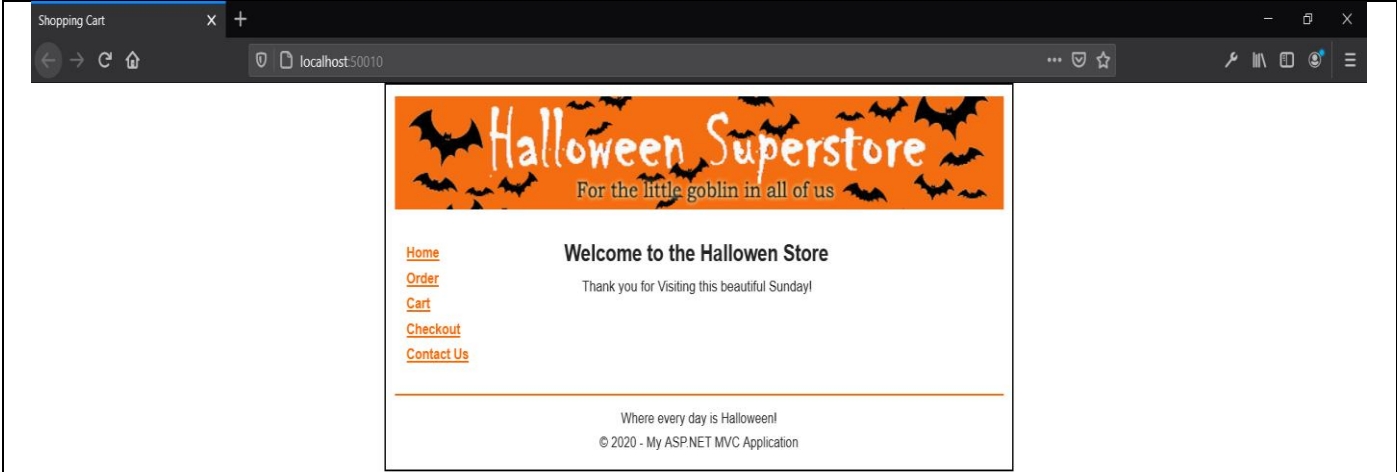


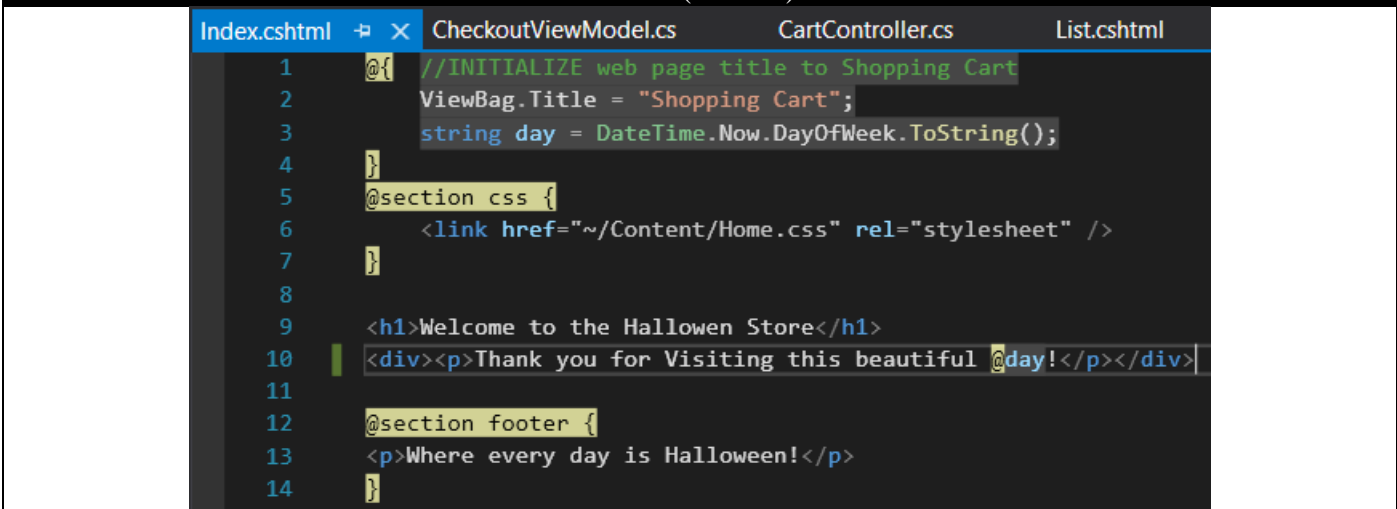
Browser Image (HOME)



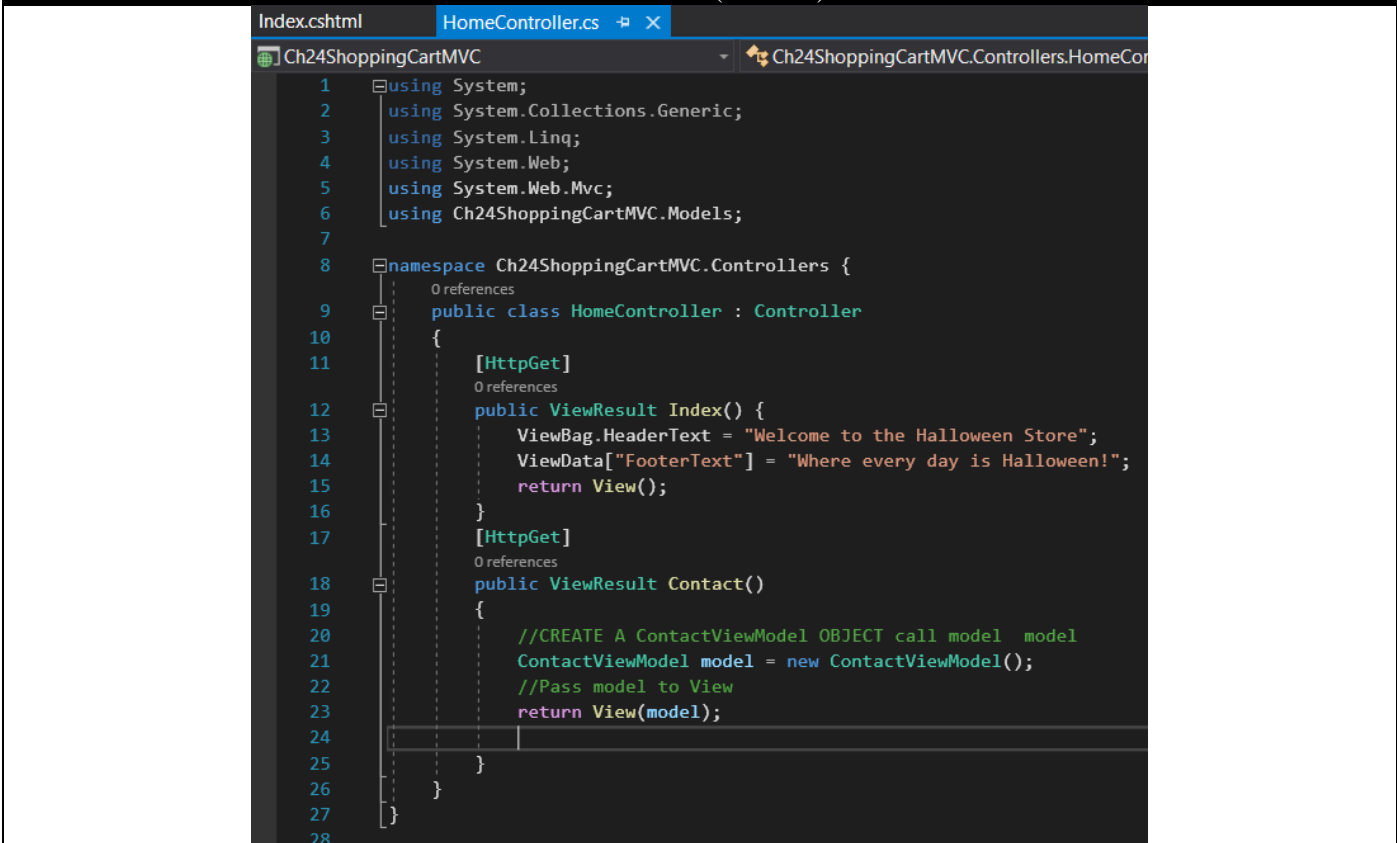
Model (HOME)

No Model for Home

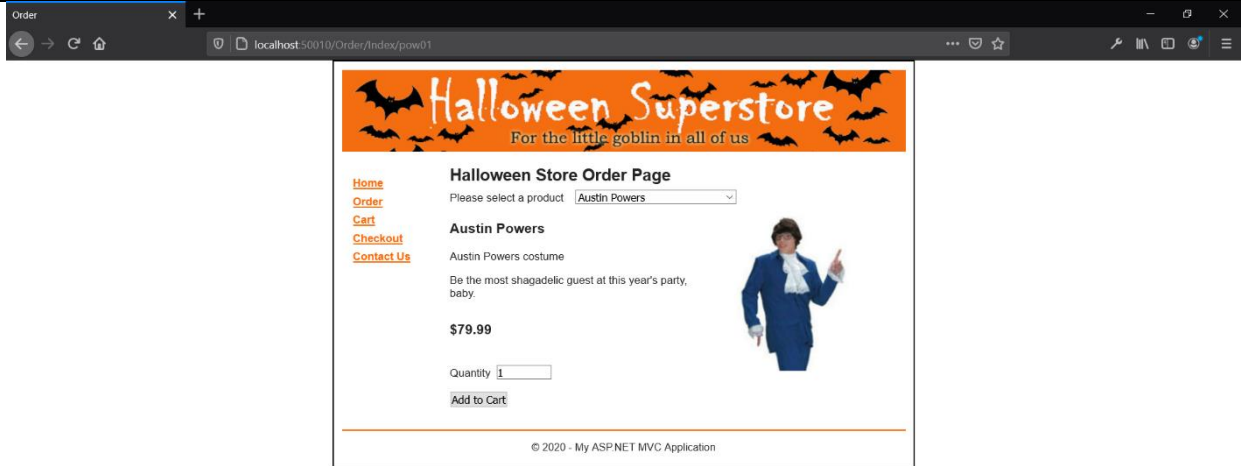
View (HOME)



Controller (HOME)



Browser Image (ORDER)



Model (ORDER)

```
OrderModel.cs | OrderViewModel.cs | Contact.cshtml | ContactViewModel.cs
Ch24ShoppingCartMVC
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Web;
5 using System.Web.Mvc;
6 using Ch24ShoppingCartMVC.Models;
7
8 namespace Ch24ShoppingCartMVC.Models {
9     3 references
10     public class OrderViewModel
11     {
12         1 reference
13         public SelectList ProductsList { get; set; }
14         4 references
15         public ProductViewModel SelectedProduct { get; set; }
16     }
17 }
```

```

OrderModel.cs  OrderViewModel.cs  Contact.cshtml  ContactViewModel.cs  Index.cshtml  HomeController.cs
Ch24ShoppingCartMVC
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Web;
5 using System.Web.Mvc;
6 using Ch24ShoppingCartMVC.Models;
7 using Ch24ShoppingCartMVC.Models.DataAccess;
8
9 namespace Ch24ShoppingCartMVC.Models {
10     public class OrderModel
11     {
12         private List<ProductViewModel> products;
13         //Implement GetAllProductsFromDataStore
14         public List<Product> GetAllProductsFromDataStore()
15         {
16             using (HalloweenEntities data = new HalloweenEntities())
17             { //get all the products from the Collection Products order by name using HalloweenEntities
18                 return data.Products.OrderBy(p => p.Name).ToList();
19             }
20         }
21         //Implement the method ConvertToViewModel
22         private ProductViewModel ConvertToViewModel(Product product)
23         {
24             ProductViewModel model = new ProductViewModel();
25             model.ProductID = product.ProductID;
26             model.Name = product.Name;
27             model.ShortDescription = product.ShortDescription;
28             model.LongDescription = product.LongDescription;
29

```

```

OrderModel.cs  OrderViewModel.cs  Contact.cshtml  ContactViewModel.cs  Index.cshtml  HomeController.cs
Ch24ShoppingCartMVC
30         model.UnitPrice = product.UnitPrice;
31         model.ImageFile = product.ImageFile;
32         model.Quantity = product.OnHand;
33         //implement other required properties
34
35         return model;
36     }
37     //Implement the method GetProductList
38     public List<ProductViewModel> GetProductsList() {
39         if (this.products == null)
40             //Call the method GetAllProducts
41             {
42                 products = GetAllProducts();
43             }
44         //Return the products
45         return products;
46     }
47     //Implement the method GetAllProducts()
48     public List<ProductViewModel> GetAllProducts()
49     {
50         List<ProductViewModel> productmodels = new List<ProductViewModel>();
51         // Call the GetAllProductsFromDataStore()
52         var products = GetAllProductsFromDataStore();
53         foreach (Product p in products)
54         { //Call the method ConvertToViewModel to convert p and pass the method ConvertToViewModel to the method add o
55             productmodels.Add(ConvertToViewModel(p));
56         }
57         return productmodels;
58     }

```

```

OrderModel.cs  OrderViewModel.cs  Contact.cshtml  ContactViewModel.cs  Index.cshtml  HomeController.cs
Ch24ShoppingCartMVC
59     public Product GetProductByIdFromDataStore(string id)
60     {
61         using (HalloweenEntities data = new HalloweenEntities())
62         { //Get a product from Products of data where ProductID is matched with id parameter
63             return data.Products.Where(p=>p.ProductID.Equals(id)).FirstOrDefault();
64         }
65     }
66     //Implement the method GetOrderInfo
67     public OrderViewModel GetOrderInfo(string id)
68     {
69         OrderViewModel order = new OrderViewModel();
70         //Call the method GetSelectedProduct and assign the return value to SelectedProduct property
71         order.SelectedProduct = GetSelectedProduct(id);
72         return order;
73     }
74     //Implement the method GetSelectedProduct
75     public ProductViewModel GetSelectedProduct(string id)
76     {
77         if (this.products == null)
78             //call the method ConvertToViewModel and pass the method GetProductByIdFromDataStore(id)
79             return ConvertToViewModel(GetProductByIdFromDataStore(id));
80         else
81             //Get the product from the products where ProductID is matched with id (Using Lambda expression)
82             return products.Where(p => p.ProductID.Equals(id)).FirstOrDefault();
83     }

```

View (ORDER)

```

Index.cshtml  OrderModel.cs  OrderViewModel.cs  Contact.cshtml  ContactViewModel.cs
1  @model Ch24ShoppingCartMVC.Models.OrderViewModel
2  @{}
3      ViewBag.Title = "Order";
4      var product = Model.SelectedProduct;
5  }
6  @section css {
7      <link href="~/Content/Order.css" rel="stylesheet" />
8  }
9  <form id="frmDropDown" action="/Order/Index/" method="post">
10     <h2>Halloween Store Order Page</h2>
11
12     <label>Please select a product&nbsp;</label>
13     <select id="ddlProducts" name="ddlProducts" class="autoPostBack">
14         @foreach (var item in Model.ProductsList) {
15             <option value="@item.Value" selected="@item.Selected">@item.Text</option>
16         }
17     </select>
18 </form>
19 @using (Html.BeginForm("List", "Cart", FormMethod.Post, new { id = "frmCart" })) {
20     <div id="productData">
21         <label id="lblName">@product.Name</label>
22         @Html.Label("", @product.ShortDescription, new { id = "lblShortDescription" })
23         <label id="lblLongDescription">@product.LongDescription</label>
24         <label id="lblUnitPrice">@product.UnitPrice.ToString("c2")</label>
25         <label id="lblQuantity">Quantity&nbsp;</label>
26         @Html.TextBoxFor(m => m.SelectedProduct.Quantity, new { id = "txtQuantity" })
27         @Html.HiddenFor(m => m.SelectedProduct.ProductID)
28         <br />
29         <input type="submit" id="btnAdd" value="Add to Cart" />
30     </div>
31     
32 }

```

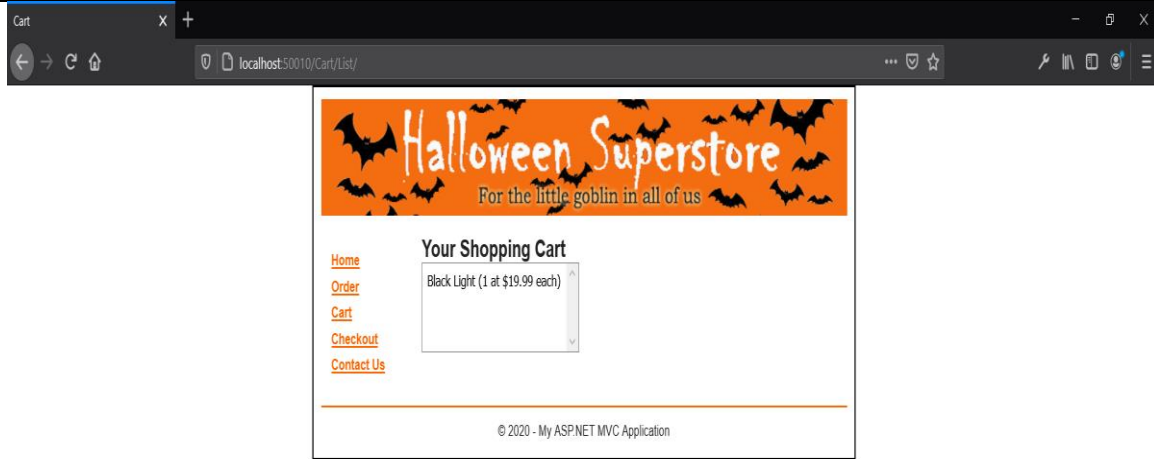
Controller (ORDER)

```

OrderController.cs  Index.cshtml  OrderModel.cs  OrderViewModel.cs  Contact.cs
Ch24ShoppingCartMVC
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5  using System.Web.Mvc;
6  using Ch24ShoppingCartMVC.Models;
7
8  namespace Ch24ShoppingCartMVC.Controllers {
9      public class OrderController : Controller
10      {
11          private OrderModel order = new OrderModel();
12
13          [HttpGet]
14          public ActionResult Index(string id)
15          {
16              //get list for drop down from temp data called products
17              SelectList products = (SelectList)TempData["products"];
18              if (products == null)
19              {
20                  //CALL THE METHOD GetProductList
21                  var list = order.GetProductsList();
22                  //CREATE THE SelectList products
23                  products = new SelectList(list, "ProductID", "Name", id);
24              }
25              //if no URL parameter, get first product from list and refresh
26              if (string.IsNullOrEmpty(id))
27              {
28                  id = products.ElementAt(0).Value;
29                  //ASSIGN products to temp data called products
30                  TempData["products"] = products;
31
32                  //Redirect to the action method Index of the Order controller with id parameter.
33                  return RedirectToAction("Index", new { id = id });
34              }
35              else
36              {
37                  //get selected product and return in view method
38                  //Call the method GetOrderInfo to get an OrderViewModel object called model
39                  var model = order.GetOrderInfo(id);
40                  //Assign products to ProductsList property of model
41                  model.ProductsList = products;
42                  //Assign the quantity of the SelectProduct of the model to 1
43                  model.SelectedProduct.Quantity = 1;
44                  //Send the model object to the view.
45                  return View(model);
46              }
47          }
48          [HttpPost] //post back - get selected ddl value and refresh
49          public RedirectToRouteResult Index(FormCollection collection)
50          {
51              string pID = collection["ddlProducts"];
52              //Redirect to the action method index of the Order controller with parameter the id assigned to pID
53              return RedirectToAction("Index", new { id = pID });
54          }
55      }
56  }

```

Browser Image (CART)



Model (CART)

```
Ch24ShoppingCartMVC
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Web;
5 using Ch24ShoppingCartMVC.Models;
6
7 namespace Ch24ShoppingCartMVC.Models {
8     public class CartViewModel
9     {
10         public List<ProductViewModel> Cart { get; set; }
11         public ProductViewModel AddedProduct { get; set; }
12     }
13 }

Ch24ShoppingCartMVC
1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Web;
5 using Ch24ShoppingCartMVC.Models;
6
7 namespace Ch24ShoppingCartMVC.Models{
8     public class CartModel
9     {
10         //Data Access methods
11         private List<ProductViewModel> GetCartFromDataStore()
12         {
13             List<ProductViewModel> cart;
14             object objCart = HttpContext.Current.Session["cart"];
15             //Convert objCart to List<ProductViewModel>
16             cart = (List<ProductViewModel>)objCart;
17             if (cart == null)
18             {
19                 //Create the object cart
20                 objCart = new List<ProductViewModel>();
21                 //Assign cart to the Session object cart
22                 cart = (List<ProductViewModel>)objCart;
23             }
24             return cart;
25         }
26         private ProductViewModel GetSelectedProduct(string id)
27         {
28             //Create an OrderModel object called order
29             OrderModel order = new OrderModel();
30             //Call the method GetSelectedProduct of the class OrderModel. Return the object returned by the call.
31             return order.GetSelectedProduct(id);
32         }
33     }
34 }
```

```

30     }
31     2 references
32     public CartViewModel GetCart(string id = "")
33     {
34         CartViewModel model = new CartViewModel();
35         //Call the method GetCartFromDataStore
36         model.Cart = GetCartFromDataStore();
37         if (!string.IsNullOrEmpty(id))
38             //Called the method GetSelectedProduct with parameter id and assign the return object to the AddedProduct
39         {
40             model.AddedProduct = GetSelectedProduct(id);
41         }
42         return model;
43     }
44     1 reference
45     private void AddItemToDataStore(CartViewModel model)
46     {
47         //Add the AddedProduct to the cart
48         model.Cart.Add(model.AddedProduct);
49         HttpContext.Current.Session["cart"] = model.Cart;
50     }
51     1 reference
52     public void AddToCart(CartViewModel model)
53     {
54         if (model.AddedProduct.ProductID != null)
55         {
56             //Get the product id of the added product
57             var pID = model.AddedProduct.ProductID;
58             //Find the product in the car that matches the id using lambda expression.
59             var inCart = GetCartFromDataStore().Where(p => p.ProductID.Equals(pID)).FirstOrDefault();
60             if (inCart == null)
61             {
62                 //Call the method AddItemToDataStore
63                 {
64                     AddItemToDataStore(model);
65                 }
66             }
67             else
68             {
69                 //Increase the Quantity by the quantity of the added product
70                 inCart.Quantity += model.AddedProduct.Quantity;
71             }
72         }
73     }

```

View (CART)

```

List.cshtml  CheckoutModel.cs  CartViewModel.cs  OrderContro
1  @model Ch24ShoppingCartMVC.Models.CartViewModel
2  @{
3      ViewBag.Title = "Cart";
4  }
5  @section css {
6      <link href="~/Content/Cart.css" rel="stylesheet" />
7  }
8  <h2>Your Shopping Cart</h2>
9  @if (Model.Cart.Count==0)
10  {
11      <p>Nothing in your cart yet.</p>
12  }
13  else
14  {
15      <select multiple="multiple">
16          @foreach (var item in Model.Cart.ToList())
17          {
18              <option>@item.Display()</option>
19          }
20      </select>
21  }
22

```

Controller (CART)

```
CartController.cs | List.cshtml | CheckoutModel.cs | CartViewModel.cs
Ch24ShoppingCartMVC
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;
using Ch24ShoppingCartMVC.Models;

namespace Ch24ShoppingCartMVC.Controllers {
    public class CartController : Controller
    {
        private CartModel cart = new CartModel();

        [HttpGet]
        public RedirectToRouteResult Index()
        {
            return RedirectToAction("List/");
        }


        [HttpGet]
        public ViewResult List()
        {
            CartViewModel model = (CartViewModel)TempData["cart"];
            //if the model is null, then call the method GetCart
            if (model==null)
            {
                model = cart.GetCart();
            }
            //Passing model to View
            return View(model);
        }
    }
}
```

```
30 [HttpPost]
31 public RedirectToRouteResult List(OrderViewModel order)
32 {
33     CartViewModel model = cart.GetCart(order.SelectedProduct.ProductID);
34     //Assign the quantity of the selected product to the quantity of the added product
35     model.AddedProduct.Quantity = order.SelectedProduct.Quantity;
36     //Call the method AddToCart
37     cart.AddToCart(model);
38     //Assign model to the TempData
39     TempData["cart"] = model;
40     return RedirectToAction("List", "Cart");
41 }
42
43
44 }
```

Browser Image (CHECKOUT)

Checkout

localhost:50010/Checkout/Index/



For the little goblin in all of us

[Home](#)
[Order](#)
[Cart](#)
[Checkout](#)
[Contact Us](#)

Ordered Item	Quantity of Item	Unit Price of Item	Total Price of Item
Black Light	1	19.9900	\$19.9900

Total Price Before Tax: \$19.99
Total Price After Tax: \$21.48925

Street Address:

City:

State:

Country:

Zip Code:

Mastercard Paypal Visa

© 2020 - My ASP.NET MVC Application

Model (CHECKOUT)

```

CheckoutViewModel.cs  CartController.cs  List.cshtml  CheckoutModel.cs
Ch24ShoppingCartMVC  Ch24ShoppingCartMVC.Models.ViewModels

1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5
6  namespace Ch24ShoppingCartMVC.Models.ViewModels
7  {
8      3 references
9      public class CheckoutViewModel
10     {
11         2 references
12         public List<ProductViewModel> Cart { get; set; }
13
14         0 references
15         public double TaxRate {
16             get
17             {
18                 return 0.075;
19             }
20         }
21
22         0 references
23         public double Total {
24             get
25             {
26                 double sum = 0;
27                 foreach(var item in Cart)
28                 {
29                     sum += (double)(item.UnitPrice) * (item.Quantity);
30                 }
31                 return sum;
32             }
33         }
34
35         public List<string> PaymentOptions
36         {
37             get
38             {
39                 List<string> options = new List<string>();
40                 options.Add("Visa");
41                 options.Add("MasterCard");
42                 options.Add("PayPal");
43                 return options;
44             }
45         }
46
47         0 references
48         public string StreetAddress { get; set; }
49
50         0 references
51         public string City { get; set; }
52
53         0 references
54         public string State { get; set; }
55
56         0 references
57         public string Country { get; set; }
58
59         0 references
60         public string ZipCode { get; set; }
61     }
62 }

```

```

CheckoutViewModel.cs  CartController.cs  List.cshtml  CheckoutModel.cs
Ch24ShoppingCartMVC  Ch24ShoppingCartMVC.Models.CheckoutMo

7  namespace Ch24ShoppingCartMVC.Models
8  {
9      1 reference
10     public class CheckoutModel
11     {
12         1 reference
13         public List<ProductViewModel> GetCartFromDataStore()
14         {
15             List<ProductViewModel> cart;
16             object objCart = HttpContext.Current.Session["cart"];
17
18             cart = (List<ProductViewModel>)objCart;
19
20             if(cart==null)
21             {
22                 objCart = new List<ProductViewModel>();
23                 cart = (List<ProductViewModel>)objCart;
24             }
25             return cart;
26         }
27
28         1 reference
29         public CheckoutViewModel GetCheckout()
30         {
31             CheckoutViewModel model = new CheckoutViewModel();
32             model.Cart = GetCartFromDataStore();
33             return model;
34         }
35     }
36 }

```


View (CHECKOUT)

```

Index.cshtml  CartModel.cs  CheckoutViewModel.cs  CartController.cs  List.cshtml  CheckoutModel
1  @model Ch24ShoppingCartMVC.Models.ViewModels.CheckoutViewModel
2  @{
3      ViewBag.Title = "Checkout";
4  }
5
6  <h2>Checkout</h2>
7  @if (Model.Cart.Count == 0)
8  {
9      <p>There is nothing to check out</p>
10 }
11 else
12 {
13     <table style="width:100%">
14     <tr>
15         <th>Ordered Item</th>
16         <th>Quantity of Item</th>
17         <th>Unit Price of Item</th>
18         <th>Total Price of Item</th>
19     </tr>
20
21     @foreach (var item in Model.Cart)
22     {
23         <tr>
24             <td style="text-align: center">@item.Name</td>
25             <td style="text-align: center">@item.Quantity</td>
26             <td style="text-align: center">@item.UnitPrice</td>
27             <td style="text-align: center">$(@item.UnitPrice * item.Quantity)</td>
28         </tr>
29     }
30 </table>
31 <p style="text-align: right">Total Price Before Tax: $@Model.Total</p>
32 <p style="text-align: right">Total Price After Tax: $@Model.Total * (1+Model.TaxRate)</p>
33 <br />

```

```

Index.cshtml  CartModel.cs  CheckoutViewModel.cs  CartController.cs  List.cshtml
34 <input type="submit" value="Visa" style="float: right">
35 <input type="submit" value="Paypal" style="float: right">
36 <input type="submit" value="Mastercard" style="float: right">
37 <form>
38     <div id="shippingInfo">
39         <table>
40         <tr>
41             <td><label for="StreetAddress">Street Address </label></td>
42             <td><input type="text" name="StreetAddress" /></td>
43         </tr>
44         <tr>
45             <td><label for="City">City </label></td>
46             <td><input type="text" name="City" /></td>
47         </tr>
48         <tr>
49             <td><label for="State">State </label></td>
50             <td><input type="text" name="State" /></td>
51         </tr>
52         <tr>
53             <td><label for="Country">Country </label></td>
54             <td><input type="text" name="Country" /></td>
55         </tr>
56         <tr>
57             <td><label for="ZipCode">Zip Code </label></td>
58             <td><input type="text" name="ZipCode" /></td>
59         </tr>
60         </table>
61     </div>
62 </form>
63

```

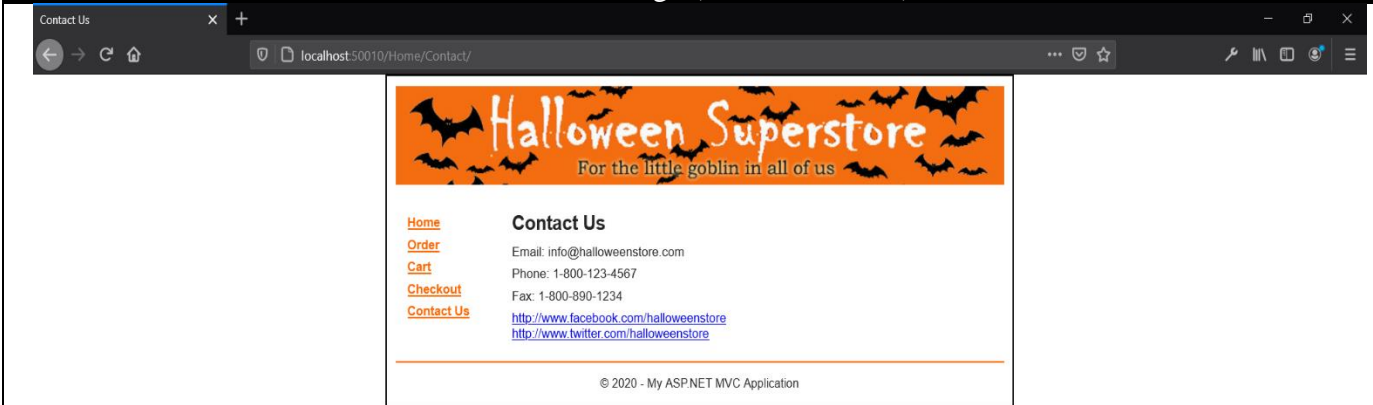
Controller (CHECKOUT)

```

Index.cshtml  HomeController.cs
Ch24ShoppingCartMVC  Ch24ShoppingCartMVC.Controllers.HomeController
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5  using System.Web.Mvc;
6  using Ch24ShoppingCartMVC.Models;
7
8  namespace Ch24ShoppingCartMVC.Controllers {
9      public class HomeController : Controller
10     {
11         [HttpGet]
12         public ActionResult Index() {
13             ViewBag.HeaderText = "Welcome to the Halloween Store";
14             ViewData["FooterText"] = "Where every day is Halloween!";
15             return View();
16         }
17         [HttpGet]
18         public ActionResult Contact()
19         {
20             //CREATE A ContactViewModel OBJECT call model model
21             ContactViewModel model = new ContactViewModel();
22             //Pass model to View
23             return View(model);
24         }
25     }
26 }
27
28

```

Browser Image (CONTACT US)



Model (CONTACT US)

```
ContactViewModel.cs | Index.cshtml | HomeController.cs
Ch24ShoppingCartMVC | Ch24ShoppingCartMVC.Models.ContactViewModel | S

1 using System;
2 using System.Collections.Generic;
3 using System.Linq;
4 using System.Web;
5
6 namespace Ch24ShoppingCartMVC.Models {
7     1 reference
8     public class ContactViewModel
9     {
10         0 references
11         public string Email { get { return "info@halloweenstore.com"; } }
12         0 references
13         public string Phone { get { return "1-800-123-4567"; } }
14         0 references
15         public string Fax { get { return "1-800-890-1234"; } }
16         0 references
17         public string Address { get { return "1234 Main Street, Anytown, USA"; } }
18         0 references
19         public List<string> SocialMediaUrls {
20             get {
21                 List<string> urls = new List<string>();
22                 urls.Add("http://www.facebook.com/halloweenstore");
23                 urls.Add("http://www.twitter.com/halloweenstore");
24                 return urls;
25             }
26         }
27     }
28 }
```

View (CONTACT US)

```
Contact.cshtml | ContactViewModel.cs | Index.cshtml | HomeCo

1 @model Ch24ShoppingCartMVC.Models.ContactViewModel
2 @{
3     ViewBag.Title = "Contact Us";
4 }
5 @* Display
6     Contact Us
7     Email
8     Phone
9     Fax
10    Social Media URLs *@
11 <h1>Contact Us</h1>
12 <p>Email: @Model.Email</p>
13 <p>Phone: @Model.Phone</p>
14 <p>Fax: @Model.Fax</p>
15
16 <p>@foreach (string url in Model.SocialMediaUrls) {
17     <a href="@url" target="_blank">@url</a> <br />
18 }
19 </p>
```

Controller (CONTACT US)

```
Index.cshtml | HomeController.cs | Ch24ShoppingCartMVC | Ch24ShoppingCartMVC.Controllers.HomeController.cs
1  using System;
2  using System.Collections.Generic;
3  using System.Linq;
4  using System.Web;
5  using System.Web.Mvc;
6  using Ch24ShoppingCartMVC.Models;
7
8  namespace Ch24ShoppingCartMVC.Controllers {
9      public class HomeController : Controller
10     {
11         [HttpGet]
12         public ActionResult Index() {
13             ViewBag.HeaderText = "Welcome to the Halloween Store";
14             ViewData["FooterText"] = "Where every day is Halloween!";
15             return View();
16         }
17         [HttpGet]
18         public ActionResult Contact()
19         {
20             //CREATE A ContactViewModel OBJECT call model model
21             ContactViewModel model = new ContactViewModel();
22             //Pass model to View
23             return View(model);
24         }
25     }
26 }
27
28
```

Assignment github repo link: <https://github.com/tanvirtanjum/ATP-MID-TERM-ASSIGNMENT>