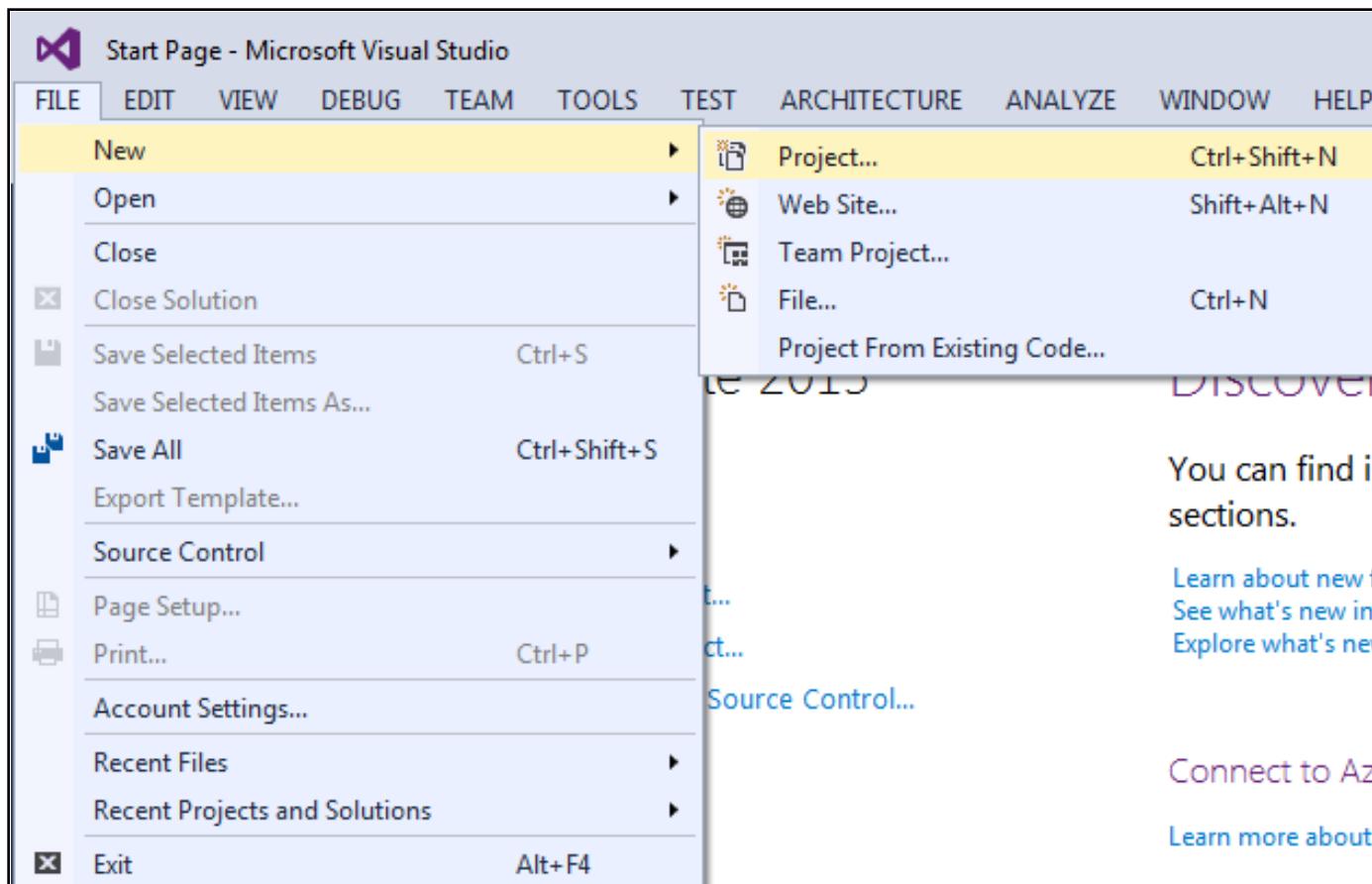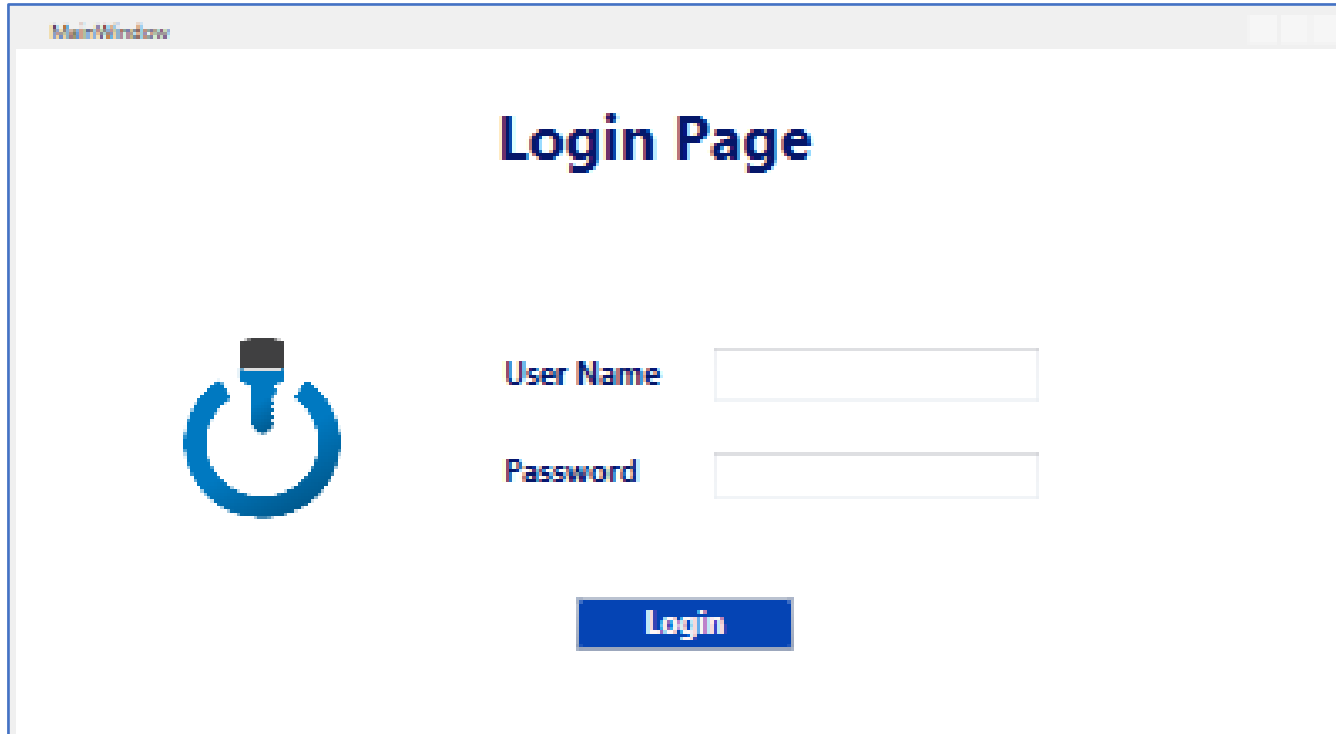# Connect SQL Server Database with Visual Studio C#

- In Microsoft Visual Studio,

- File -> New -> Project
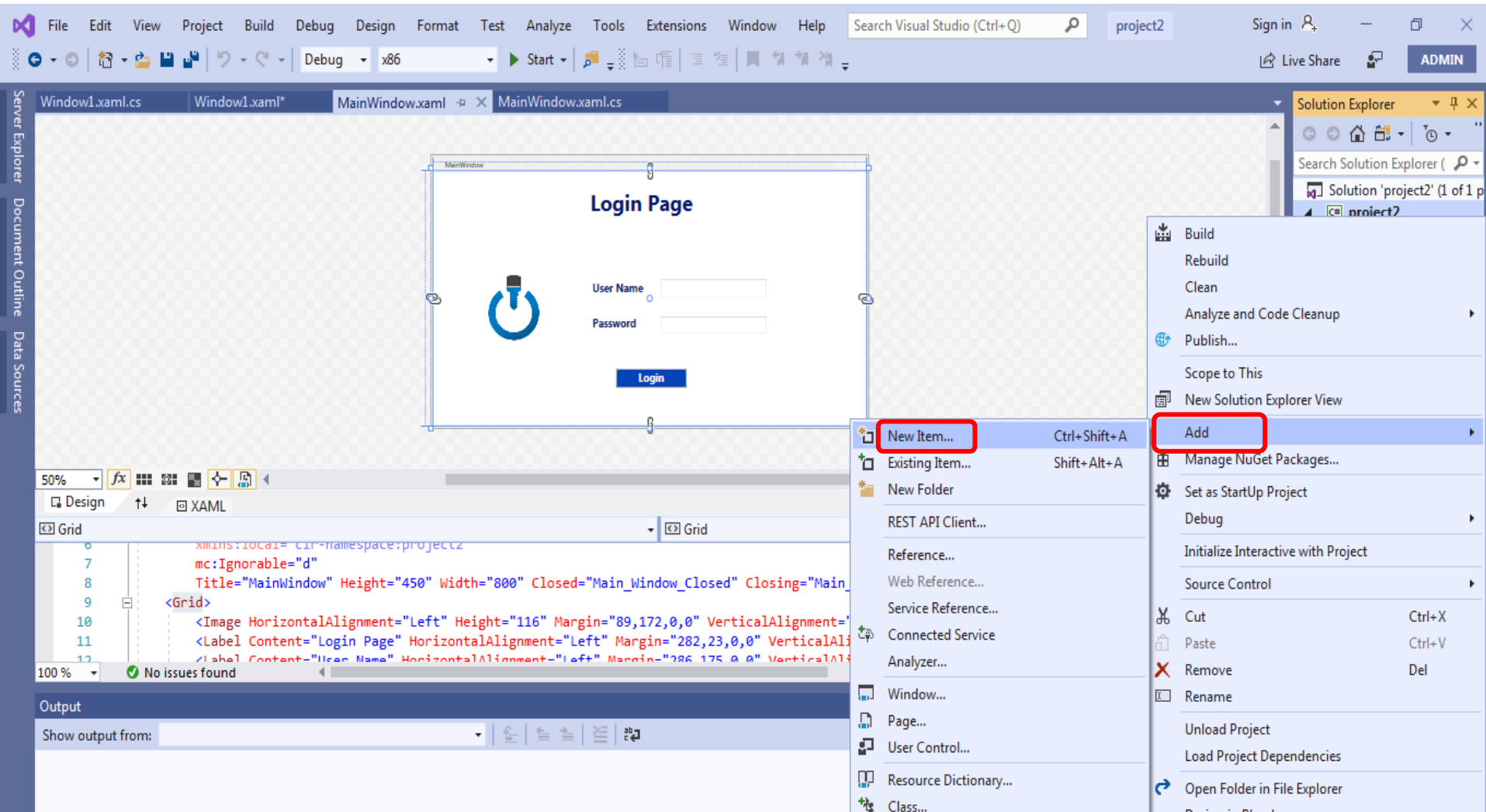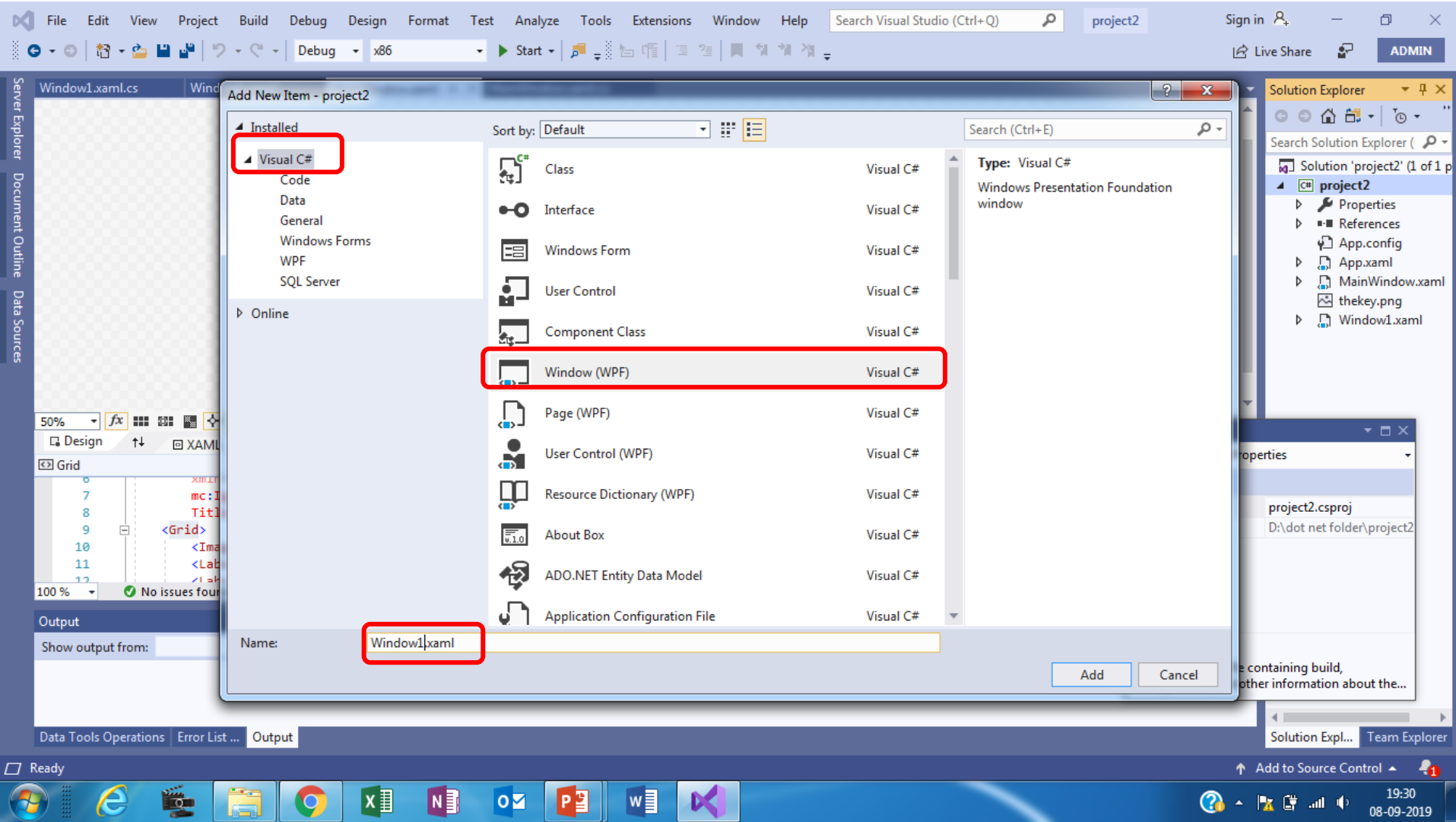
MainWindow.xaml

1. Right click on window.
2. Select properties.
3. You can set background color, foreground color, font color, font size etc.

Create another Window – Window1

Create another Window – Window1

MainWindow.xaml
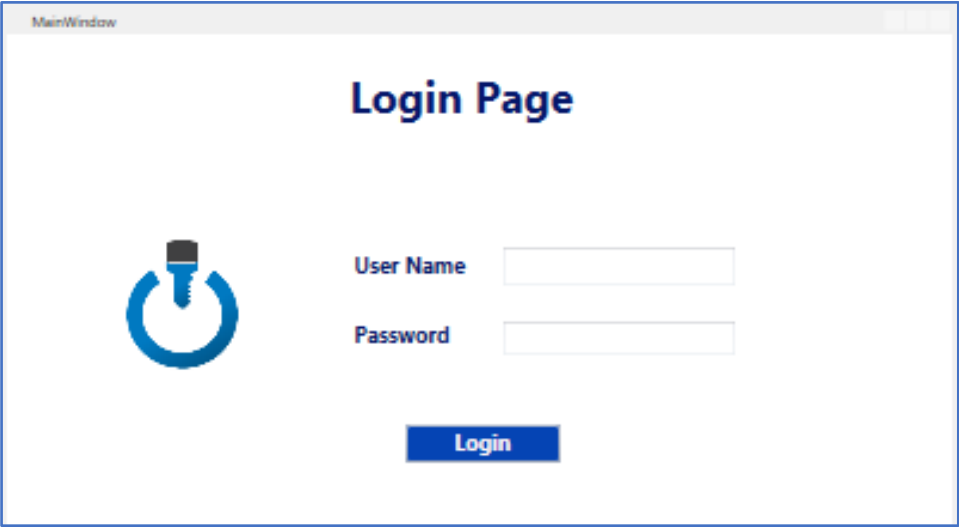
Tool	Content	Name (ID)
Label	Login Page	-
Label	User Name	-
Label	Password	-
TextBox		user
Password Box		pass
Button	Login	login

MainWindow.xaml
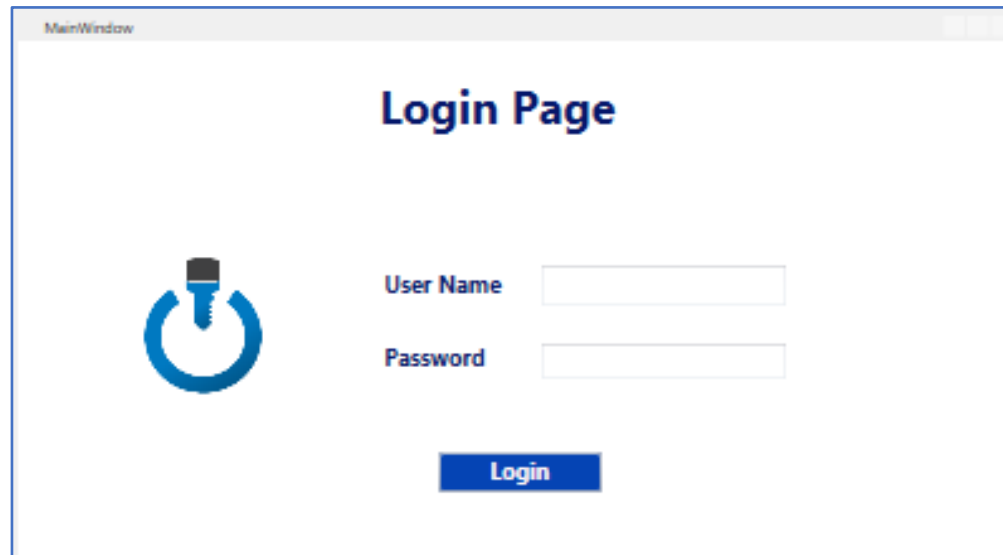
To insert image

- Drag and drop *Picture Box* from toolbox on to the Window.
- Right click picture Box.
- Click the common->Source
- Select the desired image.
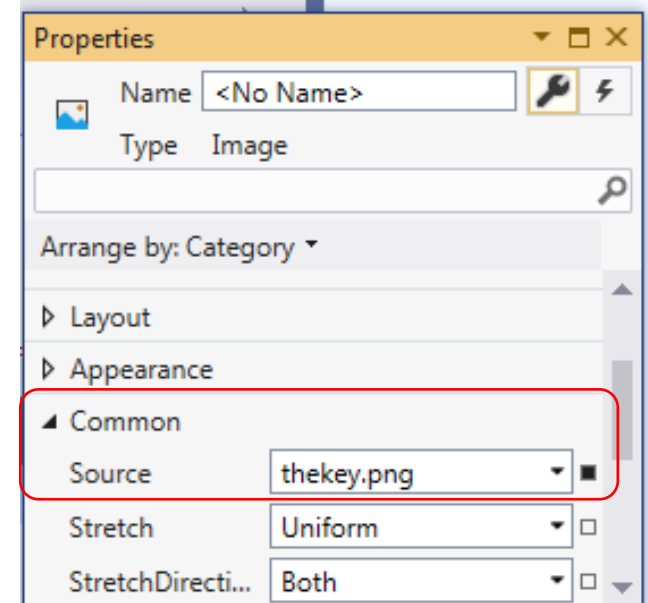
MainWindow.xaml

```xml
    xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
    xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
    xmlns:local="clr-namespace:project2"
    mc:Ignorable="d"
    Title="MainWindow" Height="450" Width="800" Closed="Main_Window_Closed" Closing="Main_Window_Closing" >
<Grid>
```

In XAML section, add the following in <window> tag
Closed="Main_Window_Closed" Closing="Main Window_Closing"
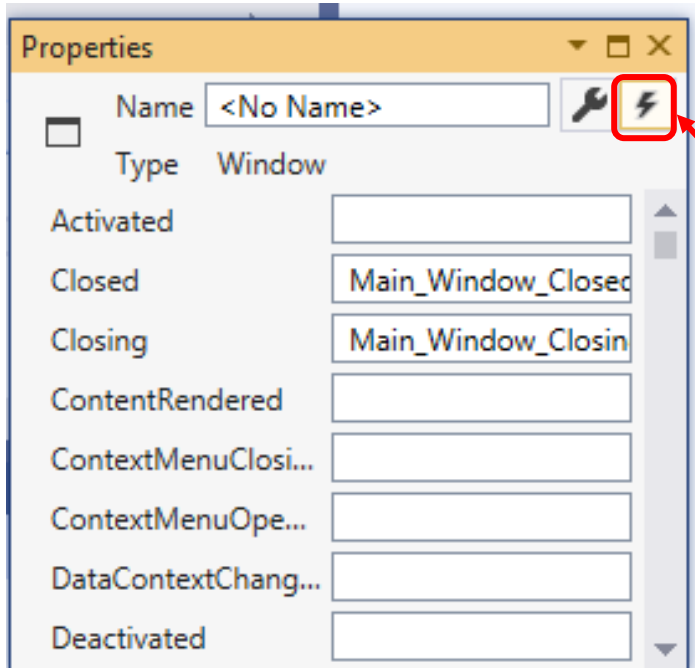
MainWindow.xaml

```xml
    xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
    xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
    xmlns:local="clr-namespace:project2"
    mc:Ignorable="d"
    Title="MainWindow" Height="450" Width="800" Closed="Main_Window_Closed" Closing="Main_Window_Closing" >
<Grid>
```

Properties

Name `<No Name>`

Type Window

Activated	
Closed	Main_Window_Closed
Closing	Main_Window_Closin
ContentRendered	
ContextMenuClosi...	
ContextMenuOpe...	
DataContextChang...	
Deactivated	

- **Select properties of MainWindow**

- **Select Events**

- **Select Closed**

- **Press Enter**

- **Method will be opened.**

MainWindow.xaml.cs

```csharp
private void Main_Window_Closed(object sender, EventArgs e)
{
        Application.Current.Shutdown();
}
```

Similarly

```csharp
private void Main_Window_Closing(object sender, System.ComponentModel.CancelEventArgs e)
{
        System.Diagnostics.Process.GetCurrentProcess().Kill();
}
```

MainWindow.xaml.cs

```csharp
private void Login_Click(object sender, RoutedEventArgs e)
{
        string n1, n2;
        n1 = user.Text;
        n2 = pass.Password;
        if (n1 == "admin" && n2 == "admin")
        {
           MessageBox.Show("Logged in successfully");
           Window1 f2 = new Window1();
           this.Hide();
           f2.ShowDialog();
        }
        else
        {
           MessageBox.Show("Invalid username/password");
           this.Hide();
           System.Diagnostics.Process.GetCurrentProcess().Kill();
        }
}
```

Window1.xaml
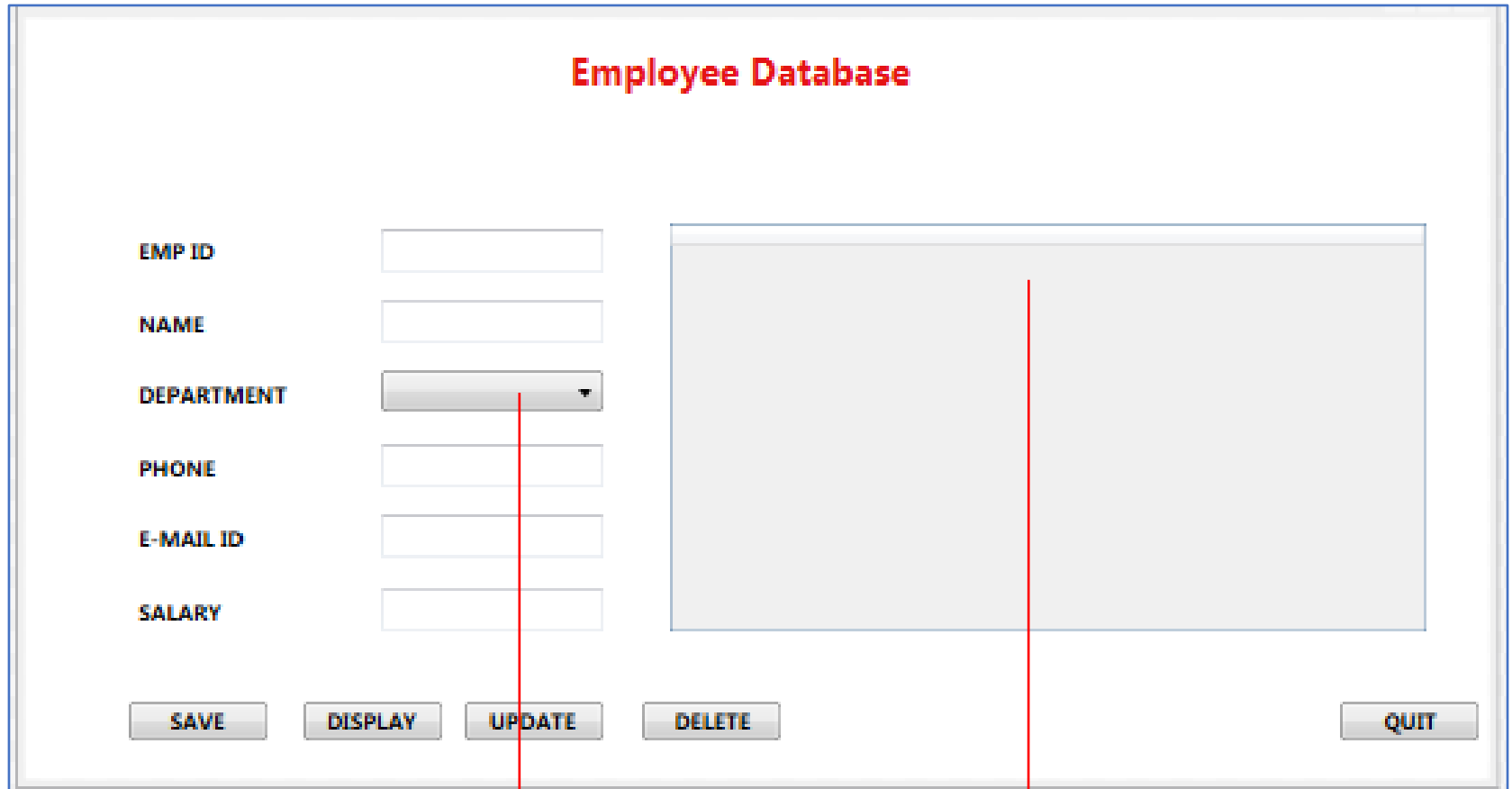
Window1.xaml

Employee Database

EMP ID

NAME

DEPARTMENT

PHONE

E-MAIL ID

SALARY

SAVE DISPLAY UPDATE DELETE QUIT

ComboBox

Data Grid View
Used to display table contents

Window1.xaml

Tool	Content
Label	Employee Database
Label	EMP ID
Label	NAME
Label	DEPARTMENT
Label	PHONE
Label	EMAIL-ID
Label	SALARY

Tool	Content	Name (ID)
DataGrid		grid
TextBox		name
ComboBox		dept
TextBox		phone
TextBox		email
TextBox		salary

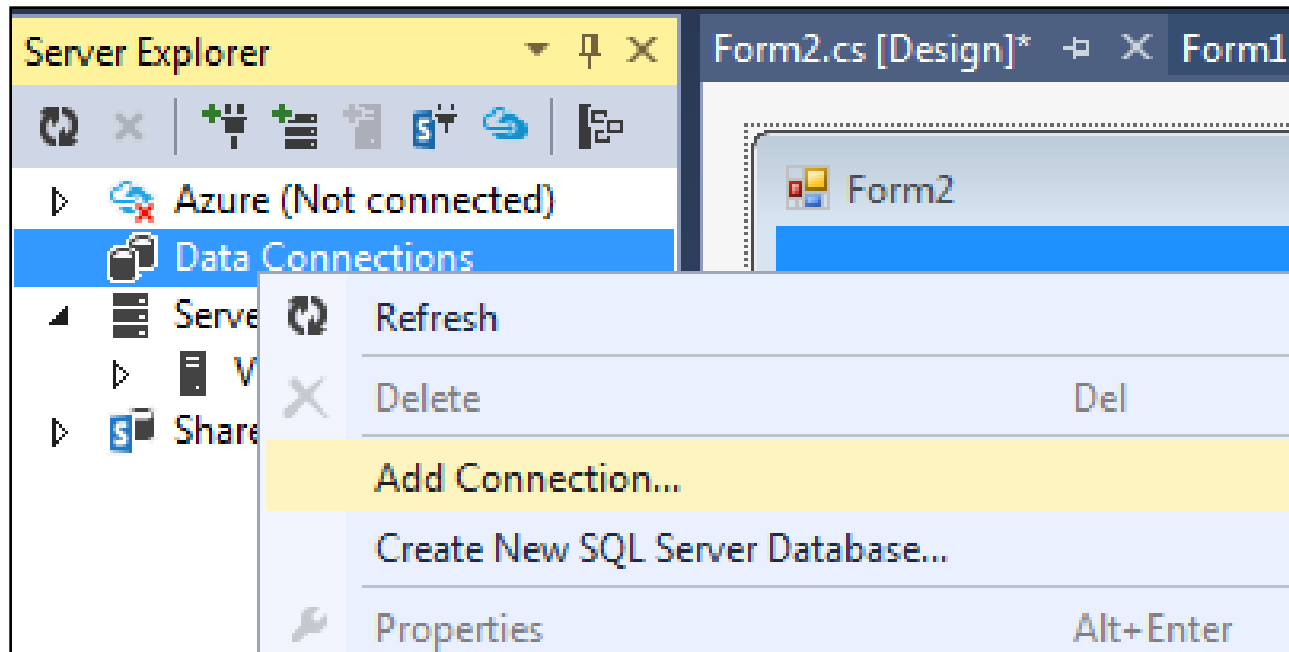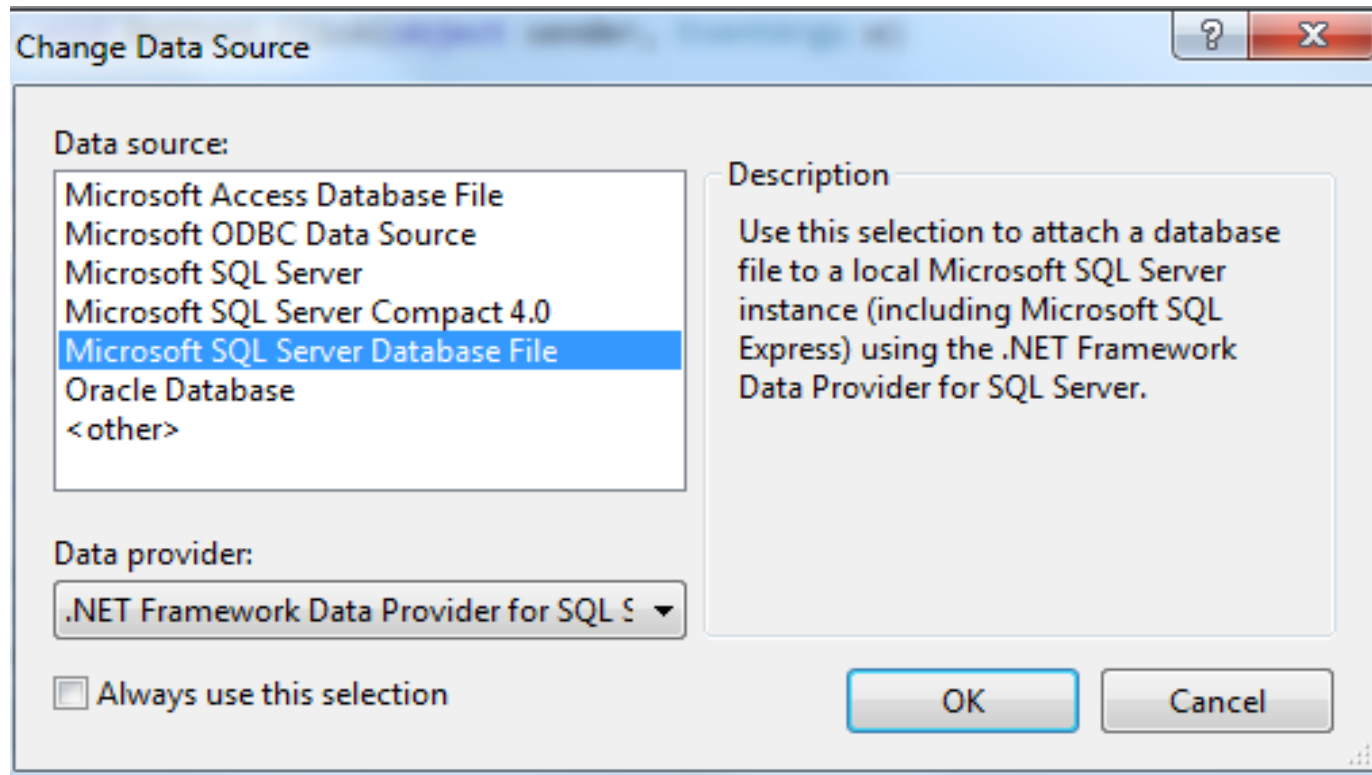| Tool	Content	Name (ID)
Button	SAVE	save
Button	DISPLAY	display
Button	UPDATE	update
Button	DELETE	delete
Button	QUIT	quit

To add items to ComboBox
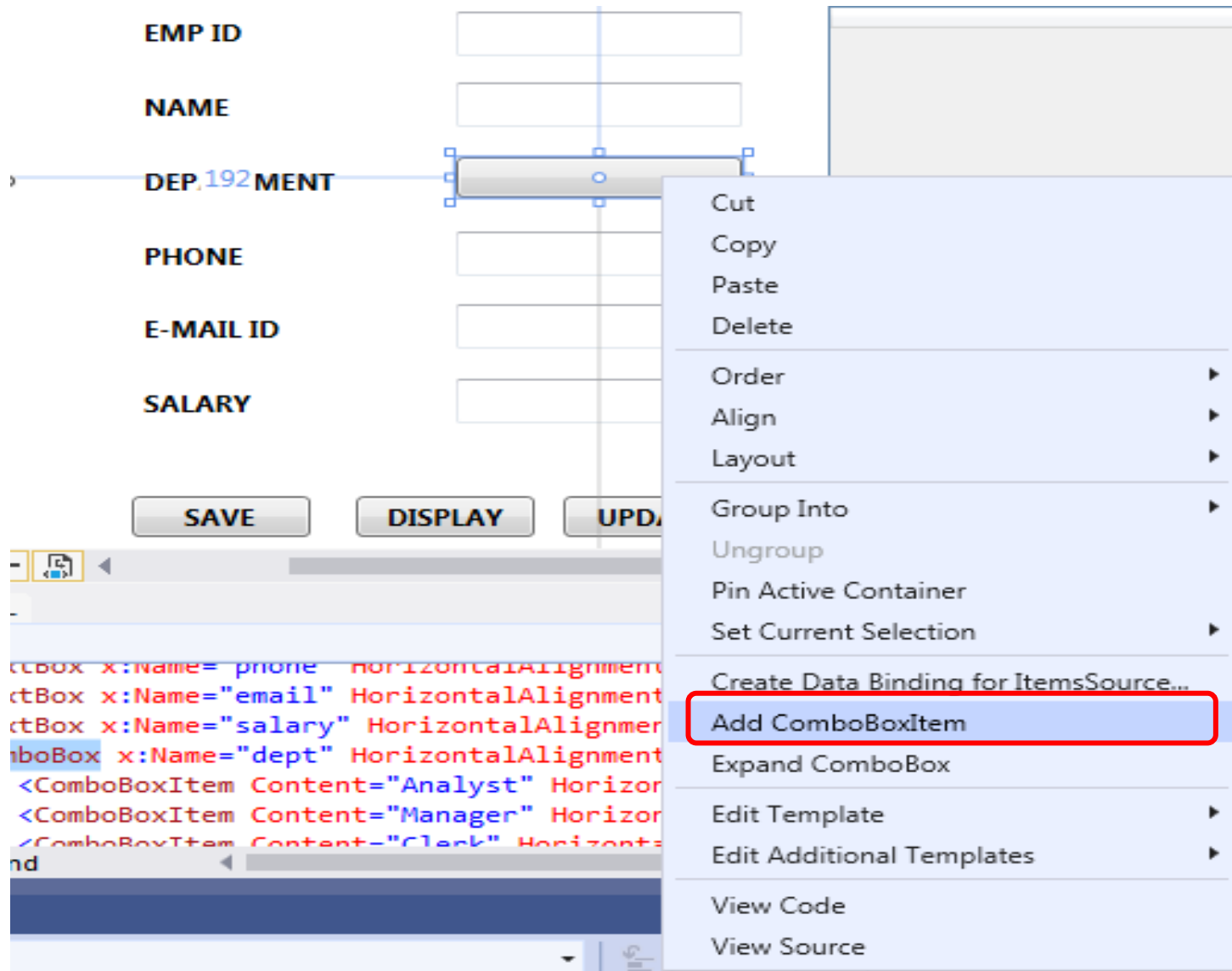
Connect to database

Connect to database

- Click View, *Server Explorer*
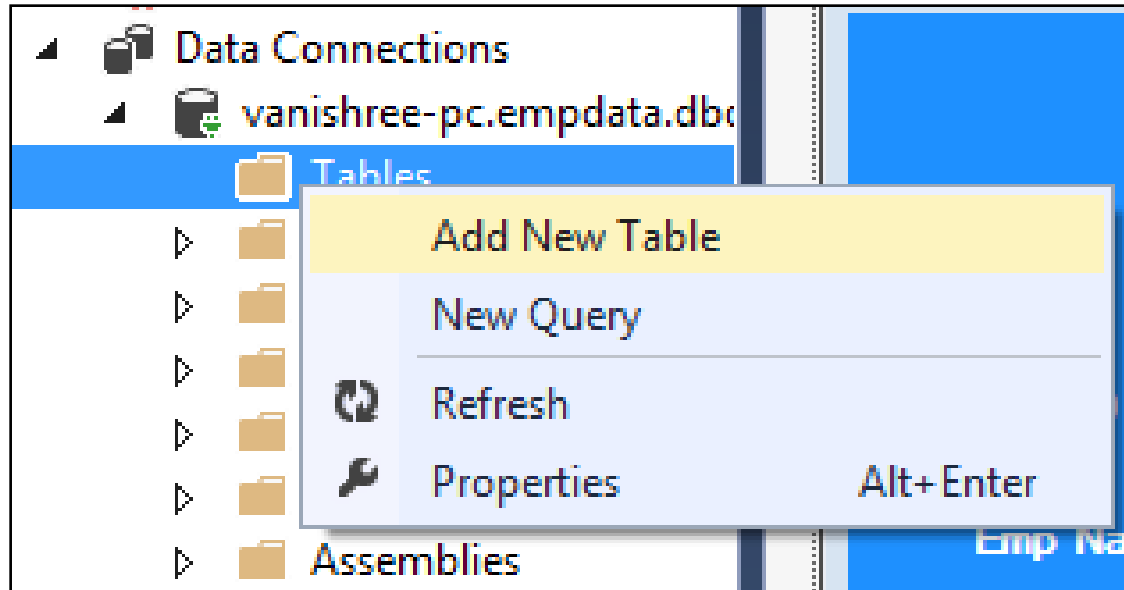- Right click on Data Connections, select *Add connection.*

Connect to database

Change Data Source

Data source:
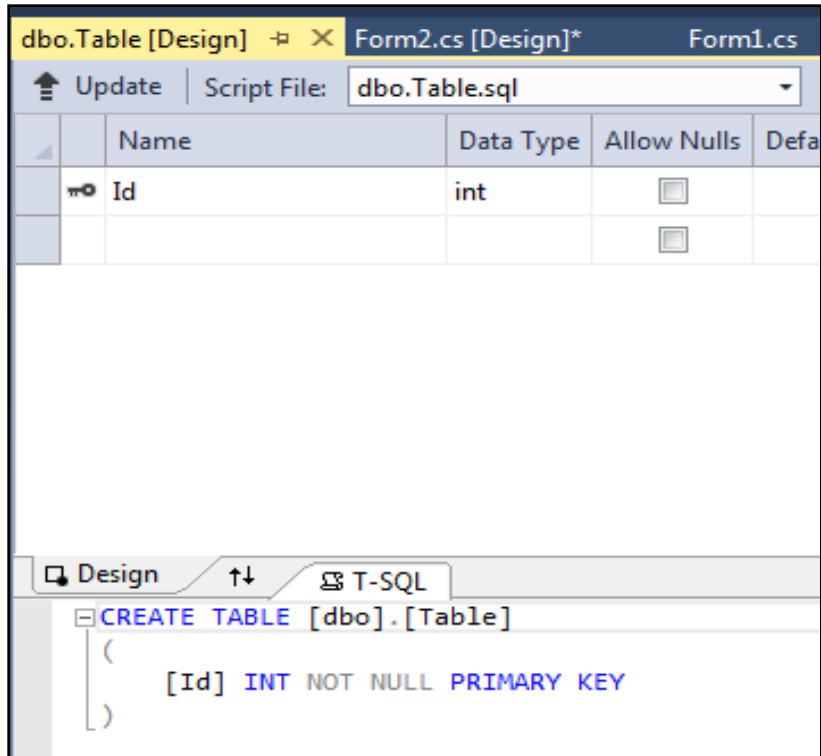
- Microsoft Access Database File
- Microsoft ODBC Data Source
- Microsoft SQL Server
- Microsoft SQL Server Compact 4.0
- **Microsoft SQL Server Database File**
- Oracle Database
- <other>

Description

Use this selection to attach a database file to a local Microsoft SQL Server instance (including Microsoft SQL Express) using the .NET Framework Data Provider for SQL Server.

Data provider:

.NET Framework Data Provider for SQL S ▼

☐ Always use this selection

OK   Cancel

Select or enter database name :
empdata

Click OK

Connect to database
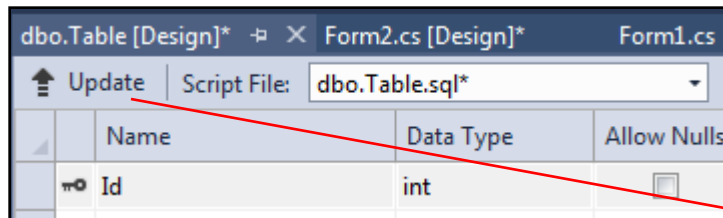
Connect to database

Add Columns and data types.

1. Delete ID Row.
   (Right click ID, select delete)

2. Select empid as primary key.
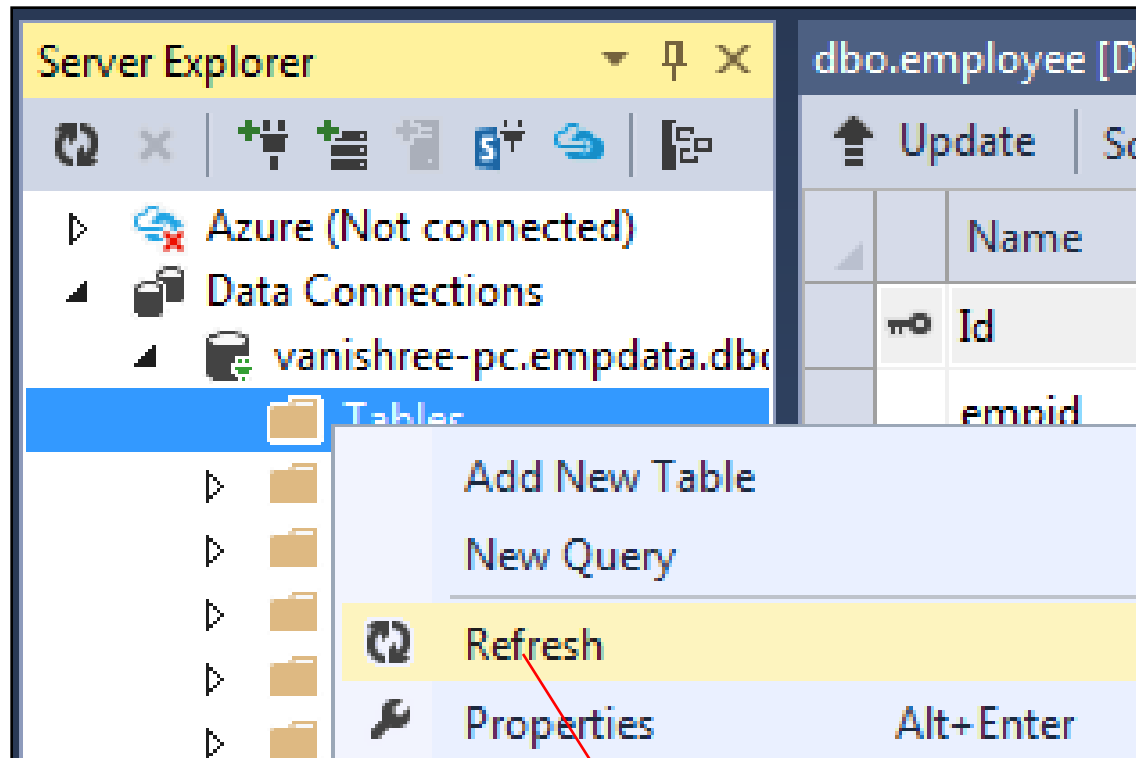   (Right click empid, select set as primary key)

Connect to database
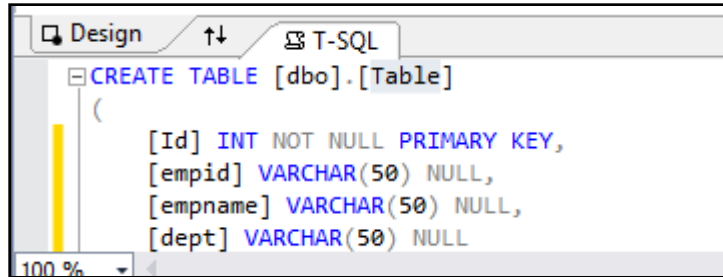
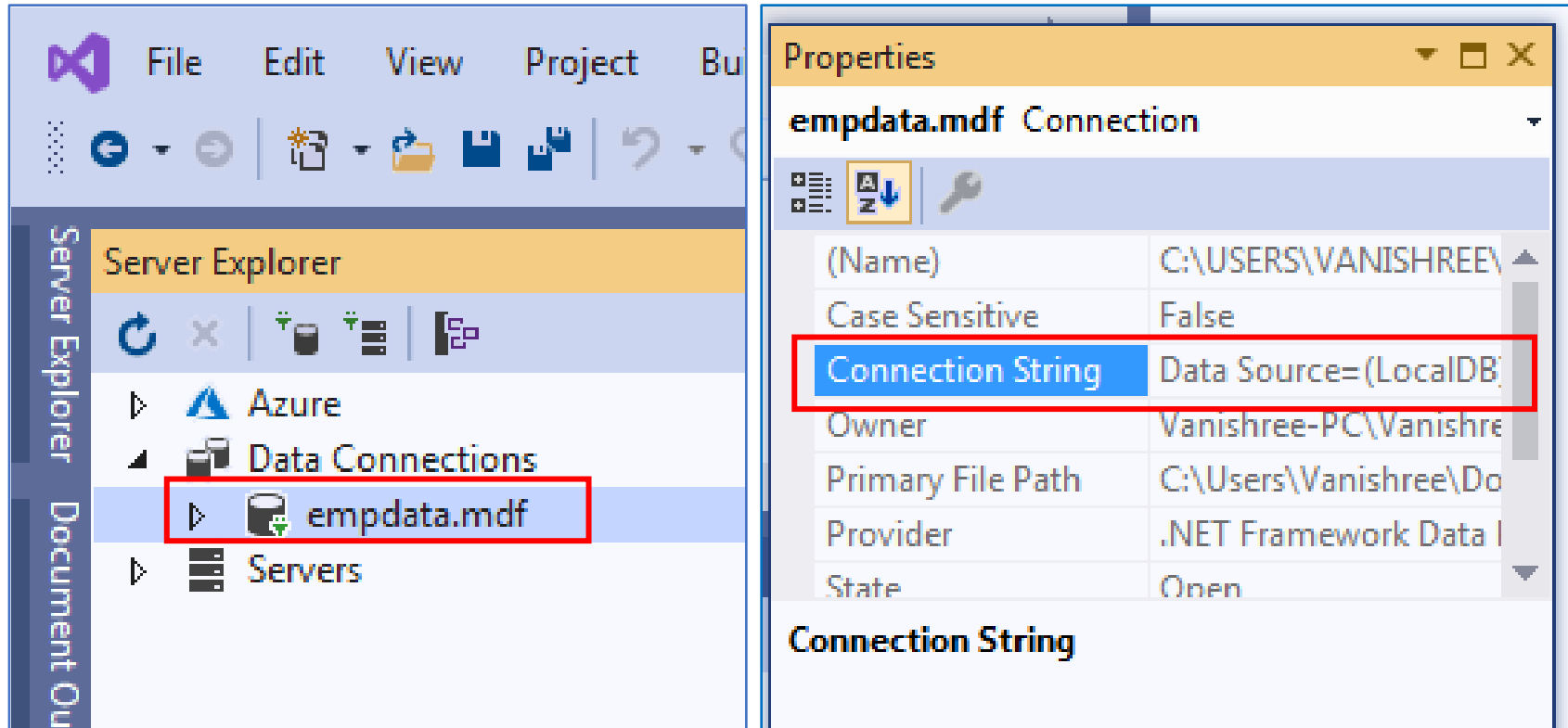Change Table as **employee**

Click on **update**

Connect to database

Click on refresh

Connection String

Click on Connection String :
Copy the string

"Data Source
=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\Vanishree\Documents\empdata.mdf;I
ntegrated Security=True;Connect Timeout=30"

And paste it in Window1.xaml.cs

Window1.xaml.cs

```csharp
using System.Data.SqlClient;
using System.Data;


public partial class Window1 : Window
{
    SqlConnection con = new SqlConnection(@"Data Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\Vanishree\Documents\empdata.mdf;Integrated Security=True;Connect Timeout=30");


......
......



}
```

Window1.xaml

```csharp
private void Save_Click(object sender, RoutedEventArgs e)
    {
        try
        {
            con.Open();
            SqlCommand cmd = con.CreateCommand();
            cmd.CommandType = CommandType.Text;
            cmd.CommandText = "insert into employee values('" + empid.Text + "','"
+     name.Text + "','" + dept.Text + "','" + phone.Text + "','" + email.Text + "','" +
salary.Text + "')";
            cmd.ExecuteNonQuery();
            con.Close();
            displayall();
            MessageBox.Show("Record inserted successfully");
        }
```

Window1.xaml

Double Click on SAVE Button

SAVE	DISPLAY	UPDATE	DELETE		QUIT

Continued........

```csharp
catch (Exception ex)
 {
    MessageBox.Show(ex.Message);
 }

}
```

Double Click on SAVE Button
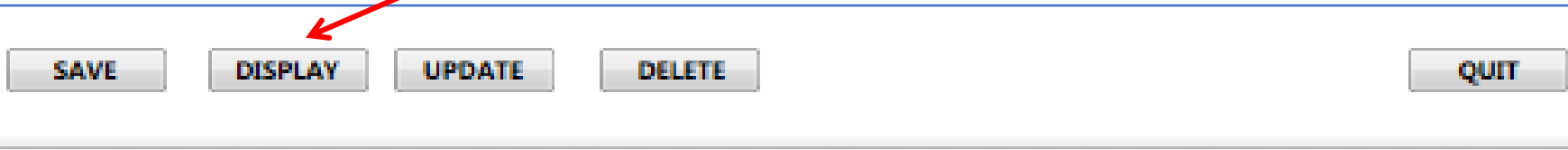

Continued........

```
public void displayall()
{
    con.Open();
    SqlCommand cmd = con.CreateCommand();
    cmd.CommandType = CommandType.Text;
    cmd.CommandText = "select * from employee";
    cmd.ExecuteNonQuery();
    DataTable dt = new DataTable();
    SqlDataAdapter da = new SqlDataAdapter(cmd);
    da.Fill(dt);
    grid.ItemsSource = dt.DefaultView;
    con.Close();
}
```
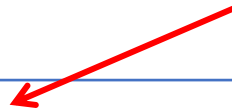
Window1.xaml

Double Click on DISPLAY Button


```
private void Display_Click(object sender, RoutedEventArgs e)
{
    displayall();
}
```
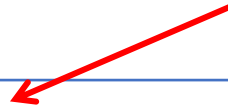
Double Click on UPDATE Button

| SAVE | DISPLAY | UPDATE | DELETE | QUIT |

```
private void Update_Click(object sender, RoutedEventArgs e)
{
    con.Open();
    SqlCommand cmd = con.CreateCommand();
    cmd.CommandType = CommandType.Text;
    cmd.CommandText = "update employee  set empname='" + name.Text + "', dept='"
+ dept.Text + "', phone='" + phone.Text + "', salary='" + salary.Text + "' where empid='" +
empid.Text + "'";
    cmd.ExecuteNonQuery();
    con.Close();
    displayall();
    MessageBox.Show("Record updated successfully");
}
```

Double Click on DELETE Button

| SAVE | DISPLAY | UPDATE | DELETE | | QUIT |

```csharp
private void Delete_Click(object sender, RoutedEventArgs e)
{
    con.Open();
    SqlCommand cmd = con.CreateCommand();
    cmd.CommandType = CommandType.Text;
    cmd.CommandText = "delete from employee where empid='" + empid.Text + "'";
    cmd.ExecuteNonQuery();
    con.Close();
    displayall();
    MessageBox.Show("Record deleted successfully");
}
```

Window1.xaml

Double Click on QUIT Button

| SAVE | DISPLAY | UPDATE | DELETE | QUIT |

```
private void Quit_Click(object sender, RoutedEventArgs e)
{
    System.Diagnostics.Process.GetCurrentProcess().Kill();
}
```

Project is ready to execute !!!

1. Build the solution
2. Start without Debugging