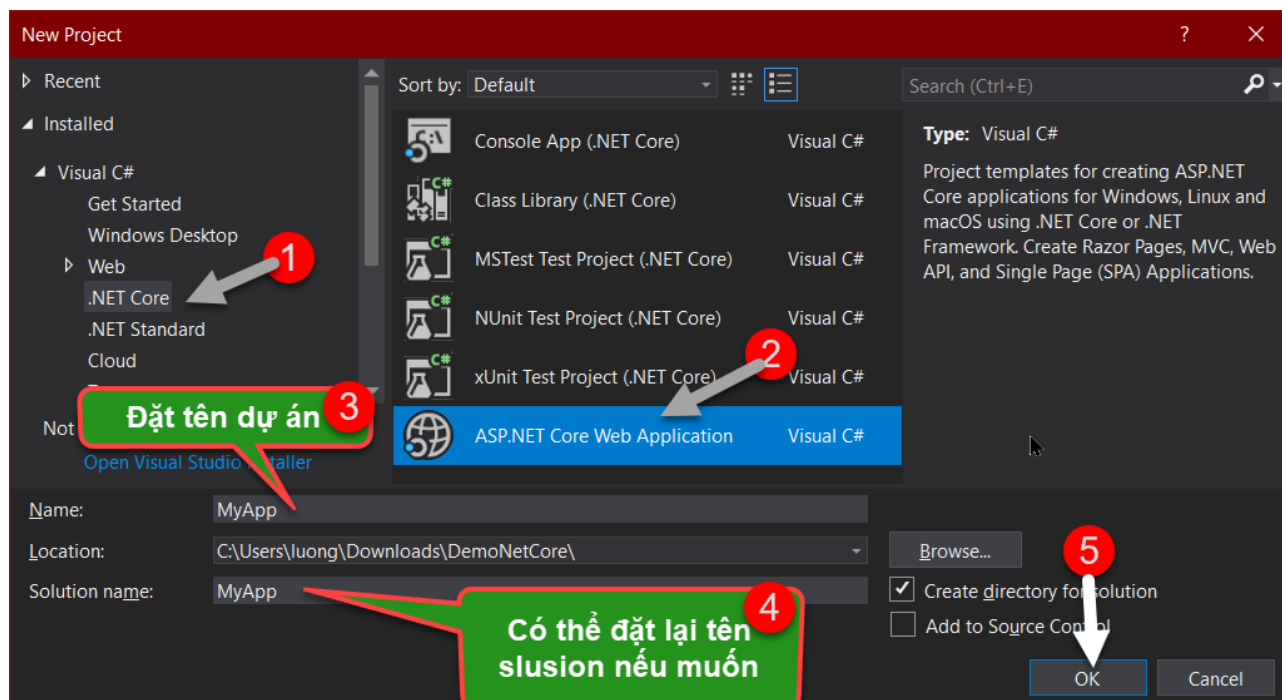


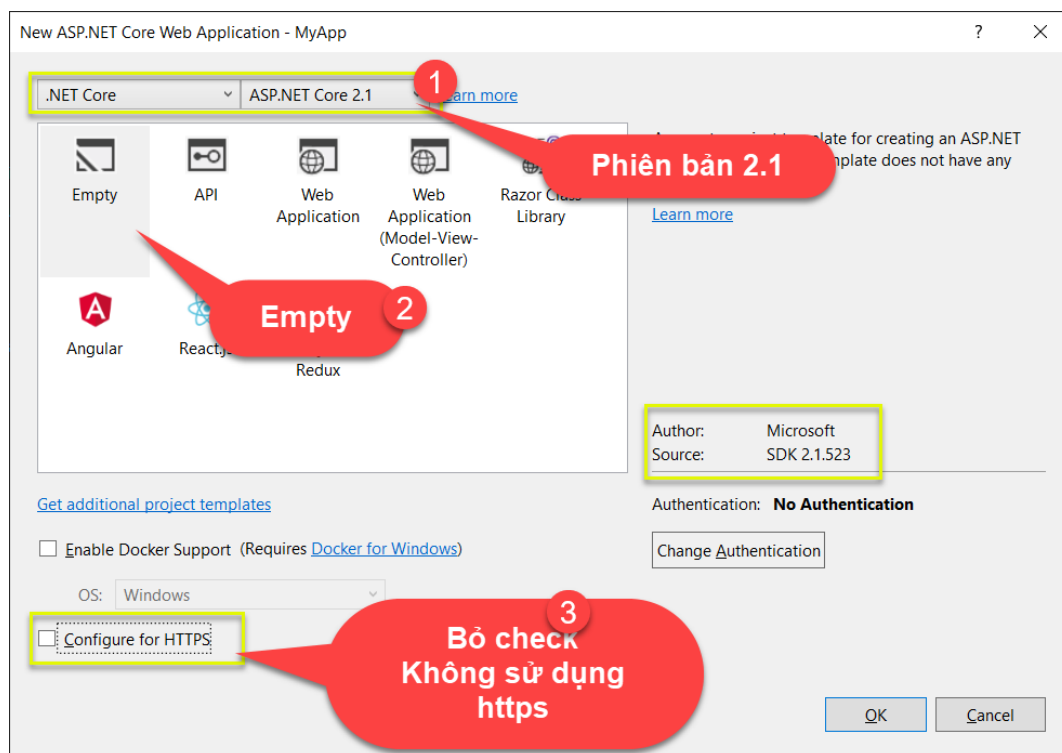
## Bài 2: Tạo ứng dụng Netcore cơ bản đầu tiên

**Bước 1:** Mở phần mềm Visual Studio 2017 sau đó tạo project mới bằng cách vào menu File -> New -> Project hoặc tổ hợp phím CTRL + SHIFT + N

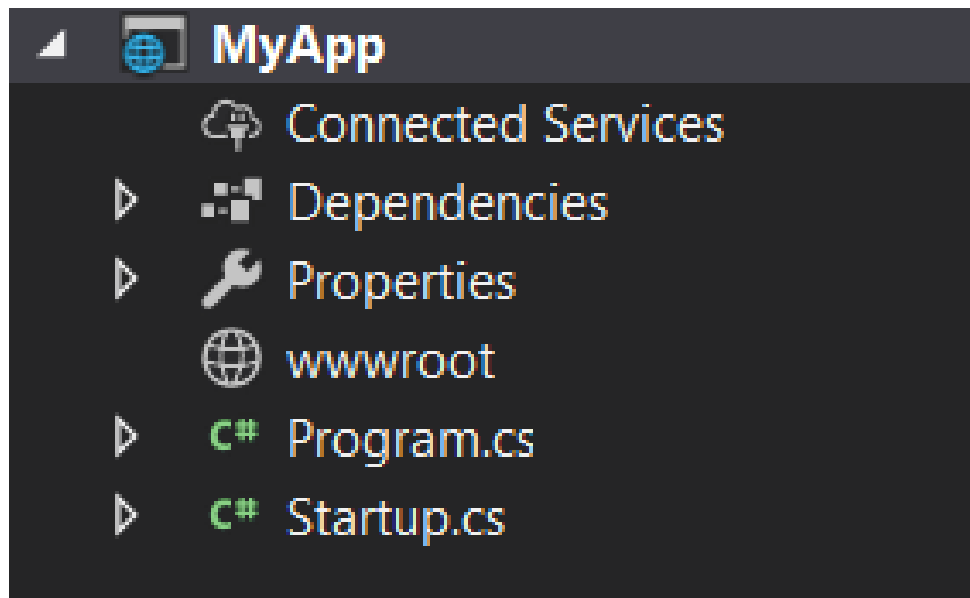


**Bước 2:** Tại màn hình tiếp theo chọn **ASP.NET Core Web Application**, đặt tên là **MyApp**



**Bước 3: Chọn template là Empty**

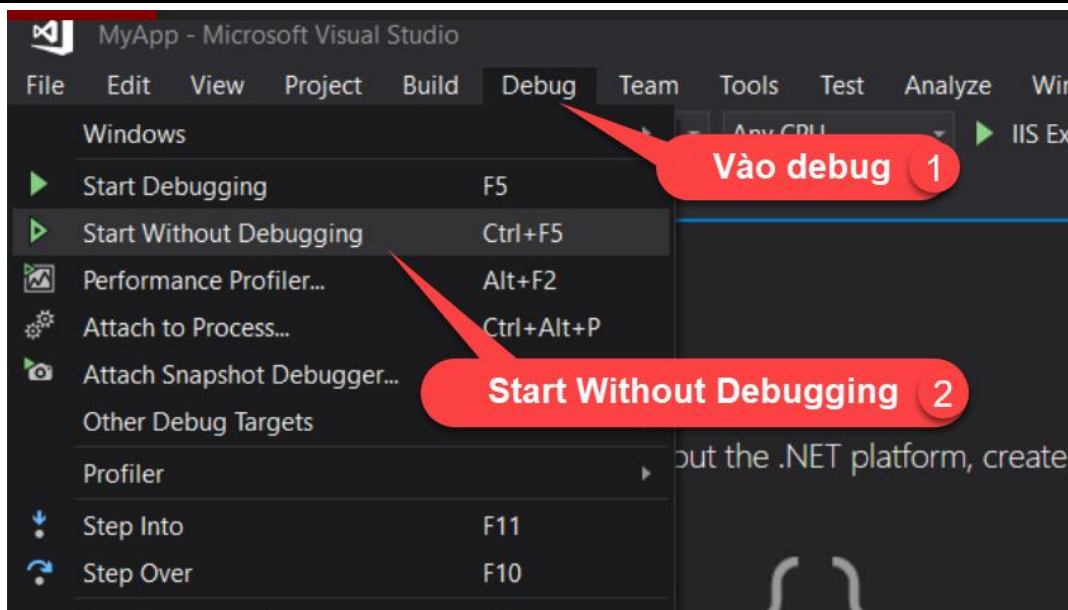
Nhấn Ok và được dự án có cấu trúc như sau



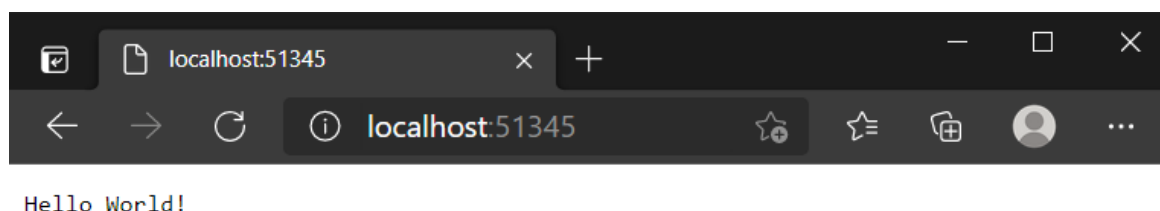
**Bước 4:** Nhấn CTRL + F5 để chạy dự án ở chế độ không debug

( Có thể vào menu Debug-> chọn Start Without Debugging )



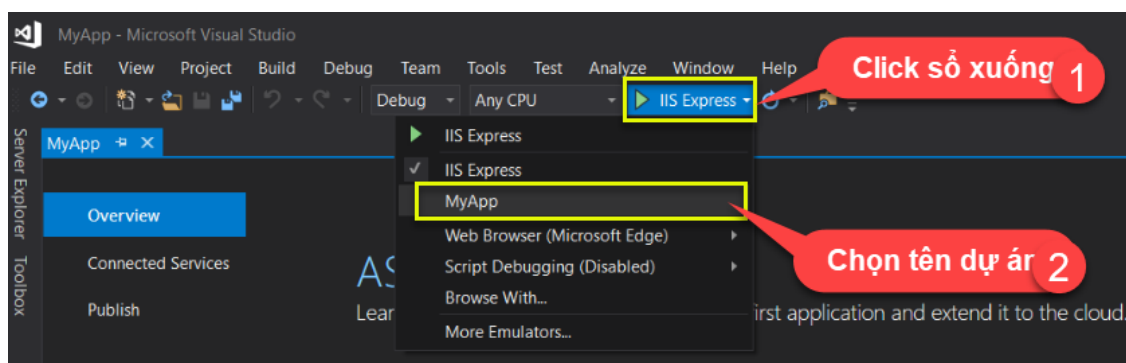


Kết quả là ứng dụng sẽ tự mở trên trình duyệt mặc định có dạng

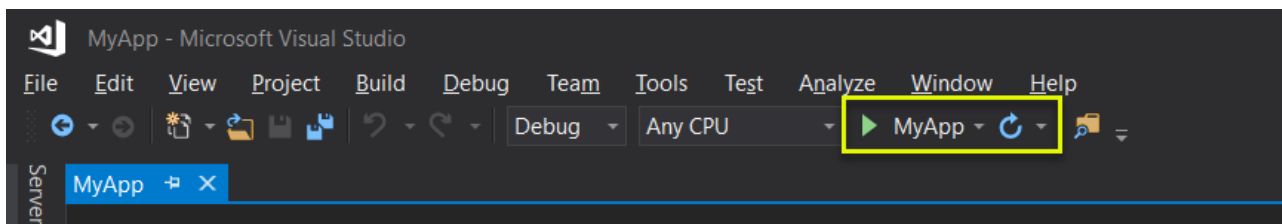


## Bài tập 2: Tìm hiểu về cấu trúc dự án và chỉnh sửa các cấu hình cơ bản

**Bước 1:** Thay đổi server mặc định là IIS sang **MyApp** là thư mục dự án



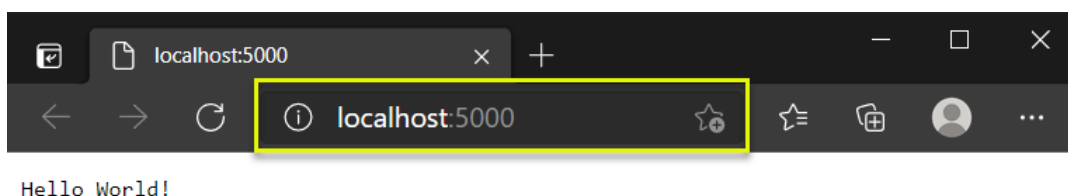
Sau khi chọn tên dự án như bước trên



**Bước 2:** CTRL + F5 để chạy dự án thì thấy bây giờ dự án sẽ khởi động với cổng mặc định là 5000 và có một màn hình command xuất hiện khi dự án khởi động

```
MyApp
Hosting environment: Development
Content root path: D:\Project\C#\NetCore\MyApp\MyApp
Now listening on: http://localhost:5000
Application started. Press Ctrl+C to shut down.
info: Microsoft.AspNetCore.Hosting.Internal.WebHost[1]
      Request starting HTTP/1.1 GET http://localhost:5000/
info: Microsoft.AspNetCore.Hosting.Internal.WebHost[2]
      Request finished in 13.6176ms 200
info: Microsoft.AspNetCore.Hosting.Internal.WebHost[1]
      Request starting HTTP/1.1 GET http://localhost:5000/favicon.ico
info: Microsoft.AspNetCore.Hosting.Internal.WebHost[2]
      Request finished in 0.2716ms 200
```

**Kết quả**



**Bước 3:** Thay đổi cổng mặc định ban đầu là 5000 sang cổng khác VD 3000

Mở file **launchSettings.json** trong thư mục **Properties**, tìm đến đoạn code có dạng sau

```
"MyApp": {
  "commandName": "Project",
  "launchBrowser": true,
```



```
"applicationUrl": "http://localhost:5000",  
"environmentVariables": {  
    "ASPNETCORE_ENVIRONMENT": "Development"  
}  
}
```

Bạn có thể thay đổi cổng 5001 hoặc 5000 sang các cổng khác nếu muốn

```
"MyApp": {  
    "commandName": "Project",  
    "launchBrowser": true,  
    "applicationUrl": "http://localhost:3000",  
    "environmentVariables": {  
        "ASPNETCORE_ENVIRONMENT": "Development"  
    }  
}
```

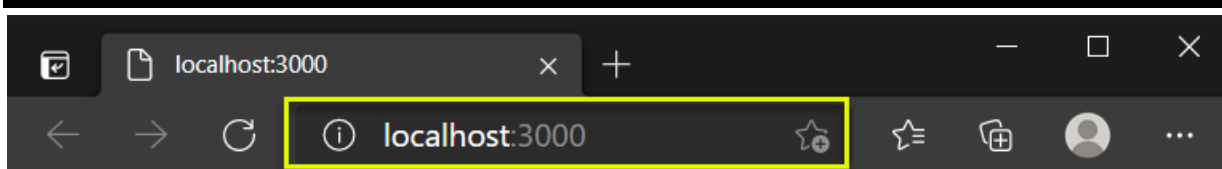
**Chú ý:** Bạn cũng có thể thay đổi cho iis Settings tương tự như trên

```
"iisSettings": {  
    "windowsAuthentication": false,  
    "anonymousAuthentication": true,  
    "iisExpress": {  
        "applicationUrl": "http://localhost:1200",  
        "sslPort": 0  
    }  
}
```

Hãy chạy lại dự án để xem kết quả

```
MyApp  
Hosting environment: Development  
Content root path: D:\Project\C#\NetCore\MyApp\MyApp  
Now listening on: http://localhost:3000  
Application started. Press Ctrl+C to shut down.  
info: Microsoft.AspNetCore.Hosting.Internal.WebHost[1]  
      Request starting HTTP/1.1 GET http://localhost:3000/  
info: Microsoft.AspNetCore.Hosting.Internal.WebHost[2]  
      Request finished in 17.4418ms 200  
info: Microsoft.AspNetCore.Hosting.Internal.WebHost[1]  
      Request starting HTTP/1.1 GET http://localhost:3000/favicon.ico  
info: Microsoft.AspNetCore.Hosting.Internal.WebHost[2]  
      Request finished in 0.1697ms 200
```

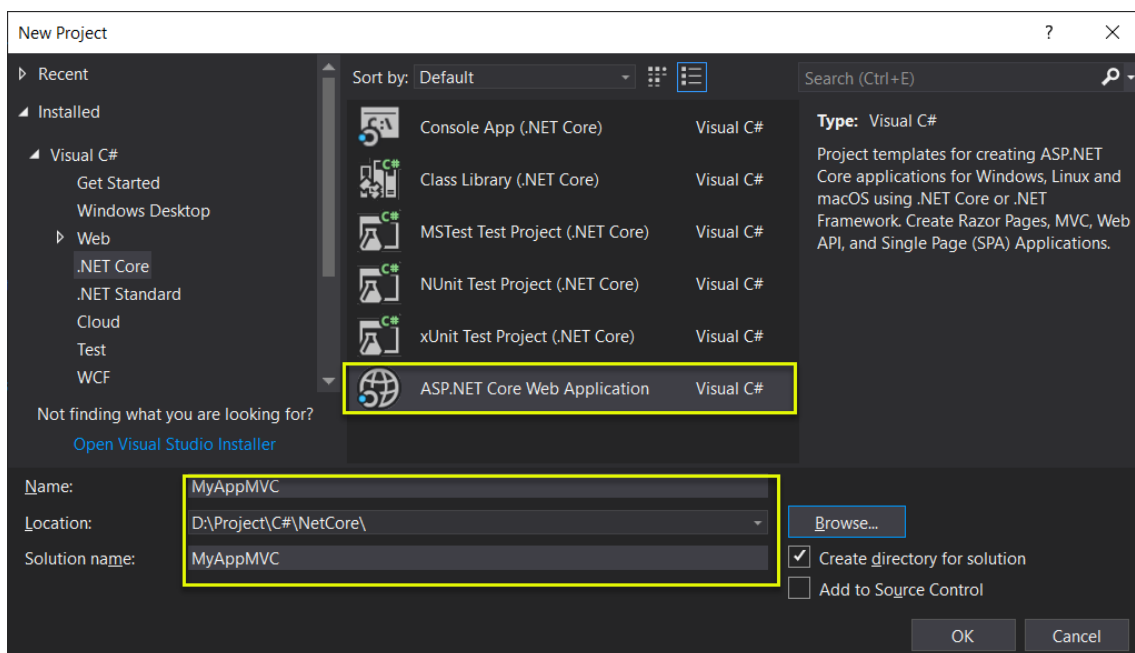




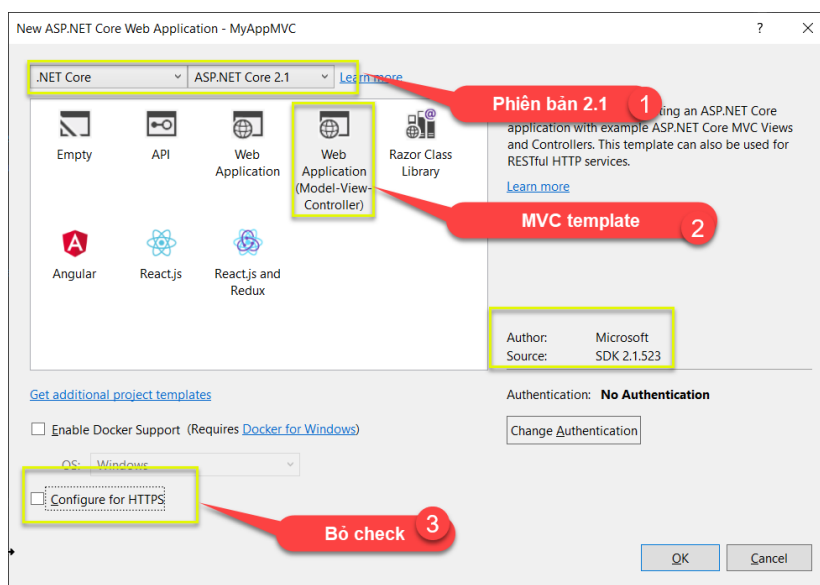
Hello World!

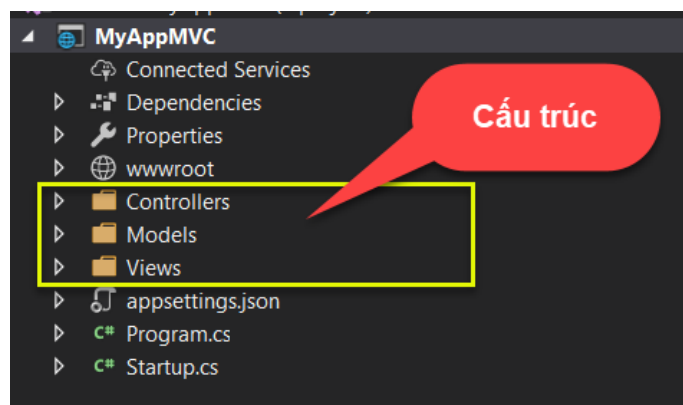
### Bài 3: Tạo dự án NetCore MVC template

**Bước 1:** Tiến hành tạo project như bài 1 đặt tên project là **MyAppMVC**

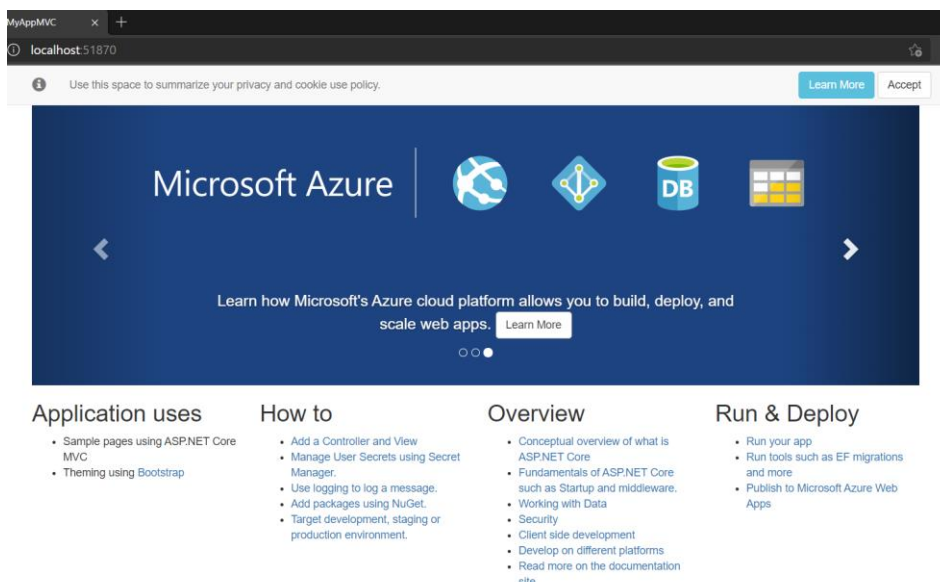


**Bước 2:** Chọn template là Web Application (Model, View, Controller)



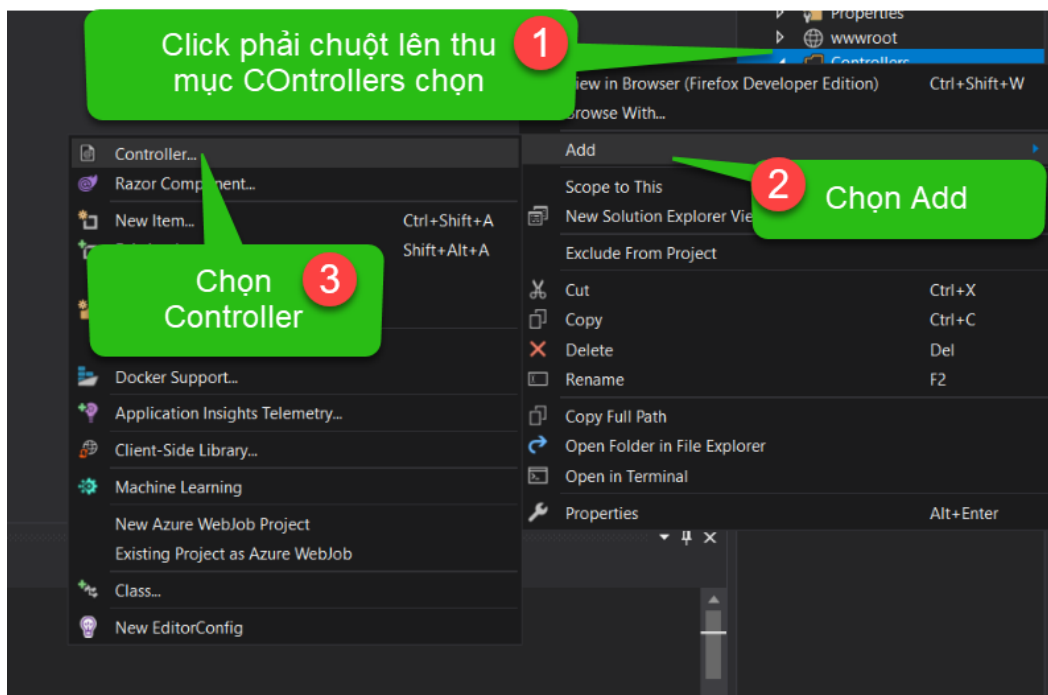
**Bước 2: Cấu trúc file và thư mục**

**CTRL F5 chạy dự án và Kết quả tương tự như hình sau**

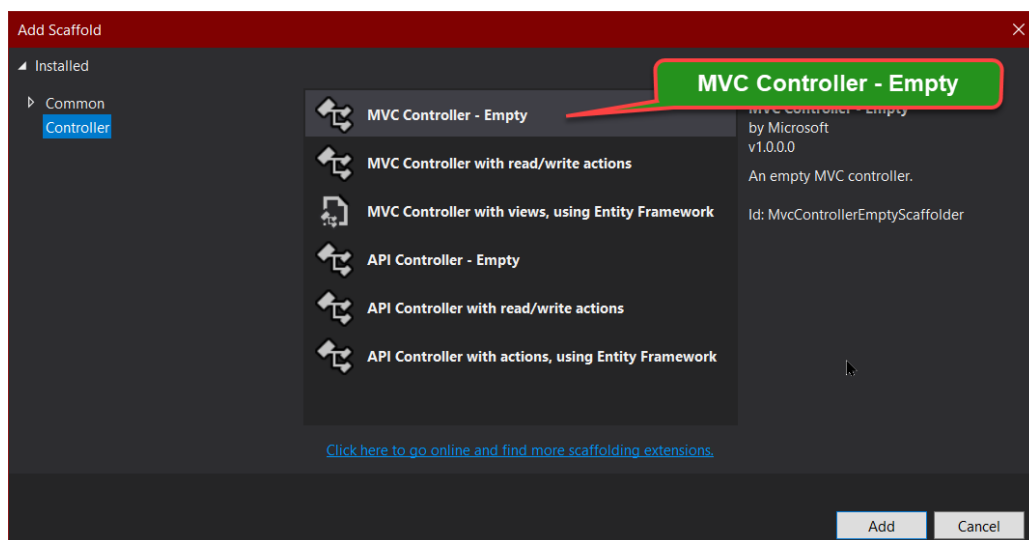
**Bài 3: Tạo Controller và view cho dự án Net Core MVC**

**Bước 1: Click phải chuột lên thư mục Controllers chọn **Add** và chọn **Controller****

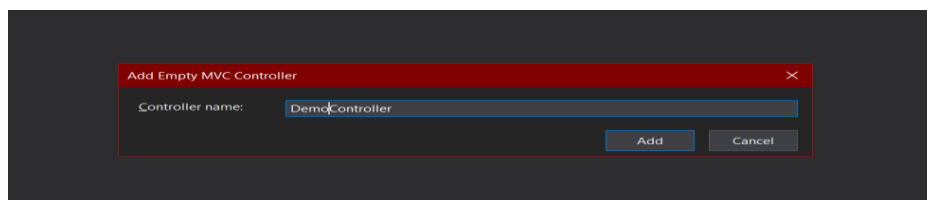




**Bước 2:** Tại màn hình tiếp theo chọn **MVC Controller - MVC**



**Bước 3:** Tại Bước tiếp theo đặt tên cho controller là DemoController



Code của **DemoController** có dạng

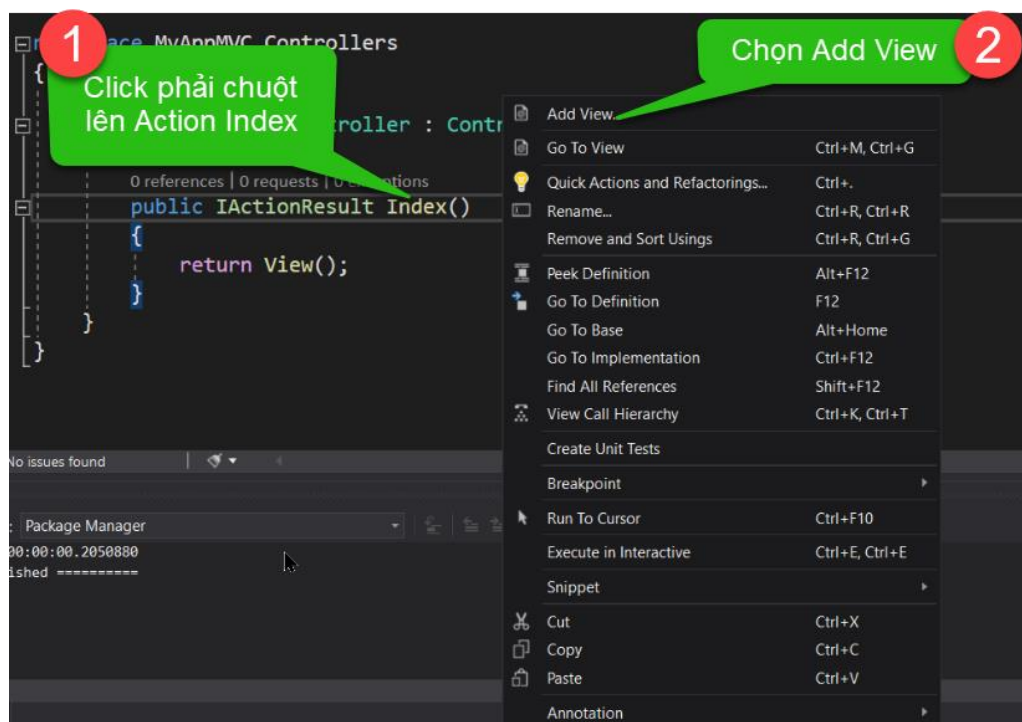
```
using Microsoft.AspNetCore.Mvc;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Threading.Tasks;

namespace MyAppMVC.Controllers
{
    0 references
    public class DemoController : Controller
    {
        0 references | 0 exceptions
        public IActionResult Index()
        {
            return View();
        }
    }
}
```

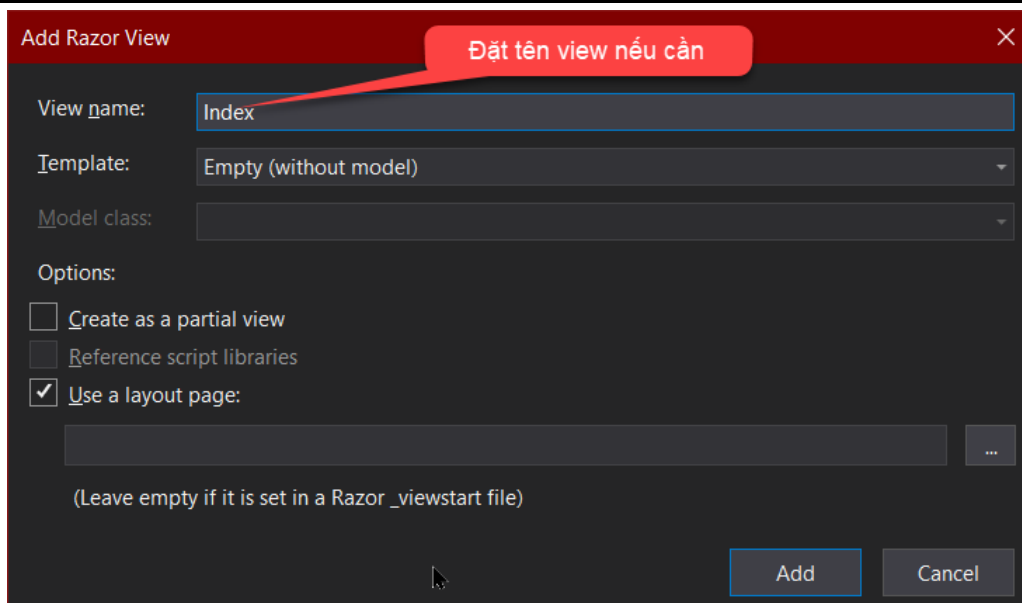
**DemoController**

#### Bước 4: Thêm view cho Action Index

1 => Click phải chuột lên action **Index** và 2 => chọn **Add view** như hình sau



#### Bước 5: Đặt tên view ( Thường để mặc định theo tên Action )



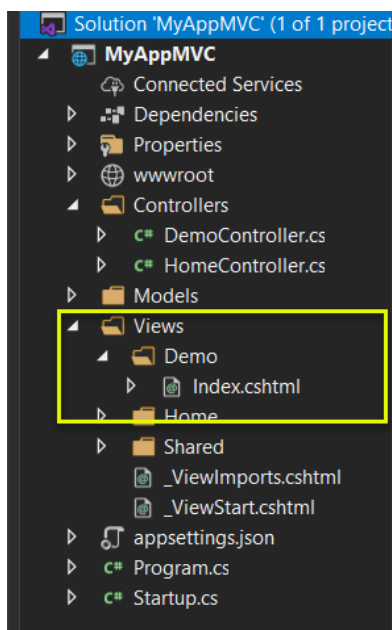
Nội dung file View có dạng sau

```
@{
    ViewData["Title"] = "Index";
}
```

<h2>Index</h2>

Nội dung view có thể chỉnh sửa tùy ý

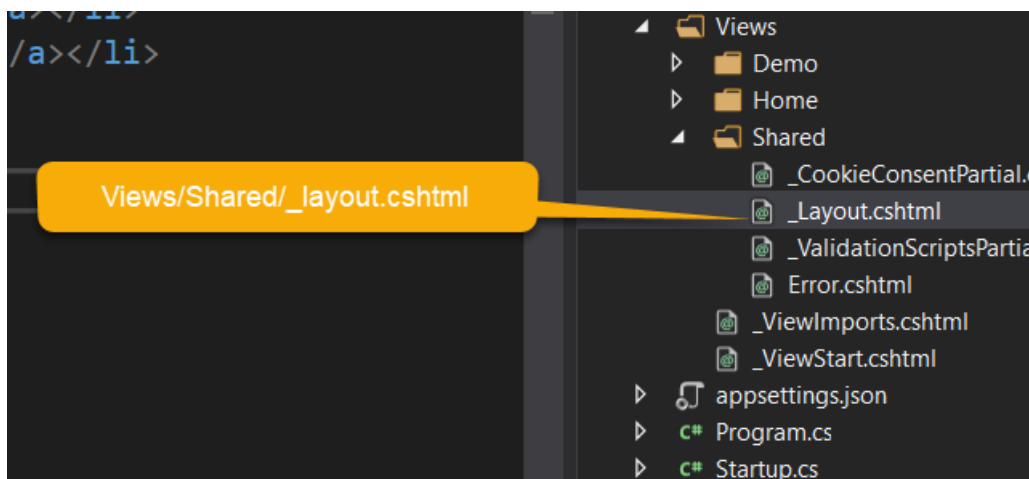
Trong thư mục Views của dự án, có thêm view vừa tạo, thư mục dự án có dạng sau



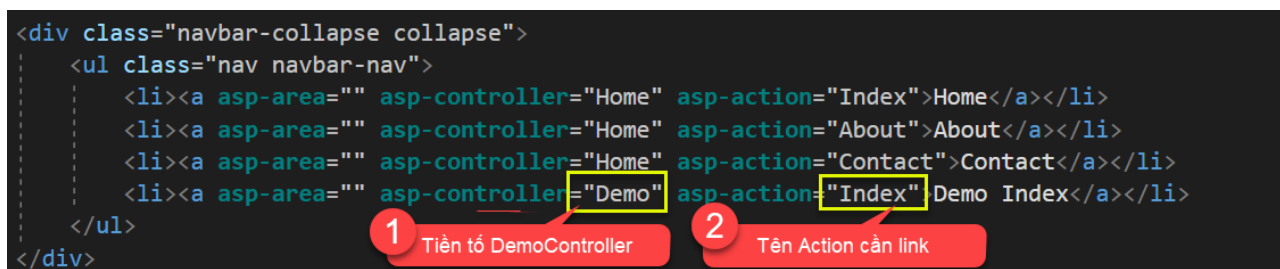
## Bước 6: Thêm link trên menu ngang



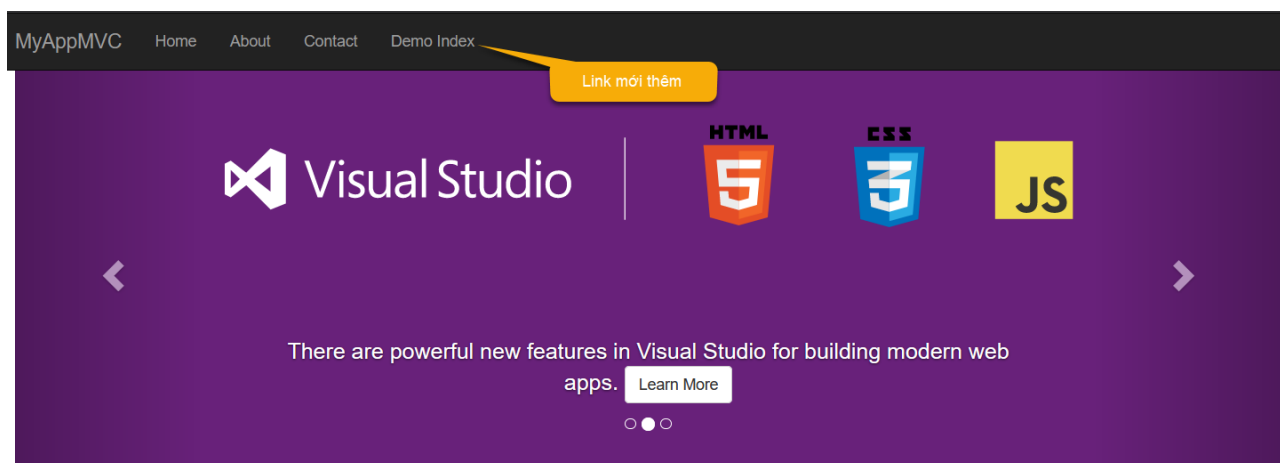
Mở file Views/Shared/\_layout.cshtml



Tìm đến đoạn code trở các link trên menu và thêm như sau



Kết quả



**Bước 7:** Vào sửa view trong thư mục (**Views/Demo/Index.cshtml**) nội dung tùy ý ví dụ

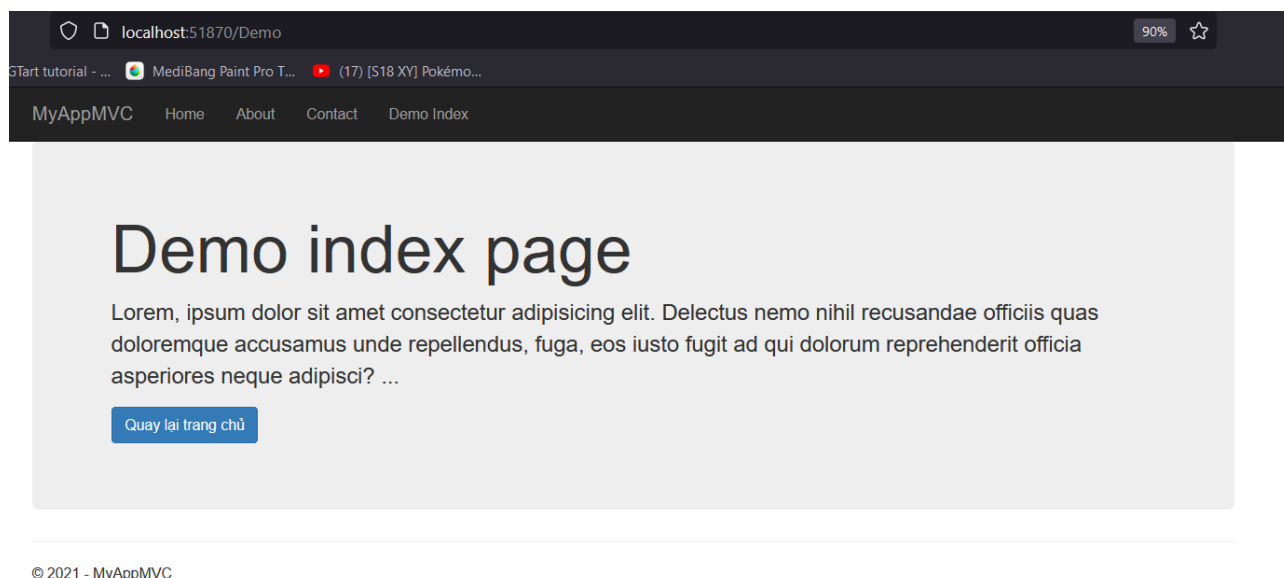
@{

```
ViewData["Title"] = "Index";
}

<div class="jumbotron">
  <div class="container">
    <h1>Demo index page</h1>
    <p>Lorem, ipsum dolor sit amet consectetur adipisicing elit. Delectus nemo nihil recusandae officiis quas doloremque accusamus unde repellendus, fuga, eos iusto fugit ad qui dolorum reprehenderit officia asperiores neque adipisci? ...</p>
    <p>
      <a asp-controller="Home" asp-action="Index" class="btn btn-primary">Quay lại trang chủ</a>
    </p>
  </div>
</div>
```

**Bước 8:** Build lại dự án nhấn CTRL + F5, Truy cập đường link để test

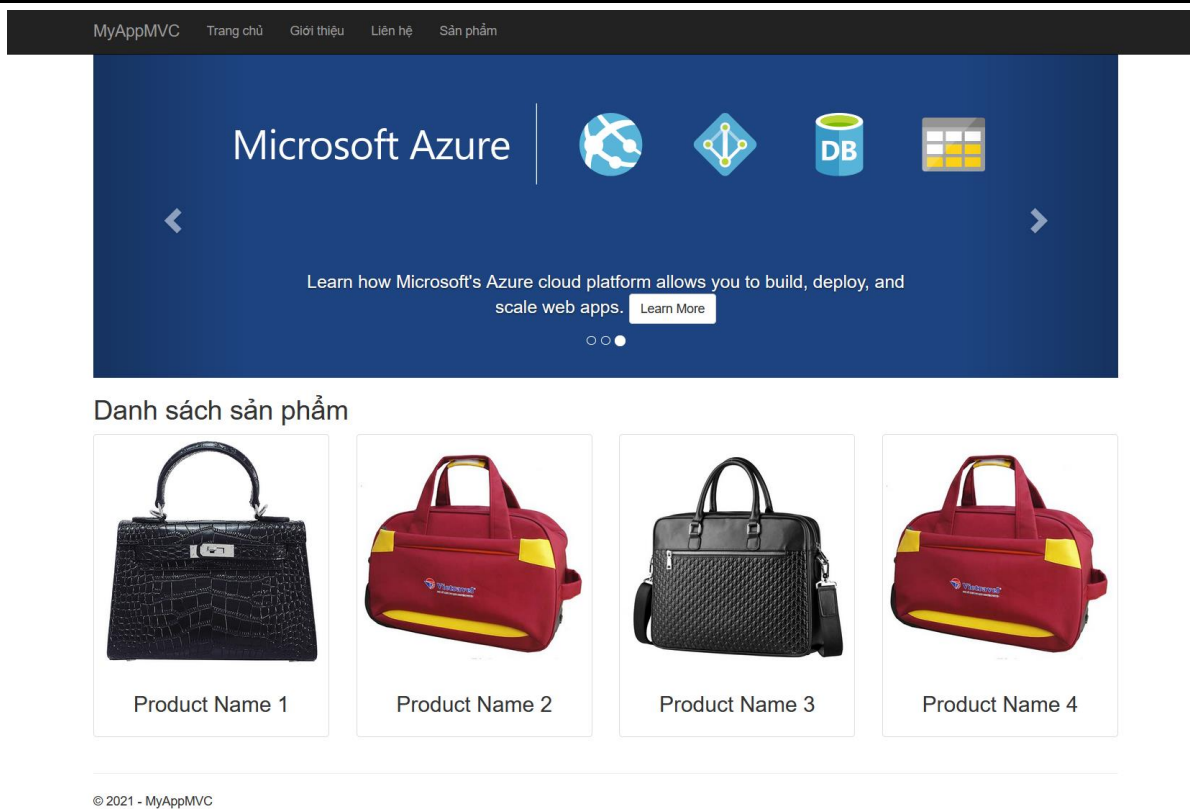
http://localhost:51870/Demo



## Bài tập tự làm

1. Sinh viên tạo dự án NetCore MVC với tên dự án theo ý muốn
2. Chỉnh sửa file view của action Index ( Index của HomeController ) có dạng sau





3. Tạo thêm controller là ProductController và tạo view Index có giao diện như sau

