(T1)在IIS執行MVC  
CourseGUID: 8503b39c-5887-4634-8291-facfb3117924  
=======================================================================  
(T1)在IIS執行MVC  
=======================================================================  
0. Summary

1. New Project - OnlineGame.Web

-----------

2. Controllers/HomeController.cs

2.1. Add Controller in VS2017

2.2. Add New File extension

2.3. Add Controller

2.4. Controllers/HomeController.cs

-----------

3. IIS

3.1. Install IIS

3.2. Start IIS

3.2.1. Run IIS: Win + R

3.2.2. Run IIS: Win + Search IIS

3.2.3. Run IIS: This PC --> Manage

3.3. Run IIS but Get Error Message

3.4. Visual Studio as Admin

3.4.1. Set Visual Studio as Admin

3.4.2. Run Visual Studio as Admin

3.4.3. Run Visual Studio as Admin without dialog

3.5. Run IIS

-----------

4. URL

4.1. Global.asax.cs

4.2. App\_Start/RouteConfig.cs

4.3. Controllers/HomeController.cs

4.4. Explain mapping URL

-----------

5. HomeController.cs

-----------

6. ASP.NET Tracing: trace.axd

6.1. App\_Start/RouteConfig.cs

6.2. Web.config

6.3. localhost/ProjectName/trace.axd

-----------

7. View

7.1. Reason to use View

7.1.1. Controllers/HomeController.cs

7.2. Views/Home/Index.cshtml

7.3. Controllers/HomeController.cs

7.4. Views/Home/Index.cshtml

-----------

8. ViewData, ViewBag

8.1. Controllers/HomeController.cs

8.2. Views/Home/Index.cshtml

-----------

9. Model

9.1. Model : Model/Gamer.cs

9.2. Controller : Controllers/GamerController.cs

9.3. View: Views/Gamer/Details.cshtml  
=======================================================================

0. Summary

=============================================

Tutorial 1: MVC基礎，IIS介紹

    \* 介紹IIS的基本觀念，使用Admin來跑Visual Studio，Model View Controller的基礎觀念介紹。

In this tutorial, we will discuss

\* Create ASP.NET MVC Project

\* Run in the IIS

\* Run VS2017 AS Admin

\* Controller, URL mapping

\* The axd file request

\* Views

\* ViewData, ViewBag

=============================================

由於MVC課程 的範例都是由Entity Framework所寫成，

所以必須要先有一點點的Entity Framework的基礎才可以繼續學MVC下去。

由於Entity Framework課程 的使用者介面的範例都是搭配MVC所寫成，

所以必須要先有一點點的MVC的基礎才可以繼續學Entity Framework下去。

原則上我的學程必須要先學MVC才能學Entity Framework

但是我擔心，有人沒學過我的MVC課程就跑去先學Entity Framework課程，

這樣肯定學Entity Framework的時候會GG。

因此

在設計課程的時候，我逼不得已，只好讓

MVC課程裡面的Tutorial 1 和 Tutorial 2 和

Entity Framework課程裡面的Tutorial 1 和 Tutorial 2 是完全一樣的。

這樣的課程設計，

一方面是讓已經學過MVC課程的學生在學Entity Framework課程的時候，

重新快速複習MVC觀念。

另一方面是讓沒學過MVC課程的學生在學Entity Framework課程的時候，

可以有稍微一點點MVC基礎觀念。

如果你事先讀過MVC的Tutorial 1 和 Tutorial 2，

那麼在學Entity Framework課程的時候，

覺得完全不需要複習MVC的基礎觀念的話，

就麻你跳過

Thanks

=============================================

更正1

在Tutorial 1

"default"這個route名稱,其實是可以改的, 用其他名稱,也會動哦

我說一定要用"Default"這個字

阿~~ 我用"一定"這個字 阿

我的錯 (攤手)

是這樣的 被公司洗腦

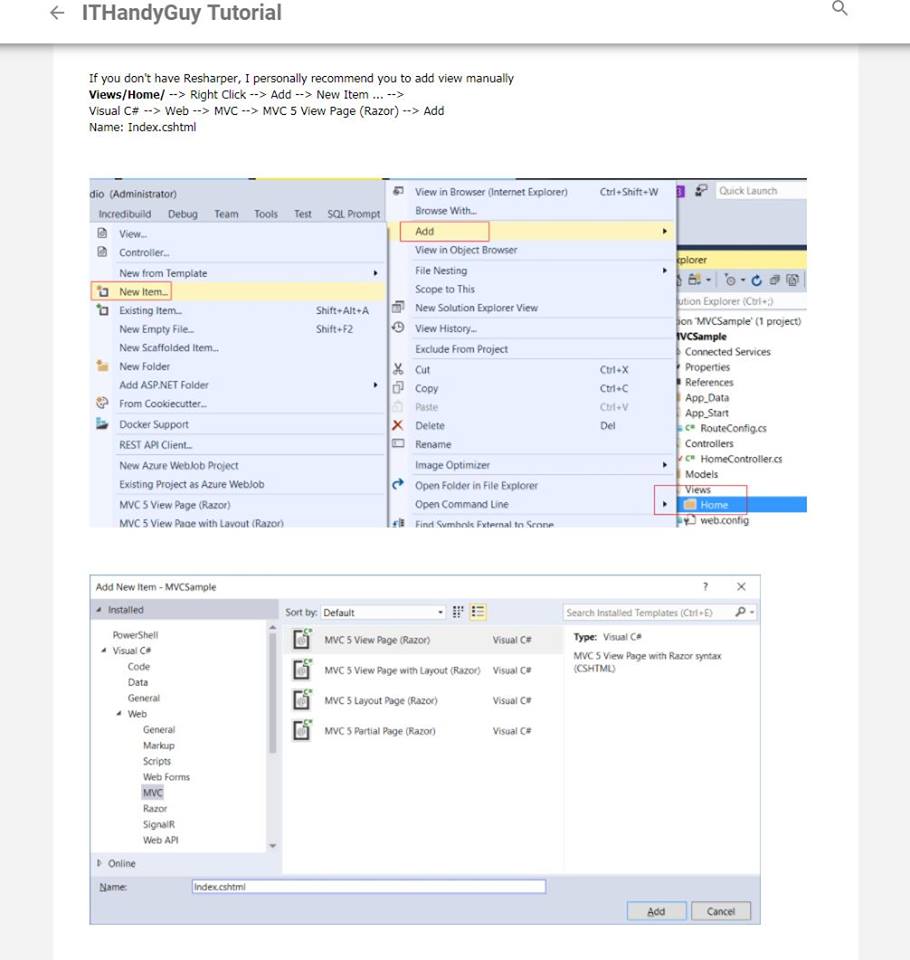
因為公司都固定用"default"

很多字都是用固定的

=============================================

更正2

在Tuutorial1-2 Video裡您示範Add View page時不小心按成Layout page，但講義裡是正確的，因為跟著影片實作發現這個小差異



=============================================

1. New Project - OnlineGame.Web

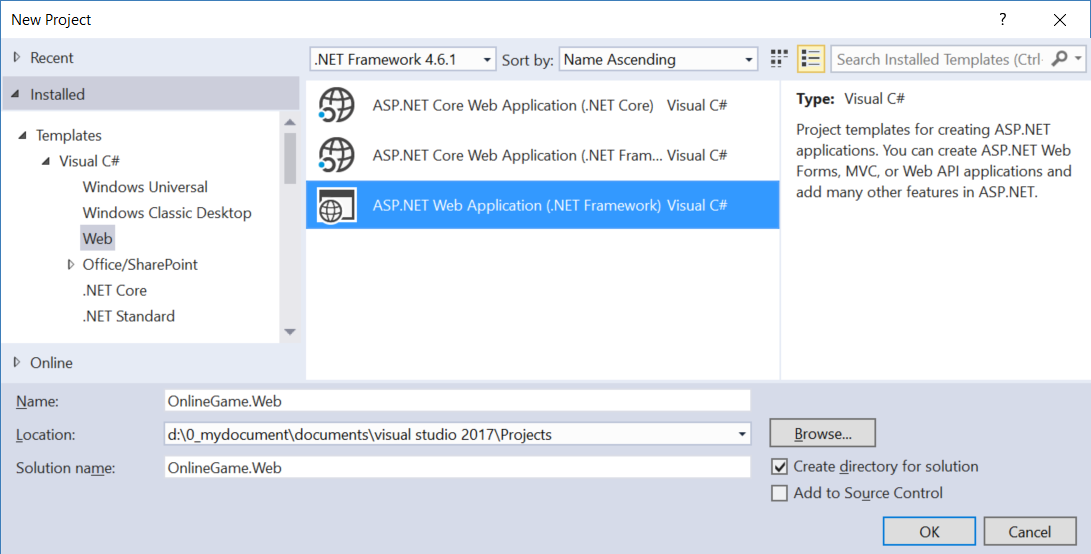
File --> New --> Project... -->

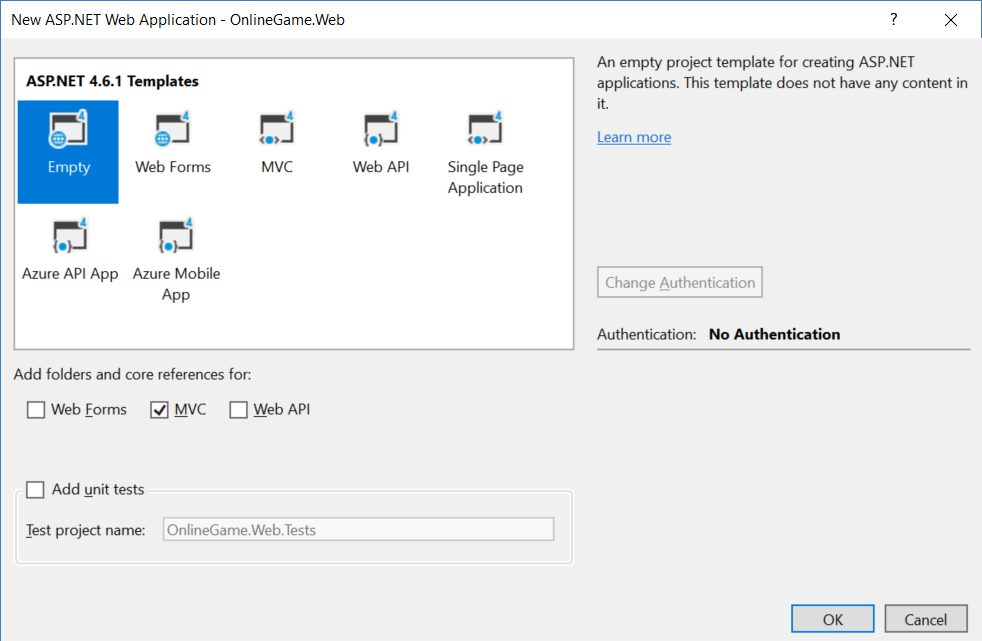
Visual C# --> Web --> ASP.NET Web Application (.Net Framework)

-->

Name: **OnlineGame.Web**

Empty --> Select "MVC" --> OK



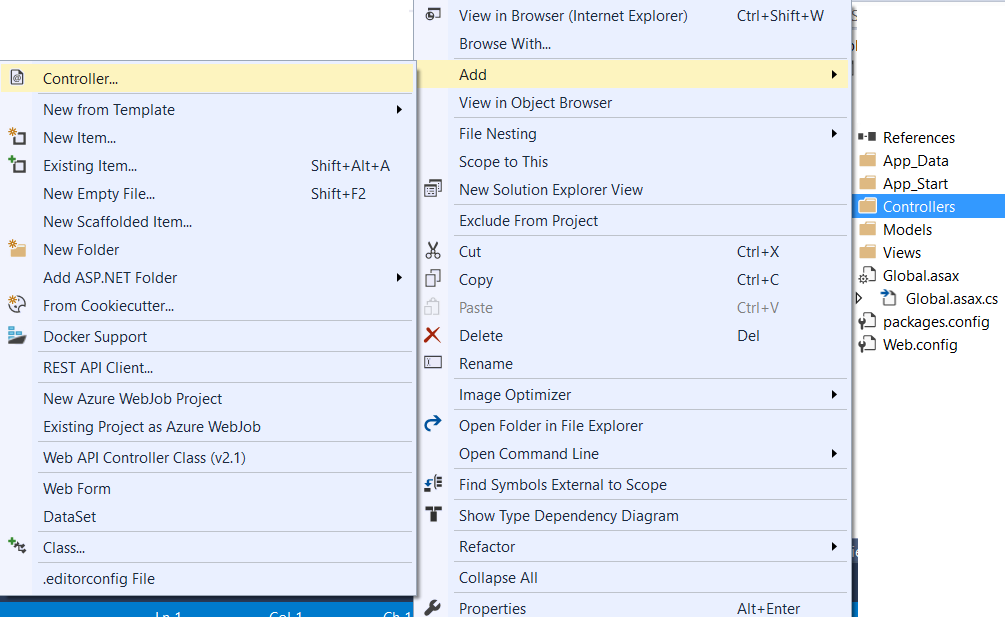


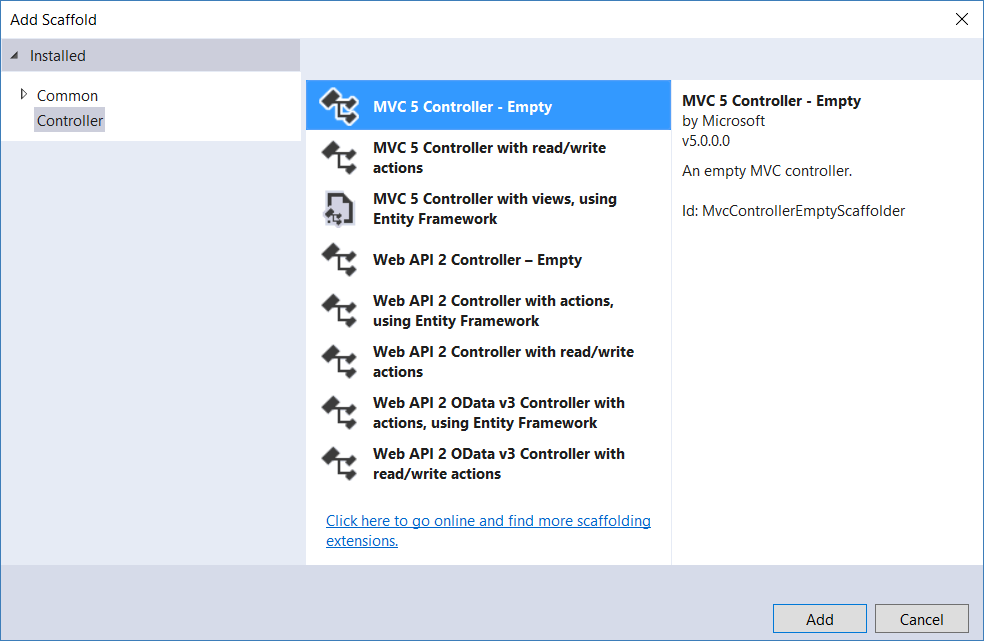
2. Controllers/HomeController.cs

2.1. Add Controller in VS2017

Controllers --> Add --> Controller --> MVC 5 Controller - Empty -->

Controller Name : **HomeController**





Application

Description automatically generated with medium confidence

2.2. Add New File extension

Add New File

Mads Kristensen

3.5.129

Visual Studio 2013, 2015, 2017

The fastest and easiest way to add new files to any project - including files that start with a dot

<https://marketplace.visualstudio.com/items?itemName=MadsKristensen.AddNewFile>

Because **Add New File** **(extension and update)**

press **Shift+F2**

**Controllers/HomeController.cs**

Graphical user interface, text, application

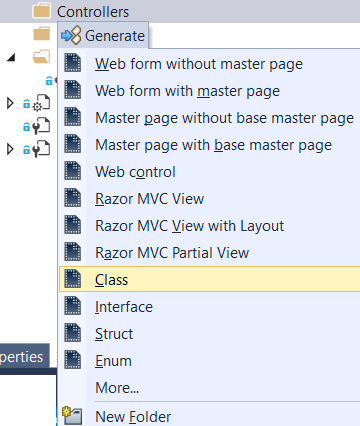
Description automatically generated

2.3. Add Controller

press **Alt + Insert**

**--> Class**

**HomeController**



Graphical user interface

Description automatically generated with low confidence

2.4. Controllers/HomeController.cs

Controllers/HomeController.cs

using System.Web.Mvc;

namespace OnlineGame.Web.Controllers

{

    public class HomeController : Controller

    {

        // GET: Home

        public ActionResult Index()

        {

            return View();

        }

    }

}

--> Change to

using System.Web.Mvc;

namespace OnlineGame.Web.Controllers

{

    public class HomeController : Controller

    {

        // GET: Home

        public string Index()

        {

            return "Hello";

        }

    }

}

3. IIS

3.1. Install IIS

Reference:

<https://superuser.com/questions/1245472/does-windows-10-home-edition-have-iis>

You may use IIS Express to Run you MVC project.

IIS and IIS express is no much different during developing process,

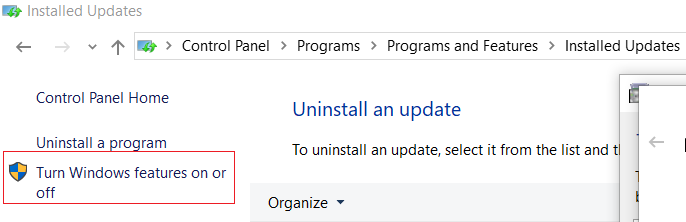
but it is still better to have IIS.

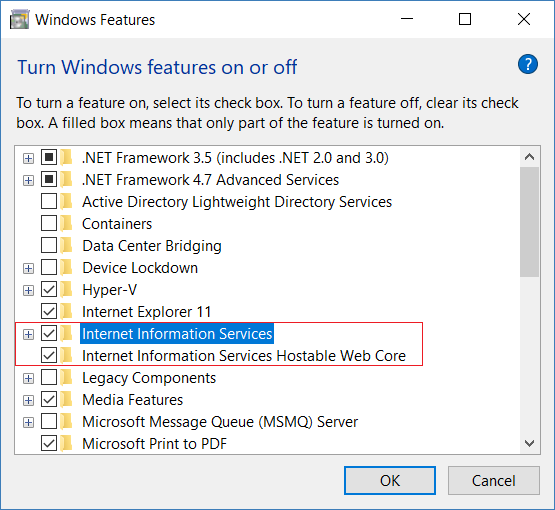
Control Panel --> Programs --> Programs and Features --> Turn Windows features on or off -->

Select:

**Internet Information Service**

**Internet Information Service Hostable Web Core**



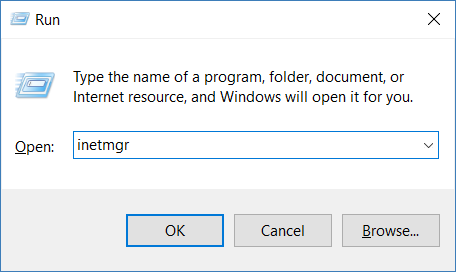


3.2. Start IIS

3.2.1. Run IIS: Win + R

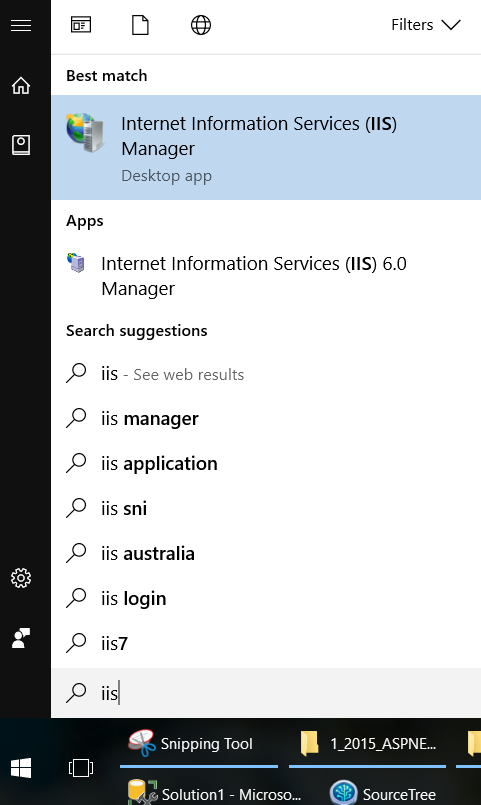
Win + R

**inetmgr**



3.2.2. Run IIS: Win + Search IIS

Win + Search IIS



3.2.3. Run IIS: This PC --> Manage

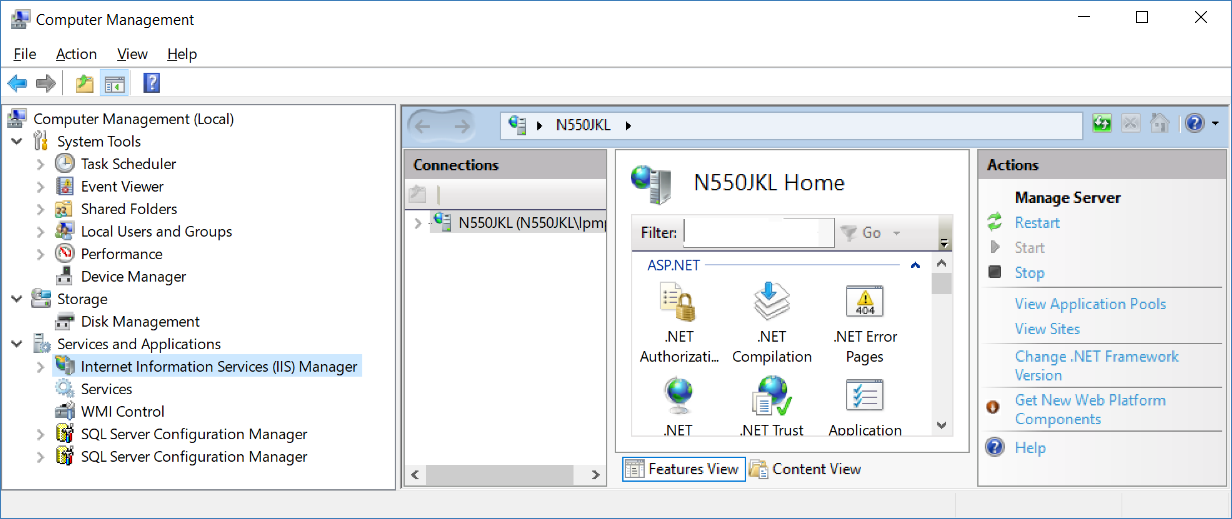
Win + E to open Explore

In the Left hand side --> This PC --> Right Click --> Manage

--> Services and Applications --> Internet Information Service

Graphical user interface, application

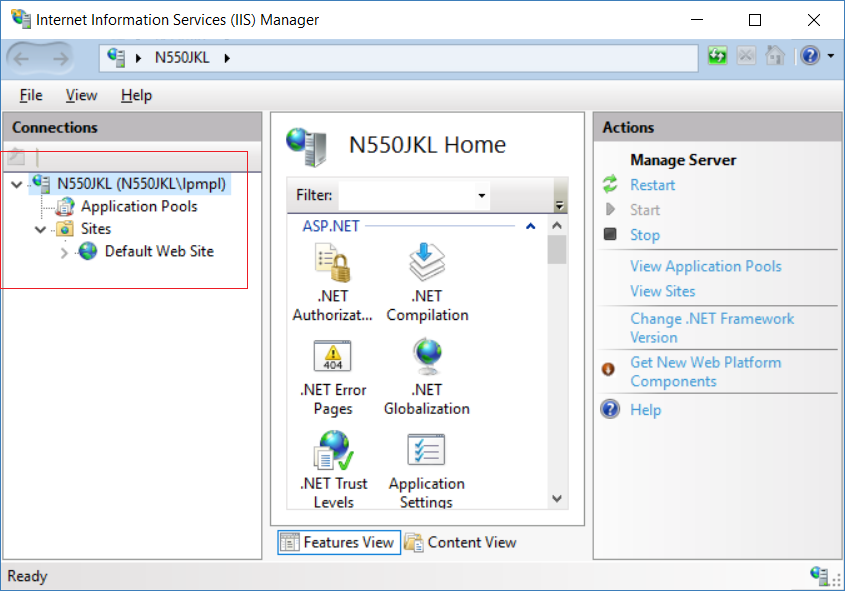
Description automatically generated



3.3. Run IIS but Get Error Message

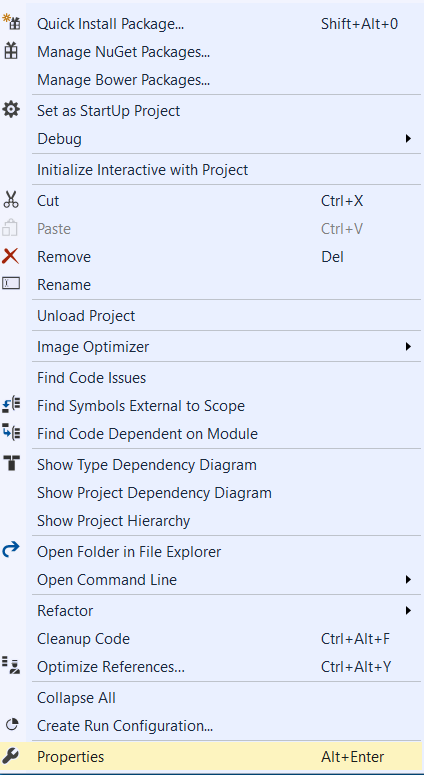
At the moment, we have no YourWebSiteName in the IIS.

Now, we want to add YourWebSiteName to IIS



Go back to Visual Studio

ProjectName --> Right Click --> Properties



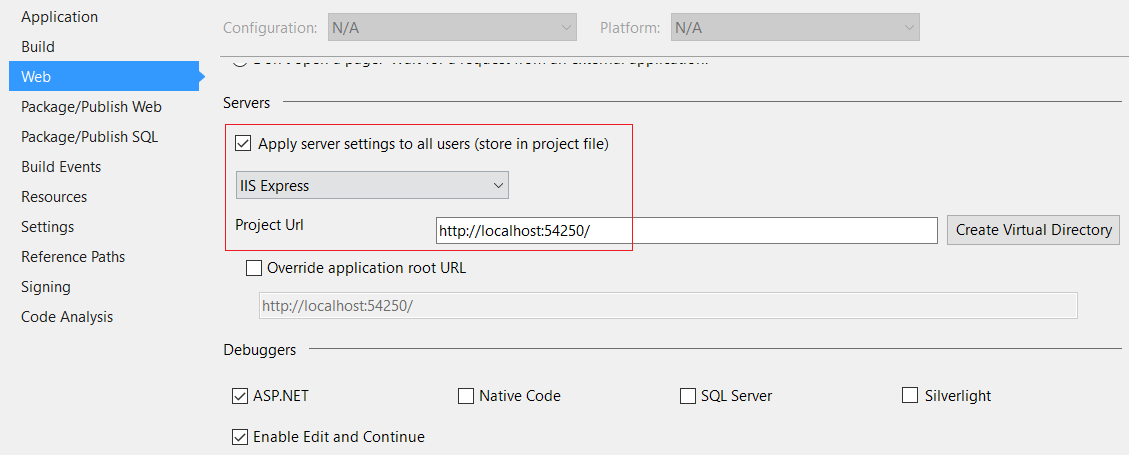
--> Web Tab --> Servers Section

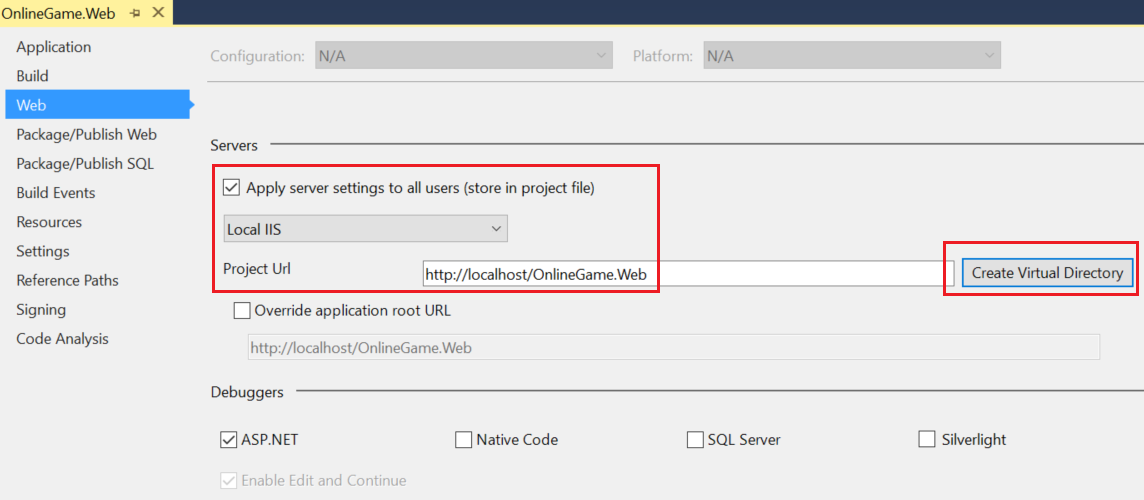
--> Dropdown box --> At the moment, we are using **IIS Express**

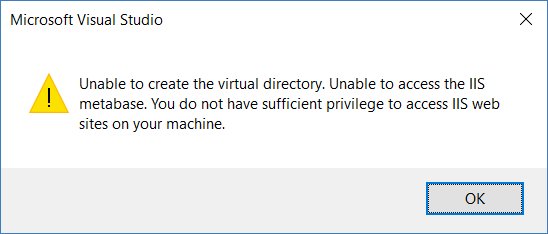
--> Change to **Local IIS**

Press "**Create Virtual Directory**" button

--> If you see the error message, please **Run Visual Studio as Admin**







3.4. Visual Studio as Admin

3.4.1. Set Visual Studio as Admin

Set    Run Visual Studio as Admin

**C:\Program Files (x86)\Microsoft Visual Studio\2017\Community\Common7\IDE\devenv.exe**

We need to set up to **devenv.exe** run as admin

--> Right Click

 --> Troubleshoot compatibility

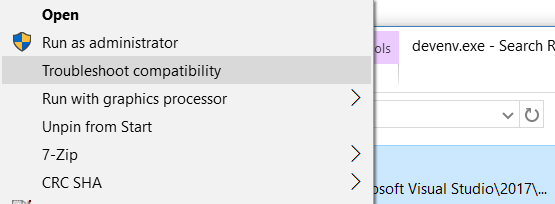
--> Troubleshoot Program

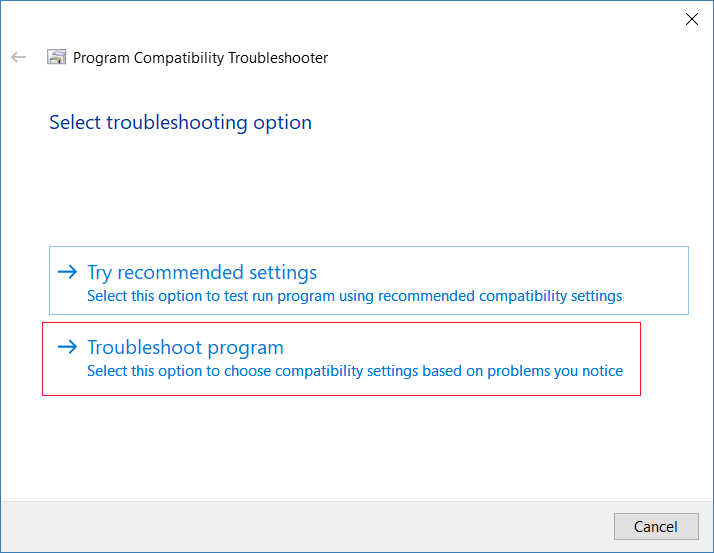
--> The program requires additional permissions

--> Test the program  --> Next

--> Yes, Save these settings for this program

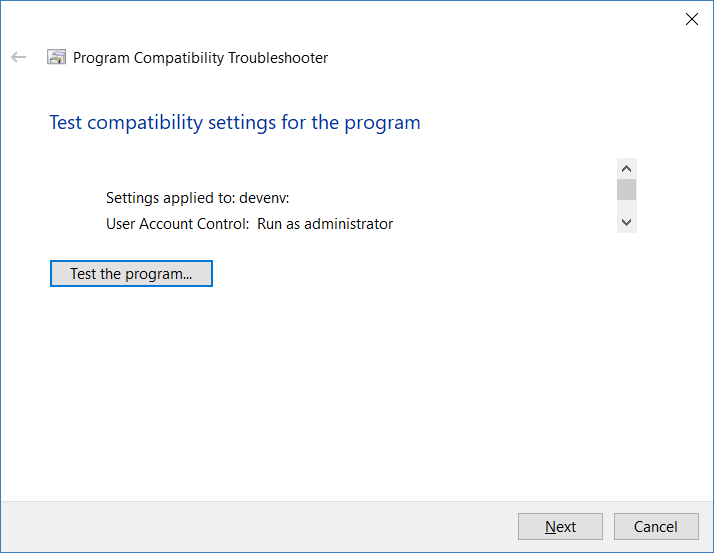
--> Close the troubleshooter

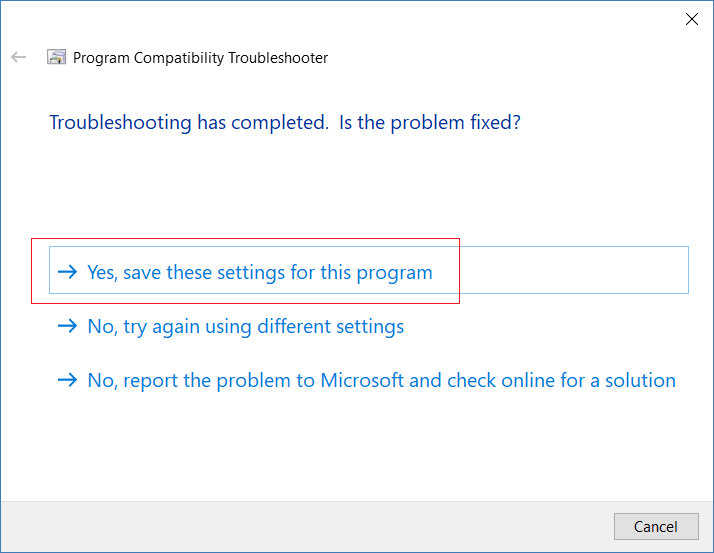


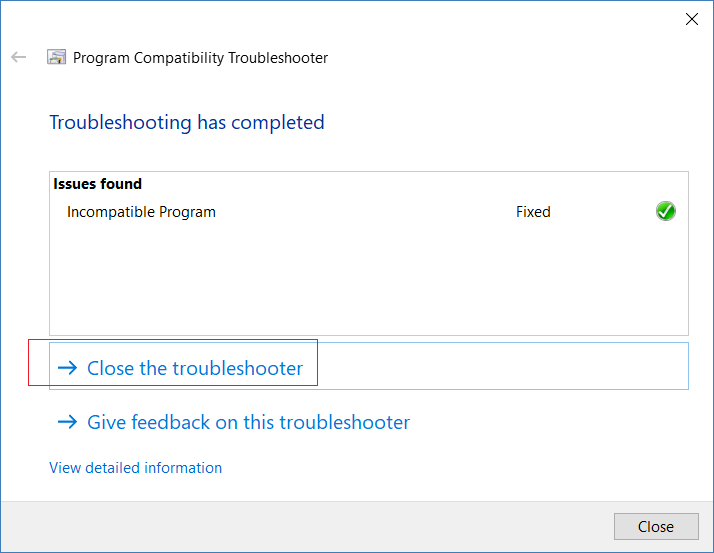


Graphical user interface, text, application, email

Description automatically generated







3.4.2. Run Visual Studio as Admin

After you Set Visual Studio as Admin.

You may Run YourWebsiteName.sln

Go to

...YourPath...\Documents\Visual Studio 2017\Projects\**YourWebsiteName**

Double click  "**YourWebsiteName**.sln"

**Do you want to allow this app to make changes to your device** --> Yes

Then Visual Studio will be opened

**Save changes to the following items?**  devenv.sln   -->  No

Graphical user interface, application, Word

Description automatically generated

3.4.3. Run Visual Studio as Admin without dialog

**Save changes to the following items?**  devenv.sln   -->  No

If you don't want to see this dialog anymore, there is a way to avoid.

Reference:

<http://rickvandenbosch.net/blog/running-visual-studio-as-an-administrator-causes-save-changes-to-devenv-sln-when-double-clicking-solutions/>

C:\Program Files (x86)\Common Files\Microsoft Shared\MSEnv\VSLauncher.exe

Set    Run **VSLauncher.exe**  Run as Admin

**C:\Program Files (x86)\Common Files\Microsoft Shared\MSEnv\VSLauncher.exe**

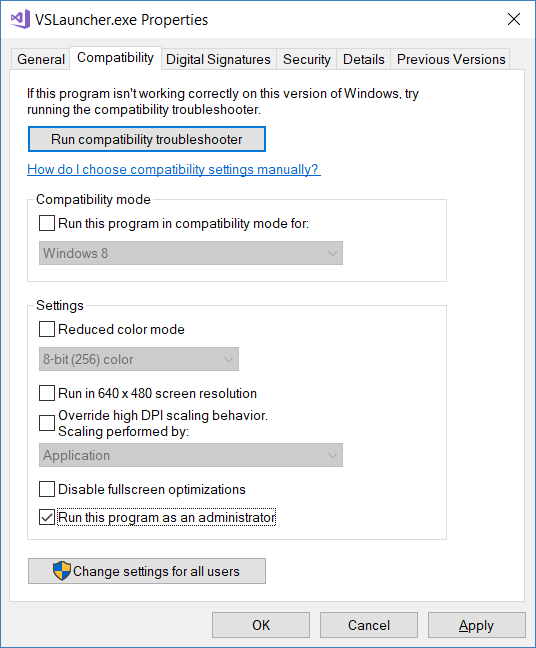
We need to set up to **VSLauncher.exe** run as admin

--> Right Click --> Compatibility Tab --> Select    Run this program as an administrator

You need to set both **devenv.exe**and **VSLauncher.exe**to Run as Admin to avoid the dialog.

**C:\Program Files (x86)\Microsoft Visual Studio\2017\Community\Common7\IDE\devenv.exe**

**C:\Program Files (x86)\Common Files\Microsoft Shared\MSEnv\VSLauncher.exe**



**Do you want to allow this app to make changes to your device** --> Yes

If you don't want to see this dialog anymore, there is a way to avoid.

Reference:

<https://answers.microsoft.com/en-us/windows/forum/windows_7-security/how-can-i-disable-the-do-you-want-to-allow-the/35a722f4-f53d-49d1-b345-c4463cdf81e1?auth=1>

Control Panel --> User Accounts --> User Accounts -->

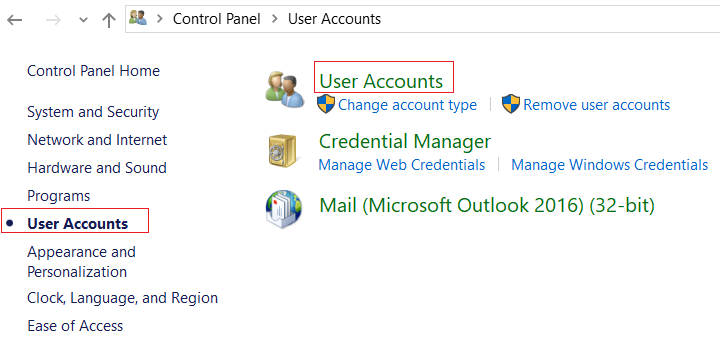
Change User Account Controller Settings --> Never notify.

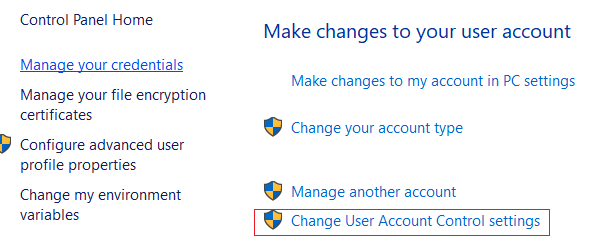
**Note:**

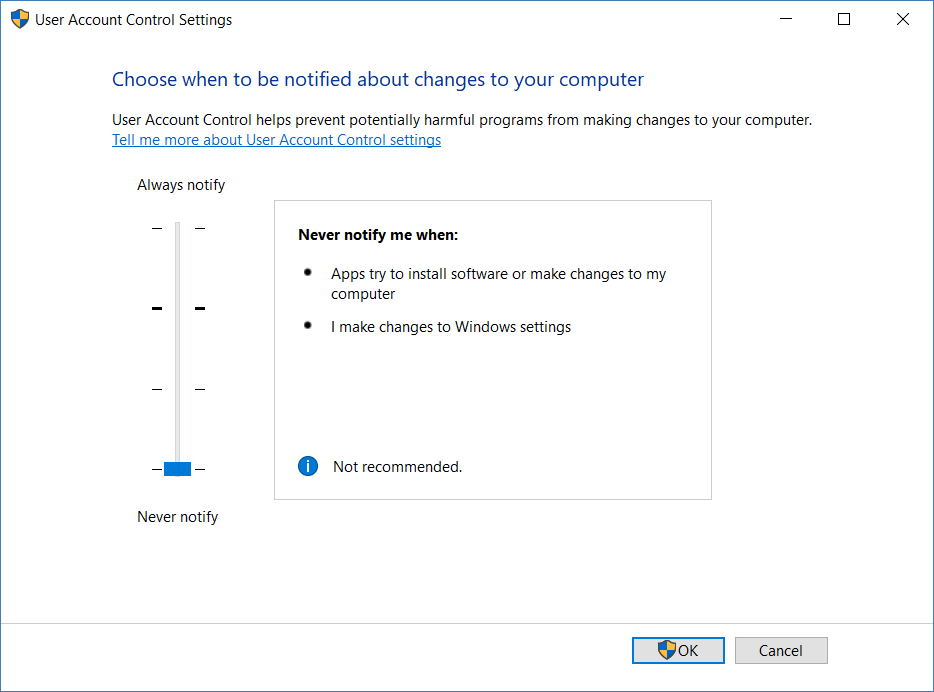
UAC helps prevent potentially harmful applications from making changes to your Computer.

Turning off UAC would turn off the feature for all programs and not for only a specific particular application.

I personally suggest just leave as the default setting.



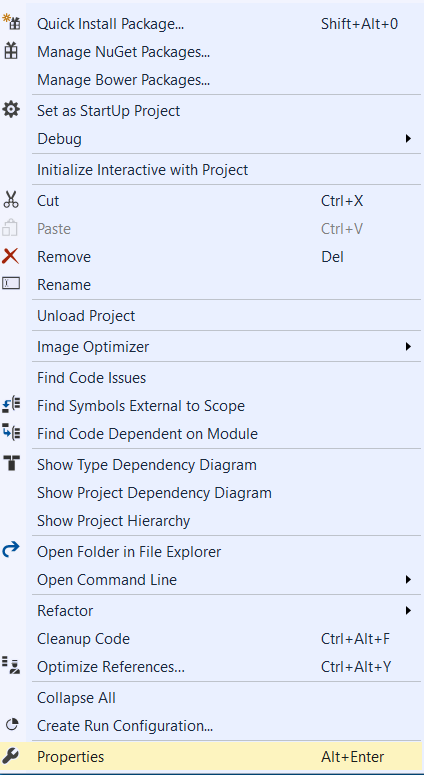




3.5. Run IIS

Go back to Visual Studio

ProjectName --> Right Click --> Properties



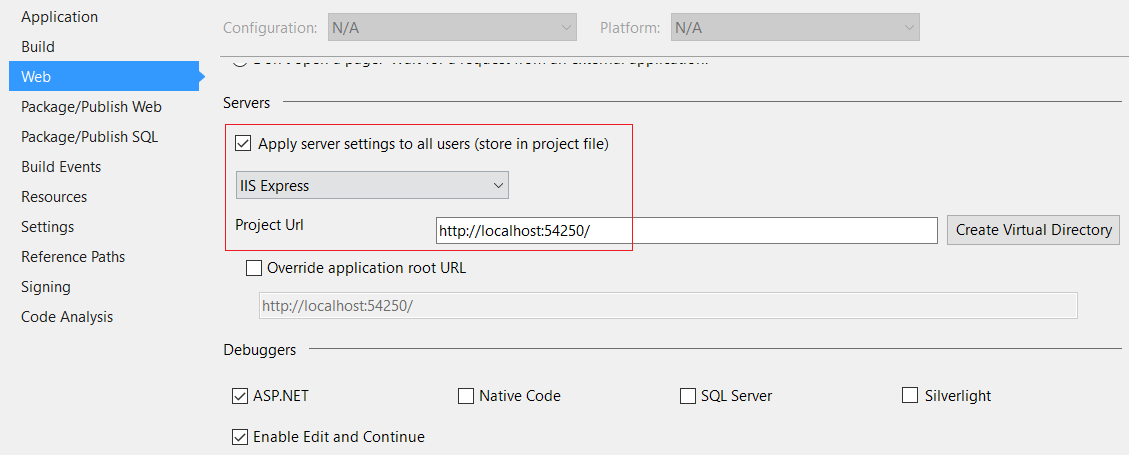
--> Web Tab --> Servers Section

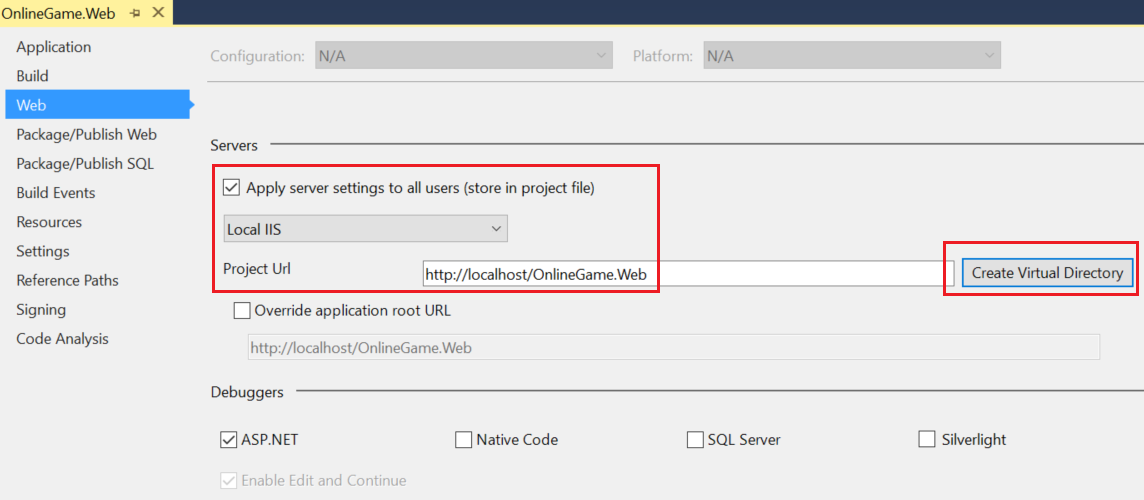
--> Dropdown box --> At the moment, we are using **IIS Express**

--> Change to **Local IIS**

Press "**Create Virtual Directory**" button

--> If you see the error message, please **Run Visual Studio as Admin**



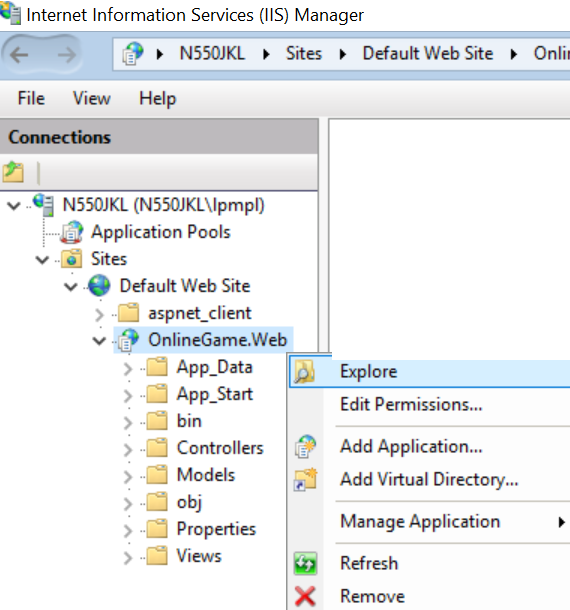


Graphical user interface, text, application

Description automatically generated

Now, In IIS

We can see "**YourWebSiteName**" is under Default Website



1.

Default Web Site --> YourWebSiteName --> Right Click --> Explore

The IIS will host your web application under **YourProjectName**directory.

2.

Default Web Site Right Click --> Explore

The IIS host default website under**C:\inetpub\wwwroot** directory

This is the place where YourWebSiteName published and hosted by IIS

The URL is [http://localhost/YourProjectName/](http://localhost/MVCDemo/)

If we are using IIS Express

The URL is [http://localhost:YourPortNumber/](http://localhost:54250/)

We can see the IIS Express will need a port number.

4. URL

4.1. Global.asax.cs

using System.Web.Mvc;

using System.Web.Routing;

namespace OnlineGame.Web

{

    public class MvcApplication : System.Web.HttpApplication

    {

        //Application\_Start() is the magic start point of this application

        protected void Application\_Start()

        {

            AreaRegistration.RegisterAllAreas();

            //1.

            //Register Route Configure in RouteConfig.cs

            //If you want to see route configuration,

            //you may find it in RouteConfig.cs

            //2.

            //System.Web.Routing.RouteCollection Routes { get; }

            //Gets a collection of objects that derive from the System.Web.Routing.RouteBase class.

            RouteConfig.RegisterRoutes(RouteTable.Routes);

        }

    }

}

4.2. App\_Start/RouteConfig.cs

using System.Web.Mvc;

using System.Web.Routing;

namespace OnlineGame.Web

{

    public class RouteConfig

    {

        public static void RegisterRoutes(RouteCollection routes)

        {

            routes.IgnoreRoute("{resource}.axd/{\*pathInfo}");

            routes.MapRoute(

                name: "Default",

                url: "{controller}/{action}/{id}",

                defaults: new { controller = "Home", action = "Index", id = UrlParameter.Optional }

            );

        }

    }

}

/\*

1.

//routes.MapRoute(

//    name: "Default",

//    url: "{controller}/{action}/{id}",

//    defaults: new { controller = "Home", action = "Index", id = UrlParameter.Optional }

//);

1.1.

When a request comes in,

it's trying to do a pattern match based on

all the templates it sees in these mapped routes.

A route is some instructions for

how to take a URI coming into a request

and map it to some code,

normally a controller.

In this case,

look at defaults parameter,

when user request <http://localhost:PortNumber/>

IIS Express will run

HomeController Index action.

It will map to Controllers/HomeController.cs

and   map to Index Method

1.2.

By convention in MVC.

All controllers will have Controller suffix.

This suffix is not required in the URL.

So, if you want to invoke Home controller,

you specify /Home and not /HomeController.

\*/

4.3. Controllers/HomeController.cs

using System.Web.Mvc;

namespace OnlineGame.Web.Controllers

{

    public class HomeController : Controller

    {

        // GET: Home

        public string Index()

        {

            return "Hello";

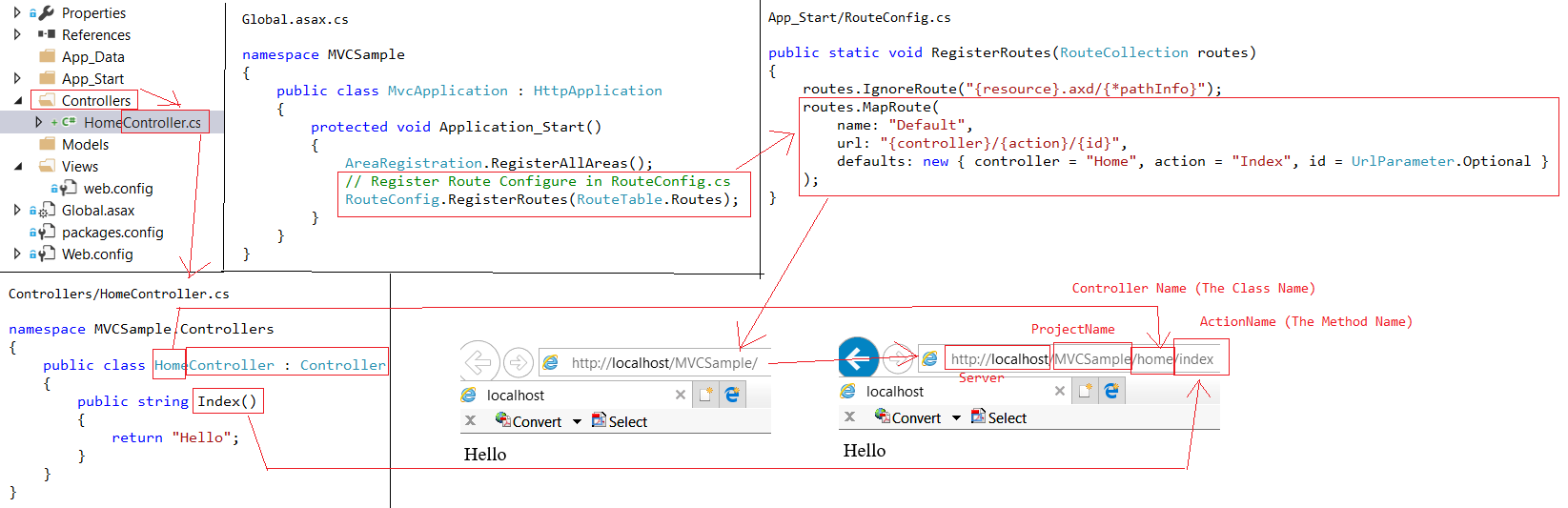
        }

    }

}

4.4. Explain mapping URL

Let's discuss the URL mapping In MVC



In  Global.asax.cs, We register the Route configure in RouteConfig.cs

// Register Route Configure in RouteConfig.cs

RouteConfig.RegisterRoutes(RouteTable.Routes);

-->

In App\_Start/RouteConfig.cs, we clear set up the mapping.

public static void RegisterRoutes(RouteCollection routes)

{

    routes.IgnoreRoute("{resource}.axd/{\*pathInfo}");

    routes.MapRoute(

        name: "Default",

        url: "{controller}/{action}/{id}",

        defaults: new { controller = "Home", action = "Index", id = UrlParameter.Optional }

    );

}

-->

1.

When a request comes in, it's trying to do a pattern match based on all the templates it sees in these mapped routes.

A route is some instructions for how to take a URL coming into a request and map it to some code, normally a controller.

In this case, look at defaults parameter, when user request <http://localhost:PortNumber/>

IIS Express will run    **HomeController Index** action.

It will map to **Controllers/HomeController.cs**     and   map to **Index** Method.

By convention in MVC.

All controllers will have Controller suffix.

This suffix is not required in the URL.

So, if you want to invoke Home controller,

you specify /Home and not /HomeController.

2.

Let's compare MVC URL and Web form application URL.

2.1.

In MVC URL are mapped to controller Action Method.

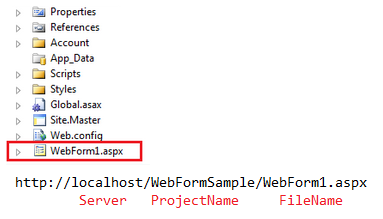
Graphical user interface, text, application

Description automatically generated

2.2.

In Web Form Application URL are mapped to physical files

E.g.



5. HomeController.cs

using System.Web.Mvc;

namespace OnlineGame.Web.Controllers

{

    public class HomeController : Controller

    {

        //// GET: Home

        //public string Index()

        //{

        //    return "Hello";

        //}

            /<http://localhost/OnlineGame.Web/home/index/aa?name=bbb>

        //<http://localhost/OnlineGame.Web/home/index/aa?name2=bbb>

        //public string Index(string id)

        //{

        //    string queryString = Request.QueryString["name"];

        //    return $"Hey, Id={id} , name={queryString}";

        //}

        //<http://localhost/OnlineGame.Web/home/index/aa?name=bbb>

        //<http://localhost/OnlineGame.Web/home/index/aa?name2=bbb>

        public string Index(string id, string name)

        {

            // return string.Format("Hey, Id ={0} , name ={1}", id, name);

            return $"Hey, Id ={id} , name ={name}";

        }

        public string GetStringA()

        {

            return "AAAAAA";

        }

    }

}

[http://localhost/OnlineGame.Web](http://localhost/OnlineGame.Web/home/index/)/home/getstringA

[http://localhost/OnlineGame.Web](http://localhost/OnlineGame.Web/home/index/aa?name=555)

[http://localhost/OnlineGame.Web](http://localhost/OnlineGame.Web/home/index/)/home/index/

[http://localhost/OnlineGame.Web](http://localhost/OnlineGame.Web/home/index/aa?name=555)/home/index/aa?name=bbb

[http://localhost/OnlineGame.Web](http://localhost/OnlineGame.Web/home/index/aa?name=555)/home/index/aa?name2=bbb

6. ASP.NET Tracing: trace.axd

If you do not understand, you may skip this part.

6.1. App\_Start/RouteConfig.cs

using System.Web.Mvc;

using System.Web.Routing;

namespace OnlineGame.Web

{

    public class RouteConfig

    {

        public static void RegisterRoutes(RouteCollection routes)

        {

            //Handle the Route of the axd request file.

            //E.g. [ASP.Net](http://asp.net/) Tracing

            routes.IgnoreRoute("{resource}.axd/{\*pathInfo}");

            //Handle the Route called "Default".

            //The mapping URL is "{controller}/{action}/{id}"

            //Set the default value of Controller, action, and id.

            routes.MapRoute(

                name: "Default",

                url: "{controller}/{action}/{id}",

                defaults: new { controller = "Home", action = "Index", id = UrlParameter.Optional }

            );

        }

    }

}

/\*

2.

//routes.IgnoreRoute("{resource}.axd/{\*pathInfo}");

2.1.

Reference:

<https://stackoverflow.com/questions/9016650/what-is-routes-ignorerouteresource-axd-pathinfo>

This line can handle the axd file request route, E.g. trace.axd

.axd files don't exist physically.

[ASP.NET](http://asp.net/) uses URLs with .axd extensions

(ScriptResource.axd and WebResource.axd) internally,

and they are handled by an HttpHandler.

Therefore, you should keep this rule,

to prevent [ASP.NET](http://asp.net/) MVC from trying to handle the request

instead of letting the dedicated HttpHandler do it.

2.2.

trace.axd

Reference:

<https://msdn.microsoft.com/en-us/library/wwh16c6c.aspx>

trace.axd trace details for a specific request.

If you want to enable trace.axd,

then you have to go to **Web.config**

Add <trace enabled="true" pageOutput="false"/> under <system.web>

Then run the project, type the following URL

[http://localhost/**YourProjectName**](http://localhost/MVCSample/trace.axd)/trace.axd

This will return [ASP.NET](http://asp.net/) trace, trace.axd.

If you do not have

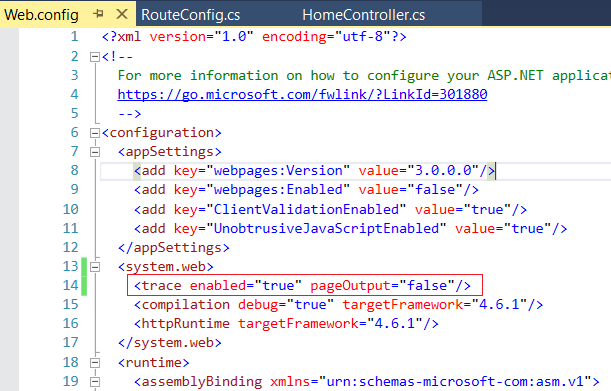
// routes.IgnoreRoute("{resource}.axd/{\*pathInfo}");

then you can not enable the trace.axd.

\*/

6.2. Web.config

**<trace** **enabled="true"** **pageOutput="false"/>**



6.3. localhost/ProjectName/trace.axd

[http://localhost/**YourProjectName**](http://localhost/MVCSample/trace.axd)/trace.axd

If you delete this line in  Home/HomeController.cs

routes.IgnoreRoute("{resource}.axd/{\*pathInfo}");

Then you will get an error message.

If you don't understand this concept, that is fine.

But please always include this line in  Home/HomeController.cs

routes.IgnoreRoute("{resource}.axd/{\*pathInfo}");

7. View

7.1. Reason to use View

7.1.1. Controllers/HomeController.cs

In the **Controllers/HomeController.cs**

When we try to return a list of data,

E.g.

return new List<string>

            {

                "Name01",

                "Name02",

                "Name03"

            };

Then, it will only display the data type of the variable

E.g.

System.Collections.Generic.List`1[System.String]

This is not what we want,

thus, we need a view to display the data in the format we want.

using System.Collections.Generic;

using System.Web.Mvc;

namespace OnlineGame.Web.Controllers

{

    public class HomeController : Controller

    {

        //// GET: Home

        //public string Index()

        //{

        //    return "Hello";

        //}

            ////<http://localhost/OnlineGame.Web/home/index/aa?name=bbb>

        ////<http://localhost/OnlineGame.Web/home/index/aa?name2=bbb>

        //public string Index(string id)

        //{

        //    string queryString = Request.QueryString["name"];

        //    return $"Hey, Id={id} , name={queryString}";

        //}

        ////<http://localhost/OnlineGame.Web/home/index/aa?name=bbb>

        ////<http://localhost/OnlineGame.Web/home/index/aa?name2=bbb>

        //public string Index(string id, string name)

        //{

        //    // return string.Format("Hey, Id ={0} , name ={1}", id, name);

        //    return $"Hey, Id ={id} , name ={name}";

        //}

        public List<string> Index()

        {

            return new List<string>

            {

                "Name01",

                "Name02",

                "Name03"

            };

            // Return System.Collections.Generic.List`1[System.String]

            // This is Wrong.

        }

        public string GetStringA()

        {

            return "AAAAAA";

        }

    }

}

/\*

1.

When we try to return a list of data,

E.g.

return new List<string>

{

    "Name01",

    "Name02",

    "Name03"

};

Then, it will only display the data type of the variable

E.g.

System.Collections.Generic.List`1[System.String]

This is not what we want,

thus, we need a view to display the data in the format we want.

2.

//public ActionResult Index()

//{

//    return View();

//}

ViewResult extend ViewResultBase

ViewResultBase extend ActionResult.

Thus, you can return View()

\*/

7.2. Views/Home/Index.cshtml

In the **Controllers/HomeController.cs**

public ActionResult Index()

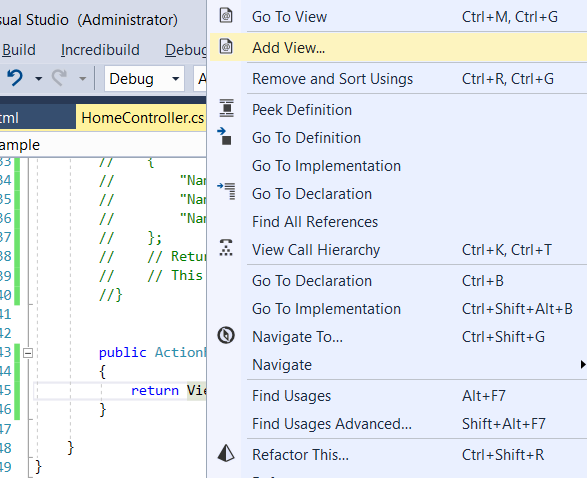
{

    return **View**();

}

Mouse point to View, and Right Click --> Add View

--> Add



Graphical user interface, text, application, email

Description automatically generated

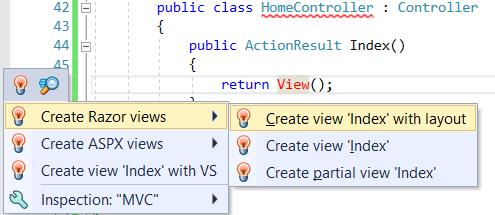
This will create  **Views/Home/Index.cshtml**

However, I personally do not like to use this way to create a view.

Sometimes, this way will have some un-necessary changes

which you can only use source control to find out and discard changes.

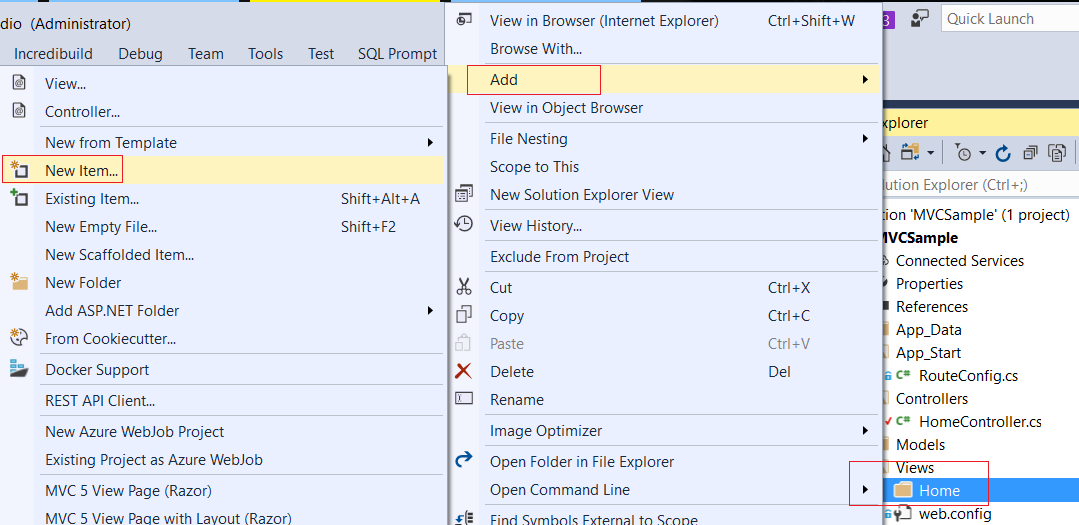
Mouse points to **View --> Alt + Enter --> Create Razor views --> Create view 'index' with layout**

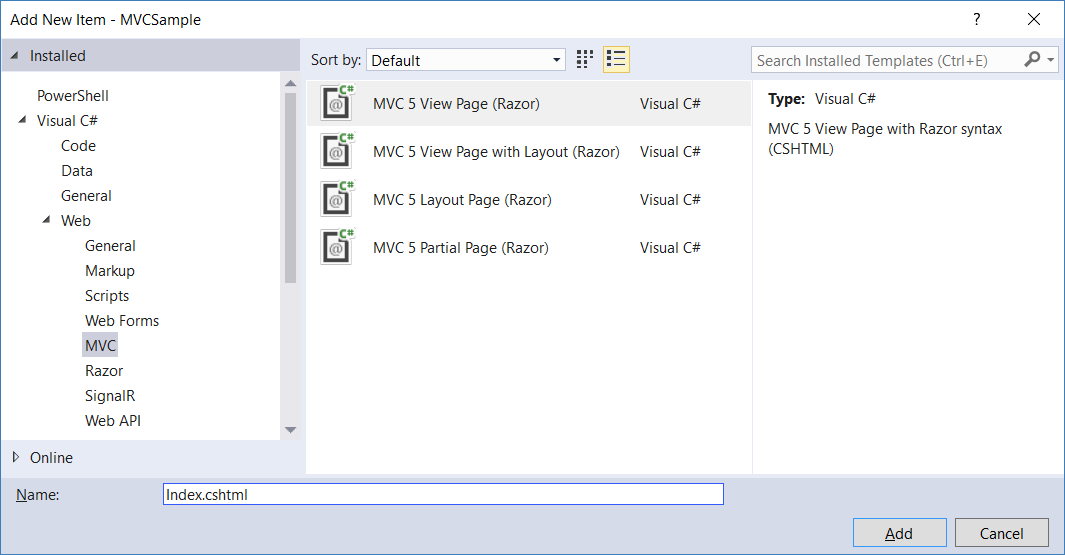


**Views/Home/**--> Right Click --> Add --> New Item ... -->

Visual C# --> Web --> MVC --> MVC 5 View Page (Razor) --> Add

Name: Index.cshtml





@{

    ViewBag.Title = "title";

}

<h2>title</h2>

7.3. Controllers/HomeController.cs

using System.Collections.Generic;

using System.Web.Mvc;

namespace OnlineGame.Web.Controllers

{

    public class HomeController : Controller

    {

        //// GET: Home

        //public string Index()

        //{

        //    return "Hello";

        //}

        //public string Index(string id)

        //{

        //    string queryString = Request.QueryString["name"];

        //    return $"Hey, Id={id} , name={queryString}";

        //}

        ////<http://localhost/OnlineGame.Web/home/index/aa?name=555>

        //public string Index(string id, string name)

        //{

        //    // return string.Format("Hey, Id ={0} , name ={1}", id, name);

        //    return $"Hey, Id ={id} , name ={name}";

        //}

        //public List<string> Index()

        //{

        //    return new List<string>

        //    {

        //        "Name01",

        //        "Name02",

        //        "Name03"

        //    };

        //    // Return System.Collections.Generic.List`1[System.String]

        //    // This is Wrong.

        //}

        public ActionResult Index()

        {

            ViewBag.Names = new List<string>

                {

                    "Name01",

                    "Name02",

                    "Name03"

                };

            return View();

        }

        public string GetStringA()

        {

            return "AAAAAA";

        }

    }

}

/\*

1.

When we try to return a list of data,

E.g.

return new List<string>

{

    "Name01",

    "Name02",

    "Name03"

};

Then, it will only display the data type of the variable

E.g.

System.Collections.Generic.List`1[System.String]

This is not what we want,

thus, we need a view to display the data in the format we want.

2.

//public ActionResult Index()

//{

//    return View();

//}

ViewResult extend ViewResultBase

ViewResultBase extend ActionResult.

Thus, you can return View()

\*/

7.4. Views/Home/Index.cshtml

@model dynamic

@{

    ViewBag.Title = "title";

}

<h2>Name List</h2>

<ul>

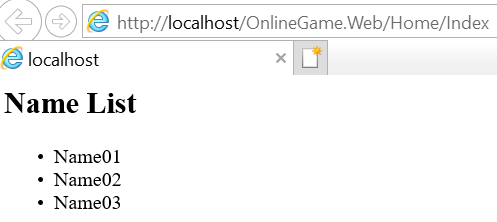
    @foreach (string strNames in ViewBag.Names)

    {

        <li>@strNames</li>

    }

</ul>



8. ViewData, ViewBag

8.1. Controllers/HomeController.cs

using System.Collections.Generic;

using System.Web.Mvc;

namespace OnlineGame.Web.Controllers

{

    public class HomeController : Controller

    {

        public string GetStringA()

        {

            return "AAAAAA";

        }

        //public string Index()

        //{

        //    return "Hello";

        //}

        //public string Index(string id)

        //{

        //    string queryString = Request.QueryString["name"];

        //    return $"Hey, Id={id} , name={queryString}";

        //}

        ////[http://localhost/onlinegame.web/home/index/aa?name=555](http://localhost/MVCSample/home/index/aa?name=555)

        //public string Index(string id, string name)

        //{

        //    // return string.Format("Hey, Id ={0} , name ={1}", id, name);

        //    return $"Hey, Id ={id} , name ={name}";

        //}

        //public List<string> Index()

        //{

        //    return new List<string>

        //    {

        //        "Name01",

        //        "Name02",

        //        "Name03"

        //    };

        //    // Return System.Collections.Generic.List`1[System.String]

        //    // This is Wrong.

        //}

        public ActionResult Index()

        {

            ////1.

            //ViewBag.Names = new List<string>

            //{

            //    "ViewBag.Names01",

            //    "ViewBag.Names02",

            //    "ViewBag.Names03"

            //};

            ////2.

            //ViewData["Names"] = new List<string>

            //{

            //    "ViewData[\"Names\"]01",

            //    "ViewData[\"Names\"]02",

            //    "ViewData[\"Names\"]03"

            //};

            ////3.

            //ViewBag.Names = new List<string>

            //{

            //    "ViewBag.Names01",

            //    "ViewBag.Names02",

            //    "ViewBag.Names03"

            //};

            //ViewData["Names"] = new List<string>

            //{

            //    "ViewData[\"Names\"]01",

            //    "ViewData[\"Names\"]02",

            //    "ViewData[\"Names\"]03"

            //};

            //4.

            ViewBag.Names = new List<string>

            {

                "ViewBag.Names01",

                "ViewBag.Names02",

                "ViewBag.Names03"

            };

            ViewData["Names2"] = new List<string>

            {

                "ViewData[\"Names\"]01",

                "ViewData[\"Names\"]02",

                "ViewData[\"Names\"]03"

            };

            return View();

        }

    }

}

/\*

1.

When we try to return a list of data,

E.g.

return new List<string>

{

    "Name01",

    "Name02",

    "Name03"

};

Then, it will only display the data type of the variable

E.g.

System.Collections.Generic.List`1[System.String]

This is not what we want,

thus, we need a view to display the data in the format we want.

2.

//public ActionResult Index()

//{

//    return View();

//}

ViewResult extend ViewResultBase

ViewResultBase extend ActionResult.

Thus, you can return View()

3.

In Home/HomeController.cs

//ViewBag.Names = new List<string>

//{

//    "ViewBag.Names01",

//    "ViewBag.Names02",

//    "ViewBag.Names03"

//};

//ViewData["Names2"] = new List<string>

//{

//    "ViewData[\"Names\"]01",

//    "ViewData[\"Names\"]02",

//    "ViewData[\"Names\"]03"

//};

In Views/HomeIndex.cshtml

//@foreach (string strNames1 in ViewBag.Names)

//{

//    <li>@strNames1</li>

//}

//<br/>

//<br/>

//@foreach (string strNames2 in (List<string>) ViewData["Names2"])

//{

//    <li>@strNames2</li>

//}

Both ViewData and ViewBag can pass values from Controller to View.

Both ViewData and ViewBag allow an object to have properties dynamically added to it.

Because of dynamic feature,

both ViewData and ViewBag does not provide compile time error checking.

Thus, it is very easy to get Null Reference Error

if misspells the property name or key name.

\*/

8.2. Views/Home/Index.cshtml

@model dynamic

@{

    ViewBag.Title = "title";

}

<h2>Name List</h2>

<ul>

    @\*

        1.

        @foreach (string strNames1 in ViewBag.Names)

        {

            <li>@strNames1</li>

        }

    \*@

    @\*

        2.

        @foreach (string strNames2 in (List<string>)ViewData["Names"])

        {

            <li>@strNames2</li>

        }

    \*@

    @\*

        3.

        @foreach (string strNames1 in ViewBag.Names)

        {

            <li>@strNames1</li>

        }

        <br />

        <br />

        @foreach (string strNames2 in (List<string>)ViewData["Names"])

        {

            <li>@strNames2</li>

        }

    \*@

    4.

    @foreach (string strNames1 in ViewBag.Names)

    {

        <li>@strNames1</li>

    }

    <br/>

    <br/>

    @foreach (string strNames2 in (List<string>) ViewData["Names2"])

    {

        <li>@strNames2</li>

    }

</ul>

Graphical user interface, text

Description automatically generated

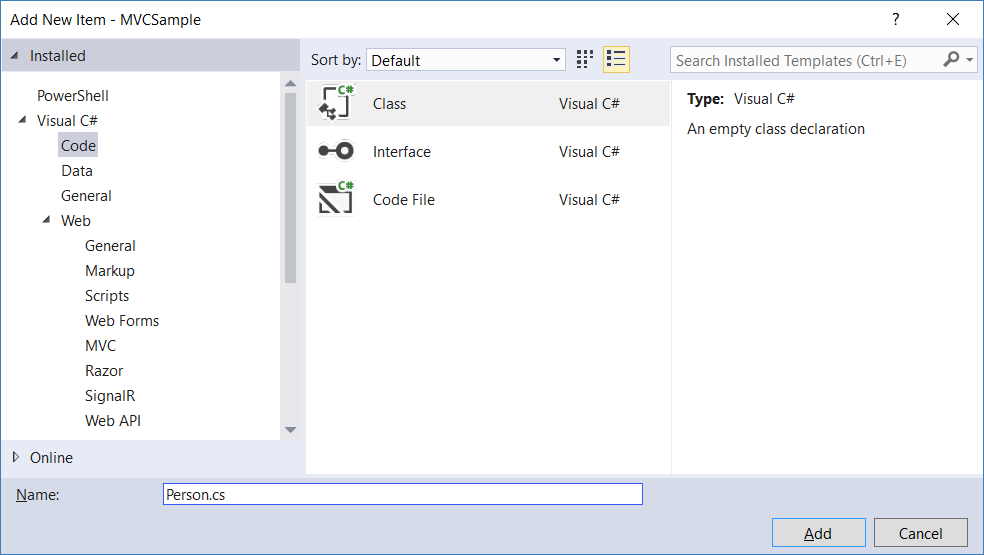
9. Model

9.1. Model : Model/Gamer.cs

Model folder --> Right Click  --> Add --> New Item  -->

Visual C# --> Code --> Class

Name : Gamer.cs



namespace OnlineGame.Web.Models

{

    public class Gamer

    {

        public int Id { get; set; }

        public string Name { get; set; }

        public string Gender { get; set; }

        public string City { get; set; }

    }

}

9.2. Controller : Controllers/GamerController.cs

Add New File

Mads Kristensen

3.5.129

Visual Studio 2013, 2015, 2017

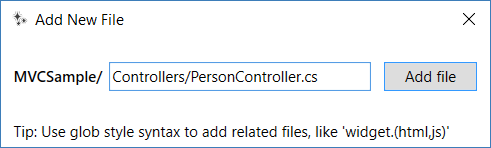
The fastest and easiest way to add new files to any project - including files that start with a dot

<https://marketplace.visualstudio.com/items?itemName=MadsKristensen.AddNewFile>

Because **Add New File** **(extension and update)**

press **Shift+F2**

**Controllers/GamerController.cs**



using System.Web.Mvc;

using OnlineGame.Web.Models;

namespace OnlineGame.Web.Controllers

{

    public class GamerController : Controller

    {

        public ActionResult Details()

        {

            var gamer = new Gamer()

            {

                Id = 1,

                Name = "Name1",

                Gender = "Male",

                City = "City1"

            };

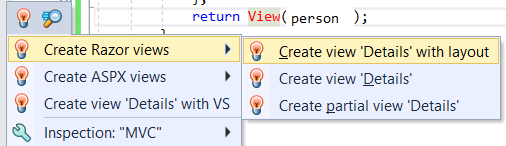
            return View(gamer);

        }

    }

}

Mouse point to View --> Alt + Enter --> Create Razor Views --> Create View 'Details' with layout



Otherwise

Because **Add New File** **(extension and update)**

press **Shift+F2**

**Views/Gamer/Details.cshtml**

Graphical user interface, text, application

Description automatically generated

9.3. View: Views/Gamer/Details.cshtml

@model OnlineGame.Web.Models.Gamer

@{

    ViewBag.Title = "Gamer Details";

}

<h2>Gamer Details</h2>

<table>

    <tr>

        <td>

            Gamer ID:

        </td>

        <td>

            @Model.Id

        </td>

    </tr>

    <tr>

        <td>

            Name:

        </td>

        <td>

            @Model.Name

        </td>

    </tr>

    <tr>

        <td>

            Gender:

        </td>

        <td>

            @Model.Gender

        </td>

    </tr>

    <tr>

        <td>

            City:

        </td>

        <td>

            @Model.City

        </td>

    </tr>

</table>

[http://localhost/OnlineGame.Web/Gamer/Details](http://localhost/MVCSample/Person/Details)

