Add to cart application

Default.aspx

<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="Default.aspx.cs" Inherits="WebApplication6.Default" %>

<!DOCTYPE html>

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

<head runat="server">

<title></title>

</head>

<body>

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

<div>

<asp:GridView ID="ProductGridView" runat="server" AutoGenerateColumns="False" DataKeyNames="ProductID">

<Columns>

<asp:BoundField DataField="ProductID" HeaderText="Product ID" />

<asp:BoundField DataField="ProductName" HeaderText="Product Name" />

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

<asp:TemplateField HeaderText="Action">

<ItemTemplate>

<asp:Button ID="AddToCartButton" runat="server" Text="Add to Cart" OnClick="AddToCartButton\_Click" />

</ItemTemplate>

</asp:TemplateField>

</Columns>

</asp:GridView>

<h2>Shopping Cart</h2>

<asp:ListView ID="CartListView" runat="server">

<ItemTemplate>

<p><%# Eval("ProductName") %>, Price: $<%# Eval("Price") %>

<asp:Button ID="RemoveFromCartButton" runat="server" Text="Remove" CommandName="RemoveFromCart" CommandArgument='<%# Eval("ProductID") %>' OnClick="RemoveFromCartButton\_Click" />

</p>

</ItemTemplate>

</asp:ListView>

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

</div>

</form>

</body>

</html>

Default.aspx.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace WebApplication6

{

public partial class Default : System.Web.UI.Page

{

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

{

if (!IsPostBack)

{

ProductGridView.DataSource = ProductRepository.GetAllProducts();

ProductGridView.DataBind();

}

}

private ShoppingCart Cart

{

get

{

if (Session["Cart"] == null)

{

Session["Cart"] = new ShoppingCart();

}

return (ShoppingCart)Session["Cart"];

}

}

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

{

// Implement code to add items to the cart when the "Add to Cart" button is clicked.

Button button = (Button)sender;

GridViewRow row = (GridViewRow)button.NamingContainer;

// Get the product ID from the GridView row

int productId = Convert.ToInt32(ProductGridView.DataKeys[row.RowIndex]["ProductID"]);

// Check if the product is already in the cart (you can modify this logic based on your needs)

bool isProductInCart = Cart.ProductsInCart.Any(p => p.ProductID == productId);

if (isProductInCart)

{

// Handle the case where the product is already in the cart (e.g., display a message)

// You can also update the quantity or implement different behavior as needed.

// For simplicity, I'll just show a message here.

LabelMessage.Text = "This product is already in the cart.";

}

else

{

// Fetch the product details from the data source (ProductRepository)

Product product = ProductRepository.GetAllProducts().Find(p => p.ProductID == productId);

// Add the selected product to the shopping cart

Cart.ProductsInCart.Add(product);

// Update the CartListView with the updated cart data

CartListView.DataSource = Cart.ProductsInCart;

CartListView.DataBind();

}

}

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

{

// Implement code to remove items from the cart when the "Remove" button is clicked.

Button button = (Button)sender;

ListViewItem item = (ListViewItem)button.NamingContainer;

int productId = Convert.ToInt32(CartListView.DataKeys[item.DataItemIndex].Value);

ShoppingCart cart = Cart;

Product productToRemove = cart.ProductsInCart.Find(p => p.ProductID == productId);

if (productToRemove != null)

{

cart.ProductsInCart.Remove(productToRemove);

}

CartListView.DataSource = cart.ProductsInCart;

CartListView.DataBind();

}

}

}

ProductRepository.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

namespace WebApplication6

{

public class ProductRepository

{

public static List<Product> GetAllProducts()

{

return new List<Product>

{

new Product { ProductID = 1, ProductName = "Product 1", Price = 10.99M },

new Product { ProductID = 2, ProductName = "Product 2", Price = 19.99M },

new Product { ProductID = 3, ProductName = "Product 3", Price = 5.99M }

};

}

}

public class Product

{

public int ProductID { get; set; }

public string ProductName { get; set; }

public decimal Price { get; set; }

}

}

ShoppingCart.cs

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

namespace WebApplication6

{

public class ShoppingCart

{

public List<Product> ProductsInCart { get; set; }

public ShoppingCart()

{

ProductsInCart = new List<Product>();

}

}

}