(T3)手寫EntityFramework連接2Tables  
CourseGUID: 8503b39c-5887-4634-8291-facfb3117924  
=======================================================================  
(T3)手寫EntityFramework連接2Tables  
=======================================================================  
0. Summary

-----------

1. Create DB: OnlineGame

1.1. Create DB: OnlineGame

1.2. Create Table

1.3. Create Database level security login

-----------

2. New Project - OnlineGame

2.1. New Project - OnlineGame.Web

2.1.1. Global.asax.cs

2.1.2. App\_Start/RouteConfig.cs

2.2. Controllers/HomeController.cs

2.3. Views/Home/Index.cshtml

2.4. Entity Framework

2.4.1. Install Entity Framework

2.4.2. Web.config : Add Connection String

2.5. Models

2.5.1. Models/Gamer.cs

2.5.2. Models/Team.cs

2.6. Data/OnlineGameContext.cs

2.7. Team

2.7.1. Controllers/TeamController.cs

2.7.2. Views/Team/Index.cshtml

2.8. Gamer

2.8.1. Controllers/GamerController.cs

2.8.2. Views/Gamer/Index.cshtml

2.8.3. Views/Gamer/Details.cshtml

2.9. Run the Web  
=======================================================================

0. Summary

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

In this tutorial, we will discuss

\* Actionlink

\* EntityFramework with 2 Tables

In order to make you more familiar with MVC,

I want you to practice to create MVC project from zero backgrounds over and over again until later tutorial.

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

補充1

在我的課程有說

For some reason, gamers會是null

然後我還說我不知道為什麼

-->

其實是因為EF預設就是lazy loading

他不會去load navigation property裡面的值

請參考

<https://dotblogs.com.tw/yc421206/2014/04/18/144784>

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

1. Create DB: OnlineGame

1.1. Create DB: OnlineGame

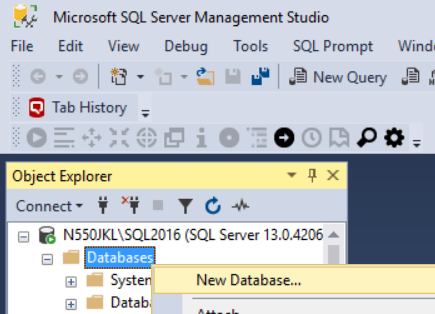
In SQL server Management Studio (SSMS)

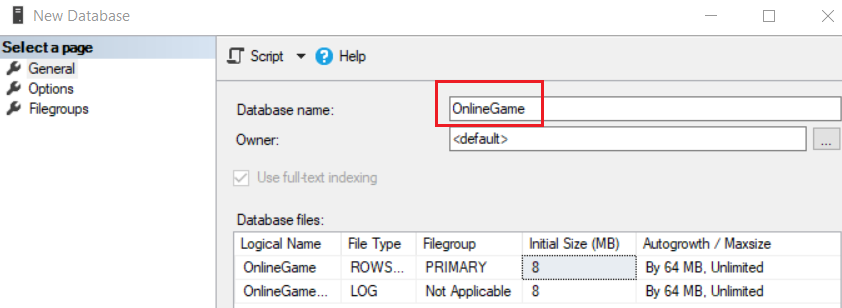
Database --> Right Click --> New Database -->

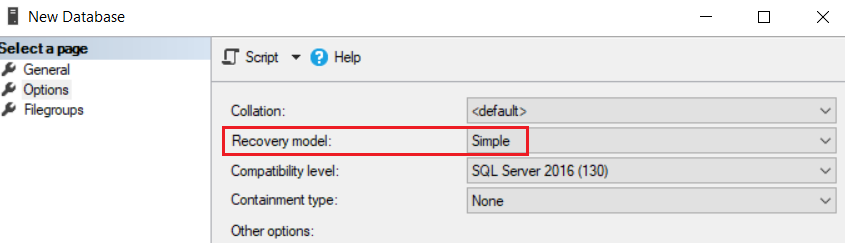
In General Tab -->

Name: **OnlineGame**

In options Tab --> Recovery model : **Simple**







1.2. Create Table

--Run the following query to create Gamer Table

--1. Drop Table if it exists

IF ( EXISTS ( SELECT    \*

              FROM      INFORMATION\_SCHEMA.TABLES

              WHERE     TABLE\_NAME = 'Gamer' ) )

    BEGIN

        TRUNCATE TABLE Gamer;

        DROP TABLE Gamer;

    END;

GO -- Run the previous command and begins new batch

IF ( EXISTS ( SELECT    \*

              FROM      INFORMATION\_SCHEMA.TABLES

              WHERE     TABLE\_NAME = 'Team' ) )

    BEGIN

        TRUNCATE TABLE Team;

        DROP TABLE Team;

    END;

GO -- Run the previous command and begins new batch

--2. Create Table

CREATE TABLE Team

    (

      Id INT PRIMARY KEY

             IDENTITY(1, 1)

             NOT NULL ,

      [Name] NVARCHAR(100) NULL

    );

GO -- Run the previous command and begins new batch

CREATE TABLE Gamer

    (

      Id INT PRIMARY KEY

             IDENTITY(1, 1)

             NOT NULL ,

      [Name] NVARCHAR(100) NULL ,

      Gender NVARCHAR(10) NULL ,

      City NVARCHAR(50) NULL,

         DateOfBirth DATETIME NULL,

         TeamId INT FOREIGN KEY REFERENCES Team(Id)

    );

GO -- Run the previous command and begins new batch

--3. Insert Data

INSERT  Team

VALUES  ( N'Team1');

INSERT  Team

VALUES  ( N'Team2');

INSERT  Team

VALUES  ( N'Team3');

INSERT  Gamer

VALUES  ( N'Name01 ABB', N'Male', N'City01', '1979/4/28', 1);

INSERT  Gamer

VALUES  ( N'Name02 CDDE', N'Female', N'City03', '1981/7/24', 2);

INSERT  Gamer

VALUES  ( N'Name03 FIJK', N'Female', N'City01', '1984/12/5', 3);

INSERT  Gamer

VALUES  ( N'Name04 LMOPPQ', N'Male', N'City02', '1983/5/29', 1);

INSERT  Gamer

VALUES  ( N'Name05 QRSTT', N'Male', N'City01', '1979/6/20', 3);

INSERT  Gamer

VALUES  ( N'Name06 TUVVX', N'Female', N'City03', '1984/5/15', 3);

INSERT  Gamer

VALUES  ( N'Name07 XYZZXX', N'Female', N'City01', '1986/4/29', 2);

INSERT  Gamer

VALUES  ( N'Name08 ABBCDE', N'Male', N'City02', '1985/7/28', 1);

INSERT  Gamer

VALUES  ( N'Name09 QRSTTUVXX', N'Male', N'City02', '1983/4/16', 1);

GO -- Run the previous command and begins new batch

1.3. Create Database level security login

In SQL server

Object Explorer --> Security --> Logins --> New Logins

-->

General Tab

Login Name :

**Tester**

Password:

**1234**

Default Database:

**OnlineGame**

-->

Server Roles Tab

Select

**sysadmin**

-->

User Mapping Tab

Select **OnlineGame**

Select every single role.









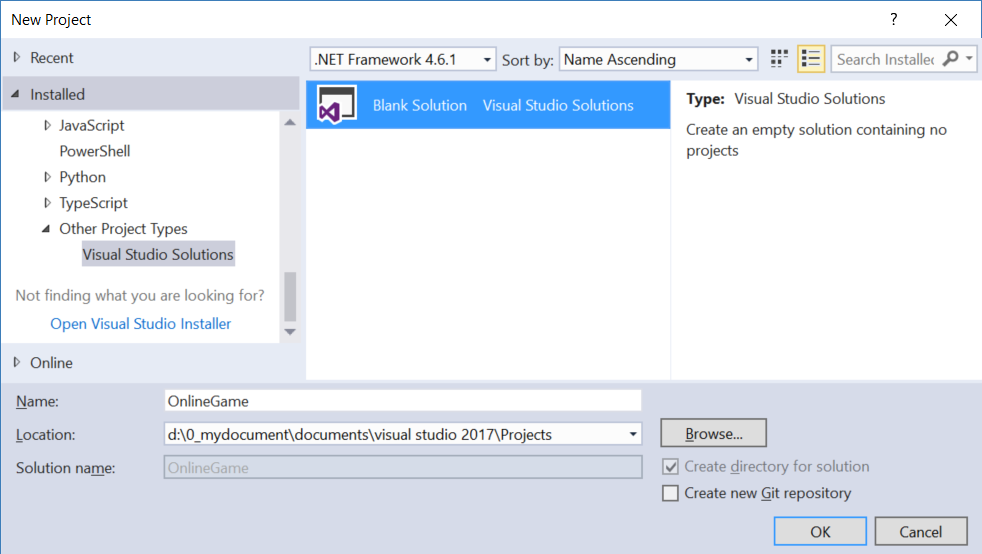
2. New Project - OnlineGame

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

Other Project Types --> Visual Studio Solutions -->  Blank Solution

-->

Name: **OnlineGame**



2.1. New Project - OnlineGame.Web

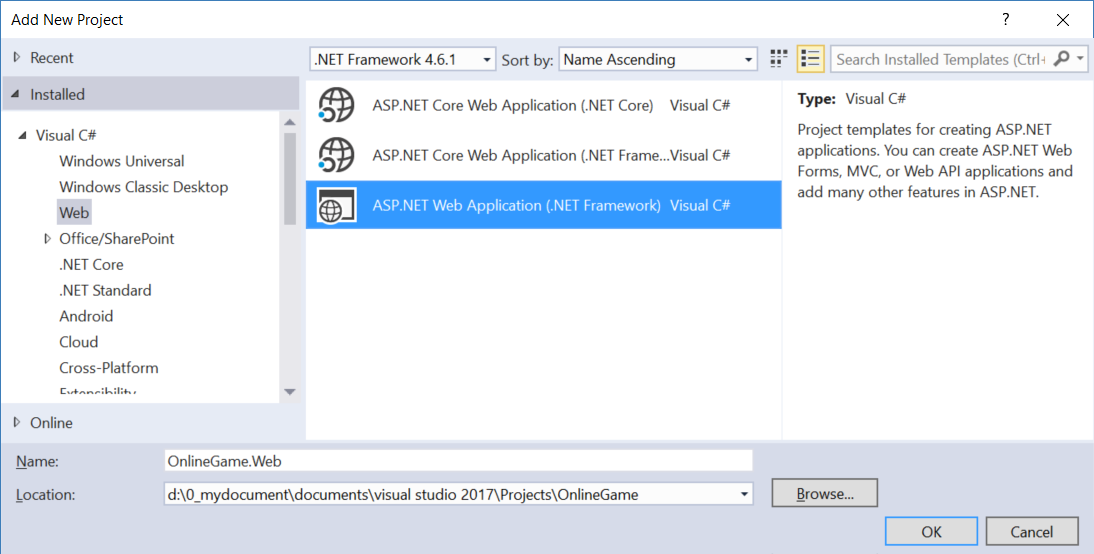
Solutions Name --> Add --> New Project -->

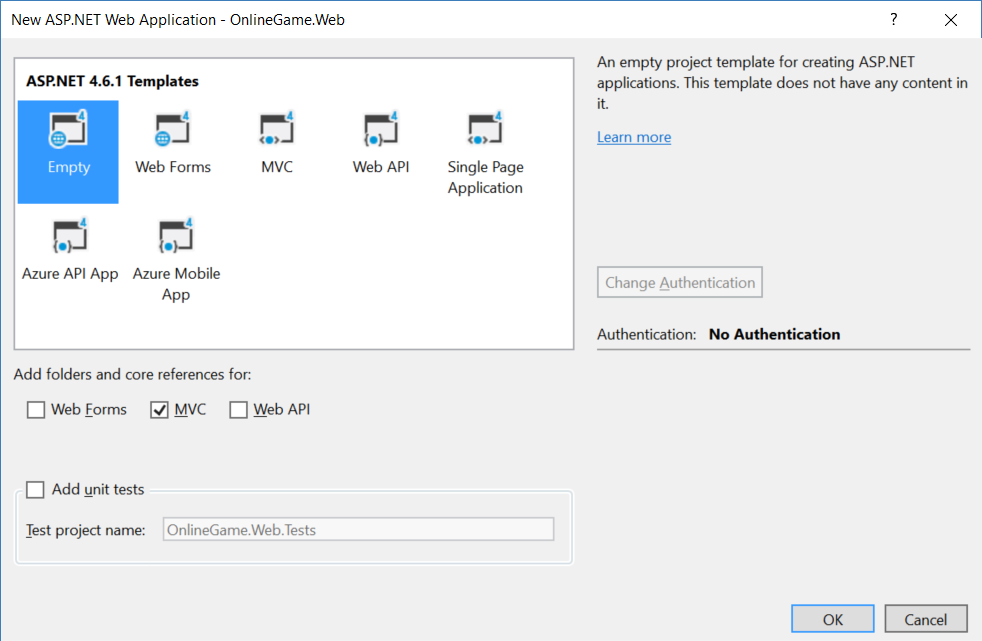
Visual C# --> Web --> [ASP.NET](http://asp.net/)Web Application (.Net Framework)

-->

Name: **OnlineGame.Web**

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





2.1.1. 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);

        }

    }

}

2.1.2. App\_Start/RouteConfig.cs

using System.Web.Mvc;

using System.Web.Routing;

namespace OnlineGame.Web

{

    public class RouteConfig

    {

        public static void RegisterRoutes(RouteCollection routes)

        {

            //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 }

            );

        }

    }

}

/\*

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.

-----------------------------------

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/OnlineGame.Web/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.

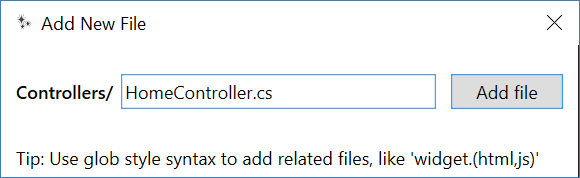
\*/

2.2. Controllers/HomeController.cs

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

press **Shift+F2**

**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";

        //}

        ////<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 ActionResult Index()

        //{

        //    return View();

        //}

        //public ActionResult Index()

        //{

        //    ViewBag.Names = new List<string>

        //        {

        //            "Name01",

        //            "Name02",

        //            "Name03"

        //        };

        //    return View();

        //}

        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();

        }

        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()

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 miss misspells the property name or key name.

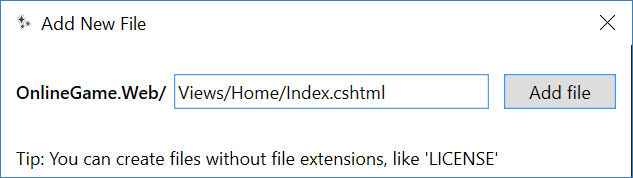
\*/

2.3. Views/Home/Index.cshtml

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

press **Shift+F2**

**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>

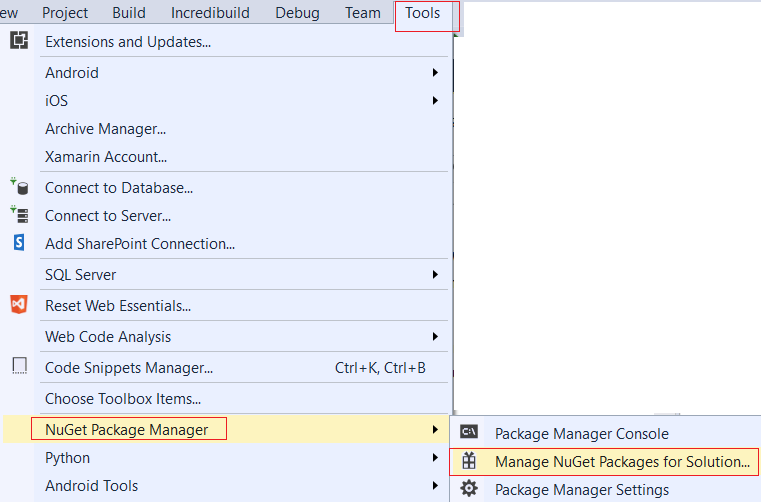
2.4. Entity Framework

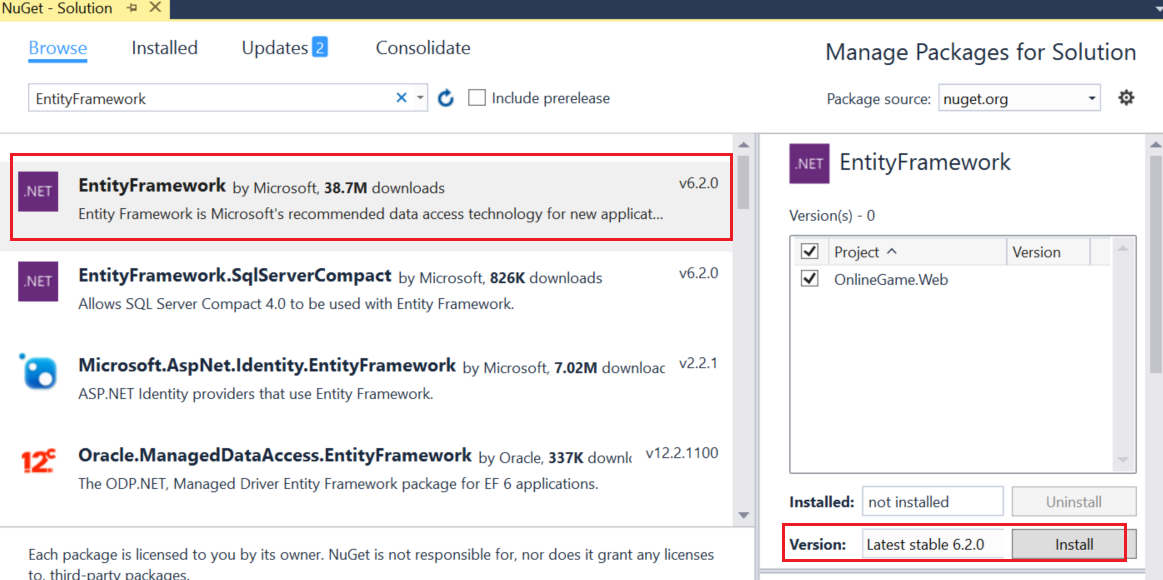
2.4.1. Install Entity Framework

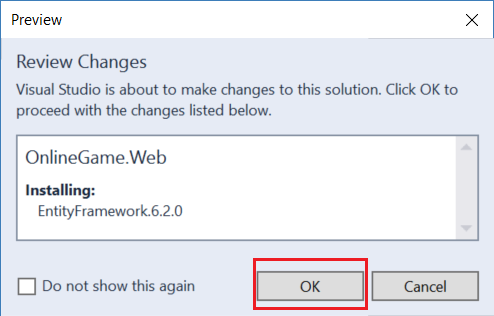
Tools --> NuGet Package Manager --> Manage NuGet Packages for Solutions...

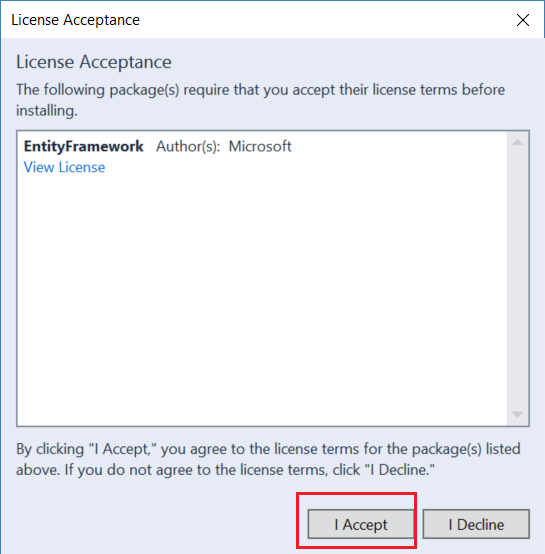
--> Browse tab --> Search  :  **EntityFramework**

--> Install it



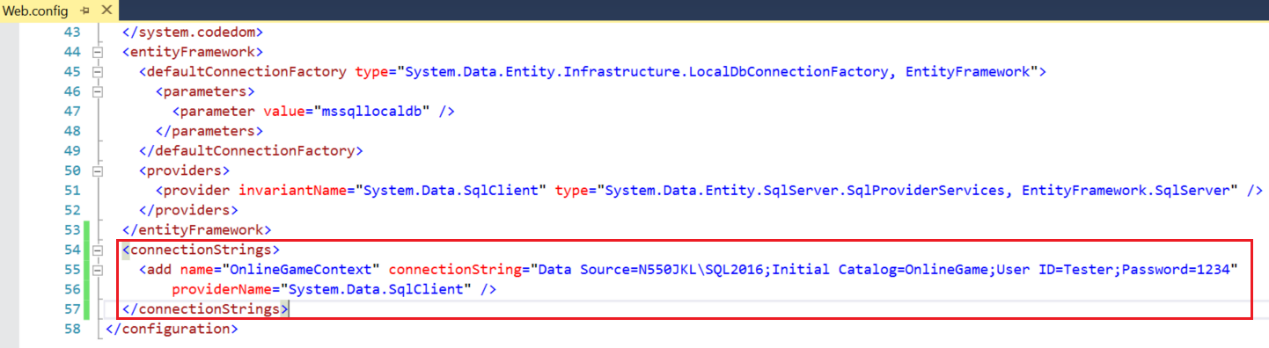






2.4.2. Web.config : Add Connection String

If you have already install EntityFramework dll, then you may skip this step.





<connectionStrings>

    <add name="OnlineGameContext" connectionString="Data Source=N550JKL\SQL2016;Initial Catalog=OnlineGame;User ID=Tester;Password=1234"

        providerName="System.Data.SqlClient" />

  </connectionStrings>

Go to Web.config

Add the connection sting on the bottom of Web.config

Please ensure the connection string name "**OnlineGameContext**" must match the DBContext Name "**OnlineGameContext**" in next section.

OnlineGameContext class extends DbContext.

When we create an instance object of OnlineGameContext,

it will search the connection string with the name "OnlineGameContext"  within web config.

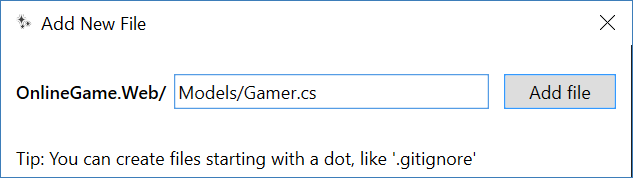
2.5. Models

2.5.1. Models/Gamer.cs

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

press **Shift+F2**

**Models/Gamer.cs**



using System;

using System.ComponentModel.DataAnnotations.Schema;

namespace OnlineGame.Web.Models

{

    //[Table("Gamer"]

    [Table("Gamer", Schema = "dbo")]

    public class Gamer

    {

        public int Id { get; set; }

        [Column("Name")]

        public string Name { get; set; }

        public string Gender { get; set; }

        public string City { get; set; }

        public DateTime DateOfBirth { get; set; }

        public int TeamId { get; set; }

    }

}

/\*

1.

////[Table("Gamer"]

//[Table("Gamer", Schema = "dbo")]

//...

//[Column("Name")]

//public string Name { get; set; }

Reference:

<http://www.entityframeworktutorial.net/code-first/table-dataannotations-attribute-in-code-first.aspx>

<http://www.entityframeworktutorial.net/code-first/column-dataannotations-attribute-in-code-first.aspx>

In order to map the Gamer Table entity into the Model, Models/Gamer.cs.

You need to add the [Table("Gamer")] attribute  in the class level.

Then the EntityFramework will automatically map the Gamer table fields into Gamer Model properties.

Gamer table field, id will automatically map to Gamer Model properties, id,

because it has the same name.

If you want to map 2 different name, then you need [Column("Name")] attribute.

//[Column("Name")]

//public string Name2 { get; set; }

This [Column("Name")] attribute

will map Table Column, Name, to Model Property, Name2.

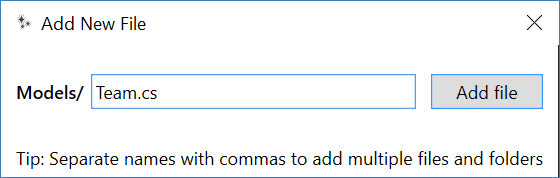
\*/

2.5.2. Models/Team.cs

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

press **Shift+F2**

**Models/Team.cs**



using System.Collections.Generic;

using System.ComponentModel.DataAnnotations.Schema;

namespace OnlineGame.Web.Models

{

    //[Table("Gamer"]

    [Table("Team", Schema = "dbo")]

    public class Team

    {

        public int Id { get; set; }

        [Column("Name")]

        public string Name { get; set; }

        public List<Gamer> Gamers { get; set; }

    }

}

2.6. Data/OnlineGameContext.cs

using System.Data.Entity;

using OnlineGame.Web.Models;

namespace OnlineGame.Web.Data

{

    public class OnlineGameContext : DbContext

    {

        public DbSet<Gamer> Gamers { get; set; }

        public DbSet<Team> Teams { get; set; }

    }

}

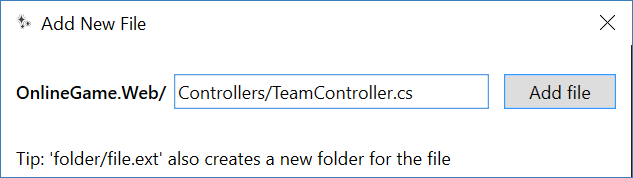
2.7. Team

2.7.1. Controllers/TeamController.cs

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

press **Shift+F2**

**Controllers/TeamController.cs**



using System.Collections.Generic;

using System.Linq;

using System.Web.Mvc;

using OnlineGame.Web.Data;

using OnlineGame.Web.Models;

namespace OnlineGame.Web.Controllers

{

    public class TeamController : Controller

    {

        public ActionResult Index()

        {

            OnlineGameContext context = new OnlineGameContext();

            List<Team> teams = context.Teams.ToList();

            return View(teams);

        }

    }

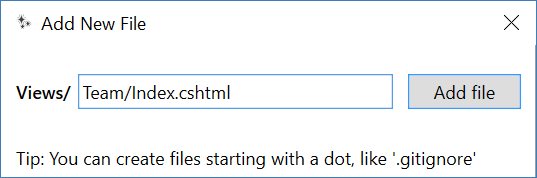
}

2.7.2. Views/Team/Index.cshtml

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

press **Shift+F2**

**Views/Team/Index.cshtml**



@using OnlineGame.Web.Models;

@model IEnumerable<Team>

@{

    ViewBag.Title = "Teams List";

}

<div style="font-family: Arial">

    <h2>@ViewBag.Title.ToString()</h2>

        @foreach (Team team in Model)

        {

            <ul>

                <li>

                    Id: @team.Id

                </li>

                <li>

                    @Html.ActionLink(team.Name, "Index", "Gamer", new {teamId = team.Id}, null)

                </li>

            </ul>

        }

</div>

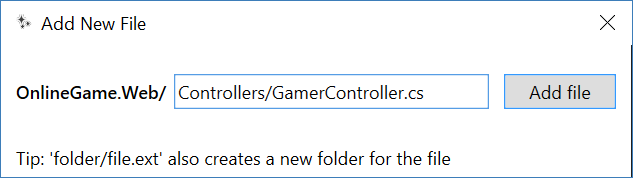
2.8. Gamer

2.8.1. Controllers/GamerController.cs

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

press **Shift+F2**

**Controllers/GamerController.cs**



using System.Collections.Generic;

using System.Linq;

using System.Web.Mvc;

using OnlineGame.Web.Data;

using OnlineGame.Web.Models;

namespace OnlineGame.Web.Controllers

{

    public class GamerController : Controller

    {

        // <http://localhost/OnlineGame.Web/Gamer/Details>

        //public ActionResult Details()

        //{

        //    var gamer = new Gamer

        //    {

        //        Id = 1,

        //        Name = "Name1",

        //        Gender = "Male",

        //        City = "City1"

        //    };

        //    return View(gamer);

        //}

        // <http://localhost/OnlineGame.Web/Gamer/Details>

        // <http://localhost/OnlineGame.Web/Gamer/Details/1>

        // <http://localhost/OnlineGame.Web/Gamer/Details/2>

        // <http://localhost/OnlineGame.Web/Gamer/Details/3>

        // <http://localhost/OnlineGame.Web/Gamer/Details/4>

        public ActionResult Details(int id = 0)

        {

            var onlineGameContext = new OnlineGameContext();

            Gamer gamer;

            if (id == 0)

            {

                gamer = new Gamer

                {

                    Id = 0,

                    Name = "Name0",

                    Gender = "NULL",

                    City = "NULL"

                };

                // or you may throw exception here.

            }

            else

            {

                gamer = onlineGameContext.Gamers.Single(p => p.Id == id);

                //Throws exception if can not find the single entity

            }

            return View(gamer);

        }

        public ActionResult Index(int teamId)

        {

            OnlineGameContext context = new OnlineGameContext();

            List<Gamer> gamers = context.Gamers.Where(gamer => gamer.TeamId == teamId).ToList();

            return View(gamers);

        }

    }

}

/\*

//var onlineGameContext = new OnlineGameContext();

//Gamer gamer = onlineGameContext.Gamers.Single(p => p.Id == id);

When user request, EntityFramework will request the data from the database

and sotre its data into a temp place called DBSet.

onlineGameContext.Gamers is a DBSet which is kind of temp place to store the Gamer Table Data.

We use LINQ to map the Gamer Table Column id to Gamer Model property, id.

Thus, we can get the gamer entity from Gamer Table by its id.

Then store gamer entity data into Gamer Model object.

Thus, each Gamer Model object is a temp place to store each Gamer Table entity from the database.

Then we pass the Gamer Model object as the ViewModel,

Thus, the Details.cshtml view can use the values from Gamer Model object

which is actually the temp place to store Gamer Table entity data.

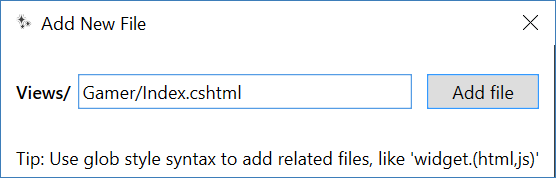
\*/

2.8.2. Views/Gamer/Index.cshtml

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

press **Shift+F2**

**Views/Gamer/Index.cshtml**



@using OnlineGame.Web.Models;

@model List<Gamer>

@{

    ViewBag.Title = "Gamers List";

}

<div style="font-family: Arial">

    <h2>@ViewBag.Title.ToString()</h2>

    @foreach (Gamer gamer in Model)

    {

        <ul>

            <li>

                Team Id: @gamer.TeamId

            </li>

            <li>

                Gamer Id: @gamer.Id

            </li>

            <li>

                Name: @Html.ActionLink(gamer.Name, "Details", "Gamer", new { id = gamer.Id }, null)

            </li>

        </ul>

        <br />

    }

</div>

@Html.ActionLink("Back to Gamers List", "Index", "Team")

2.8.3. Views/Gamer/Details.cshtml

@using OnlineGame.Web.Models;

@model Gamer

@{

    ViewBag.Title = "Gamer Details";

}

<h2>Gamer Details</h2>

<ul>

    <li>Id: @Model.Id</li>

    <li>Name: @Model.Name</li>

    <li>Gender: @Model.Gender</li>

    <li>City: @Model.City</li>

    <li>DateOfBirth: @Model.DateOfBirth</li>

    <li>TeamId: @Model.TeamId</li>

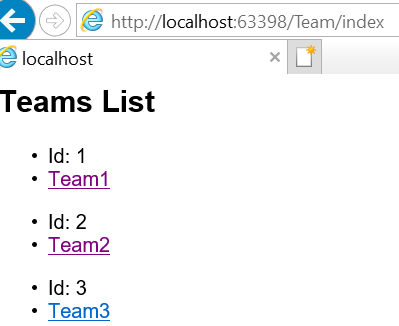
</ul>

@Html.ActionLink("Back to Gamers List", "Index", "Gamer", new { teamId = Model.TeamId }, null)

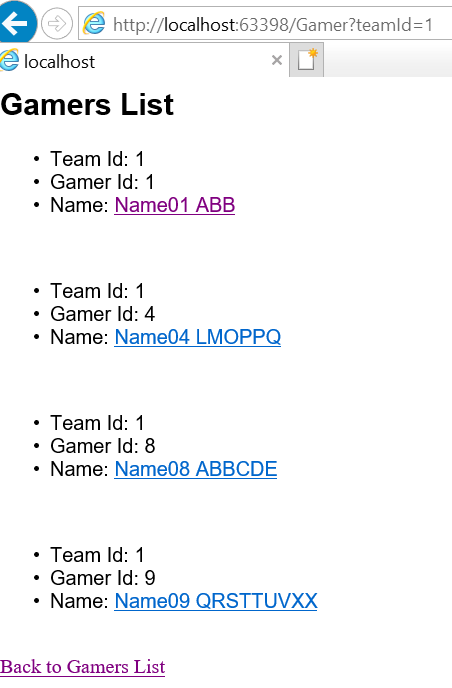
2.9. Run the Web

<http://localhost:63398/Team>

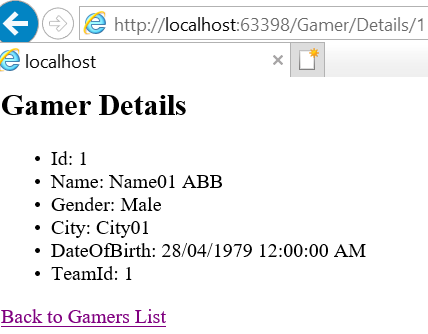
<http://localhost:63398/Team/index>



<http://localhost:63398/Gamer?teamId=1>



<http://localhost:63398/Gamer/Details/1>



<http://localhost:63398/>

