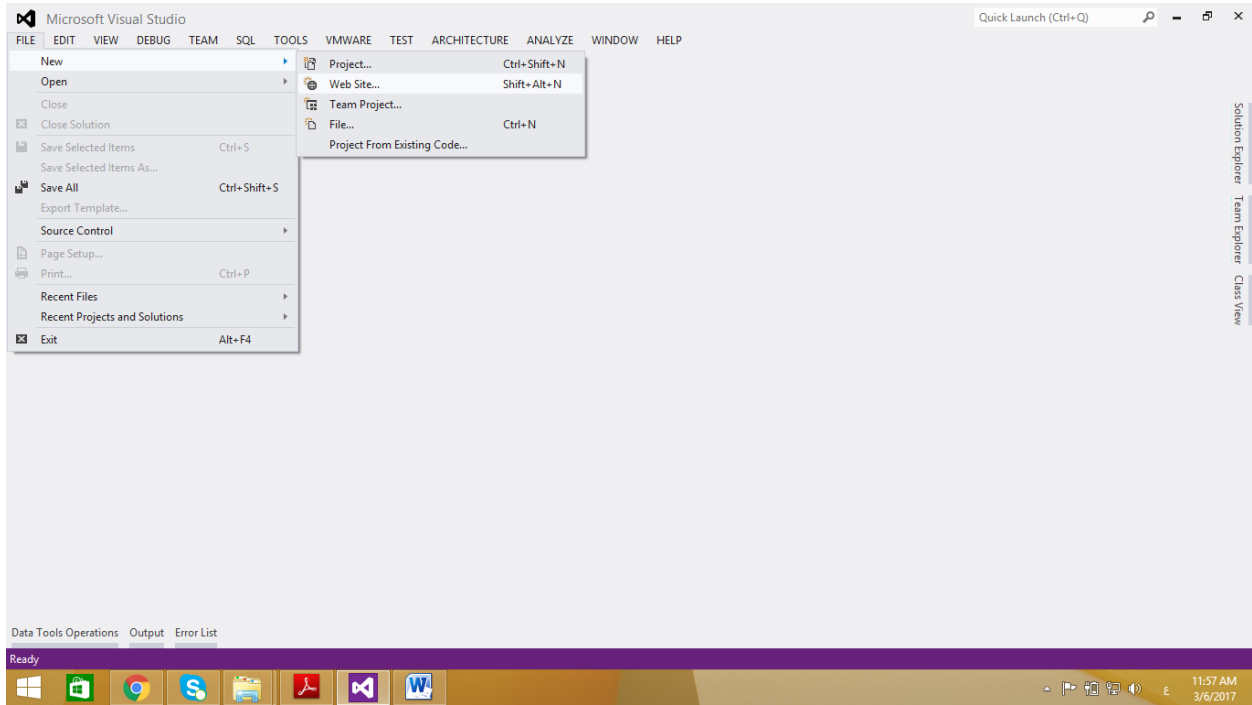


## 3-Tier In Asp.net

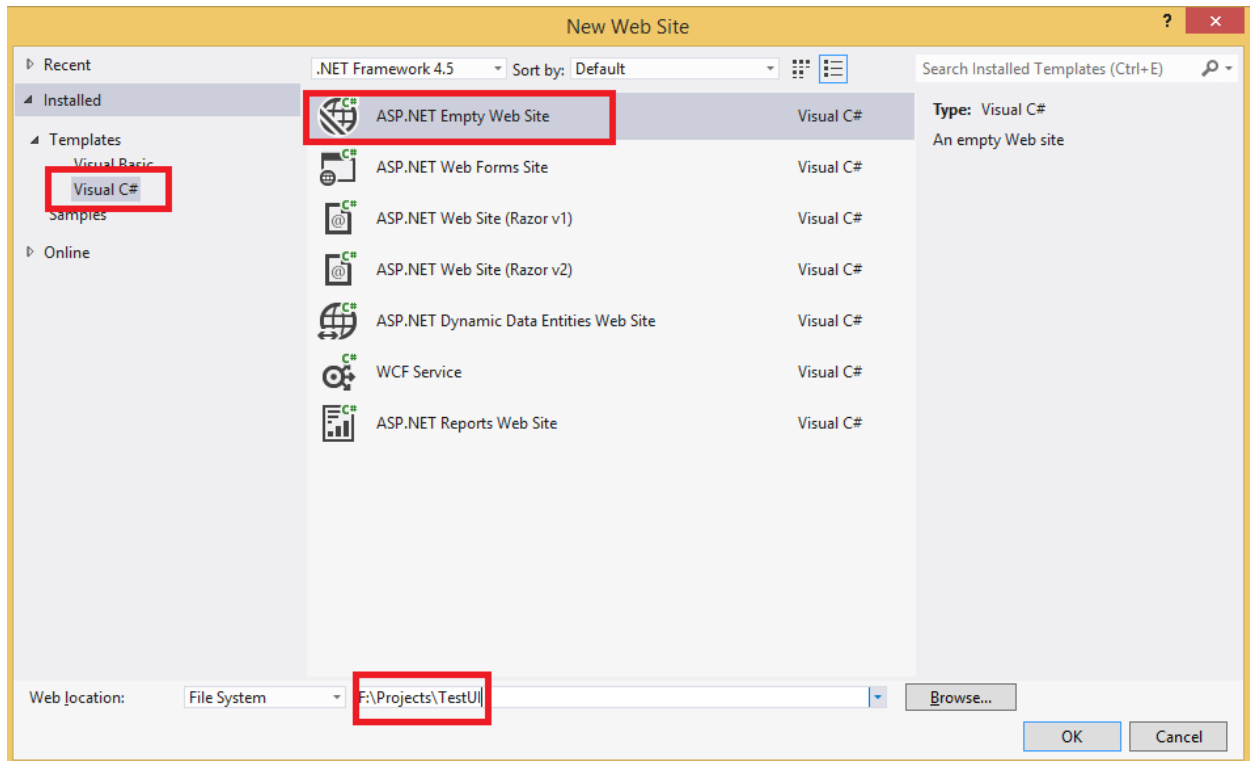
محمد الساعدي

<https://mohamediddan.wordpress.com/>

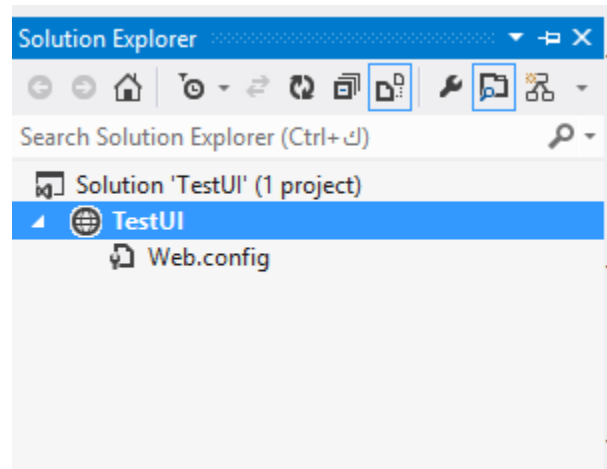
أولا: ننشئ مشروع ويب جديد :



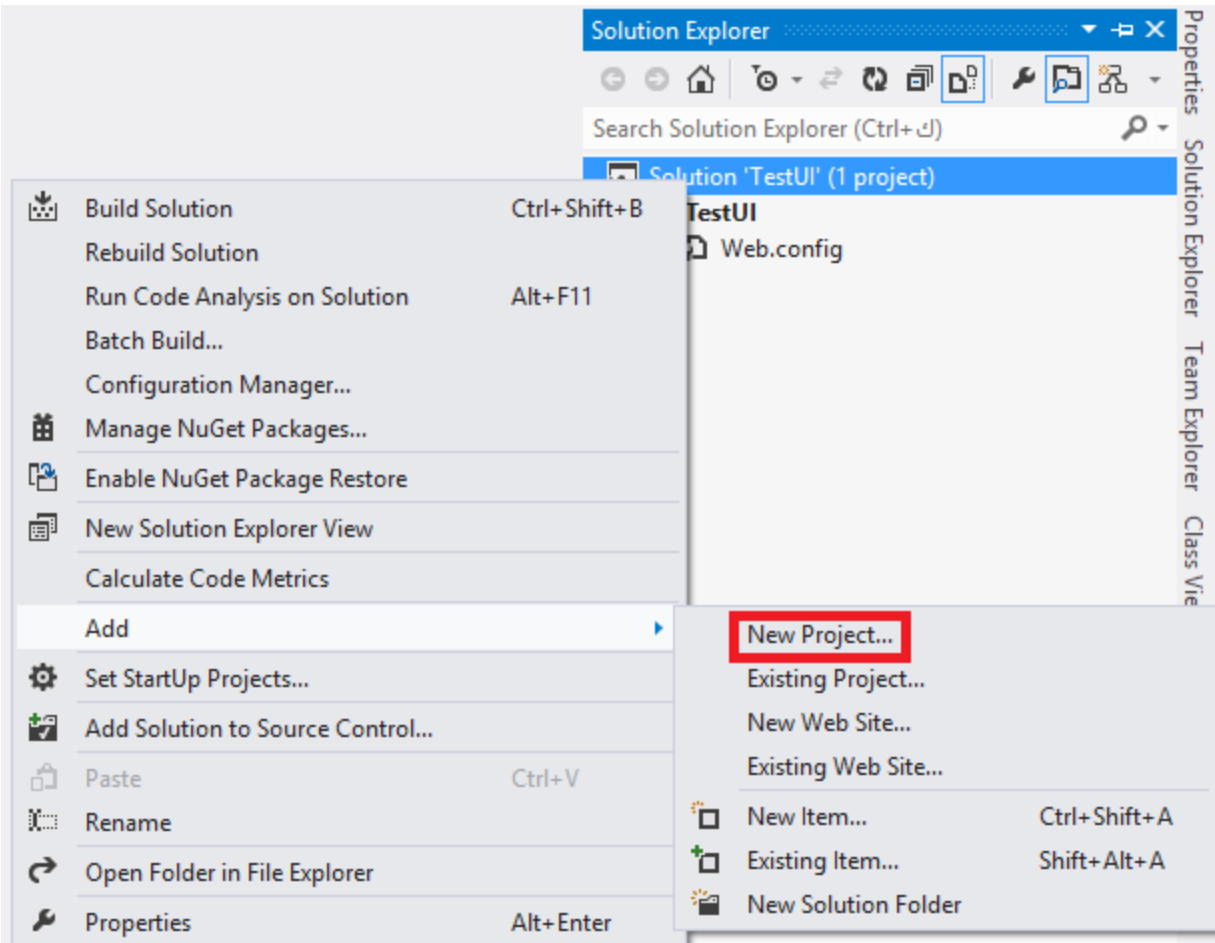
نختار ( Asp.net Empty WebSite ) , ونسميه ( TestUI ) .



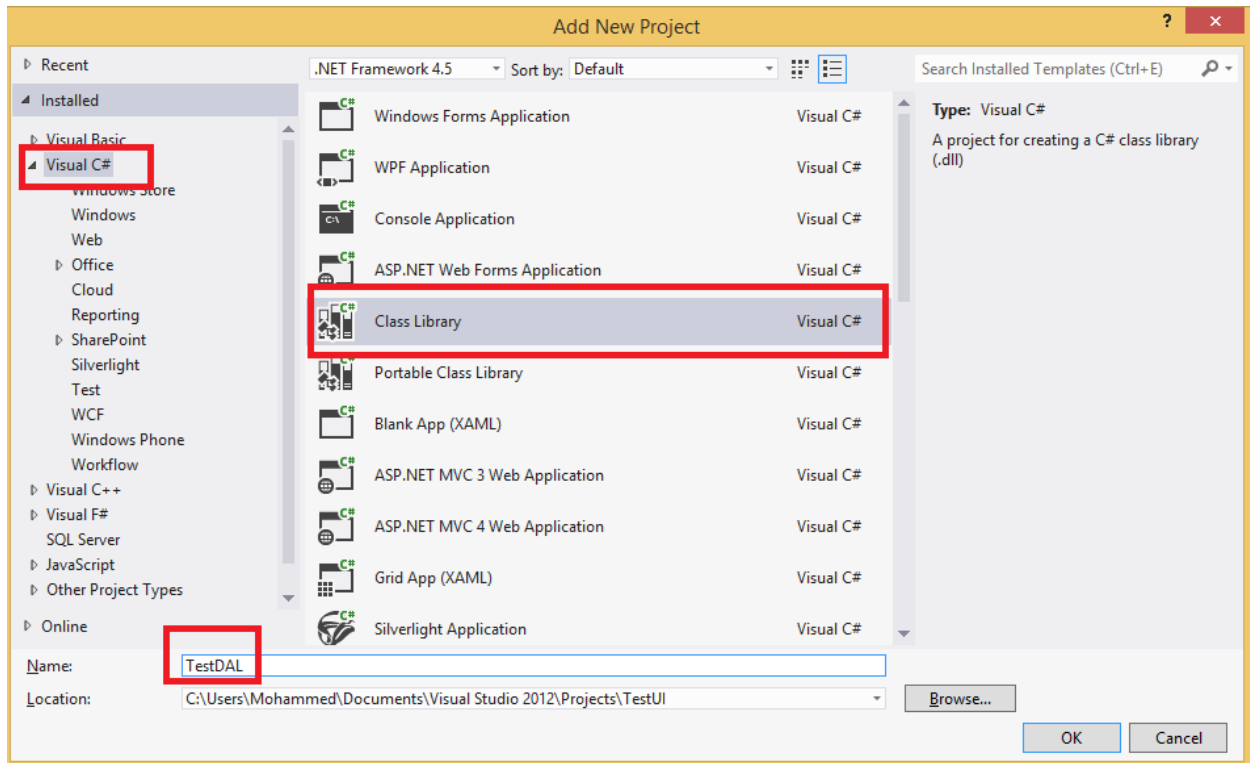
كما نلاحظ :



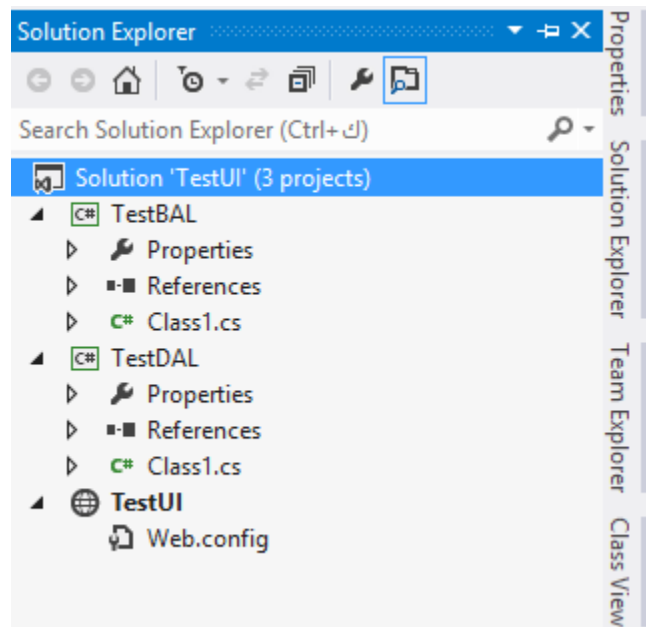
ثانيا : ننشئ داخل ال Solution مشروعين جديدين , من خلال كلك يمين على ( Solution 'TestUI' )  
ونختار ( Add ) ثم ( New Project ) :



نوع المشروع ( Class Library ) ونسميه ( TestDAL ) :

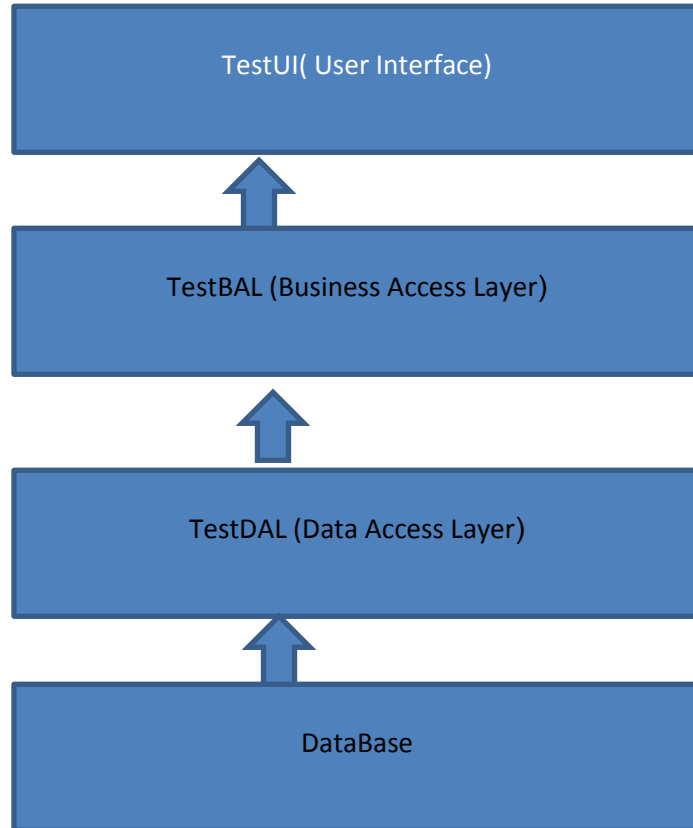


نعيد الخطوة الثانية وننشئ مشروع جديد من نوع ( Class Library ) ونسميه ( TestBAL ) , بالتالي ال Solution صار فيه ثلاث مشاريع , الان نتكلم عن هذه المشاريع الثلاث :

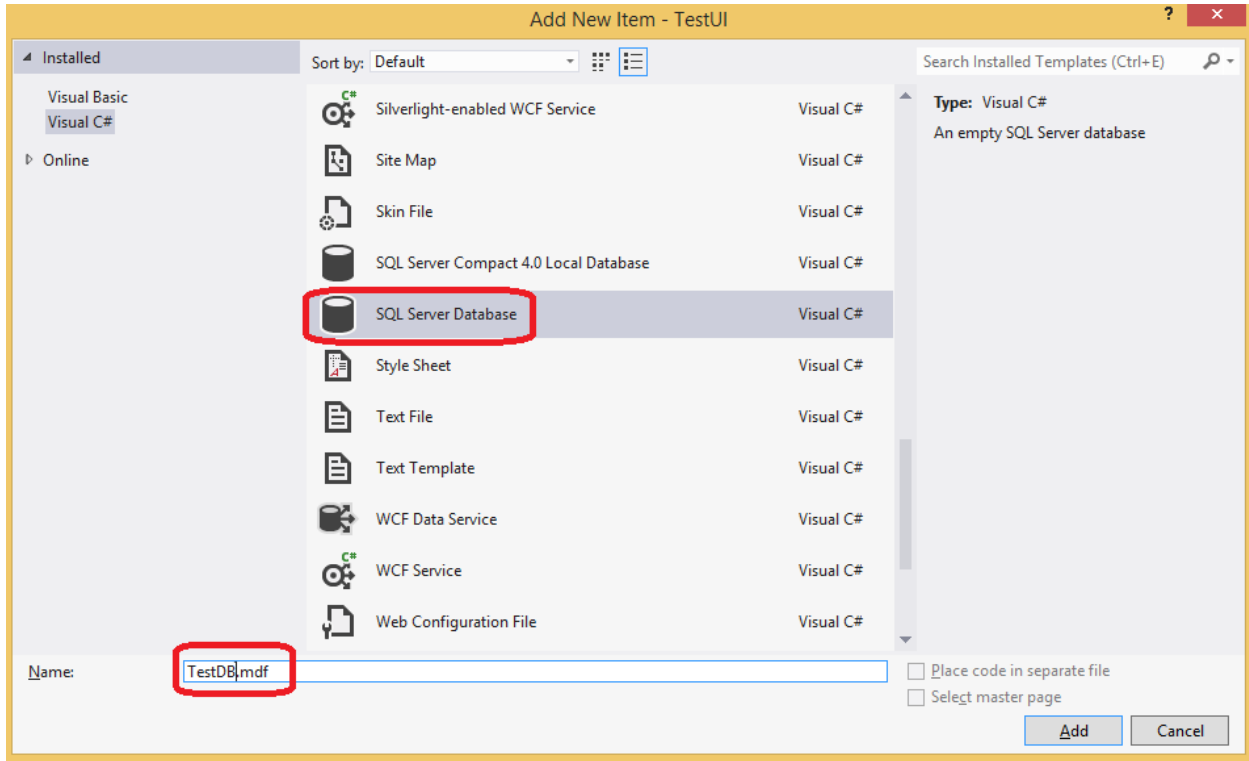


فكرة الطبقات ( Layers ) وهي تقسيم المشروع الى ثلاث أجزاء , أحدهما للتعامل مع قاعدة البيانات , والآخر للتعامل مع المستخدمين , والثالث وسيط بين الأول والثاني , الغاية من هذا الطريقة البرمجية هي تقليل الخطأ , وسهولة الصيانة , وكذلك تتيح المجال لأكثر من شخص ان يعملو على نفس المشروع , ومنها أشتقت فكرة ( MVC ) .

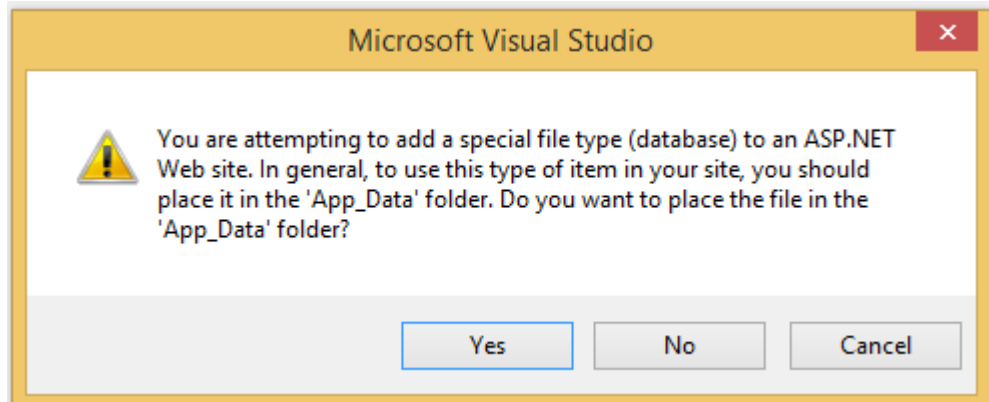
- TestUI : هذا الجزء الخاص بالتعامل مع المستخدم , يعني واجهات وصفحات وهكذا شي .
  - TestDAL : هذه الطبقة عبارة عن كلاسات ( Classes ) , تتعامل مباشرة مع قاعدة البيانات .
  - TestBAL : كذلك هي كلاسات , وتكون وسيطة بين ( TestUI ) و ( TestDAL ) .
- كما تلاحظ بالمخطط أدناه , فإن البرمجة تبدأ من قاعدة البيانات صعودا الى طبقة الواجهات :



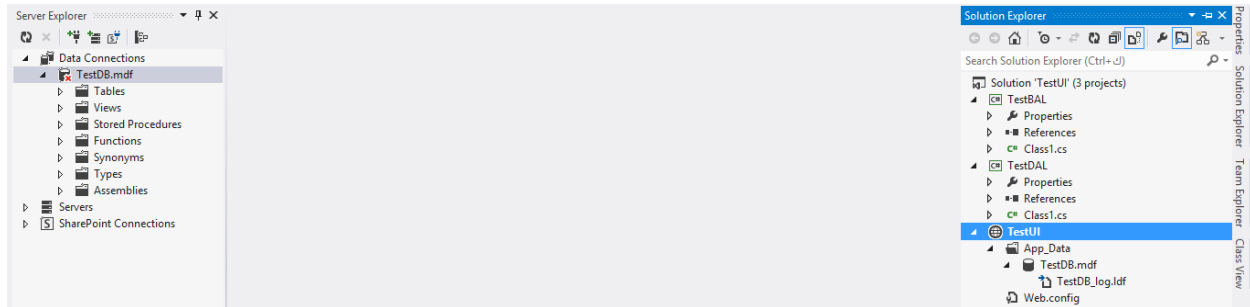
ثالثاً: ننشئ قاعدة البيانات , نختار (TestUI) ونختار (Add) ثم (new item) , ونختار ( SQL Server database) ونسميها (TestDB) كما في الصورة :



ستظهر النافذة التالية , التي تخبرنا بأن الفيچوال ستوديو سيقوم بوضع القاعدة بمجلد اسمه ( App\_Data )  
ننقر على ( Yes ) :

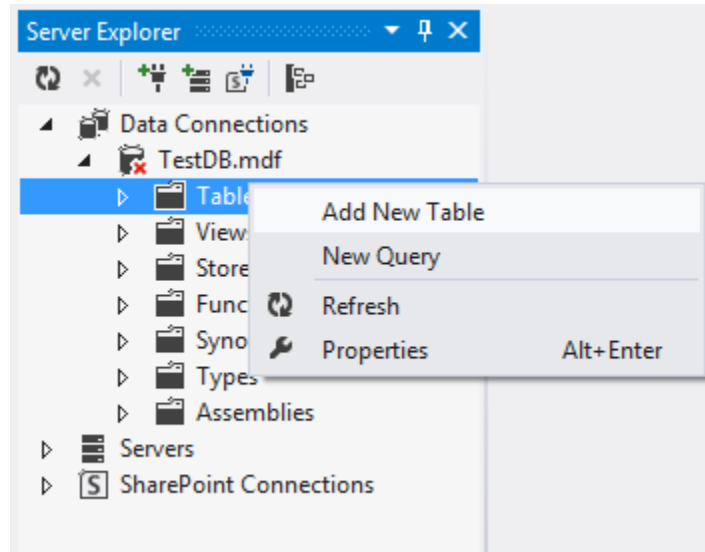


كما تلاحظ تم إضافة قاعدة البيانات , نقرتين عليها حتى تفتح ب Server Explorer ,



ملاحظة: القاعدة أمتداها ( .mdf ) اما الملف ( TestDB\_Log.ldf ) هذا عبارة عن ملف تسجل فيه التغييرات التي تطرأ على قاعدة البيانات .

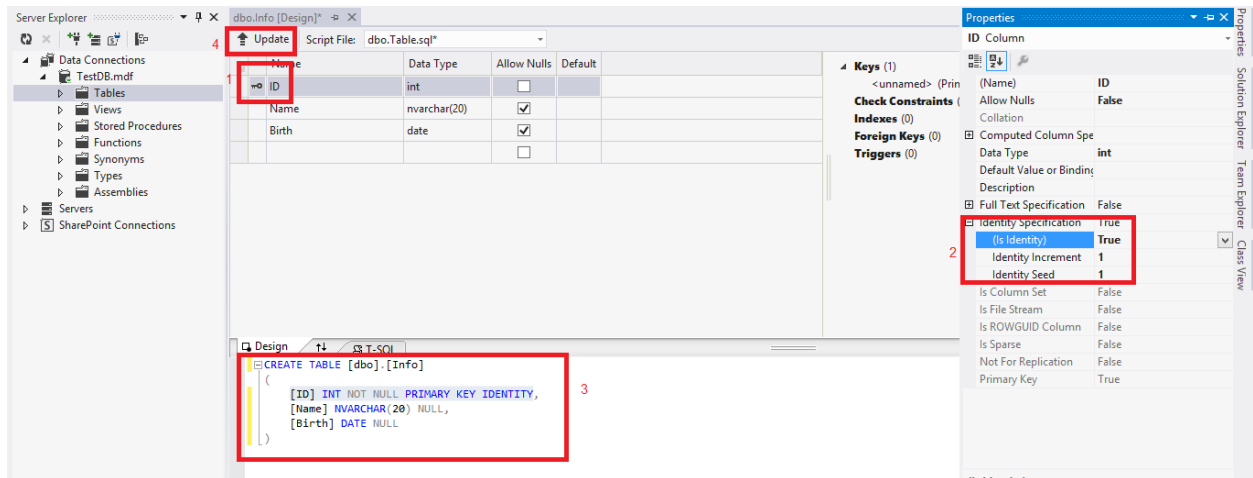
من ( Server Explorer ) كلك يمين على ( Tables ) ونختار ( Add New Table ) :



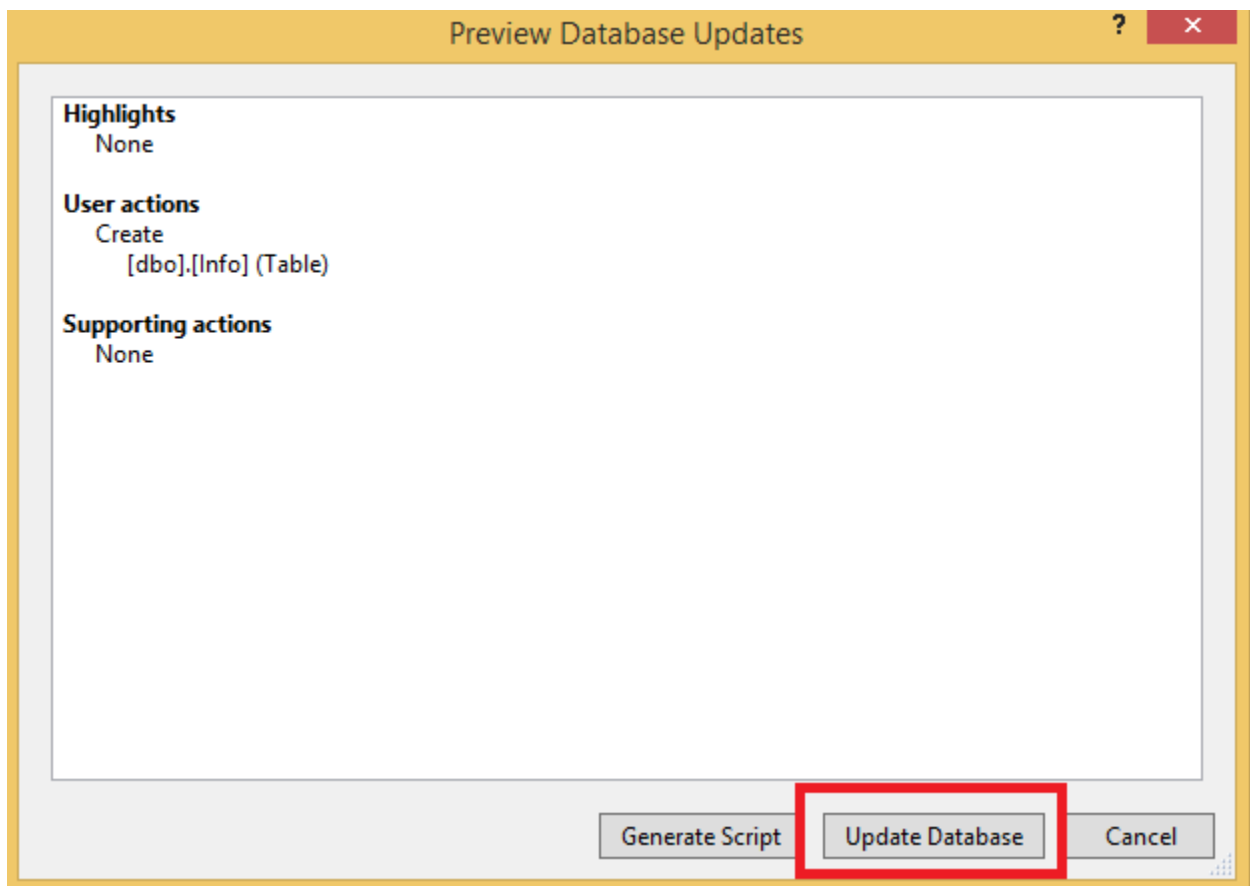
سيكون أسمه ( Info ) , وفيه الحقول كما موضح بالصورة أدناه :

```
CREATE TABLE [dbo].[Info]
(
    [ID] INT NOT NULL PRIMARY KEY IDENTITY,
    [Name] NVARCHAR(20) NULL,
    [Birth] DATE NULL
)
```

ولا ننسى أن نجعل ( ID ) يزداد تلقائياً بمقدار واحد ثم ننقر على ( Update ) :

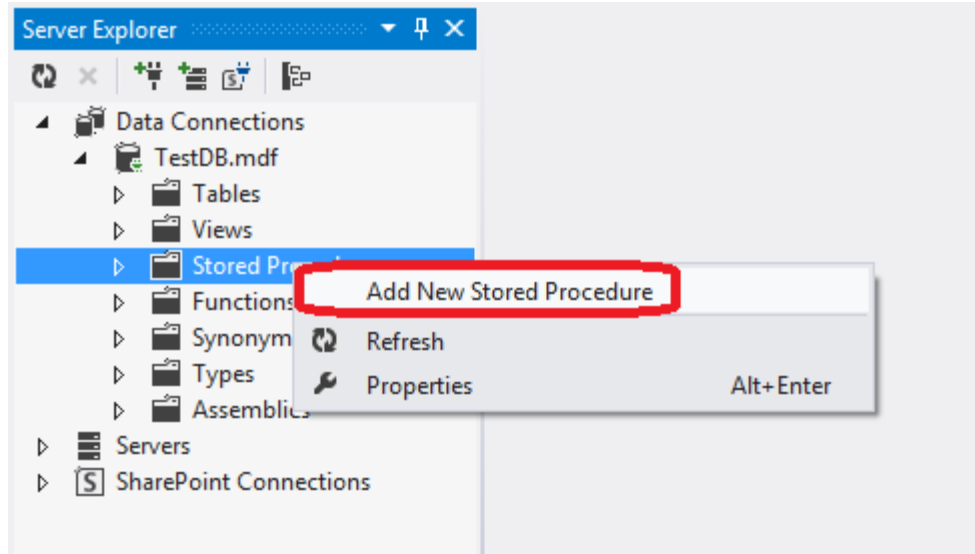


وننقر على ( Update data Base ) :



ثم من ( Server Explorer ) كلك يمين على ( Stored Procedure ) ونختار ( Add New Stored Procedure )





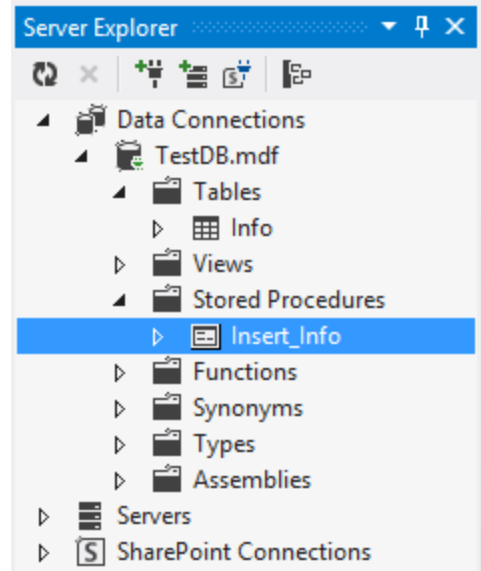
نسمح ما بداخلها ونكتب بروسيجر خاصة بالاضافة :

```
CREATE PROCEDURE [dbo].[Insert_Info]
    @Name nvarchar(50),
    @Birth date
AS
    Insert into Info(Name , Birth)
    Values (@Name ,@Birth)
```

بالامكان الاستغناء عن البروسيجر , لكن فائدتها تعطيك مرونة أكبر في استعلامات TSQL وكذلك تقلل من  
. SQL Injection

وننقر على ( Update ) .

الان انتهى عملنا على قاعدة البيانات .

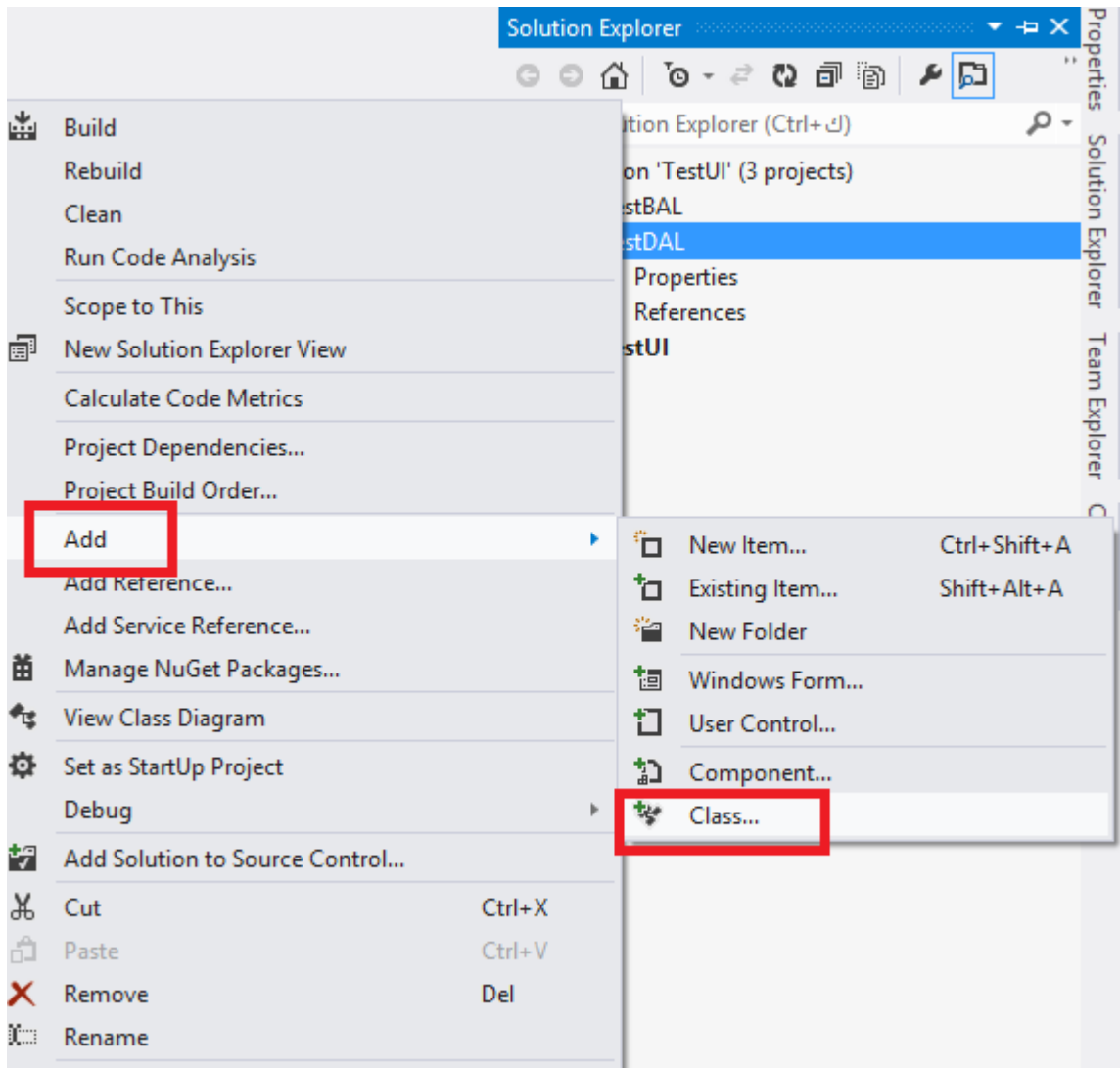


بقي شيء أخير في هذه الطبقة , نضيف نص الاتصال مع القاعدة في ( Web.config ) .

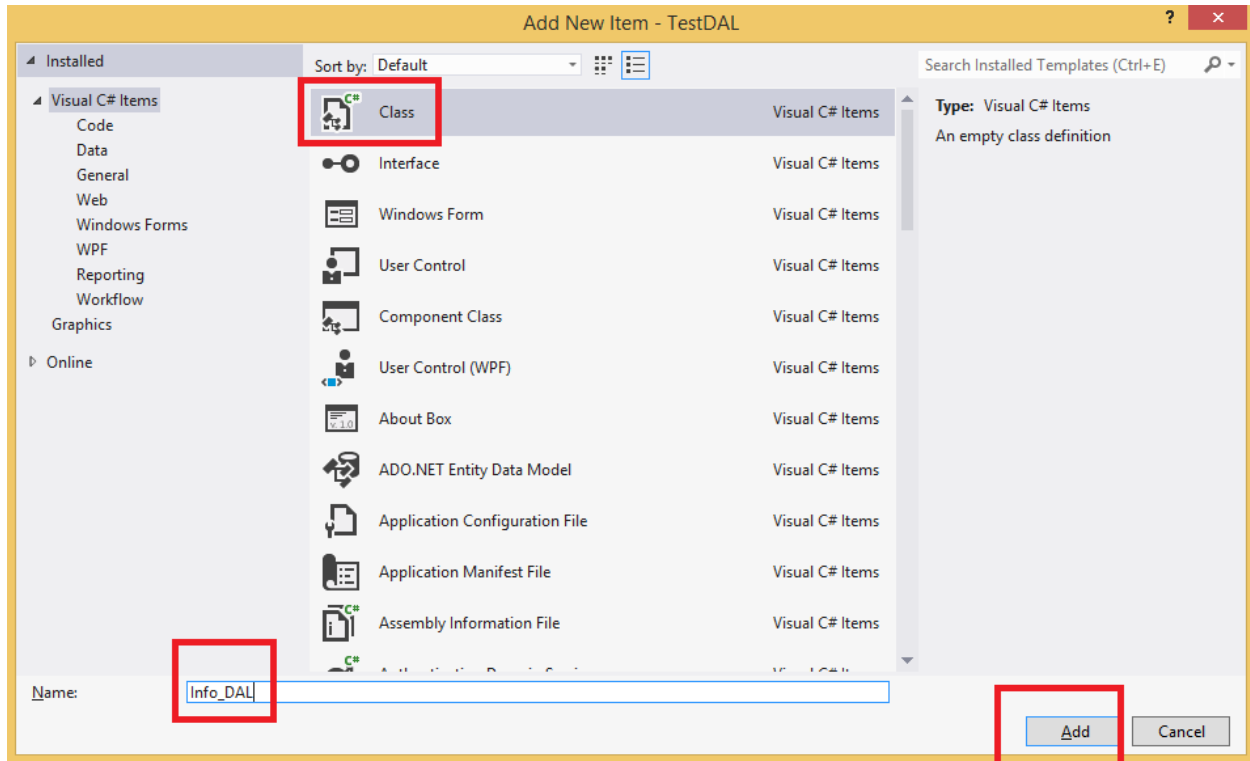
```
<connectionStrings>
  <add name="TestConString" connectionString="Data
Source=(LocalDB)\v11.0;AttachDbFilename=|DataDirectory|\TestDB.mdf;Integrated
Security=True"
      providerName="System.Data.SqlClient" />
</connectionStrings>
```

```
<?xml version="1.0"?>
<!--
For more information on how to configure your ASP.NET application, please visit
http://go.microsoft.com/fwlink/?LinkId=169433
-->
<configuration>
  <connectionStrings>
    <add name="TestConString" connectionString="Data Source=(LocalDB)\v11.0;AttachDbFilename=|DataDirectory|\TestDB.mdf;Integrated Security=True"
      providerName="System.Data.SqlClient" />
  </connectionStrings>
  <system.web>
    <compilation debug="false" targetFramework="4.5" />
    <httpRuntime targetFramework="4.5" />
  </system.web>
</configuration>
```

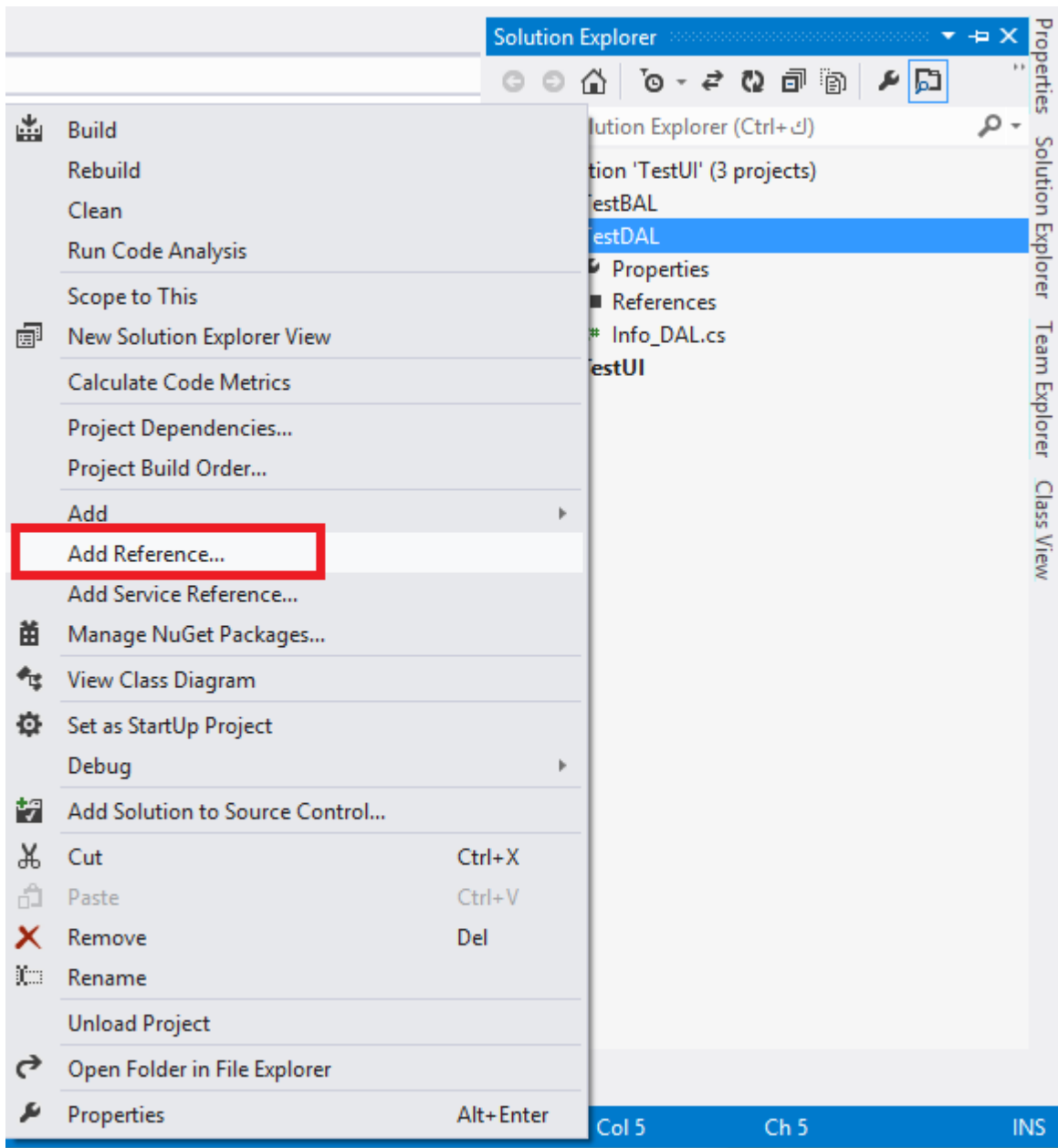
رابعاً: في الطبقة ( TestDAL ) نحذف الكلاس ( Class.cs ) , ومن خلال كليك يمين على ( TestDAL ) ونختار ( Add ) ثم ( Class ) حتى نضيف كلاس جديد :

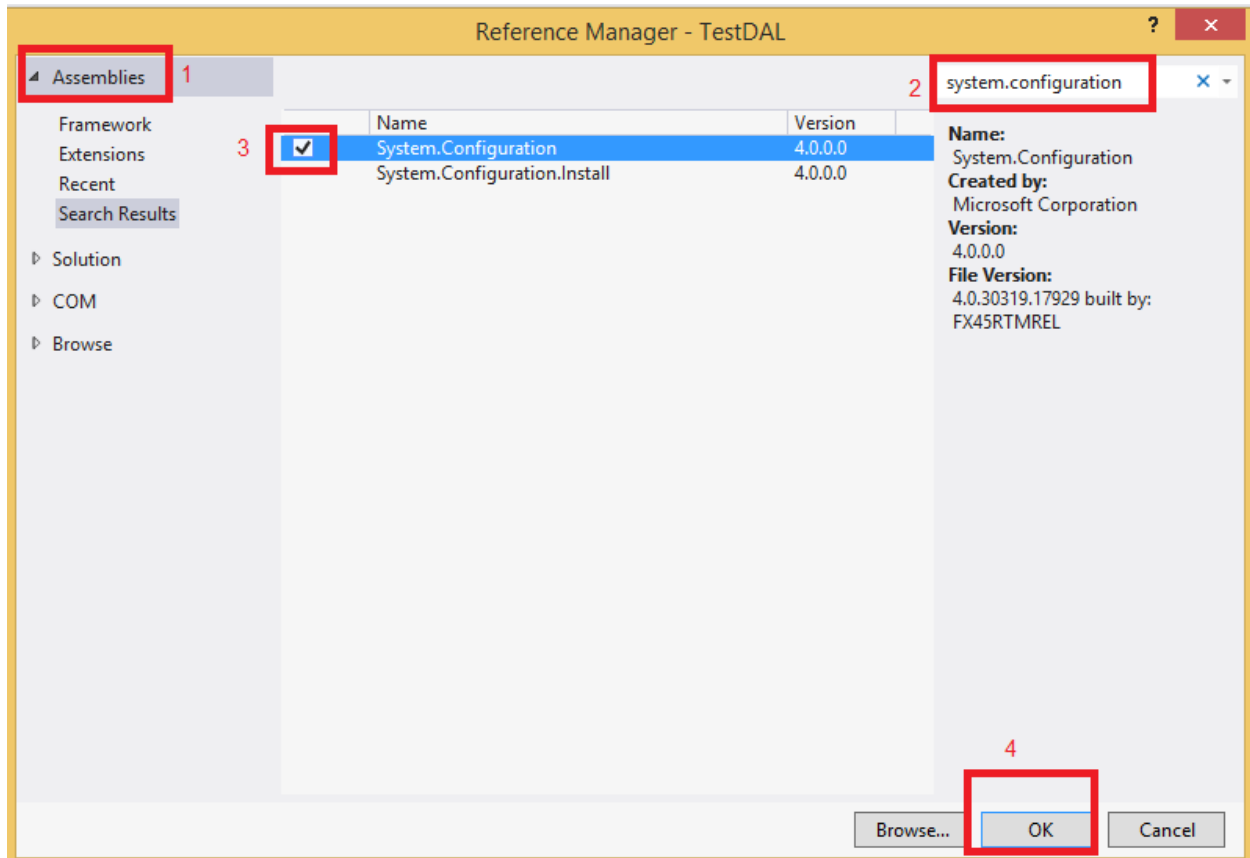


نسميه (Info\_DAL):



ثم كلك يمين على ( TestDAL ) ونختار ( Add References ) :





ثم نفتح الكلاس (Info\_DAL) من خلال النقر عليه مرتين ونضيف داخله :

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Data;
using System.Data.SqlClient;
using System.Configuration ;
namespace TestDAL
{
    public class Info_DAL
    {
        string CnnString =
ConfigurationManager.ConnectionStrings["TestConString"].ToString();
        public Info_DAL() { }
        public int Insert(string Name,DateTime Birth)
        {
            SqlConnection SQLCnn = new SqlConnection(CnnString);
            SQLCnn.Open();
            SqlCommand SQLCmd = new SqlCommand("Insert_Info", SQLCnn);
            SQLCmd.CommandType = CommandType.StoredProcedure;
            try

```

```

        {
            SQLCmd.Parameters.AddWithValue("@Name", Name);
            SQLCmd.Parameters.AddWithValue("@Birth", Birth);

            return SQLCmd.ExecuteNonQuery();

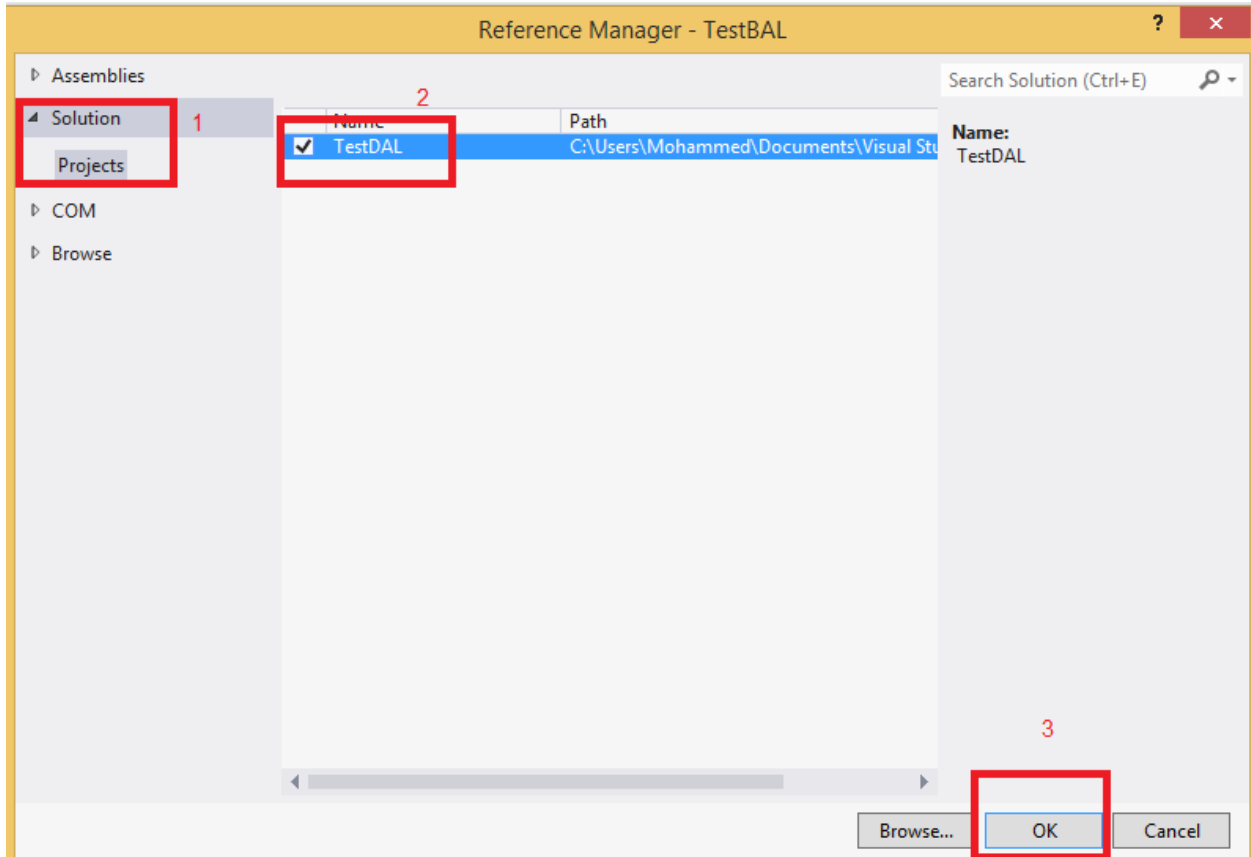
        }

    catch
    {
        throw;
    }
    finally
    {
        SQLCmd.Dispose();
        SQLCnn.Close();
        SQLCnn.Dispose();
    }
}
}
}
}
}

```

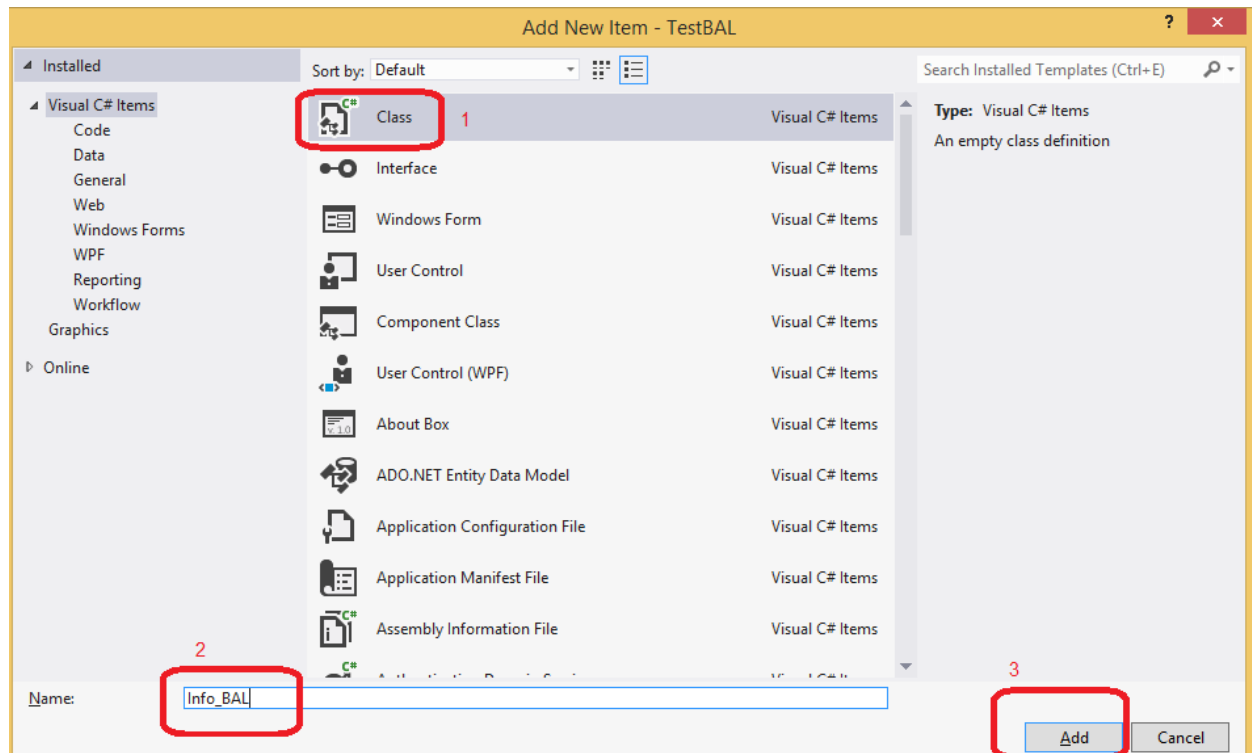
بالبداية قمنا بأستدعاء ثلاث مكتبات ( Name Spaces ) هم ( System.Data , System.Configuration و System.Data.SqlClient ) لغرض استخدامهم فيما بعد , ثم قمنا بتعريف متغير نصي أسمه ( CnnString ) لكي يحمل نص الاتصال مع قاعدة البيانات , بعدها عملنا دالة نوعها ( Integer ) ومررنا لها عنصرين هما ( الاسم والمواليد ) وعرفنا داخلها متغير ( SQLCnn ) ويعني الاتصال مع القاعدة وفتحنا بعدها الاتصال , وعرفنا متغير ( SQLCmd ) الذي سيحمل الاستعلام الذي سوف نرسله لقاعدة البيانات ومررنا عنصرين الى قاعدة البيانات , وقد حددنا نوع الاستعلام ( Stored Procedure ) التي عملناها في الخطوة السابقة ثم نفذنا وغلقنا الاتصال .

خامسا : ننتقل الى طبقة ( TestBAL ) وكذلك نضيف فيها ( References ) ( System.Configuration ) و ( TestDAL ) كمورد .



ونحذف الكلاس ( Class.cs ) ونضيف بدله كلاس جديد أسمه ( Info\_BAL )



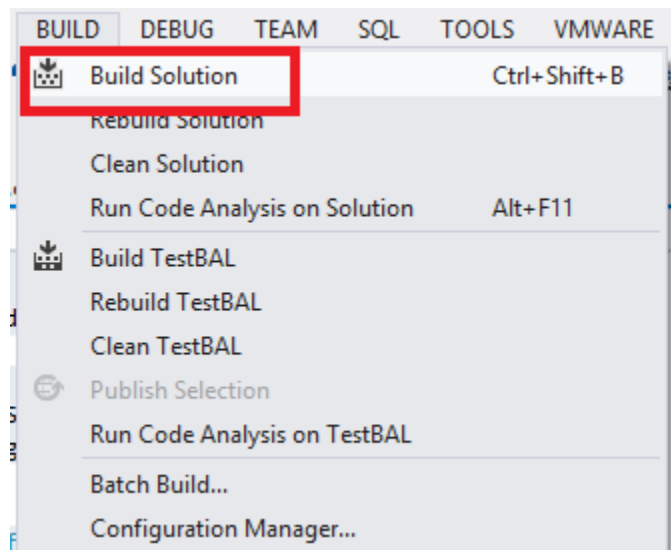


ونکتب داخله :

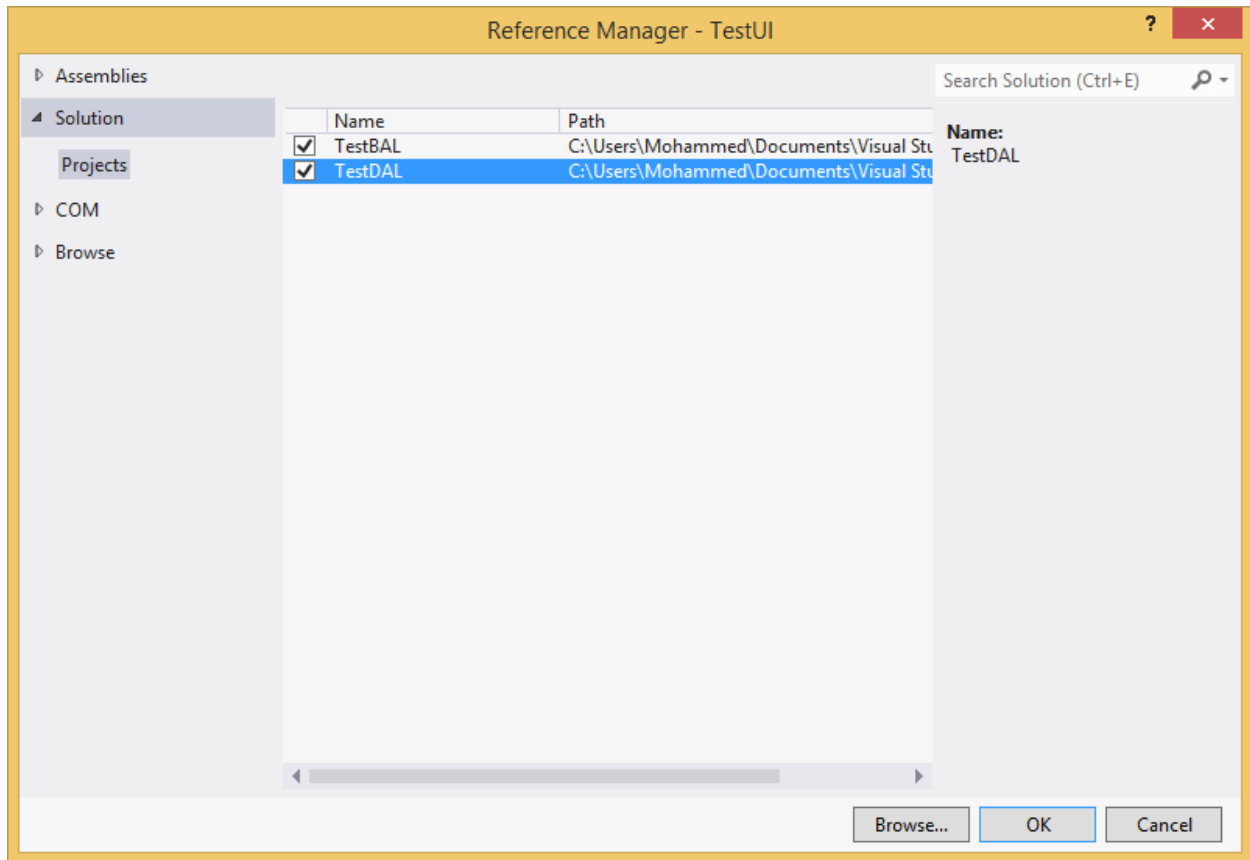
```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using TestDAL;
using System.Data;
using System.Data.SqlClient;
using System.Configuration;
namespace TestBAL
{
    public class Info_BAL
    {
        public Info_BAL() { }
        public int Insert(string Name, DateTime Birth)
        {
            Info_DAL Info = new Info_DAL();
            try
            {
                return Info.Insert(Name, Birth);
            }
            catch
            {
                throw;
            }
            finally
            {
            }
        }
    }
}
```

```
    {  
        Info = null;  
    }  
}  
}
```

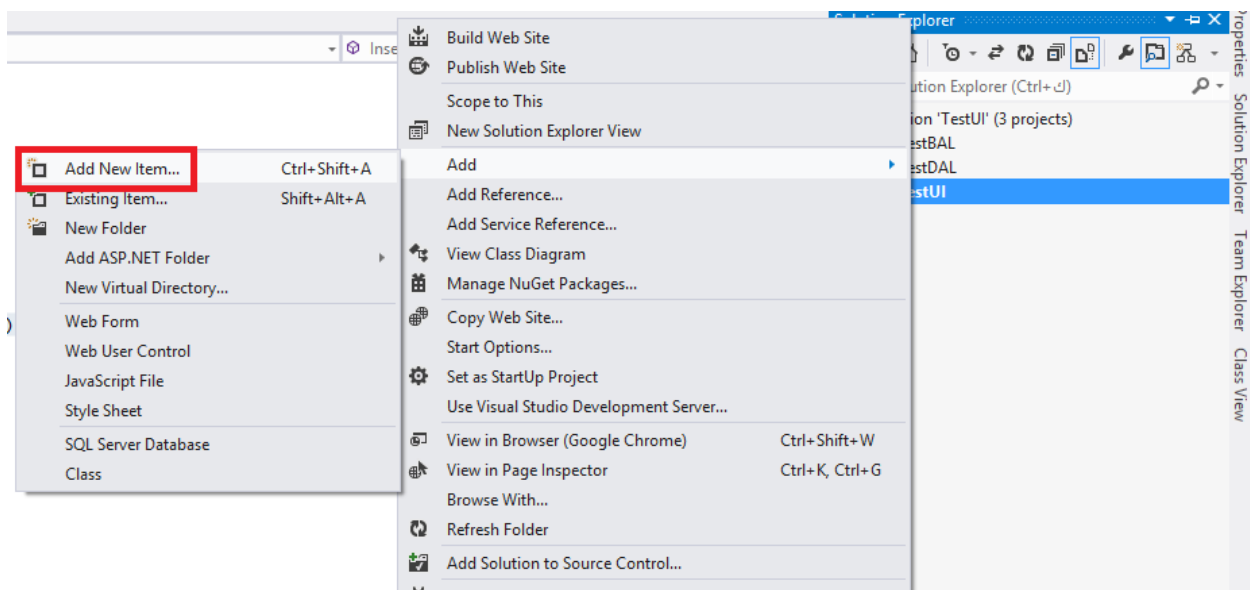
الآن من خلال (Build) نختار (Build Solution) :

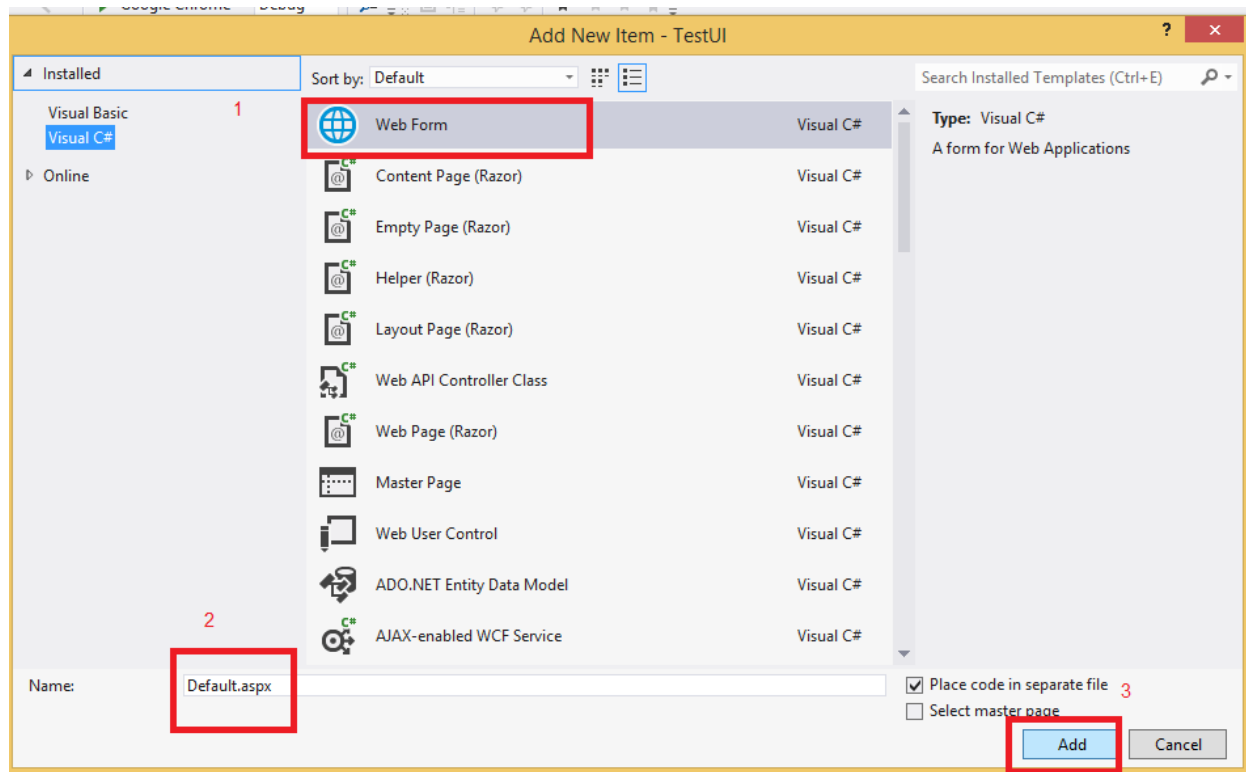


سادسا : نضيف في (TestUI) كلاً من (TestDAL) و (TestBAL) ك (References) :



ونضيف صفحة جديدة , من خلال كلك يمين على ( TestUI ) ونختار ( Add ) ثم ( Add New Item ) ,





ونضيف داخلها :

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Default.aspx.cs"
Inherits="_Default" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
  <title></title>
</head>
<body>
  <form id="form1" runat="server">
    <div>

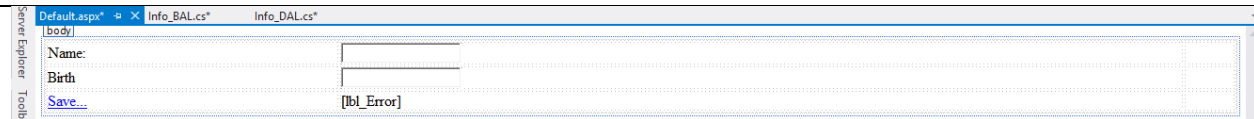
      <table style="width:100%;">
        <tr>
          <td>Name: </td>
          <td>
            <asp:TextBox ID="txt_name" runat="server"></asp:TextBox>
          </td>
          <td>&nbsp;</td>
        </tr>
      </table>
    </div>
  </form>
</body>
</html>
```

```

        </tr>
        <tr>
            <td>Birth</td>
            <td>
                <asp:TextBox ID="txt_Birth" runat="server"
TextMode="Date"></asp:TextBox>
            </td>
            <td>&nbsp;</td>
        </tr>
        <tr>
            <td>
                <asp:LinkButton ID="lbtn_Save"
runat="server">Save...</asp:LinkButton>
            </td>
            <td>
                <asp:Label ID="lbl_Error" runat="server"></asp:Label>
            </td>
            <td>&nbsp;</td>
        </tr>
    </table>

</div>
</form>
</body>
</html>

```



نقرتين على (Link Button) الذي أسميناه (Save) حتى يظهر الكود الخلفي، ونكتب داخله :

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data;
using System.Data.SqlClient;
using System.Configuration;
using TestBAL;
using TestDAL;

public partial class _Default : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
    }
    protected void lbtn_Save_Click(object sender, EventArgs e)
    {
        if (!Page.IsValid)
            return;
        int intResult = 0;
    }
}

```

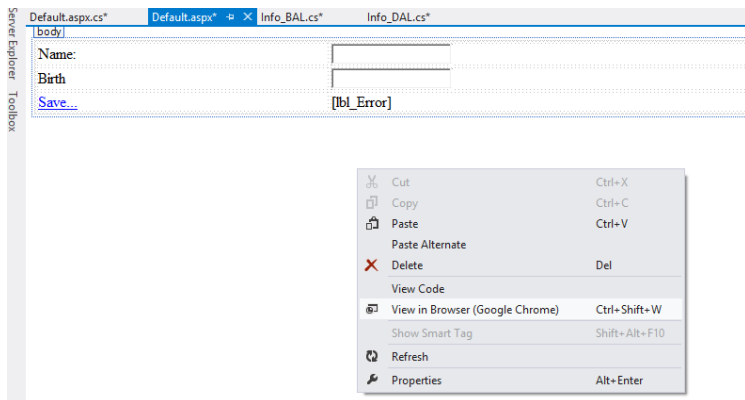
```

Info_BAL Info = new Info_BAL();
string Name = txt_name.Text;
DateTime Birth =Convert.ToDateTime( txt_Birth.Text);
try
{
    intResult = Info.Insert(Name, Birth);
    if (intResult > 0)
        lbl_Error.Text = "New record Saved successfully.";
    else
        lbl_Error.Text = "FirstName [<>" + txt_name.Text + "</b>] already exists,
try another name";
}
catch (Exception ex)
{
    lbl_Error.Text = ex.Message.ToString();
}

finally
{
    Info = null;
}
}
}

```

الان نستعرض الصفحة :



Name:

Muhammed

Birth

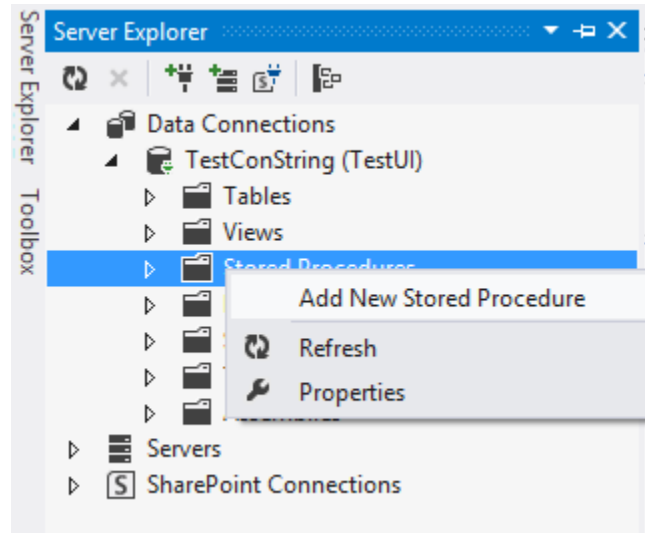
٠٦ / ٠٢ / ٢٠١٧

[Save...](#)

New record Saved successfully.

الان دعونا نعمل صفحة للبحث وعرض معلومات شخص معين حسب أسم الشخص ,

اول شي نعمل ( Stored Procedure ) نسميها ( Search ) :



كود البروسيجر :

```
CREATE PROCEDURE [dbo].[Search]
(
    @Name nvarchar(50)
)
AS
begin
if not exists (select ID from Info where Name = @Name )
    begin
select * from Info
    End
else
    begin
select * from Info
where Name=@Name
    End
End
```

في البروسيجر اعلاه سيتحقق هل المعامل (@Name) فيه قيمه أو لا , اذا فيه هل مطابقة لأحد سجلات الاسم بالجدول , واذا لا أرجع لنا جميع البيانات بالجدول .

الان نفتح الكلاس ( Info\_DAL ) من داخل ( TestDAL ) ونضيف فيه الدالة التالية :

```
public DataTable Search(string Name)
{
    SqlConnection SQLCnn = new SqlConnection(CnnString);
    SqlDataAdapter SQLAdapter = new SqlDataAdapter("Search", SQLCnn);
    SQLAdapter.SelectCommand.CommandType = CommandType.StoredProcedure;
    SQLAdapter.SelectCommand.Parameters.AddWithValue("@Name", Name);
    DataSet SQLDs = new DataSet();
    try
```

```

        {
            SQLAdapter .Fill(SQLds, "Info");
            return SQLds.Tables["Info"];
        }
        catch
        {
            throw;
        }
        finally
        {
            SQLds.Dispose();
            SQLAdapter.Dispose();
            SQLCnn.Close();
            SQLCnn.Dispose();
        }
    }
}

```

الان نفتح كلاس ( Info\_BAL ) داخل ( TestBAL ) :

```

public DataTable Search(string Name)
{
    Info_DAL Id = new Info_DAL();
    try
    {
        return Id.Search(Name);
    }
    catch
    {
        throw;
    }
    finally
    {
        Id = null;
    }
}

```

الان نظيف صفحة الى مشروع ( TestUI ) نسميها ( Search ) :

```

<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Search.aspx.cs" Inherits="Search"
%>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title></title>
</head>
<body>
    <form id="form1" runat="server">
        <div dir="rtl">

            <table align="right" style="width: 100%; font-weight: 700;">
                <tr>

```



```

        <td>&nbsp;</td>
        <td>
            <asp:TextBox ID="txt_Name" runat="server"></asp:TextBox>
            <asp:Button ID="btn_Search" runat="server" onClick="btn_Search_Click"
Text="بحث" />
        </td>
    </tr>
    <tr>
        <td colspan="3">
            <hr />
        </td>
    </tr>
    <tr>
        <td colspan="3">
            <asp:GridView ID="GridView1" runat="server" CellPadding="4"
ForeColor="#333333" GridLines="None" Width="100%">
                <AlternatingRowStyle BackColor="White" />
                <EditRowStyle BackColor="#2461BF" />
                <FooterStyle BackColor="#507CD1" Font-Bold="True"
ForeColor="White" />
                <HeaderStyle BackColor="#507CD1" Font-Bold="True"
ForeColor="White" />
                <PagerStyle BackColor="#2461BF" ForeColor="White"
HorizontalAlign="Center" />
                <RowStyle BackColor="#EFF3FB" />
                <SelectedRowStyle BackColor="#D1DDF1" Font-Bold="True"
ForeColor="#333333" />
                <SortedAscendingCellStyle BackColor="#F5F7FB" />
                <SortedAscendingHeaderStyle BackColor="#6D95E1" />
                <SortedDescendingCellStyle BackColor="#E9EBEF" />
                <SortedDescendingHeaderStyle BackColor="#4870BE" />
            </asp:GridView>
        </td>
    </tr>
</table>

</div>
</form>
</body>
</html>

```

ضفنا جدول وداخله ثلاث عناصر ( Button , TextBox ,Grid View ) , نقرتين على اليبوتون ونضيف داخله الكود التالي :

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Configuration;
using System.Data;

```

```

using System.Data.SqlClient;
using TestDAL;
using TestBAL;

public partial class Search : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
    }

    protected void btn_Search_Click(object sender, EventArgs e)
    {
        if (!Page.IsValid)
            return;
        Info_BAL Ib = new Info_BAL();
        string Name = txt_Name.Text;
        try
        {
            GridView1.DataSource = Ib.Search(Name);
            GridView1.DataBind();
        }
        catch (Exception ex) { }
        finally
        {
            Ib = null;
        }
    }
}

```

بعد التحقق من صحة الصفحة استدعينا كلاس المعلومات الذي فيه دالة البحث وجعلناه كمصدر بيانات للغرد فيو .

استعرض الصفحة

بحث

Birth	Name	ID
AM 12:00:00 2/6/2017	Muhammed	1
AM 12:00:00 2/6/2017	Ali	2

بحث

Birth	Name	ID
AM 12:00:00 2/6/2017	Muhammed	1