

Assignment #3

Objective: Learn about HTML forms, server-side processing (using Microsoft’s ASP.NET Web Pages), implement user registration and login. Other techniques learned include DB store and lookup, and using permanent cookies.

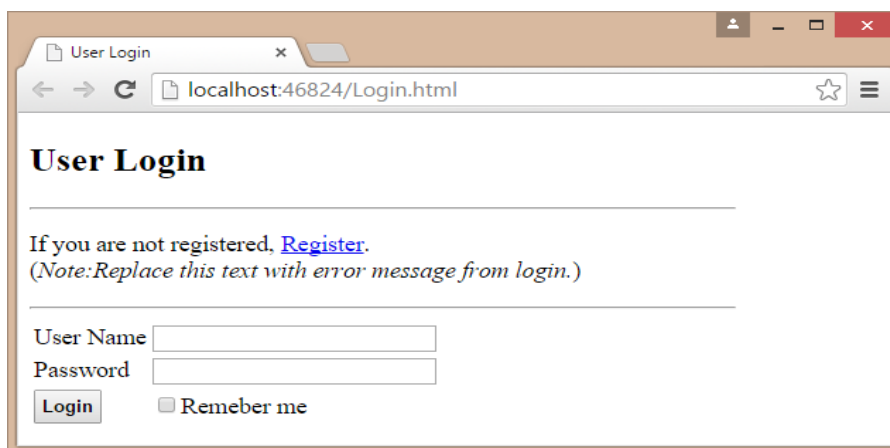
Submission & Deadline: Send an e-mail to instructor with the URL of your site by Feb. 18, 2016.

You are to develop a basic web application. The application’s “**default.cshtml**” page automatically displays “Welcome <the user name>” if the incoming HTTP request includes a cookie named “UserID” with a valid user id; otherwise, the login page is shown (use *Response.Redirect*). The application also includes two more web pages: “**login.cshtml**” and “**register.cshtml**”. *The login page has a link to the registration page.* The login page is shown below where you have to supply the server-side code to process the submitted data as follows:

If the user information is invalid then reply with the login page again with a message at top of the page (*invalid user name OR invalid password*); otherwise, redirect to the default page; in addition, if the user has checked “Remember me” then send a permanent cookie (*UserID=<UserID>*). *This task requires DB lookup.*

The user’s login information is stored into database (use a DB table named “tblUsers”) when registration form is submitted. It consists of three fields: *UserID*, *UserName*, *Passwd*, where UserID (set as Primary Key) is a string of 12 random digits.

Note: You can test the login page by entering dummy data into your DB directly.



The screenshot shows a web browser window titled "User Login" with the address bar displaying "localhost:46824/Login.html". The page content is as follows:

User Login

If you are not registered, [Register](#).
(Note: Replace this text with error message from login.)

User Name

Password

Remember me

Recommended Readings on Razor

1. <http://www.asp.net/web-pages/tutorials/basics/2-introduction-to-asp-net-web-programming-using-the-razor-syntax>
2. <http://www.asp.net/web-pages/overview/data/5-working-with-data>
3. Do Google search for “*tutorial razor filetype: pdf*”
The first hit (*download.microsoft.com*) is a link to a good reference on WebMatix and Razor.

Login.html

```
<!DOCTYPE html>
<html>
<head><title>User Login</title></head>
<body style="width:500px;font-size: 1.1em">
<h2>User Login</h2>
<form method="GET" action="Login.cshtml" >
<input type="hidden" name="hvar" value="hval" >
<hr>
<p>If you are not registered, <a href="Register.cshtml" >Register</a>.
  <br>(<em>Note:Replace this text with error message from login.</em>)
</p>
<hr>
<table>
<tr><td>User Name</td><td><input type="text" name="UserName" size="24" >
</td></tr>
<tr><td>Password</td><td><input type="text" name="Passwd" size="24" >
</td></tr>
<tr><td><input type="Submit" value="Login" style="padding:4px;font-
weight:bold" ></td>
  <td><input type="CheckBox" name="ChkRemember" >Remember me</td></tr>
</table>
</form>
</body>
</html>
```

```
// --- Use the following function to generate a User ID
string genUserID()
{
  Random rand = new Random();
  string str = "";
  for (int i = 1; i <= 12; i++)
  { str = str + rand.Next(10); }
  return str;
}
```