

MA Practical 1

1) Install .Net Core Sdk (Link:

<https://dotnet.microsoft.com/learn/dotnet/hello-world-tutorial/install>) 2) create folder MyMVC

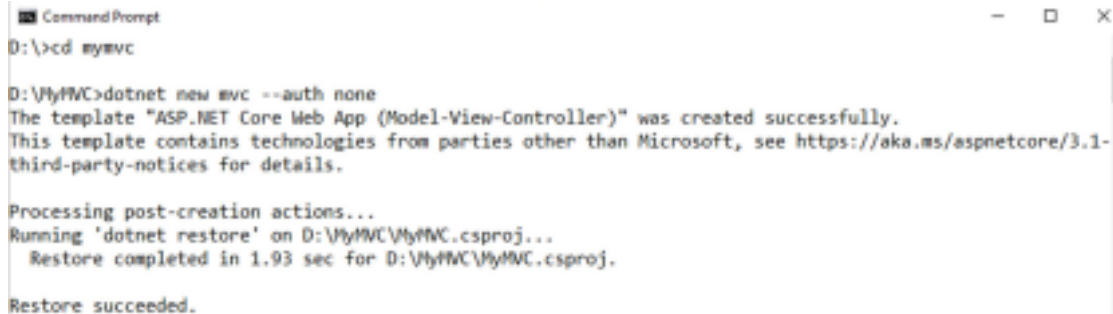
folder in D: drive or any other drive

3) open command prompt and perform following operations

Command: to create mvc project

`dotnet new mvc --auth none`

output:



```
Command Prompt
D:\>cd mymvc

D:\MyMVC>dotnet new mvc --auth none
The template "ASP.NET Core Web App (Model-View-Controller)" was created successfully.
This template contains technologies from parties other than Microsoft, see https://aka.ms/aspnetcore/3.1-third-party-notices for details.

Processing post-creation actions...
Running 'dotnet restore' on D:\MyMVC\MyMVC.csproj...
  Restore completed in 1.93 sec for D:\MyMVC\MyMVC.csproj.

Restore succeeded.
```

4) Go

to controllers folder and modify HomeController.cs file to match following: using System;

using System.Collections.Generic;

using System.Diagnostics;

using System.Linq;

using System.Threading.Tasks;

using Microsoft.AspNetCore.Mvc;

using Microsoft.Extensions.Logging;

using MyMVC.Models;

namespace MyMVC.Controllers

{ public class HomeController : Controller

{

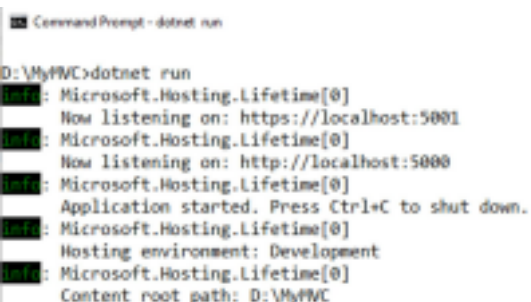
public String Index()

{ return "Hello World"; }

}

}

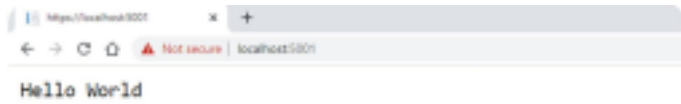
5) Run the project



```
Command Prompt - dotnet run
D:\MyMVC>dotnet run
Microsoft.Hosting.Lifetime[0]: Now listening on: https://localhost:5001
Microsoft.Hosting.Lifetime[0]: Now listening on: http://localhost:5000
Microsoft.Hosting.Lifetime[0]: Application started. Press Ctrl+C to shut down.
Microsoft.Hosting.Lifetime[0]: Hosting environment: Development
Microsoft.Hosting.Lifetime[0]: Content root path: D:\MyMVC
```

Now open

browser and type URL: localhost:5000



6) Now go back to command prompt and stop running project using CTRL+C

```
Command Prompt
D:\MyMVC>dotnet run
Microsoft.Hosting.Lifetime[0]:
Now listening on: https://localhost:5001
Microsoft.Hosting.Lifetime[0]:
Now listening on: http://localhost:5000
Microsoft.Hosting.Lifetime[0]:
Application started. Press Ctrl+C to shut down.
Microsoft.Hosting.Lifetime[0]:
Hosting environment: Development
Microsoft.Hosting.Lifetime[0]:
Content root path: D:\MyMVC
Microsoft.Hosting.Lifetime[0]:
Application is shutting down...
D:\MyMVC>
```

7)Go to models folder and add new file StockQuote.cs to it with following content using System; namespace MyMVC.Models

```
{
    public class StockQuote
    { public string Symbol {get;set;}
      public int Price{get;set;}
    }
}
```

8)Now Add View to folder then home folder in it and modify index.cshtml file to match following @{

```
    ViewData["Title"] = "Home Page";
}
<div>
    Symbol: @Model.Symbol <br/>
    Price: $@Model.Price <br/>
</div>
```

9)Now modify HomeController.cs file to match following:

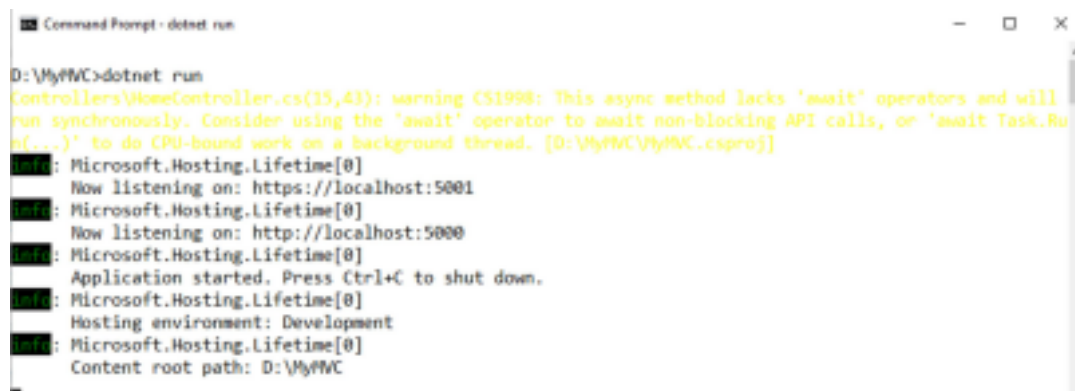
```
using System;
using System.Collections.Generic;
using System.Diagnostics;
using System.Linq;
using System.Threading.Tasks;
using Microsoft.AspNetCore.Mvc;
using Microsoft.Extensions.Logging;
using MyMVC.Models;
```

```

namespace MyMVC.Controllers
{
    public class HomeController : Controller
    { public async Task <IActionResult> Index()
    {
        var model= new StockQuote{ Symbol='HLLO', Price=3200};
        return View(model);
    }
}

```

10)Now run the project using
dotnet run



```

D:\MyMVC>dotnet run
Controllers\HomeController.cs(15,43): warning CS1998: This async method lacks 'await' operators and will
run synchronously. Consider using the 'await' operator to await non-blocking API calls, or 'await Task.Run(
n(...))' to do CPU-bound work on a background thread. [D:\MyMVC\MyMVC.csproj]
info: Microsoft.Hosting.Lifetime[0]
      Now listening on: https://localhost:5001
info: Microsoft.Hosting.Lifetime[0]
      Now listening on: http://localhost:5000
info: Microsoft.Hosting.Lifetime[0]
      Application started. Press Ctrl+C to shut down.
info: Microsoft.Hosting.Lifetime[0]
      Hosting environment: Development
info: Microsoft.Hosting.Lifetime[0]
      Content root path: D:\MyMVC

```

11)Now go back to browser and refresh to get modified view response

