

Practical 7: Login form using session management

Login.aspx:

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Login.aspx.cs"
Inherits="WebFormsNoDB.Login" %>
<!DOCTYPE html>
<html>
<head runat="server">
<title>Login</title>
<style>
body {
font-family: Arial, sans-serif;
background-color: #f2f2f2;
text-align: center;
padding: 50px;
}
form {
background: #fff;
padding: 20px;
border-radius: 8px;
display: inline-block;
box-shadow: 0px 0px 8px rgba(0,0,0,0.1);
}
h2 {
color: #333;
}
asp\:TextBox {
margin: 5px 0;
padding: 5px;
}
asp\:Button {
background-color: #4CAF50;
color: white;
padding: 8px 15px;
border: none;
border-radius: 5px;
cursor: pointer;
}
asp\:Button:hover {
background-color: #45a049;
}
a {
color: #0066cc;
text-decoration: none;
}
a:hover {
text-decoration: underline;
}
</style>
</head>
<body>
<form id="form1" runat="server">
<h2>Login</h2>

<asp:Label Text="Username:" runat="server" />
<asp:TextBox ID="txtUsername" runat="server" /><br />
<asp:Label Text="Password:" runat="server" />
<asp:TextBox ID="txtPassword" runat="server" TextMode="Password" /><br />
<asp:Button ID="btnLogin" runat="server" Text="Login" OnClick="btnLogin_Click"
/>
<br /><br />
```

```

<asp:Label ID="lblMessage" runat="server" ForeColor="Red" />
<!-- Register link -->
<asp:HyperLink ID="hlRegister" runat="server" NavigateUrl="Register.aspx">
Don't have an account? Register here
</asp:HyperLink>
</form>
</body>
</html>

```

Login.aspx.cs:

```

using System;
using System.Collections.Generic;
namespace WebFormsNoDB
{
    public partial class Login : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            if (Request.QueryString["session"] == "expired")
            {
                lblMessage.Text = "Your session has expired. Please log in
again.";
            }
        }
        protected void btnLogin_Click(object sender, EventArgs e)
        {
            if (Application["Users"] != null)
            {
                var users = (Dictionary<string, Tuple<string,
string>>)Application["Users"];
                if (users.ContainsKey(txtUsername.Text) &&
users[txtUsername.Text].Item1 ==
txtPassword.Text)
                {
                    Session["Username"] = txtUsername.Text;
                    Session.Timeout = 10; // Explicitly set 10 minutes here as
well
                    Response.Redirect("Welcome.aspx");
                }
                else
                {
                    lblMessage.Text = "Invalid username or password!";
                }
            }
            else
            {
                lblMessage.Text = "No users registered yet!";
            }
        }
    }
}

```

Register.aspx:

```

<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Register.aspx.cs"
Inherits="WebFormsNoDB.Register" %>
<!DOCTYPE html>
<html>
<head runat="server">
<title>Register</title>

```

```

<style>
body {
font-family: Arial, sans-serif;
background-color: #f2f2f2;
text-align: center;
padding: 50px;
}
form {
background: #fff;
padding: 20px;
border-radius: 8px;
display: inline-block;
box-shadow: 0px 0px 8px rgba(0,0,0,0.1);
}
h2 {
color: #333;
}
asp\:TextBox {
margin: 5px 0;
padding: 5px;
}
asp\:Button {
background-color: #007bff;
color: white;
padding: 8px 15px;
border: none;
border-radius: 5px;
cursor: pointer;
}
asp\:Button:hover {
background-color: #0056b3;
}
a {
color: #0066cc;
text-decoration: none;
}
a:hover {
text-decoration: underline;
}
</style>
</head>
<body>
<form id="form1" runat="server">
<h2>Registration</h2>

<asp:Label Text="Username:" runat="server" />
<asp:TextBox ID="txtUsername" runat="server" /><br />
<asp:Label Text="Password:" runat="server" />
<asp:TextBox ID="txtPassword" runat="server" TextMode="Password" /><br />
<asp:Label Text="Email:" runat="server" />
<asp:TextBox ID="txtEmail" runat="server" /><br />
<asp:Button ID="btnRegister" runat="server" Text="Register"
OnClick="btnRegister_Click" />
<br /><br />
<asp:Label ID="lblMessage" runat="server" ForeColor="Red" /><br /><br />
<asp:HyperLink ID="hlLogin" runat="server" NavigateUrl="Login.aspx">
Already have an account? Login here
</asp:HyperLink>
</form>
</body>
</html>

```

Register.aspx.cs:

```
using System;
using System.Collections.Generic;
namespace WebFormsNoDB
{
    public partial class Register : System.Web.UI.Page
    {
        protected void btnRegister_Click(object sender, EventArgs e)
        {
            if (Application["Users"] == null)
            {
                Application["Users"] = new Dictionary<string, Tuple<string,
string>>();
            }
            var users = (Dictionary<string, Tuple<string,
string>>)Application["Users"];
            if (users.ContainsKey(txtUsername.Text))
            {
                lblMessage.Text = "Username already exists!";
            }
            else
            {
                users[txtUsername.Text] = Tuple.Create(txtPassword.Text,
txtEmail.Text);
                lblMessage.Text = "Registration successful! You can now login.";
                lblMessage.ForeColor = System.Drawing.Color.Green;
            }
        }
    }
}
```

Welcome.aspx:

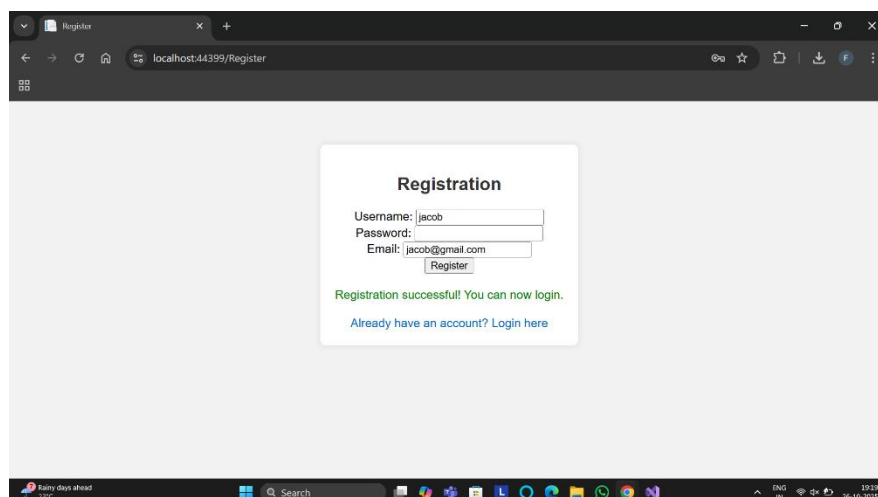
```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Welcome.aspx.cs"
Inherits="WebFormsNoDB.Welcome" %>
<!DOCTYPE html>
<html>
<head runat="server">
    <title>Welcome</title>
</head>
<body>
    <form id="form1" runat="server">
        <h2>Welcome, <asp:Label ID="lblUser" runat="server" /></h2>
        <asp:Button ID="btnLogout" runat="server" Text="Logout"
OnClick="btnLogout_Click" /><br />
        <p>ASP.NET is a web application framework designed and developed by
Microsoft.<br />
ASP.NET is open source and a subset of the .NET Framework and successor of the
<br />
classic ASP(Active Server Pages). With version 1.0 of the .NET Framework,<br />
it was first released in January 2002. So a question comes to mind that <br />
which technology we were using before the year 2002 for developing web
applications <br />
and services? Answer is Classic ASP. So before .NET and ASP.NET there was
Classic ASP. <br />
ASP.NET is built on the CLR(Common Language Runtime) which allows the
programmers to execute <br />
its code using any .NET language(C#, VB etc.). It is specially designed to work
with HTTP and <br />
```

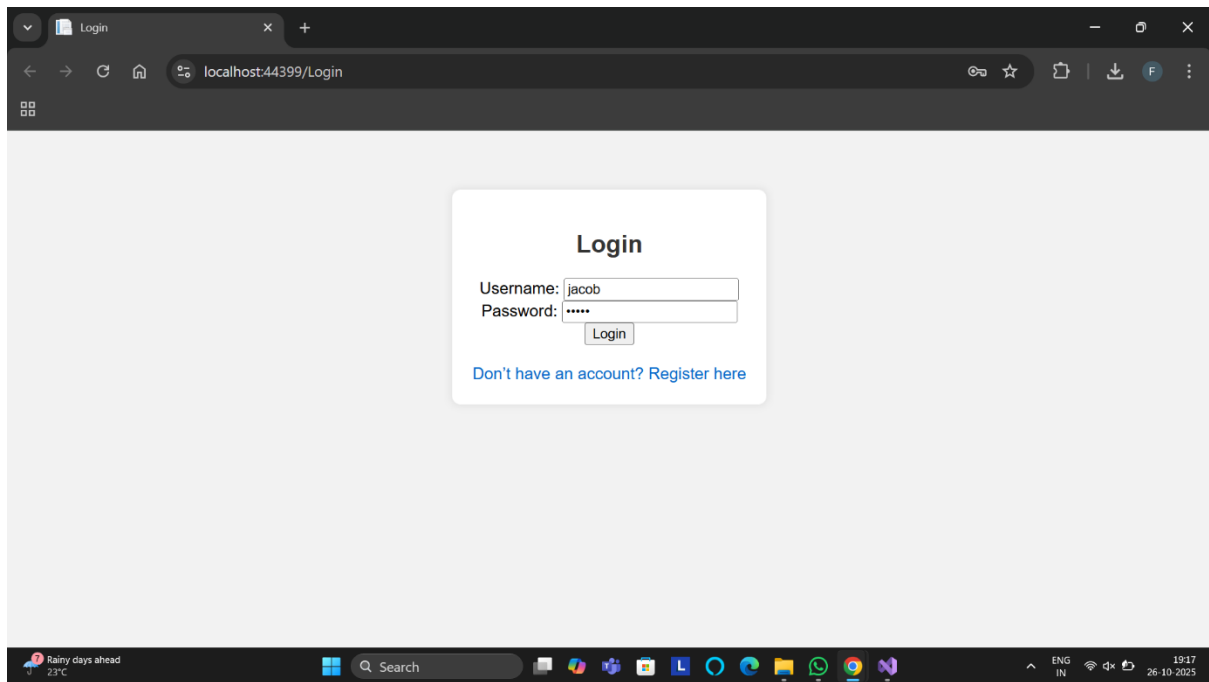
for web developers to create dynamic web pages, web applications, web sites, and web services as `
`
 it provides a good integration of HTML, CSS, and JavaScript. .NET Framework is used to create a `
`
 variety of applications and services like Console, Web, and Windows, etc. But ASP.NET is only `
`
 used to create web applications and web services. That's why we termed ASP.NET as a subset of the . `
`
 NET Framework.`</p>`
`</form>`
`</body>`
`</html>`

Welcome.aspx.cs:

```
using System;
namespace WebFormsNoDB
{
    public partial class Welcome : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            if (Session["Username"] != null)
            {
                lblUser.Text = Session["Username"].ToString();
            }
            else
            {
                // Session expired or user not logged in
                Response.Redirect("Login.aspx?session=expired");
            }
        }
        protected void btnLogout_Click(object sender, EventArgs e)
        {
            Session.Abandon();
            Response.Redirect("Login.aspx");
        }
    }
}
```

Output:





Welcome, jacob

[Logout](#)

ASP.NET is a web application framework designed and developed by Microsoft. ASP.NET is open source and a subset of the .NET Framework and successor of the classic ASP(Active Server Pages). With version 1.0 of the .NET Framework, it was first released in January 2002. So a question comes to mind that which technology we were using before the year 2002 for developing web applications and services? Answer is Classic ASP. So before .NET and ASP.NET there was Classic ASP. ASP.NET is built on the CLR(Common Language Runtime) which allows the programmers to execute its code using any .NET language(C#, VB etc.). It is specially designed to work with HTTP and for web developers to create dynamic web pages, web applications, web sites, and web services as it provides a good integration of HTML, CSS, and JavaScript. .NET Framework is used to create a variety of applications and services like Console, Web, and Windows, etc. But ASP.NET is only used to create web applications and web services. That's why we termed ASP.NET as a subset of the .NET Framework.

