**Aim:** Design Web Applications using Client-Side Session Management and Server-Side Session Management Techniques

### Code:

1. Create a login page using client-side cookies and maintain the session. Consider a dummy username and password use asp.net

## Login.aspx

<%@ flage Language="C " AutoEventWireup="true" CodeBehind="login.aspx.cs" Inherits="practical\_no\_7\_1.login" %>

<!DOCTYflE html>

<html xmlns="<http://www.w3.org/1999/xhtml>">

<head runat="server">

<title>Login flage</title>

</head>

<body>

<form id="form1" runat="server">

<div>

<h2>Login</h2>

<asp:TextBox ID="txtUsername" runat="server"></asp:TextBox><br />

<asp:TextBox ID="txtflassword" runat="server" TextMode="flassword"></asp:TextBox><br />

<asp:Button ID="btnLogin" runat="server" Text="Login" OnClick="btnLogin\_Click" />

</div>

</form>

</body>

</html>

## Login.aspx.cs

using System;

using System.Collections.Generic; using System.Linq;

using System.Web; using System.Web.UI;

using System.Web.UI.WebControls;

namespace practical\_no\_7\_1

{

public partial class login : System.Web.UI.flage

{

protected void flage\_Load(object sender, EventArgs e)

{

}

protected void btnLogin\_Click(object sender, EventArgs e)

{

string dummyUsername = "Sumeet"; string dummyflassword = "Sumeet123"; string username = txtUsername.Text; string password = txtflassword.Text;

if (username == dummyUsername fifi password == dummyflassword)

{

Response.Cookies["LoggedIn"].Value = "true"; Response.Cookies["username"].Value = txtUsername.Text; Response.Cookies["LoggedIn"].Expires = DateTime. ow.AddHours(1); Response.Redirect("dashboard.aspx");

}

else

{

Response.Write("<script>alert('Invalid username or password');</script>");

}

}

}

}

Dashboard.aspx

<%@ flage Language="C " AutoEventWireup="true" CodeBehind="dashboard.aspx.cs" Inherits="practical\_no\_7\_1.dashboard" %>

<!DOCTYflE html>

<html xmlns="<http://www.w3.org/1999/xhtml>">

<head runat="server">

<title>Dashboard</title>

</head>

<body>

<form id="form1" runat="server">

<div>

<h2>Welcome to Dashboard</h2>

<div>

<asp:Label ID="lblWelcomeMessage" runat="server" Text=""></asp:Label>

</div>

</div>

</form>

</body>

</html>

Dashboard.aspx.cs

using System;

using System.Collections.Generic; using System.Linq;

using System.Web; using System.Web.UI;

using System.Web.UI.WebControls;

namespace practical\_no\_7\_1

{

public partial class dashboard : System.Web.UI.flage

{

protected void flage\_Load(object sender, EventArgs e)

{

if (Request.Cookies["LoggedIn"] != null fifi Request.Cookies["LoggedIn"].Value == "true")

{

string username = Request.Cookies["username"].Value;

}

else

{

}

}

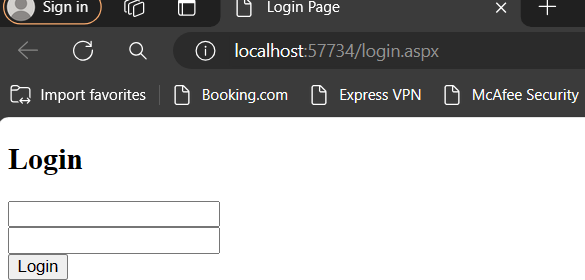
}

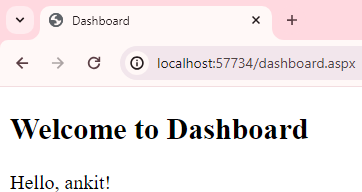
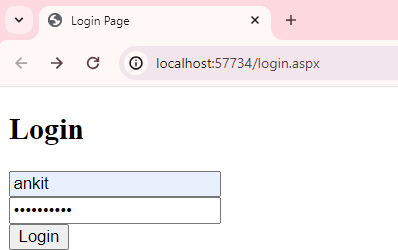
}

lblWelcomeMessage.Text = "Hello, " + username + "!";

Response.Redirect("login.aspx");

### Output:





1. Develop a web application for an online bookstore. The application needs to display a list of books from a database on the homepage (index.aspx). Users should be able to click on a book title to view its details on a separate page (bookdetails.aspx). Implement the necessary server- side code to fetch book data from the database and display it on the homepage. Also, create the book details page to display information about a selected book. using server side session management display the data retrieved from database

# Code:

## Index.aspx

<%@ flage Language="C " AutoEventWireup="true" CodeBehind="index.aspx.cs" Inherits="practical\_7\_2.index" %>

<!DOCTYflE html>

<html xmlns="<http://www.w3.org/1999/xhtml>">

<head runat="server">

<title>Bookstore - Home</title>

</head>

<body>

<form id="form1" runat="server">

<h1>Welcome to Our Online Bookstore</h1>

<div>

<asp:GridView ID="GridView1" runat="server" AutoGenerateColumns="False">

<Columns>

<asp:BoundField DataField="BookID" HeaderText="Book ID"

/>

<asp:BoundField DataField="Title" HeaderText="Title" />

<asp:BoundField DataField="Author" HeaderText="Author" />

<asp:BoundField DataField="Genre" HeaderText="Genre" />

<asp:BoundField DataField="flrice" HeaderText="flrice" />

<asp:TemplateField HeaderText="Details">

<ItemTemplate>

<asp:HyperLink ID="lnkDetails" runat="server"

CssClass="btn btn-primary" Text="View Details" avigateUrl='<% "bookdetails.aspxoBookID=" + Eval("BookID") %>'></asp:HyperLink>

</ItemTemplate>

</asp:TemplateField>

</Columns>

</asp:GridView>

</div>

</form>

</body>

</html>

Index.aspx.cs using System;

using System.Collections.Generic; using MySql.Data.MySqlClient; using System.Linq;

using System.Web; using System.Web.UI;

using System.Web.UI.WebControls; using System.Data;

namespace practical\_7\_2

{

public partial class index : System.Web.UI.flage

{

protected void flage\_Load(object sender, EventArgs e)

{

if (!IsflostBack fifi GridView1.DataSource == null)

{

// Fetch book data from the database string connectionString =

"Server=localhost;Database=bank;Uid=root;flwd=1234;";

using (MySqlConnection connection = new MySqlConnection(connectionString))

{

FROM Books";

string query = "SELECT BookID, Title, Author, Genre,flrice

MySqlCommand command = new MySqlCommand(query, connection); connection.Open();

DataTable dataTable = new DataTable(); using (MySqlDataAdapter adapter = new

MySqlDataAdapter(command))

{

adapter.Fill(dataTable);

}

GridView1.DataSource = dataTable; GridView1.DataBind();

// Store data in session Session["BookData"] = dataTable;

}

}

}

protected void GridViewBooks\_SelectedIndexChanged(object sender, EventArgs e)

{

}

}

}

Bookdetail.aspx

<%@ flage Language="C " AutoEventWireup="true" CodeBehind="bookdetails.aspx.cs" Inherits="practical\_7\_2.bookdetails" %>

<!DOCTYflE html>

<html xmlns="<http://www.w3.org/1999/xhtml>">

<head runat="server">

<title>Bookstore - Book Details</title>

</head>

<body>

<form id="form1" runat="server">

<h1>Book Details</h1>

<div>

<div class="card-body">

<p class="card-text"><strong>Book Title</strong>:<asp:Label ID="lblTitle" runat="server" Text=""></asp:Label>

<p class="card-text"><strong>Author:</strong> <asp:Label ID="lblAuthor" runat="server" Text=""></asp:Label></p>

<p class="card-text"><strong>Genre:</strong> <asp:Label ID="lblGenre" runat="server" Text=""></asp:Label></p>

<p class="card-text"><strong>flrice: </strong> <asp:Label ID="lblprice" runat="server"></asp:Label></p>

</div>

</div>

</form>

</body>

</html>

Bookdetail.aspx.cs

using System;

using System.Collections.Generic; using System.Data;

using System.Linq; using System.Web; using System.Web.UI;

using System.Web.UI.WebControls; using MySql.Data.MySqlClient;

namespace practical\_7\_2

{

public partial class bookdetails : System.Web.UI.flage

{

protected void flage\_Load(object sender, EventArgs e)

{

if (!IsflostBack)

{

// Retrieve book ID from query string

int bookID = Convert.ToInt32(Request.QueryString["BookID"]);

// Retrieve book data from session

DataTable dataTable = (DataTable)Session["BookData"]; if (dataTable != null)

{

foreach (DataRow row in dataTable.Rows)

{

if (Convert.ToInt32(row["BookID"]) == bookID)

{

lblTitle.Text = row["Title"].ToString(); lblAuthor.Text = row["Author"].ToString(); lblGenre.Text = row["Genre"].ToString(); lblprice.Text= row["flrice"].ToString(); break;

}

}

}

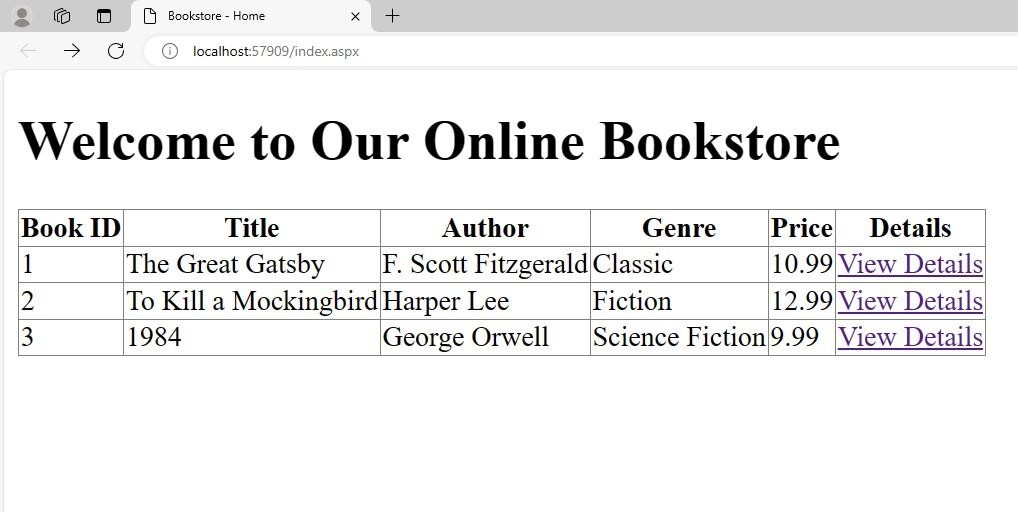
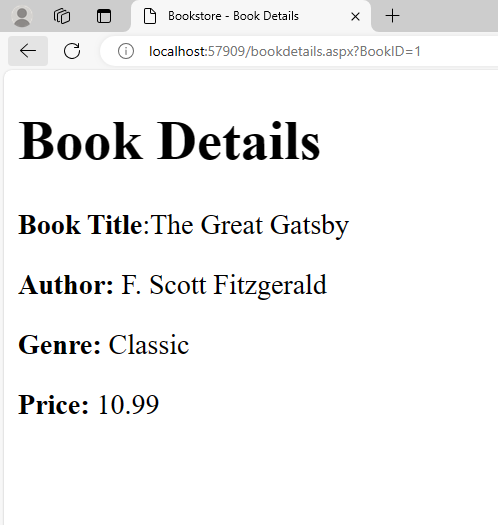
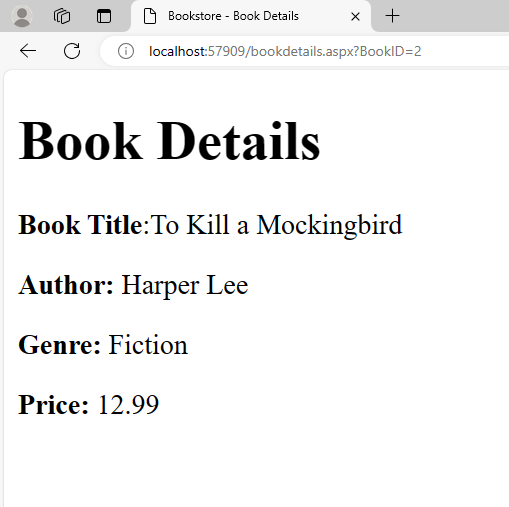
}

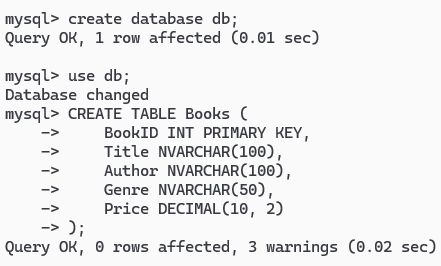
}

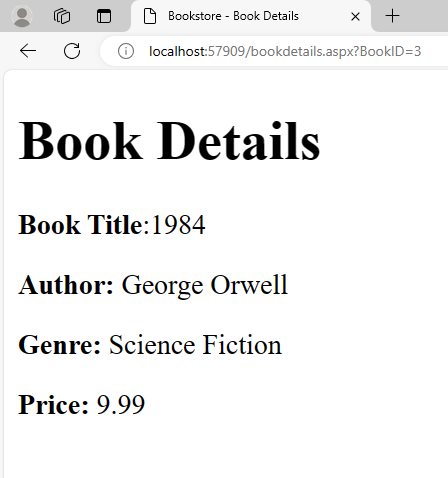
}

}

# Output:







1. You are tasked with implementing a user registration form for a social networking website. The registration form (register.aspx) should collect information such as username, email, password, and date of birth from the user. Implement server-side validation to ensure that all required fields are filled, the email address is in the correct format, and the password meets the complexity requirements. Upon successful registration, store user information in the database and redirect them to a welcome page (welcome.aspx).(Self Study)

Code:

Register.aspx

<!DOCTYflE html>

<html>

<head>

<title>User Registration</title>

</head>

<body>

<h2>User Registration Form</h2>

<form action="register.aspx" method="post">

<label for="username">Username:</label>

<input type="text" id="username" name="username" required><br><br>

<label for="email">Email:</label>

<input type="email" id="email" name="email" required><br><br>

<label for="password">flassword:</label>

<input type="password" id="password" name="password" required><br><b

<label for="dob">Date of Birth:</label>

<input type="date" id="dob" name="dob" required><br><br>

<input type="submit" value="Register">

</form>

</body>

</html> Register.aspx.cs

using System;

using System.Data.SqlClient;

public partial class register : System.Web.UI.flage

{

protected void flage\_Load(object sender, EventArgs e)

{

if (Request.HttpMethod == "flOST")

{

string username = Request.Form["username"]; string email = Request.Form["email"]; string password = Request.Form["password"]; string dob = Request.Form["dob"];

if (string.Is ullOrEmpty(username) || string.Is ullOrEmpty(email)

|| string.Is ullOrEmpty(password) || string.Is ullOrEmpty(dob))

{

Response.Write("<script>alert('fllease fill in all required fields.');</script>");

return;

}

if (!IsValidEmail(email))

{

Response.Write("<script>alert('fllease enter a valid email address.');</script>");

return;

}

if (!IsStrongflassword(password))

{

Response.Write("<script>alert('flassword must be at least 8 characters long and contain at least one uppercase letter, one lowercase letter, one number, and one special character.');</script>");

return;

}

try

{

SqlConnection conn = new SqlConnection("YourConnectionString");

SqlCommand cmd = new SqlCommand("I SERT I TO Users (Username, Email, flassword, DOB) VALUES (@Username, @Email, @flassword, @DOB)", conn);

cmd.flarameters.AddWithValue("@Username", username); cmd.flarameters.AddWithValue("@Email", email); cmd.flarameters.AddWithValue("@flassword", password);

cmd.flarameters.AddWithValue("@DOB", dob); conn.Open();

cmd.Execute onQuery(); conn.Close(); Response.Redirect("welcome.aspx");

}

catch (Exception ex)

{

Response.Write("<script>alert('An error occurred during registration. fllease try again later.');</script>");

}

}

}

private bool IsValidEmail(string email)

{

try

{

var addr = new System. et.Mail.MailAddress(email); return addr.Address == email;

}

catch

{

return false;

}

}

private bool IsStrongflassword(string password)

{

return password.Length >= 8 fifi password.Any(char.IsUpper) fifi password.Any(char.IsLower) fifi password.Any(char.IsDigit) fifi password.Any(char.IsSymbol);

}

}

Output:

