(T1)自動生成 LinqToSQL 搭配 AspNetWebForm。討論 SqlProfiler 監視資料庫
CourseGUID: 5ba9a6fe-7475-4b0c-8b99-bbcf7f5e2e1c
==========
(T1)自動生成 LinqToSQL 搭配 AspNetWebForm。討論 SqlProfiler 監視資料庫
==========

==========

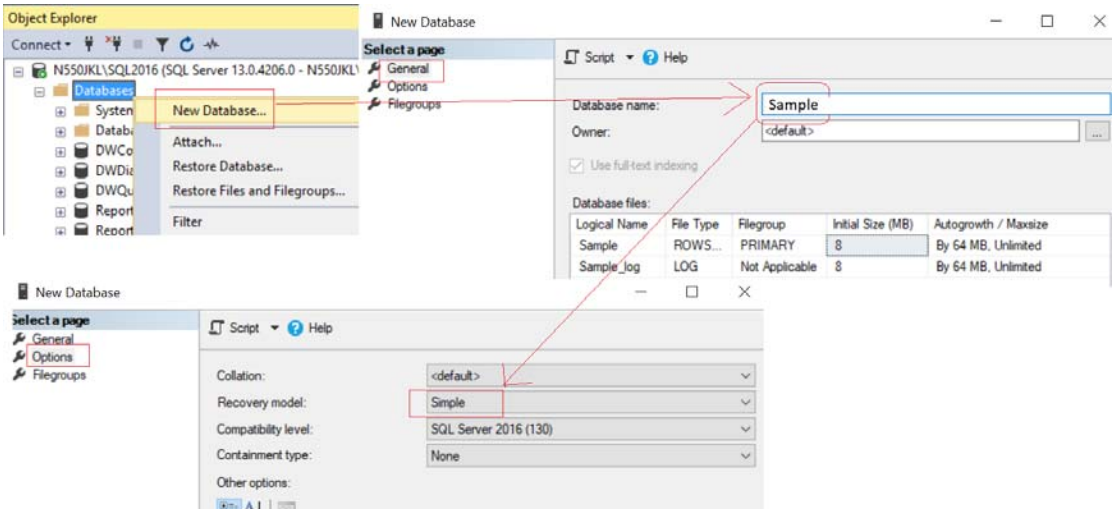
0 Summary

1.
1.1.
Language Integrated 整體 Query (LINQ) is a component between the LINQ query and the actual data source which includes SQL Server, XML documents, Objects in memory etc.
E.g. Linq to SQL provider can convert a Linq query to TSQL.
1.2.
LINQ query can be written by any .NET supported programming language, and it provides compile time error checking.

1. Web Form Application - Linq Query

1.1. TSQL

Database --> Right Click --> New Database -->
Database Name : Sample
Options --> Recovery Model : Simple



--Create an Sample DataBase and Run the following TSQL

```
Create Table Gamer
(
      Id int primary key IDENTITY(1,1),
     Name nvarchar(100),
      Gender nvarchar(50)
)
GO
Insert into Gamer values ('Name01', 'Male')
Insert into Gamer values ('Name02', 'Female')
Insert into Gamer values ('Name03', 'Male')
Insert into Gamer values ('Name04', 'Female')
Insert into Gamer values ('Name05', 'Female')
GO
```

1.2. Set up SQL Authentication

In SQL server

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

-->

General Tab

Login Name :

Tester

Password:

1234

Default Database:

Sample

-->

Server Roles Tab

Select

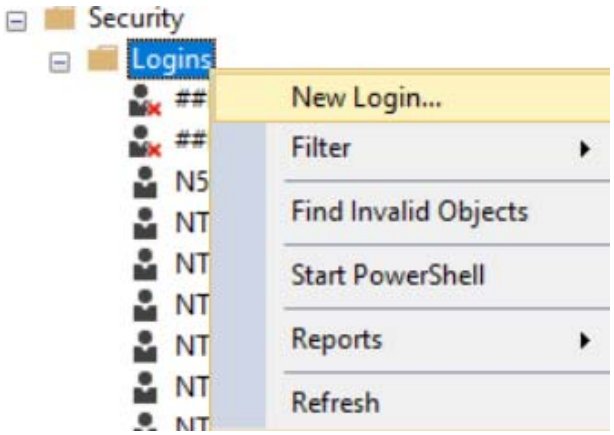
sysadmin

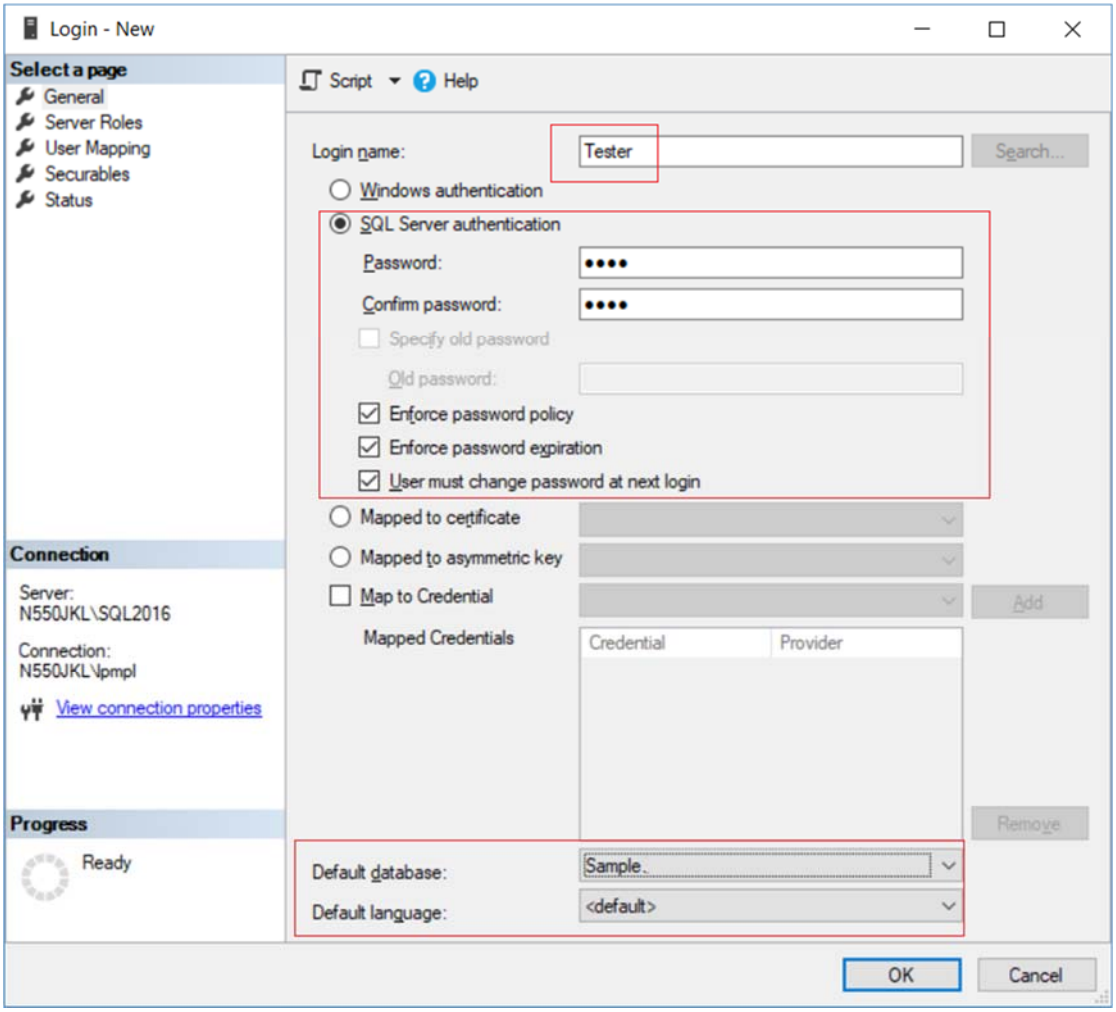
-->

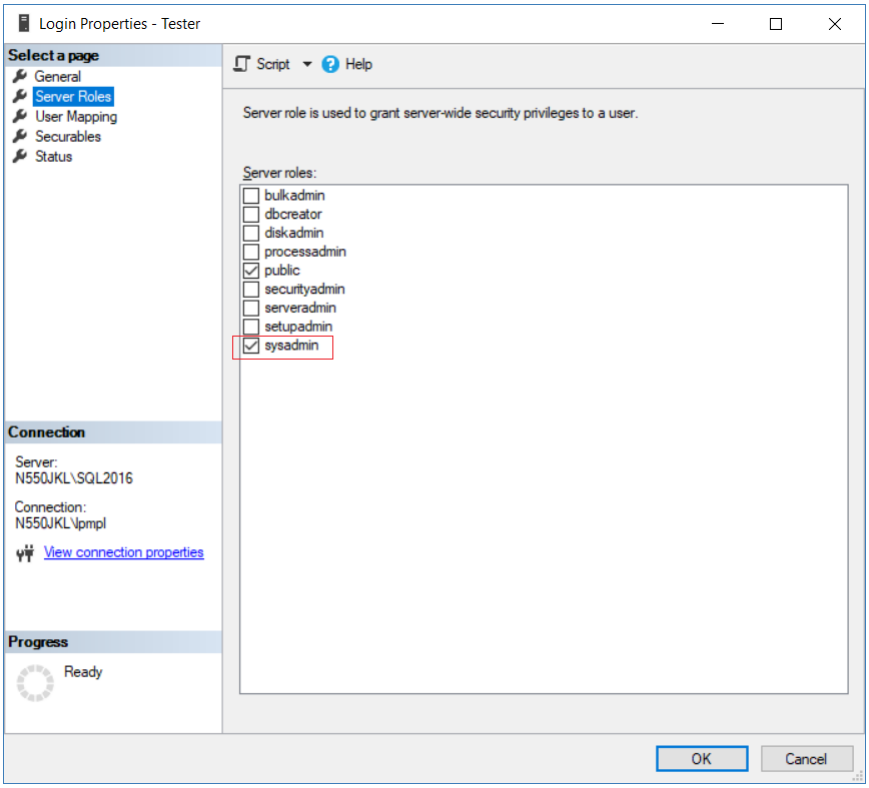
User Mapping Tab

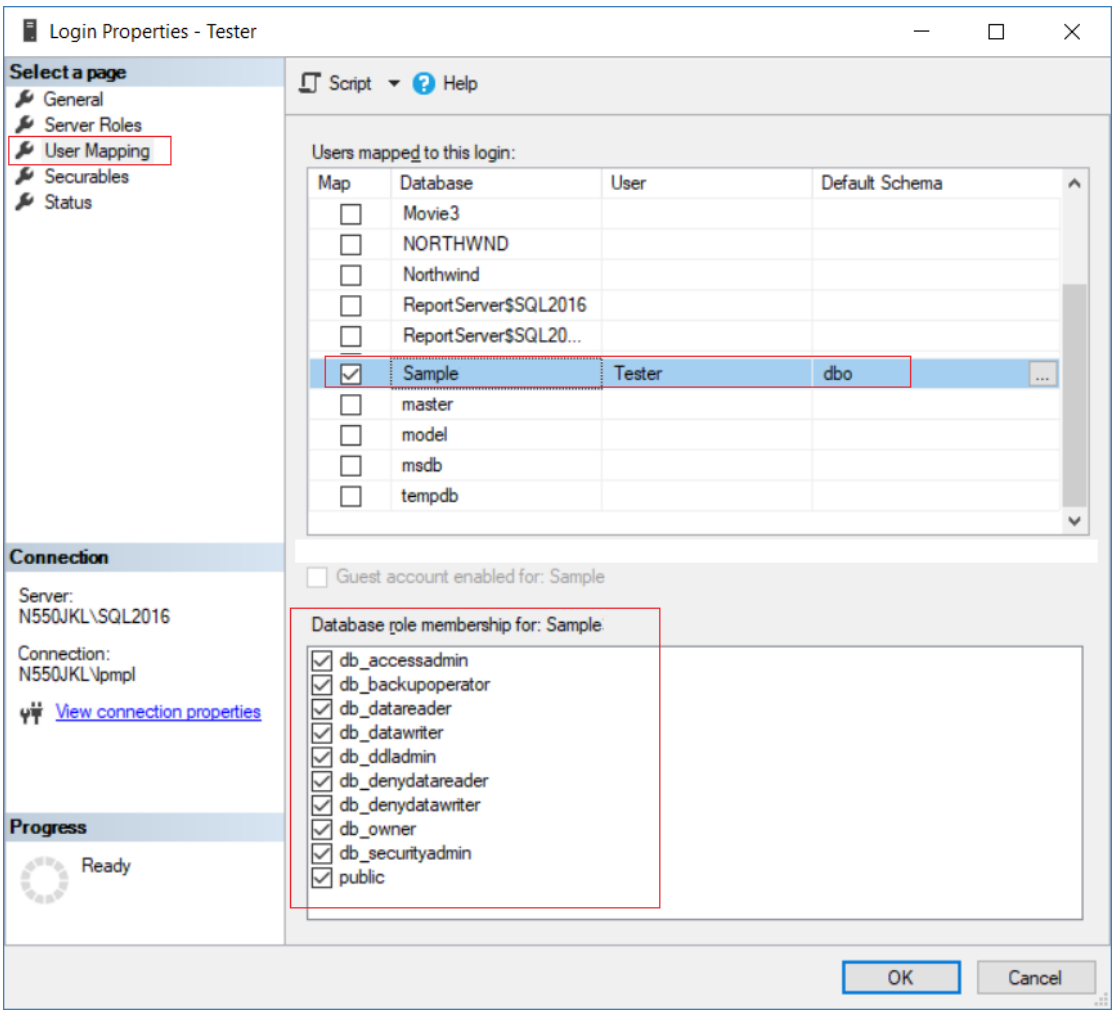
Select **Sample**

Select every Roles.







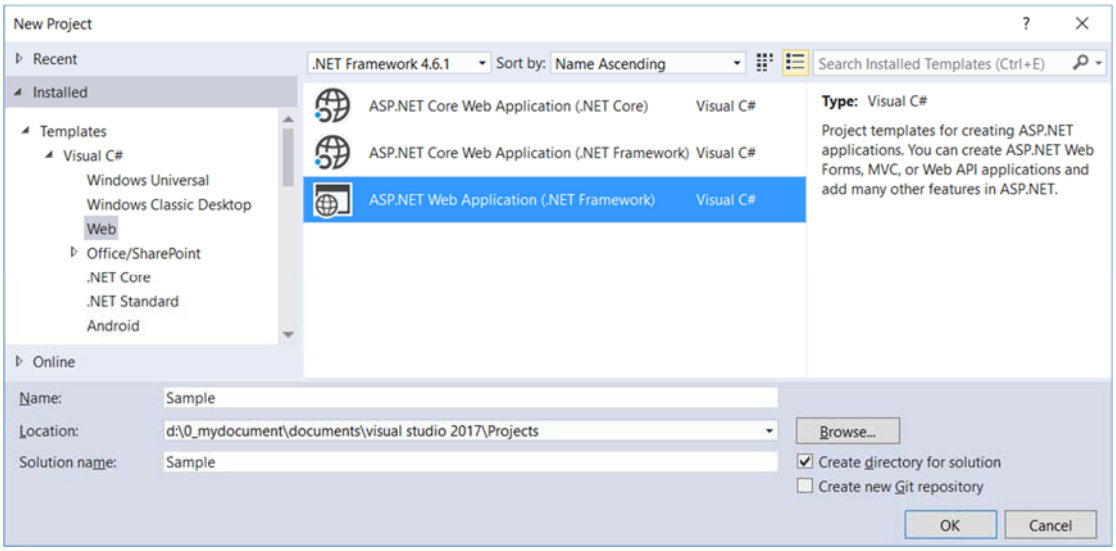


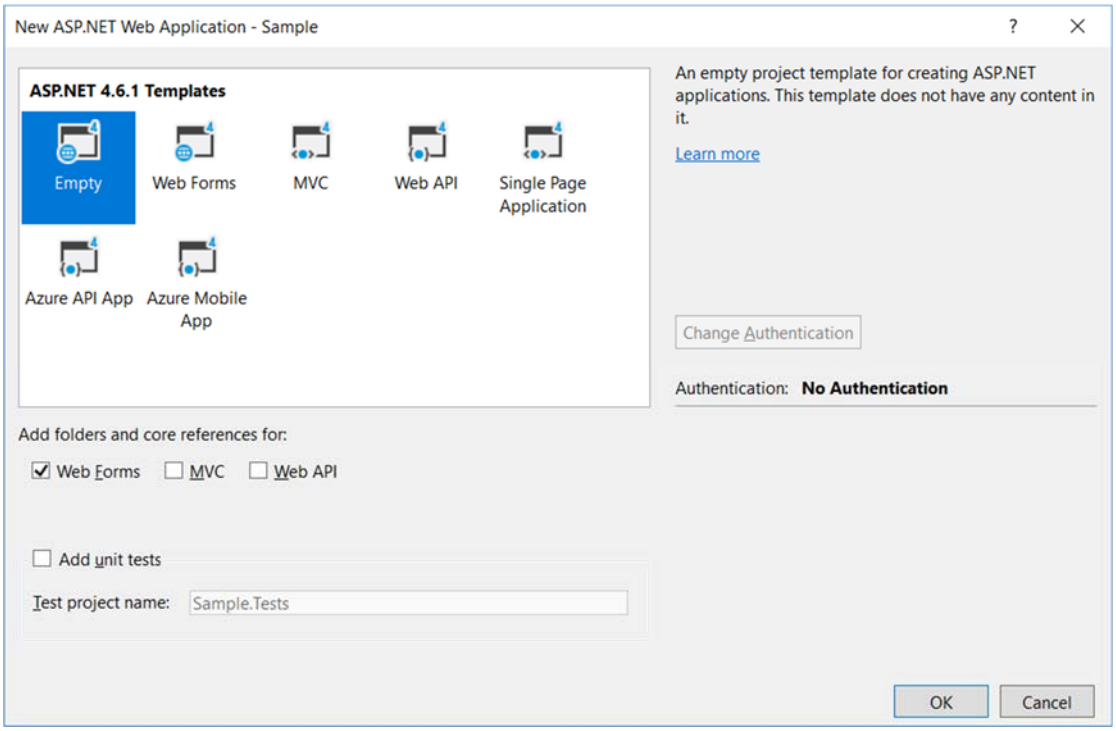
1.3. Create Web Application

Open Visual Studio, I am currently using VS2017

If you don't have it, you may following the instruction here to download.
http://ithandyguytutorial.blogspot.com/2017/10/ch00install-visual-studio-2017-offline.html

New Project --> Web --> ASP.NET **Web Application (.Net Framework)**
-->
Name:
Sample
--> **Empty** --> Select "**Web Forms**" --> OK





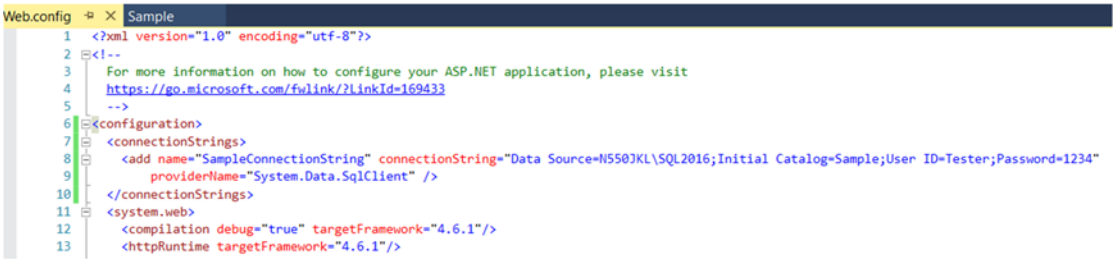
1.4. Code

1.4.1. Web.config

Add connection String

```
<configuration>
  <connectionStrings>
```

```
    <add name="SampleConnectionString" connectionString="Data Source=N550JKL\SQL2016;Initial
Catalog=Sample;User ID=Tester;Password=1234"
         providerName="System.Data.SqlClient" />
  </connectionStrings>
```



1.4.2. WebForm1.aspx

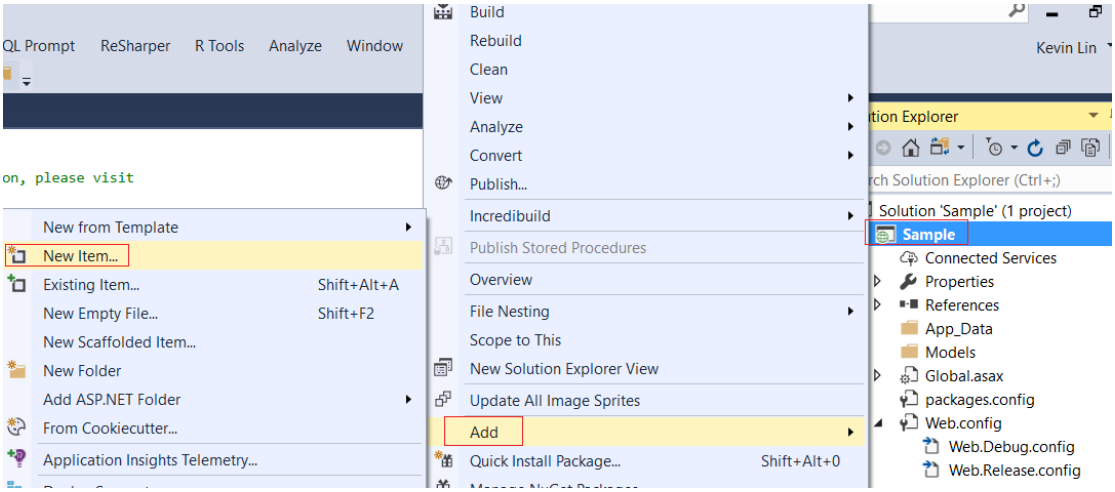
ProjectName --> Right Click --> Add --> New Item...

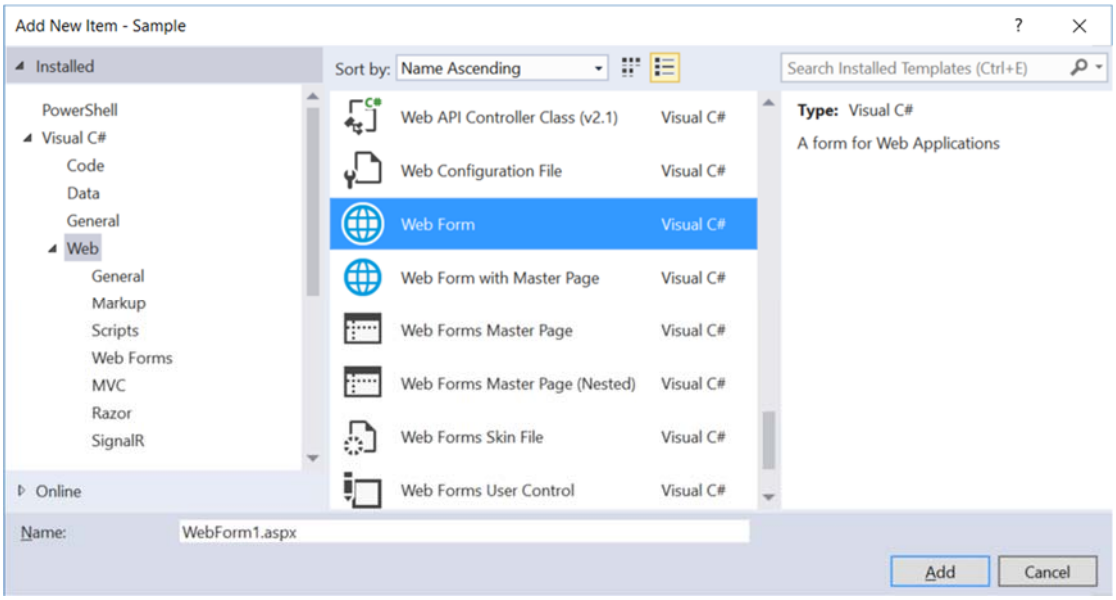
-->

WebForm

Name :

WebForm1.aspx

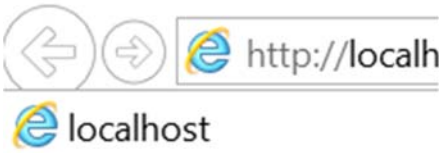




```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="WebForm1.aspx.cs" Inherits="Sample.WebForm1" %>
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            <asp:GridView ID="GridView1" runat="server"></asp:GridView>
        </div>
    </form>
</body>
</html>
```

1.4.3. WebForm1.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Configuration;
using System.Data.SqlClient;
namespace Sample
{
    public partial class WebForm1 : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            string cs = ConfigurationManager.ConnectionStrings["SampleConnectionString"].ConnectionString;
            SqlConnection sqlConnection = new SqlConnection(cs);
            SqlCommand sqlCommand = new SqlCommand
                ("Select Id, Name, Gender from Gamer where Gender='Female'", sqlConnection);
            List<GamerA> listGamers = new List<GamerA>();
            sqlConnection.Open();
            SqlDataReader sqlDataReader = sqlCommand.ExecuteReader();
            while (sqlDataReader.Read())
            {
                GamerA gamerItem = new GamerA();
                gamerItem.Id = Convert.ToInt32(sqlDataReader["Id"]);
                gamerItem.Name = sqlDataReader["Name"].ToString();
                gamerItem.Gender = sqlDataReader["Gender"].ToString();
                listGamers.Add(gamerItem);
            }
            sqlConnection.Close();
            GridView1.DataSource = listGamers;
            GridView1.DataBind();
        }
    }
    public class GamerA
    {
        public int Id { get; set; }
        public string Name { get; set; }
        public string Gender { get; set; }
    }
}
```

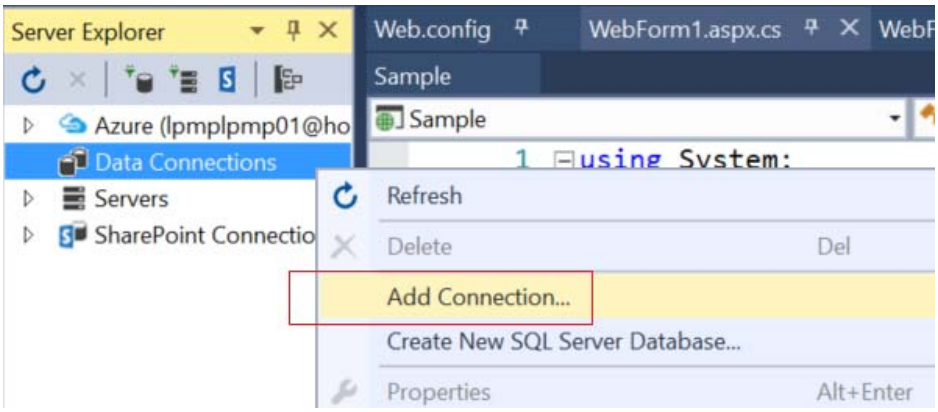


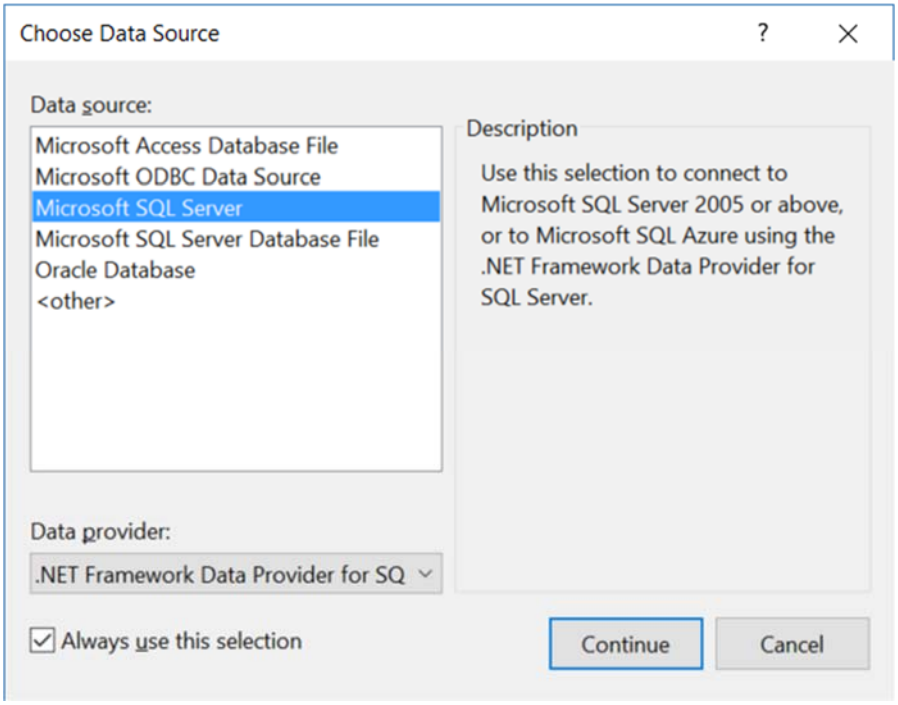
Id	Name	Gender
2	Name02	Female
4	Name04	Female
5	Name05	Female

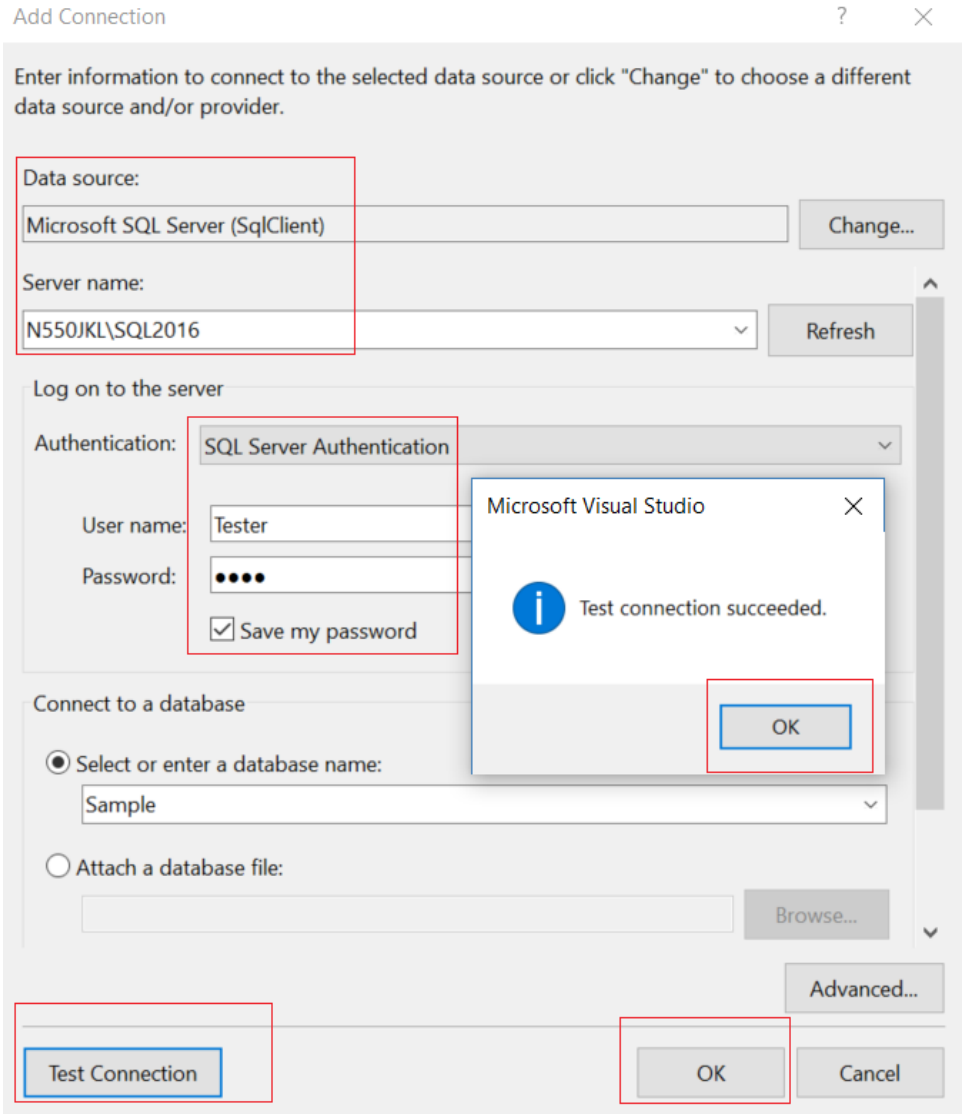
2. Linq to SQL

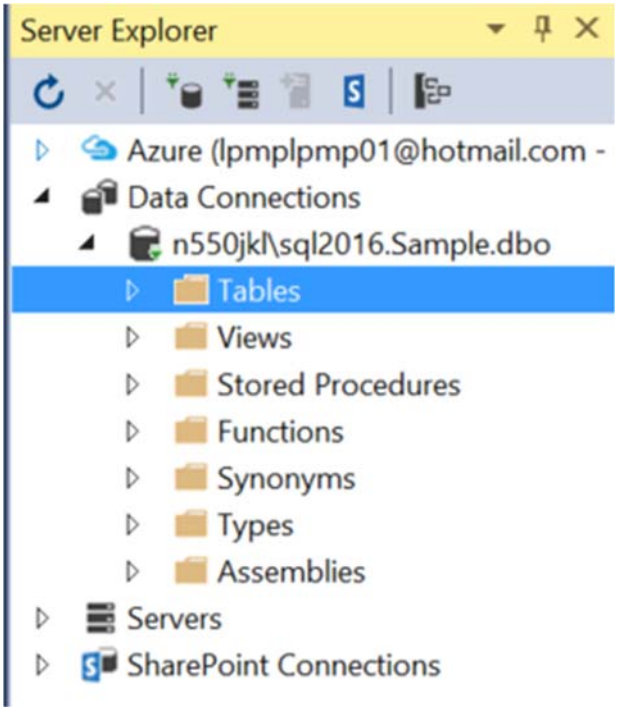
2.1. Add Connection

Server Explorer --> Data Connections --> Right click --> Add Connection...
--> Microsoft SQL server -->
Enter your server and database details ....




Choose Data Source
Data source:
Microsoft Access Database File
Microsoft ODBC Data Source
Microsoft SQL Server
Microsoft SQL Server Database File
Oracle Database
<other>
Description
Use this selection to connect to Microsoft SQL Server 2005 or above, or to Microsoft SQL Azure using the .NET Framework Data Provider for SQL Server.
Data provider:
.NET Framework Data Provider for SQ
Always use this selection
Continue
Cancel

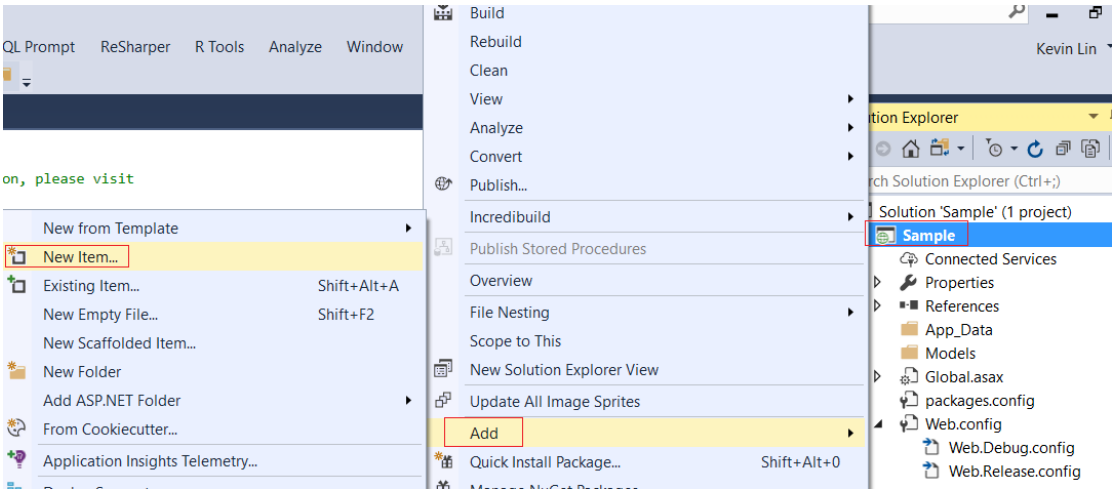

Add Connection
Enter information to connect to the selected data source or click "Change" to choose a different data source and/or provider.
Data source:
Microsoft SQL Server (SqlClient)
Change...
Server name:
N550JKL\SQL2016
Refresh
Log on to the server
Authentication:
SQL Server Authentication
User name:
Tester
Password:
Save my password
Microsoft Visual Studio
Test connection succeeded.
OK
Connect to a database
Select or enter a database name:
Sample
Attach a database file:
Browse...
Advanced...
Test Connection
OK
Cancel

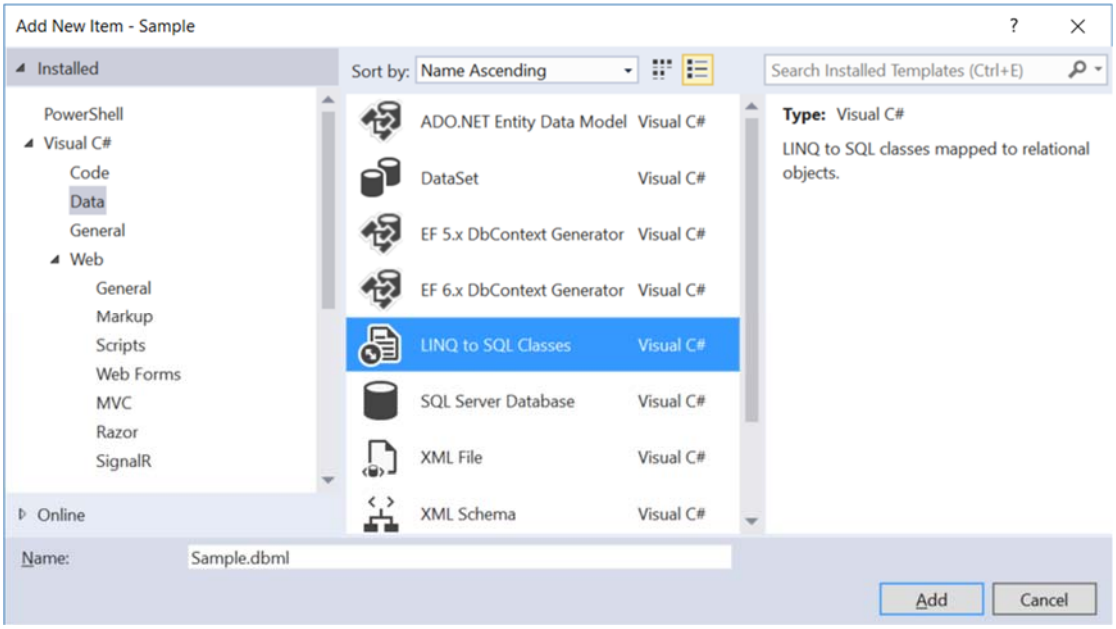


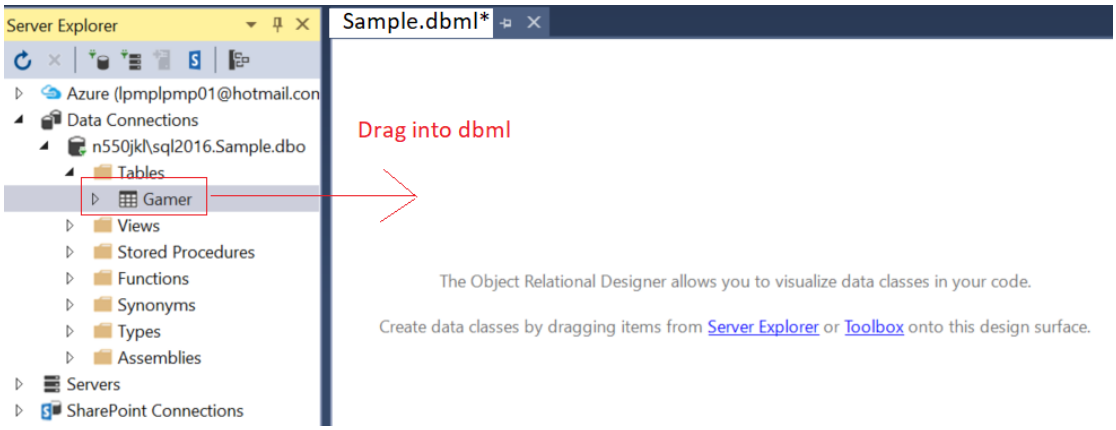
2.2. DataClasses1.dbml

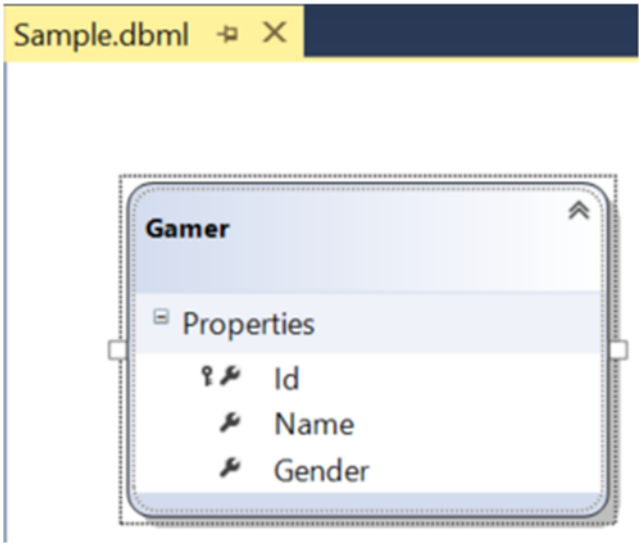
ProjectName --> Right Click --> Add --> New Item...
--> Linq to SQL classes -->
Name : **Sample.dbml**

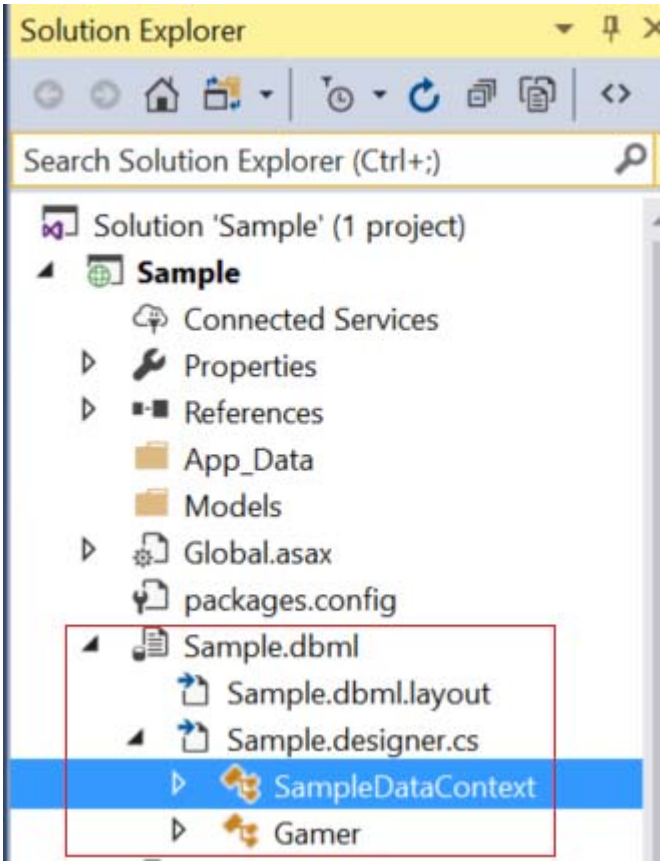
-->
Drag Table from Server Explorer into DBML




Add New Item - Sample
Installed
PowerShell
Visual C#
Code
Data
General
Web
General
Markup
Scripts
Web Forms
MVC
Razor
SignalR
Online
Sort by: Name Ascending
Search Installed Templates (Ctrl+E)
ADO.NET Entity Data Model Visual C#
DataSet Visual C#
EF 5.x DbContext Generator Visual C#
EF 6.x DbContext Generator Visual C#
LINQ to SQL Classes Visual C#
SQL Server Database Visual C#
XML File Visual C#
XML Schema Visual C#
Type: Visual C#
LINQ to SQL classes mapped to relational objects.
Name: Sample.dbml
Add
Cancel


Server Explorer
Azure (lpmplpmp01@hotmail.con
Data Connections
n550jkl\sql2016.Sample.dbo
Tables
Gamer
Views
Stored Procedures
Functions
Synonyms
Types
Assemblies
Servers
SharePoint Connections
Sample.dbml*
Drag into dbml
The Object Relational Designer allows you to visualize data classes in your code.
Create data classes by dragging items from Server Explorer or Toolbox onto this design surface.

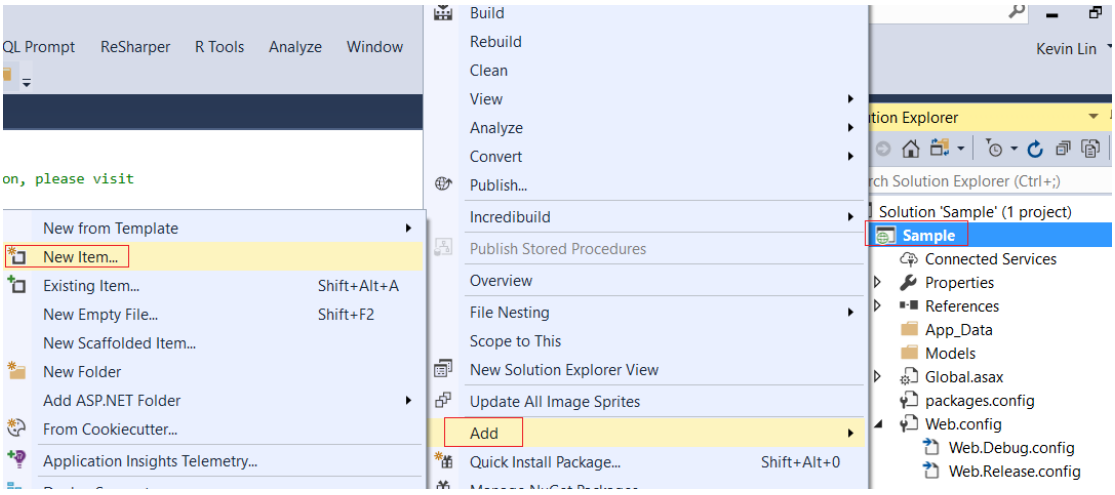

Sample.dbml
Gamer
Properties
Id
Name
Gender

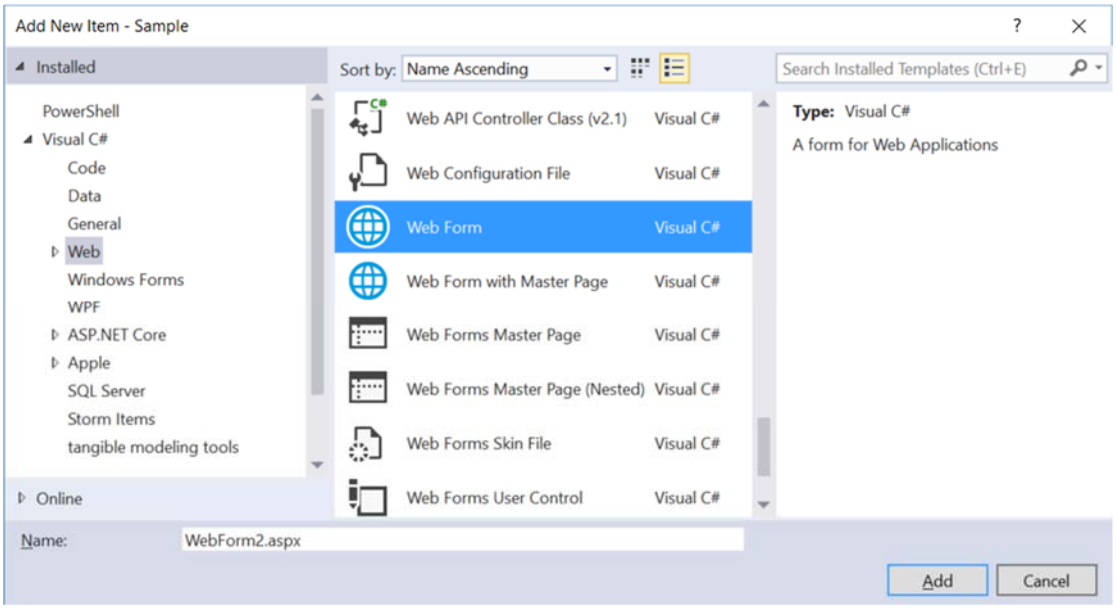


2.3. WebForm2

2.3.1. WebForm2.aspx

ProjectName --> Right Click --> Add --> New Item...
-->
WebForm
Name :
WebForm2.aspx






```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="WebForm2.aspx.cs" Inherits="Sample.WebForm2" %>
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            <asp:GridView ID="GridView1" runat="server"></asp:GridView>
        </div>
    </form>
</body>
</html>
```

2.3.2. WebForm2.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
namespace Sample
{
    public partial class WebForm2 : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            SampleDataContext dataContext = new SampleDataContext();
            GridView1.DataSource = from gamer in dataContext.Gamers
                where gamer.Gender == "Female"
                select gamer;
            //GridView1.DataSource = dataContext.Gamers.Where(gamer => gamer.Gender == "Female");
```

```
            GridView1.DataBind();
        }
    }
}
```

Id	Name	Gender
2	Name02	Female
4	Name04	Female
5	Name05	Female

1.

1.1.

Language Integrated 整體 Query (LINQ) is a component between the LINQ query and the actual data source which includes SQL Server, XML documents, Objects in memory etc.

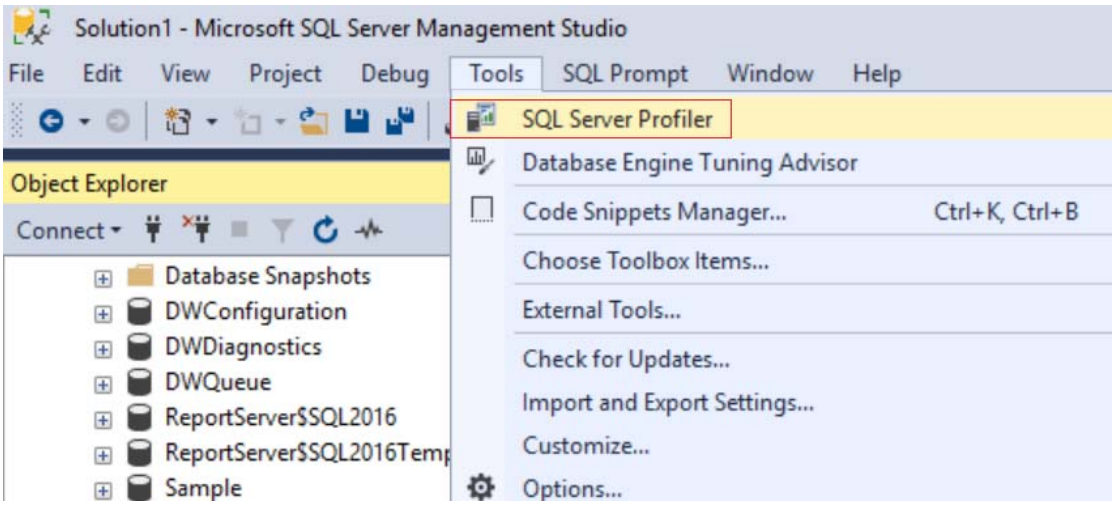
E.g. Linq to SQL provider can convert a Linq query to TSQL.

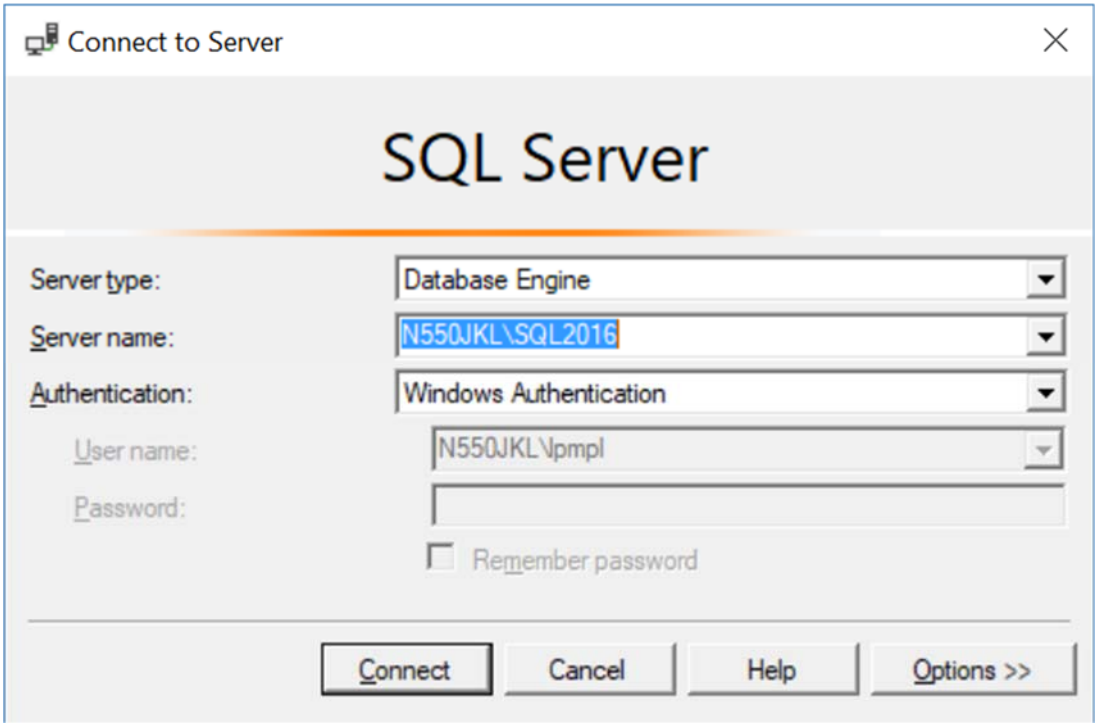
1.2.

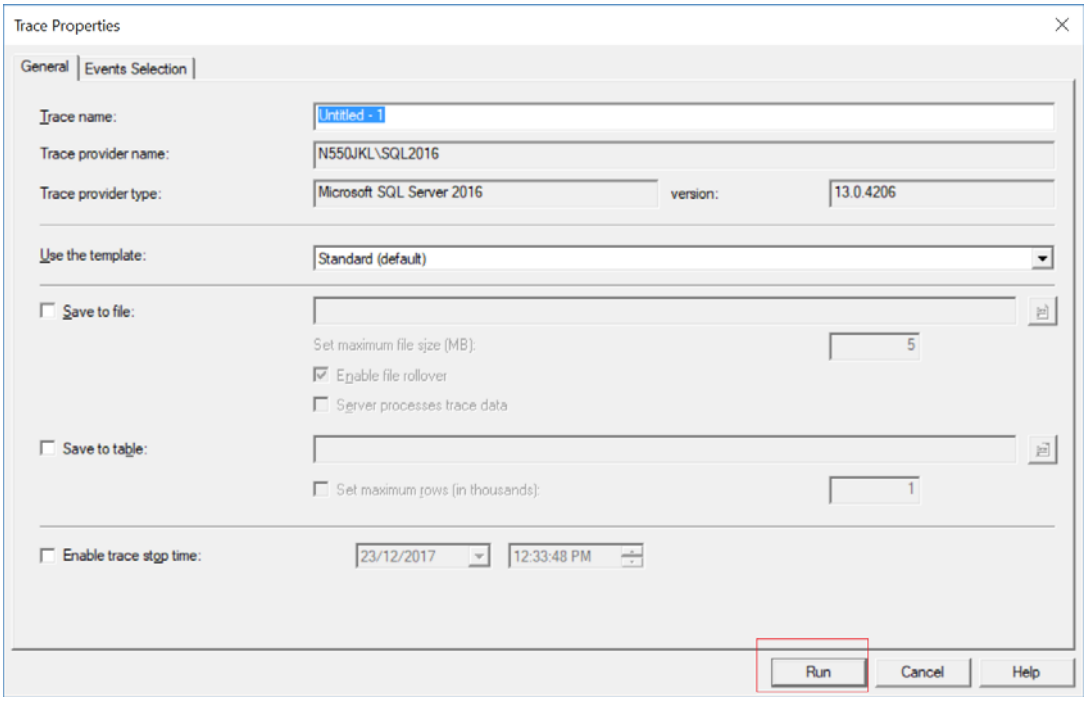
LINQ query can be written by any .NET supported programming language, and it provides compile time error checking.

2.3.3. SQL Profiler

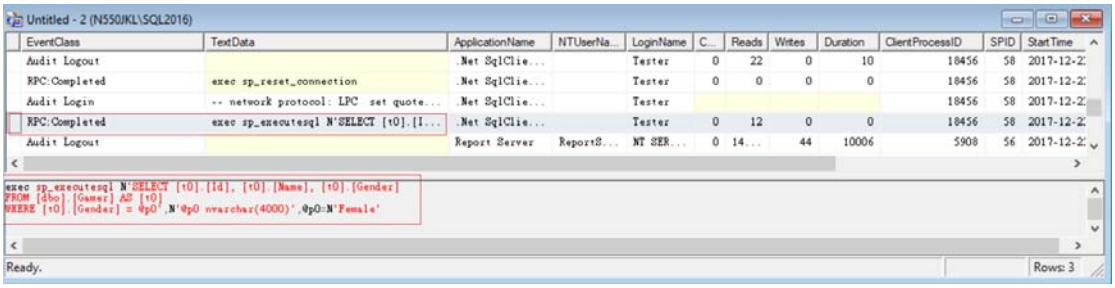
Tools --> SQL Server Profiler







Now, go back to VS2017, and run WebForm2.aspx again
You will see Linq to SQL provider convert Linq to TSQL.



2.4. WebForm3

2.4.1. WebForm3.aspx

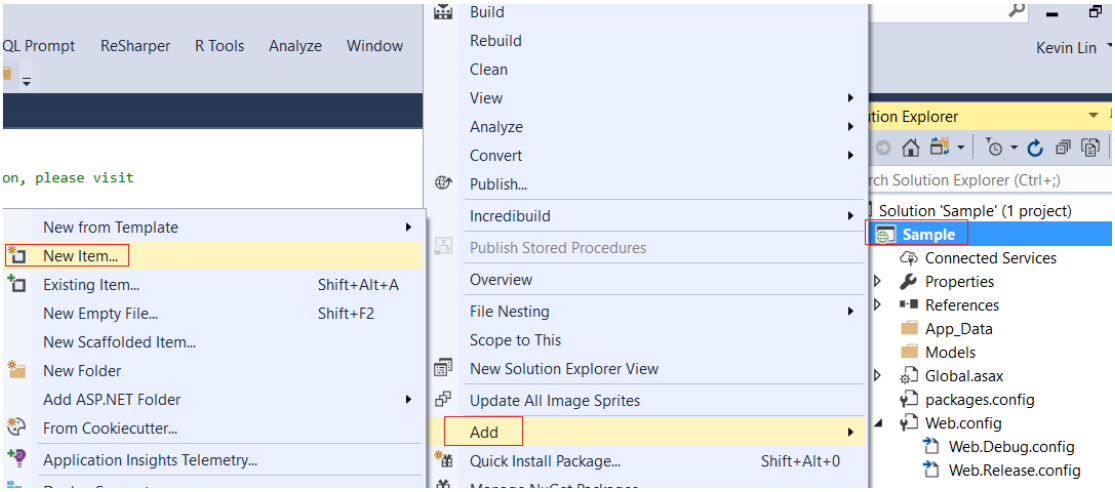
ProjectName --> Right Click --> Add --> New Item...

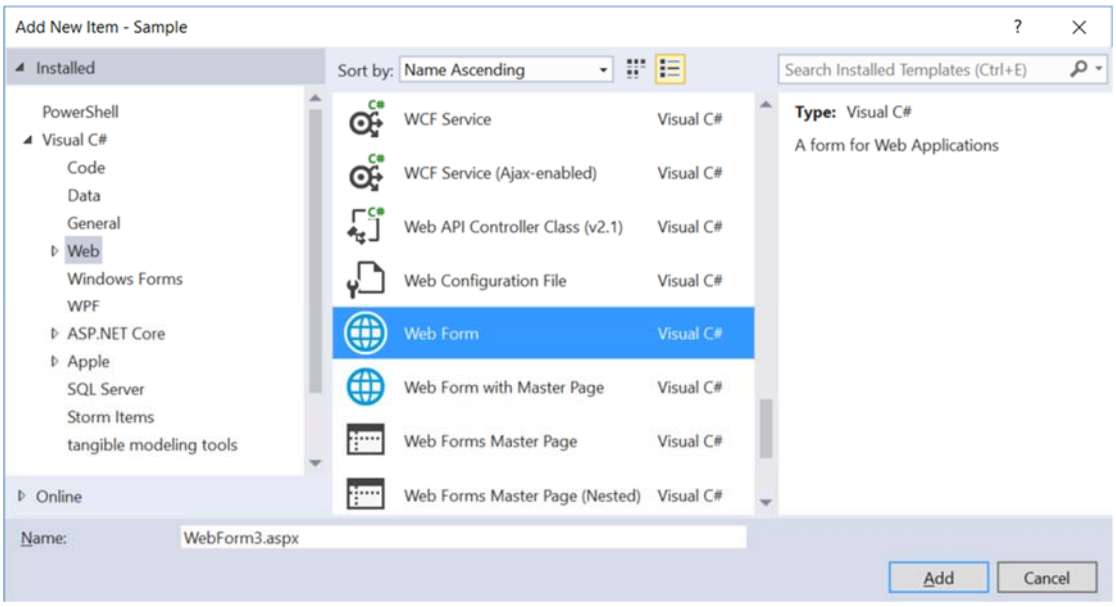
-->

WebForm

Name :

WebForm3.aspx



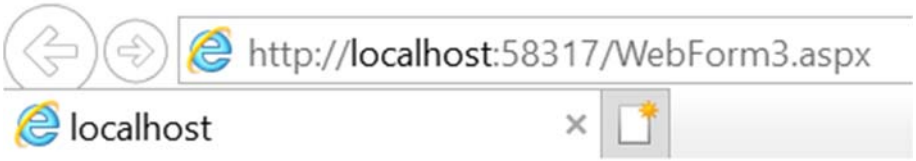



```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="WebForm3.aspx.cs" Inherits="Sample.WebForm3" %>
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            <asp:GridView ID="GridView1" runat="server"></asp:GridView>
        </div>
```

```
    </form>
</body>
</html>
```

2.3.2. WebForm3.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
namespace Sample
{
    public partial class WebForm3 : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            SampleDataContext dataContext = new SampleDataContext();
            int[] intArr = {10, 9, 8, 7, 6, 5, 4, 3, 2, 1};
            GridView1.DataSource = from intItem in intArr
                                   where intItem >= 5
                                   select intItem;
            //GridView1.DataSource = intArr.Where(intItem => intItem >= 5);
            GridView1.DataBind();
        }
    }
}
```



Item
10
9
8
7
6
5