**In the name of God**

**Learning ASP.NET Core MVC (.NET 5)**

**Mr. Dariush Tasdighi**

**+98-9121087461**

[**DariushT@GMail.com**](mailto:DariushT@GMail.com)

[**https://DTAT.ir**](https://DTAT.ir)

[**https://github.com/Dariush.Tasdighi**](https://github.com/Dariush.Tasdighi)

**Branch: STEP\_1**

1. **In visual studio 2019**
   1. **Create a new project**
      1. **ASP.NET Core Empty**
         1. **.NET 5.0**
         2. **Configure for HTTPS**
2. **Use Application instead of IIS Express**
3. **Fix Application**
4. **Fix appsettings.json**
5. **In Properties Folder** 🡪 **Fix launchSettings.json**
6. **Right Click on Application (Project) 🡪 Select Properties Menu Item 🡪 Startup object: Application.Program**
7. **Fix Program.cs**

using Microsoft.AspNetCore.Hosting;

using Microsoft.Extensions.Hosting;

namespace Application

{

public class Program : object

{

public Program() : base()

{

}

public static void Main(string[] args)

{

// Run -> using Microsoft.Extensions.Hosting;

CreateHostBuilder(args).Build().Run();

}

public static Microsoft.Extensions.Hosting.IHostBuilder CreateHostBuilder(string[] args)

{

var hostBuilder =

Microsoft.Extensions.Hosting.Host.CreateDefaultBuilder(args)

.ConfigureWebHostDefaults(webBuilder =>

{

// UseStartup -> using Microsoft.AspNetCore.Hosting;

webBuilder.UseStartup<Startup>();

});

return hostBuilder;

}

}

}

1. **Fix Startup.cs**

using Microsoft.AspNetCore.Http;

using Microsoft.AspNetCore.Builder;

using Microsoft.Extensions.Hosting;

namespace Application

{

public class Startup : object

{

public Startup() : base()

{

}

public void ConfigureServices

(Microsoft.Extensions.DependencyInjection.IServiceCollection services)

{

}

public void Configure

(Microsoft.AspNetCore.Builder.IApplicationBuilder app,

Microsoft.AspNetCore.Hosting.IWebHostEnvironment env)

{

// IsDevelopment -> using Microsoft.Extensions.Hosting;

if (env.IsDevelopment())

{

// UseDeveloperExceptionPage -> using Microsoft.AspNetCore.Builder;

app.UseDeveloperExceptionPage();

}

// UseRouting -> using Microsoft.AspNetCore.Builder;

app.UseRouting();

// UseEndpoints -> using Microsoft.AspNetCore.Builder;

app.UseEndpoints(endpoints =>

{

endpoints.MapGet("/", async context =>

{

// WriteAsync -> using Microsoft.AspNetCore.Http;

await context.Response.WriteAsync("Hello World!");

});

});

}

}

}

**Branch: STEP\_2**

1. **Create Controllers Folder**
2. **Right click on Controllers Folder** 🡪 **Select Add Menu Item** 🡪 **Select Controller Menu Item** 🡪 **Select MVC Controller – Empty Item 🡪 Create HomeController**
3. **Fix HomeController**

namespace Application.Controllers

{

public class HomeController : Microsoft.AspNetCore.Mvc.Controller

{

public HomeController() : base()

{

}

public Microsoft.AspNetCore.Mvc.ViewResult Index()

{

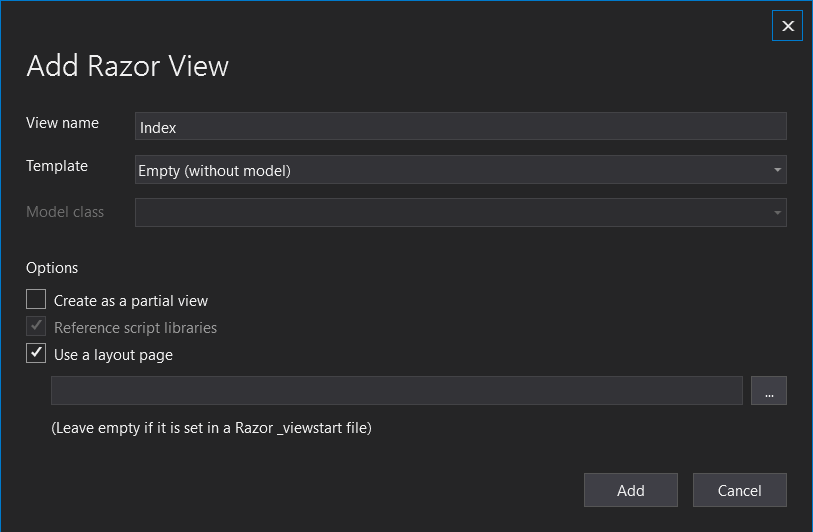
return View();

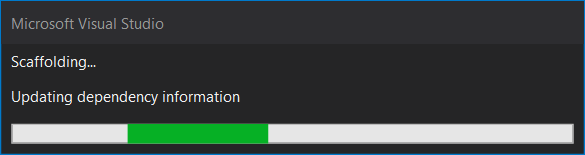
}

}

}

1. **Right click on Index() 🡪 Select Add View menu item 🡪 Select Razor View 🡪 Click on [Add] Button**





1. **In created folder: Views** 🡪 **In created folder Home** 🡪 **Fix Index.cshtml**

@{

ViewData["Title"] = "Index";

}

<h1>

Home

</h1>

1. **In [Views] folder 🡪 Create [\_ViewStart.cshtml]**

@{

Layout = "\_Layout";

}

1. **In [Views] folder 🡪 Create [\_ViewImports.cshtml]**

@using Application

@addTagHelper \*, Microsoft.AspNetCore.Mvc.TagHelpers

1. **Fix Application**
2. **Fix Startup.cs:**

using Microsoft.AspNetCore.Builder;

using Microsoft.Extensions.Hosting;

using Microsoft.Extensions.DependencyInjection;

namespace Application

{

public class Startup : object

{

public Startup() : base()

{

}

public void ConfigureServices

(Microsoft.Extensions.DependencyInjection.IServiceCollection services)

{

// using Microsoft.Extensions.DependencyInjection;

services.AddControllersWithViews();

}

public void Configure

(Microsoft.AspNetCore.Builder.IApplicationBuilder app,

Microsoft.AspNetCore.Hosting.IWebHostEnvironment env)

{

// IsDevelopment -> using Microsoft.Extensions.Hosting;

if (env.IsDevelopment())

{

// UseDeveloperExceptionPage -> using Microsoft.AspNetCore.Builder;

app.UseDeveloperExceptionPage();

}

// UseHttpsRedirection -> using Microsoft.AspNetCore.Builder;

app.UseHttpsRedirection();

// UseStaticFiles -> using Microsoft.AspNetCore.Builder;

app.UseStaticFiles();

// UseRouting -> using Microsoft.AspNetCore.Builder;

app.UseRouting();

// UseRouting -> using Microsoft.AspNetCore.Builder;

//app.UseCookiePolicy();

// UseRouting -> using Microsoft.AspNetCore.Builder;

//app.UseAuthentication();

// UseRouting -> using Microsoft.AspNetCore.Builder;

//app.UseAuthorization();

// UseEndpoints -> using Microsoft.AspNetCore.Builder;

app.UseEndpoints(endpoints =>

{

// MapControllerRoute -> using Microsoft.AspNetCore.Builder;

endpoints.MapControllerRoute(

name: "default",

pattern: "{controller=Home}/{action=Index}/{id?}");

});

}

}

}