

[Learn to Code and Setup Your ASP.NET Platform \(http://freeasphosting.net/register.aspx\)](http://freeasphosting.net/register.aspx)[Login \(http://freeasphosting.net/cp/login.aspx\)](http://freeasphosting.net/cp/login.aspx)

ASP.NET Exercises

- [Getting Started with Visual Studio \(getting-started-with-visual-studio.html\)](#)
- [Creating a Login System \(login-system-exercise.html\)](#)
- [Creating an Online Users Counter \(online-counter-exercise.html\)](#)
- [Creating a Contact Form \(contact-form-exercise.html\)](#)
- [Creating a Comment Box \(comment-box-exercise.html\)](#)
- [ASP.NET Grid Control \(asp-net-grid-control.html\)](#)

Creating a Login System

[code/Login_System.zip](#)[Download Sample Source Code \(https://freeasphosting.net/sample-](https://freeasphosting.net/sample-code/Login_System.zip)

Greetings, and welcome back to FreeASPHosting.net (<https://freeasphosting.net/>) trainings with ASP.NET. In this tutorial, we will go through the steps of setting up an Authentication system, or simply put “Login System”.

In order to help you get started quickly, we have prepared a “getting started” package, that you may freely download. (https://freeasphosting.net/sample-code/Login_System.zip)

Prerequisites

This tutorial is very simple, but assumes that you already have the following:

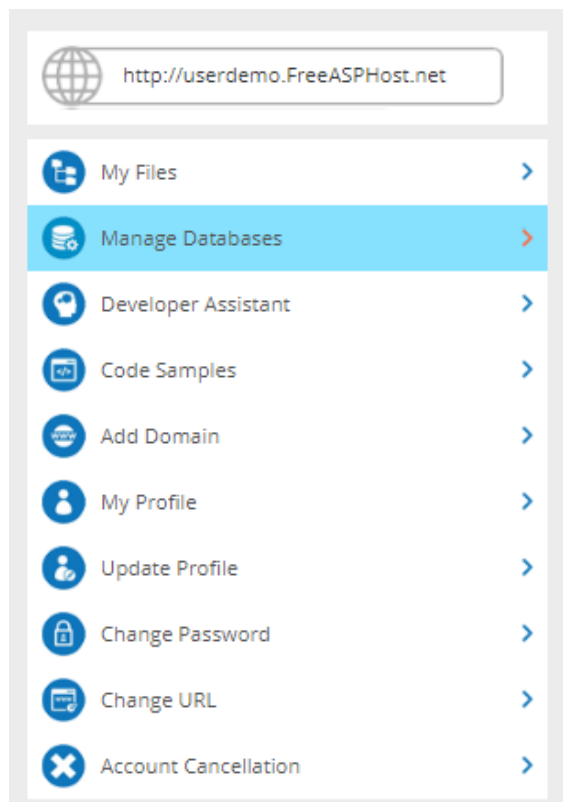
- Visual Studio 2015 or newer installed on your machine.

- Microsoft SQL Server Management Studio.
- An Account at FreeASPHosting.net. (<https://freeasphosting.net/>) If you don't have one already, please subscribe here. (<https://freeasphosting.net/register.aspx>)

Adding the Login System

1 Creating a Database

First, login to your FreeASPHosting.net (<https://freeasphosting.net/register.aspx>) Account. From the Dashboard, on the sidebar, click on "Manage Databases"



Databases

Here you can create new or manage existing databases.

[Add Database](#)[Create Sample Database](#)

Next, click on the “Create Sample Database” button, make sure the “Create Sample Database” tab is active, then press “OK” to proceed.



Developer Code Samples

Deploying an ASP.NET Web Application with SQL Server

This sample will show you how to connect to your MS SQL Server database through the use of ASP.NET. This code will also show the results from your sample SQL Server database.

[Create Sample Database](#)[Instructions](#)

Click OK to create your first MS SQL sample database project that can be accessed and edited directly from your file manager. You can also manager your databases via your "Manage Databases" console.

For more detailed instructions how to create your ASP.NET application please click on the instructions tab.

[OK](#)[Cancel](#)

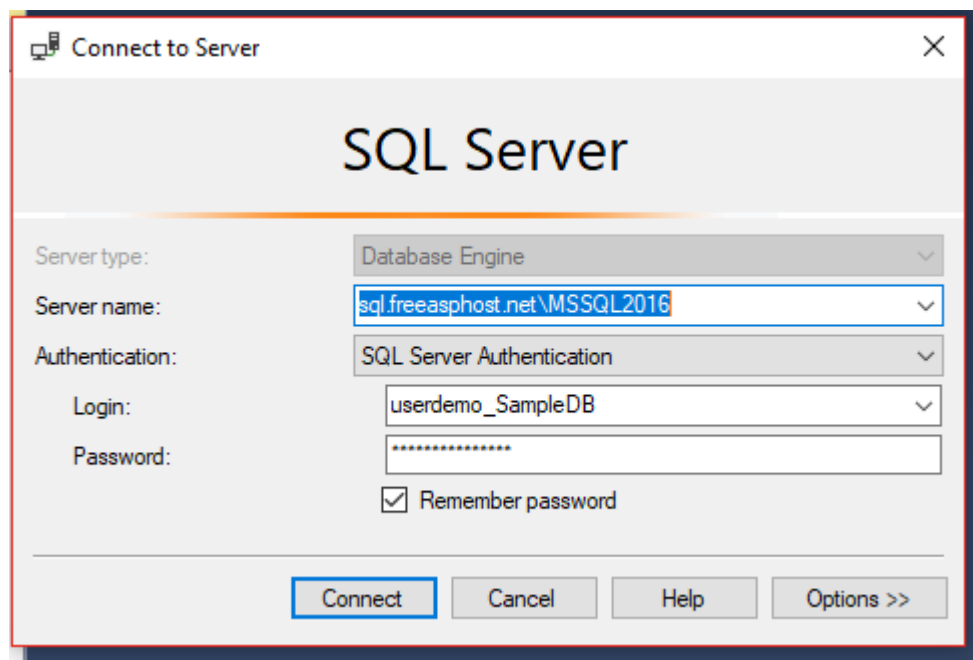
2 Setting up Microsoft SQL Server Management Studio

Start your SQL Server Management Studio (SSMS). Use the data from your Sample database to login. You will need the following:

- *Server Name*

- Username
- Password

Once you are done, click on “Connect” to proceed.



3 Creating the Database Schema

Now it's time to create the tables that will hold the user data. Open the query panel and paste the following code. This SQL query simply creates a Table called "Ulogin", with a couple of fields: UserId and Password. Then it creates a test user with username "testuser".

```
Use userdemo_SampleDB      --Select database or use database

Create table Ulogin        -- create table Ulogin is my table name
(
    UserId varchar(50) primary key not null,  --primary key
    Password varchar(100) not null
)

insert into  Ulogin values ('testuser','123456') --insert value in Ulogin table
```

```
Use userdemo_SampleDB      --Select database or use database
Create table Ulogin        -- create table Ulogin is my table name
(
    UserId varchar(50) primary key not null,  --primary key not accept null value
    Password varchar(100) not null
)
insert into  Ulogin values ('testuser','123456') --insert value in Ulogin table
SELECT * FROM Ulogin
```

.00 %

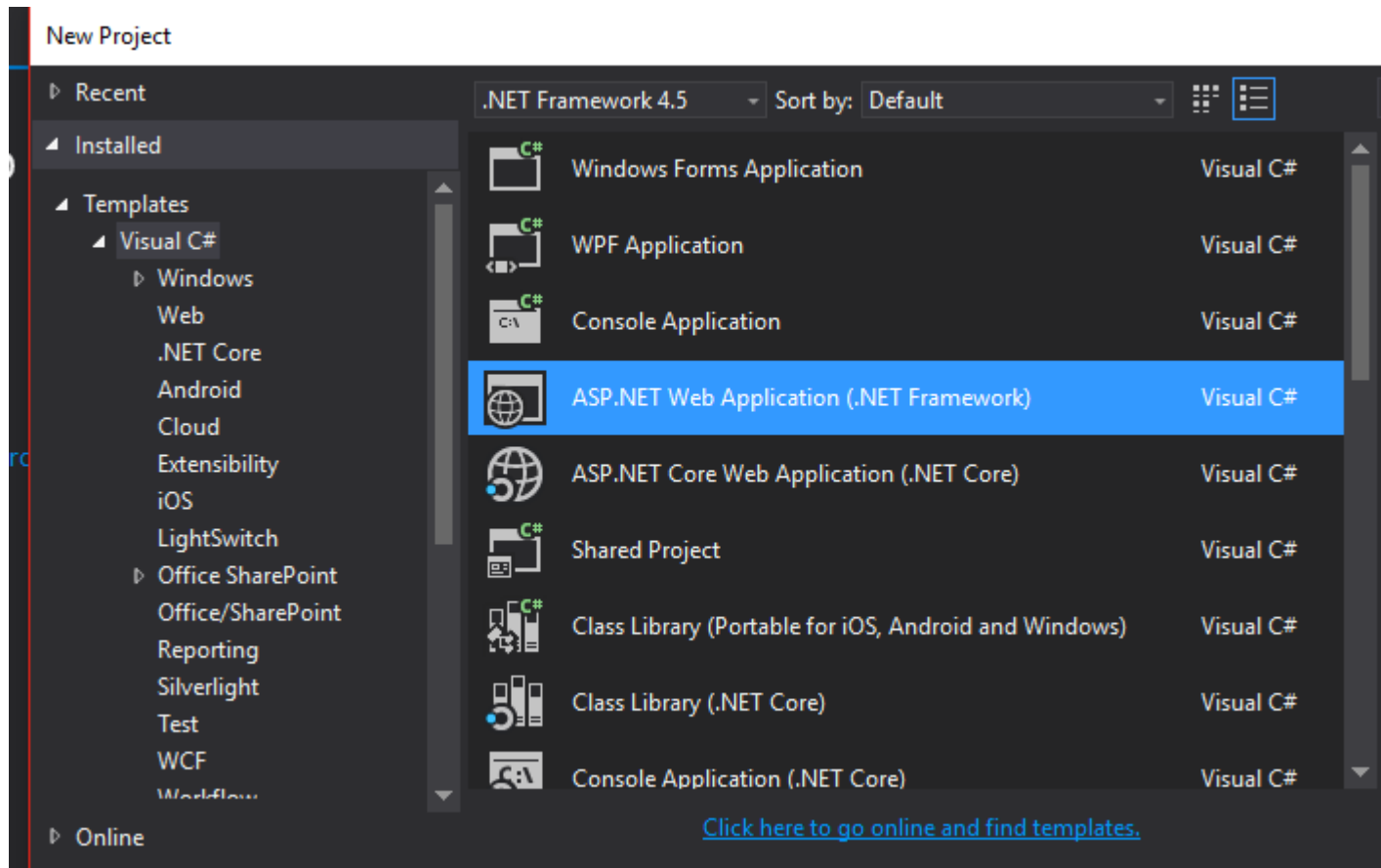
Results Messages

	UserId	Password
1	testuser	123456

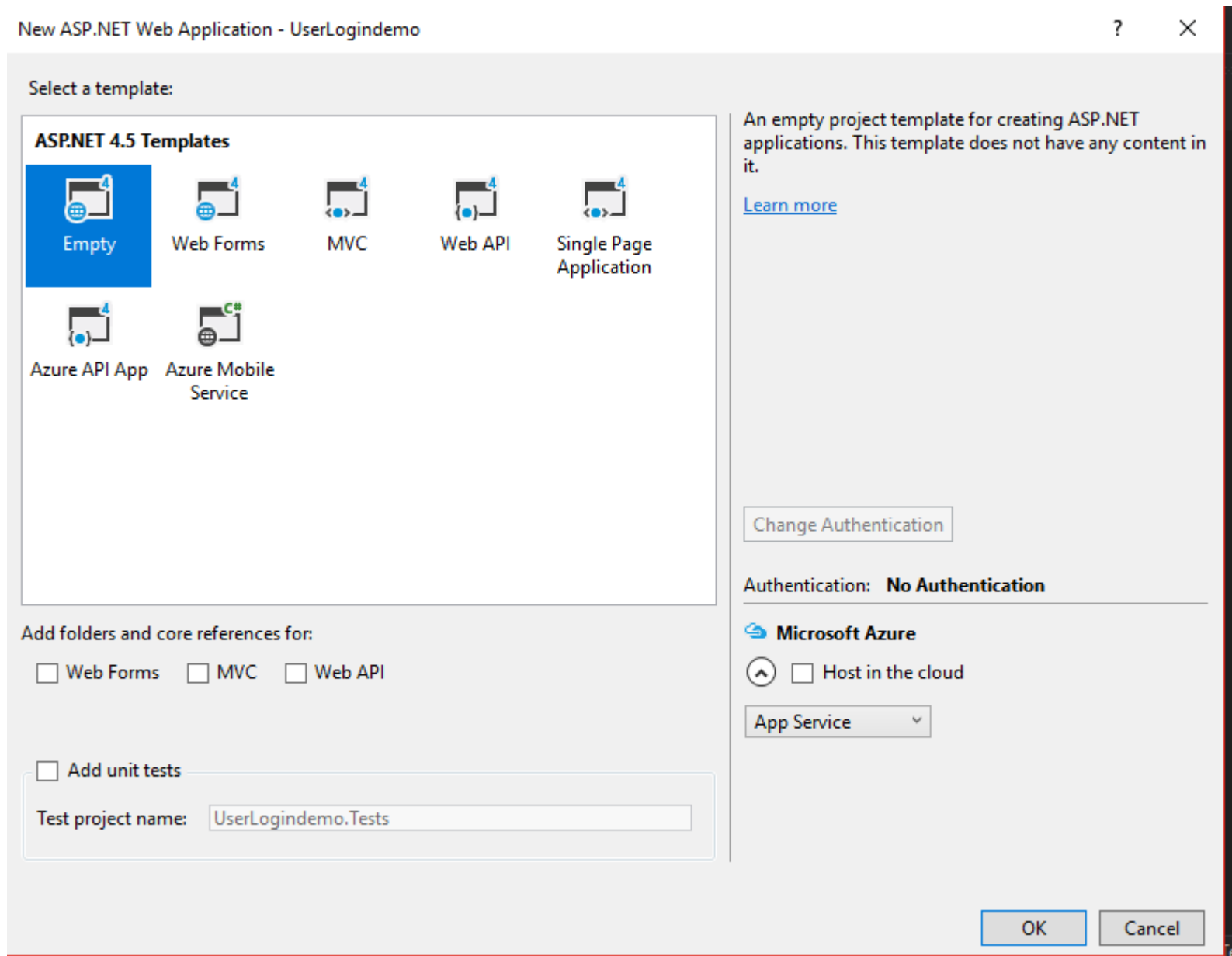
4 The Code

Now that we have set the database, it's time to start working on the ASP.NET code. We will be using a simple design to illustrate how the login page will work, as the purpose of this tutorial is demonstrating the Authentication system basics. So let's open Visual Studio.

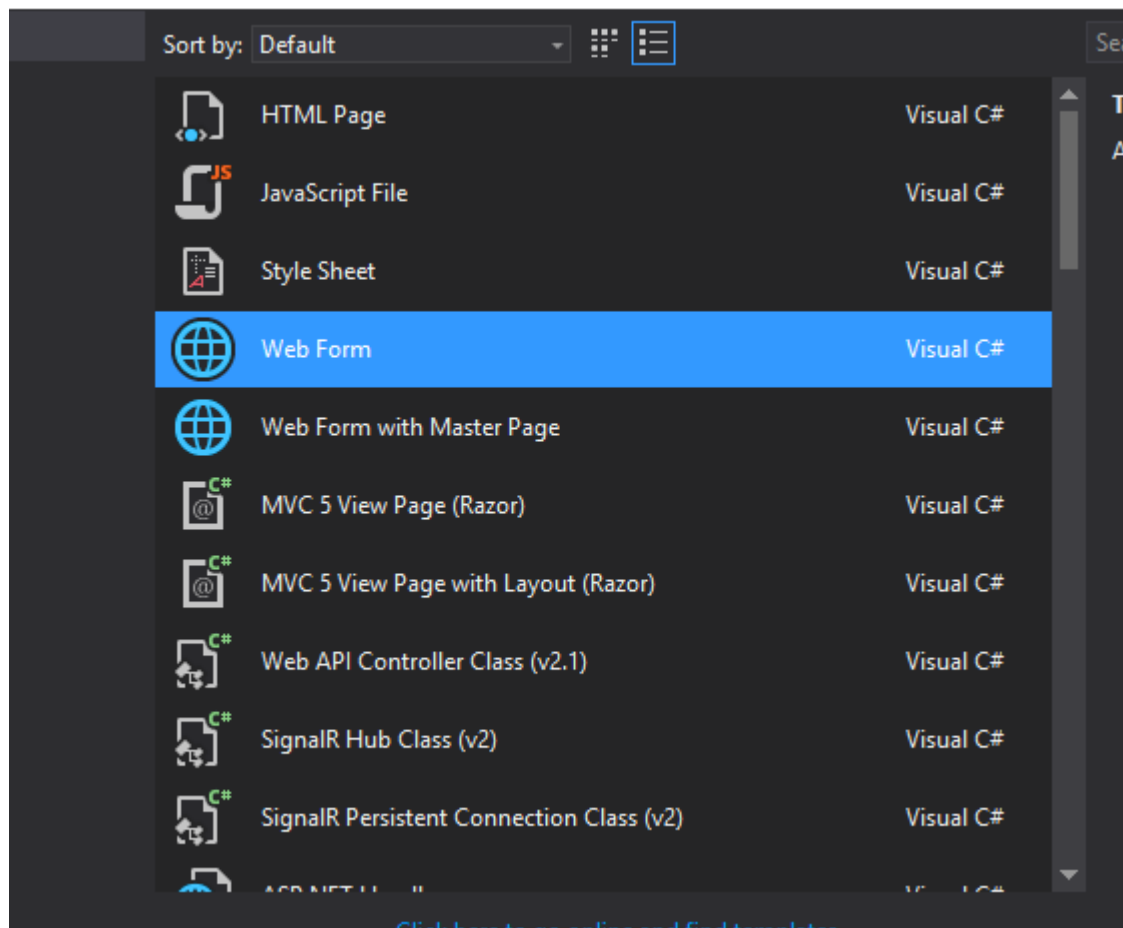
Create a new project, by selecting the "ASP.NET Web Application" template from the list of the available project types.



On the next window, select an the "Empty" template



Next, add a "Web Form" to your project.



At this point, we may start designing our login.aspx page. To lay the different elements on the screen, we will be using a Table. If you feel like making the elements in the middle of the screen, use the Table properties and set the text-align style property. Next, from the Toolbox, drag two labels, two textboxes and a single button, as illustrated in the sources code and image below. The labels and textboxes are used for the login and password respectively. The button will be used to submit the form's content to the server for processing.

```

<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="login.aspx.cs" Inherits="UserLogindemo.WebForm1" %>

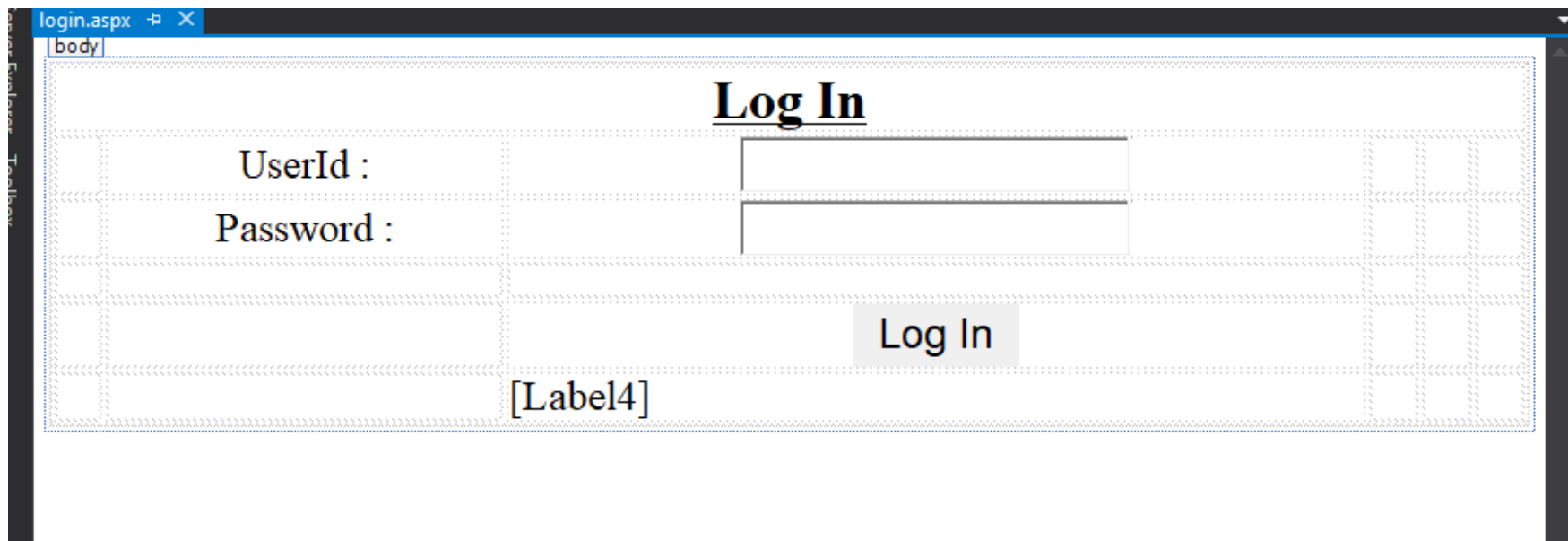
<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
<title></title>
<style type="text/css">
.auto-style1 {
    width: 100%;
}
</style>
</head>
<body>
<form id="form1" runat="server">
    <div>
        <table class="auto-style1">
            <tr>
                <td colspan="6" style="text-align: center; vertical-align: top"><asp:Label ID="Label1" runat="server" Font-Bold="true" Text="Login" /></td>
            </tr>
            <tr>
                <td></td>
                <td style="text-align: center"><asp:Label ID="Label2" runat="server" Font-Size="X-Large" Text="UserId :"></asp:Label>
                <td style="text-align: center"><asp:TextBox ID="TextBox1" runat="server" Font-Size="X-Large"></asp:TextBox></td>
                <td></td>
                <td></td>
                <td></td>
            </tr>
            <tr>
                <td></td>
                <td style="text-align: center"><asp:Label ID="Label3" runat="server" Font-Size="X-Large" Text="Password :"></asp:Label>
                <td style="text-align: center"><asp:TextBox ID="TextBox2" runat="server" Font-Size="X-Large" TextMode="Password"></asp:TextBox></td>
                <td></td>
                <td></td>
                <td></td>
            </tr>
            <tr>
                <td></td>
                <td></td>
                <td></td>
                <td></td>
                <td></td>
                <td></td>
            </tr>
        </table>
    </div>
</form>

```

```
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td style="text-align: center"><asp:Button ID="Button1" runat="server" BorderStyle="None" Font-Size="X-Large" OnC
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td><asp:Label ID="Label4" runat="server" Font-Size="X-Large"></asp:Label></td>
<td></td>
<td></td>
<td></td>
</tr>
</table>
</div>
</form>
</body>
</html>
```

The login form should look like the following, as described earlier, the couple of inputs (textboxes) will be used by the user to type account credentials.



Before going any further with the code, we should connect our ASP.NET application to the database server. In order to do that, we should create a Connection String. In the Solution Explorer, browse down to the web.config file, and add the following Connection String :

```
<connectionStrings>
  <add name="mycon" connectionString="Data Source=sql.freeasphost.net\MSSQL2016;Initial Catalog=userdemo_SampleDB;Persi
</connectionStrings>
```

Now, we can add the code that will handle the connection process, open login.aspx.cs in Visual Studio, and insert the following code:

```
using System.Data.SqlClient;
using System.Configuration;
```

Then, let's create an SQL connection object. The Connection's name should be the same as the one in the web.config file. In this case: "mycon".

```
SqlConnection con = new  
SqlConnection(ConfigurationManager.ConnectionStrings["mycon"].ToString());
```

Remember that login button we created on the page's design? Now it's time to make it work, by adding a click event handler as illustrated in the following code:

```
protected void Button1_Click(object sender, EventArgs e)
{
    try
    {
        string uid = TextBox1.Text;
        string pass = TextBox2.Text;
        con.Open();
        string qry = "select * from Ulogin where UserId='" + uid + "' and Password='" + pass + "'";
        SqlCommand cmd = new SqlCommand(qry, con);
        SqlDataReader sdr = cmd.ExecuteReader();
        if (sdr.Read())
        {
            Label4.Text = "Login Success.....!!";
        }
        else
        {
            Label4.Text = "UserId& Password Is not correct Try again..!!";
        }
    }
    con.Close();
}
catch (Exception ex)
{
    Response.Write(ex.Message);
}
```

The code is very simple to understand, the "Button1_Click" method is triggered when the user presses the login button. Once inside, we retrieve the account credentials typed by the user from the TextBox objects.

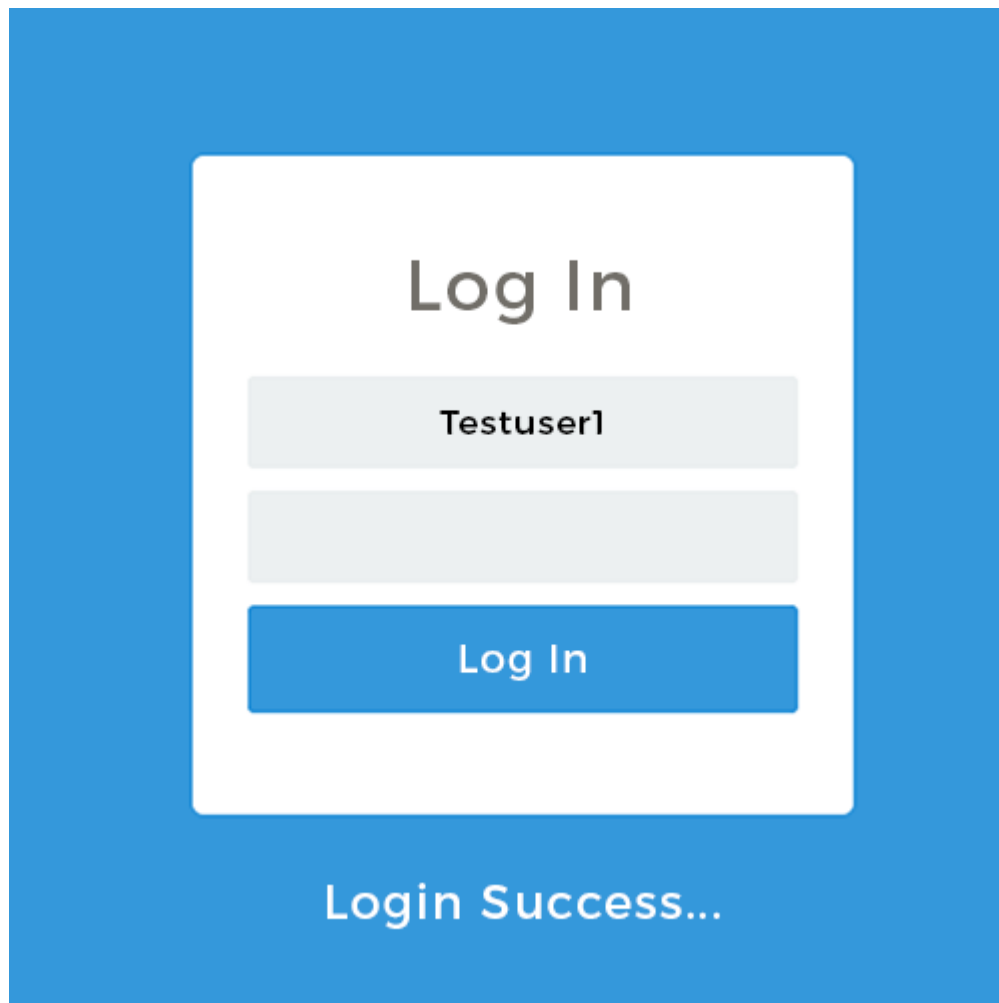
Then we simply check the database, through a simple "SELECT" query, if a user with that login AND password exists. If the check passes then the login succeeds.

Log In

Testuser1

.....

Log In



« Previous ([getting-started-with-visual-studio.html](#))

Next » ([online-counter-exercise.html](#))

[Home \(http://freeasphosting.net\)](http://freeasphosting.net) | [About Us \(http://freeasphosting.net/about-us.html\)](http://freeasphosting.net/about-us.html) | [Contact \(http://freeasphosting.net/contact.aspx\)](http://freeasphosting.net/contact.aspx) |
[Privacy \(http://freeasphosting.net/privacy.html\)](http://freeasphosting.net/privacy.html) | [Terms of Use \(http://freeasphosting.net/terms-of-use.html\)](http://freeasphosting.net/terms-of-use.html)

© 2019 FreeASPHosting.net