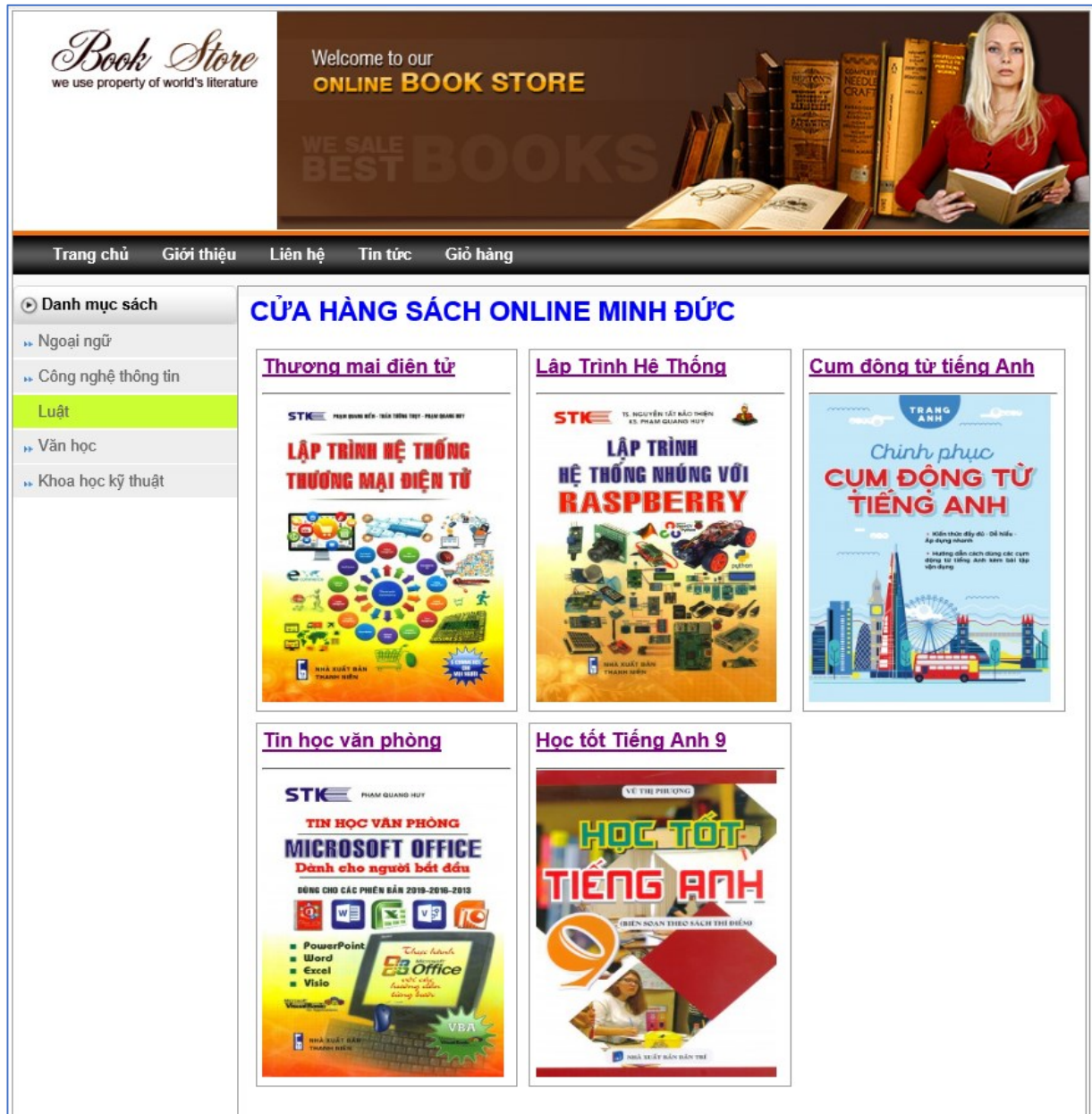


Chuẩn bị:

- Tạo CSDL **QLBanSach** (chạy file *QLBanSach.sql*)
- Tạo một project đặt tên là **BookStore**, chọn mẫu **MVC**.
- Copy thư mục **images** và **CSS** vào thư mục **wwwroot** trong Project
- Tạo trang **_LayoutClient** (Copy file **_LayoutClient.cshtml** vào **Views/Shared...**)

XÂY DỰNG TRANG CHỦ HIỂN THỊ VÀ XEM CHI TIẾT SẢN PHẨM

1. Hiển thị 5 sản phẩm mới nhất lên trang chủ



- Cài đặt EntityFramework sử dụng NuGet Package Manager
- Sử dụng EF code first để kết nối với cơ sở dữ liệu **BookStore** (đặt tên ADO.NET Entity Model là **QLBanSachDB**) chọn tất cả các bảng trong cơ sở dữ liệu.
- Kích **ReBuild Solution** để build lại project (mỗi khi sửa model cần Rebuild lại hệ thống)
- Tạo mới Controller: **BookStoreController**
- Tạo View: **Index** từ phương thức **Index** của **BookStoreController** (kế thừa từ **_LayoutClient.cshtml**)
- Trong **BookStoreController** cập nhật code như sau:

```
using BookStore.Models;
namespace BookStore.Controllers
{
```

```
public class BookStoreController : Controller
{
```

```
    QLBanSachDB db = new QLBanSachDB();
    private List<SACH> Laysachmoi(int count)
    { //Sắp xếp giảm dần theo Ngaycapnhat, lay count dòng đầu
        return db.SACHes.OrderByDescending(a => a.Ngaycapnhat).Take(count).ToList();
    }
    public ActionResult Index()
    { //Lấy 5 cuốn sách mới nhất
        var sachmoi = Laysachmoi(5);
        return View(sachmoi);
    }
}
```

- Trong View **Index** của BookStore cập nhật Code như sau:

```
@model IEnumerable<BookStore.Models.SACH>
```

```
@{
```

```
    ViewBag.Title = "Index";
```

```
    Layout = "~/Views/Shared/_LayoutClient.cshtml";
```

```
}
```

```
<style type="text/css">
```

```
    #products {
```

```
        width: 760px;
```

```
    }
```

```
    #product {
```

```
        width: 222px;
```

```
        height: 320px;
```

```
        float: left;
```

```
        display: block;
```

```
        padding: 5px;
```

```
        margin: 5px;
```

```
        text-align: left;
```

```
        background-color: #fff;
```

```
        border: 1px solid #999;
```

```
    }
```

```
</style>
```

```
<h1 style="color:blue">CỬA HÀNG SÁCH ONLINE MINH ĐỨC</h1> <br/>
```

```
<div id="products">
```

```
    @foreach (var sach in Model)
```

```
    {
```

```
        <div id="product">
```

```
            <a href="@Url.Action("Details", "BookStore", new { id = sach.Masach })"/>
```

```
                <h2> @sach.Tensach </h2> <br /> <hr />
```

```
                
```

```
            </div>
```

```
        }
```

```
        <div style="clear:left;"></div>
```

```
</div>
```

- Chạy xem kết quả:

2. Xem chi tiết sản phẩm: khi click vào tên sách, hình bìa của trang chủ.



- Ta cần tạo 1 Action tên **Details**, trong **BookStoreController**, tiếp nhận **id=Masach** để thực hiện lấy đầy đủ thông tin về 1 quyển sách để trình bày chi tiết.
- Cập nhật bổ sung Action **Details** trong **BookStoreController** như sau:

```
using BookStore.Models;
namespace BookStore.Controllers
{
    public class BookStoreController : Controller
    {
        // GET: BookStore

        QLBanSachDB db = new QLBanSachDB();
        private List<SACH> Laysachmoi(int count)
        { //Sắp xếp giảm dần theo Ngaycapnhat, lay count dòng đầu
            return db.SACHes.OrderByDescending(a => a.Ngaycapnhat).Take(count).ToList();
        }

        public ActionResult Index()
        { //Lấy 5 cuốn sách mới nhất
            var sachmoi = Laysachmoi(5);
            return View(sachmoi);
        }

        public ActionResult Details(int id)
        {
            var sach = from s in db.SACHes
                        where s.Masach == id
                        select s;
            return View(sach.Single());
        }
    }
}
```

- Tạo View hiển thị thông tin 1 quyển sách này: Tạo mới View **Details** sử dụng **_LayoutClient**.
- Cập nhật Code hiển thị trên View **Details**:

```
@{
    ViewBag.Title = "Details";
    Layout = "~/Views/Shared/_LayoutClient.cshtml";
}
```

```
<style type="text/css">
    #Datmua a {
        clear: both;
        background-color: red;
        color: white;
        width: 120px;
        height: 35px;
        display: block;
        float: right;
        text-align: center;
        padding-top: 10px;
        text-decoration: none;
        font-weight: bold;
        font-size: 14pt;
    }

    #Datmua a:hover {
        background-color: gray;
        color: red;
    }
</style>

<h2>THÔNG TIN CHI TIẾT SÁCH</h2>
<br />
<table border="0" style="width:100%">
    <tr>
        <td style="width:60% valign="top">
            <h1 style="font-weight:bold;color:red">Tên sách: @Model.Tensach</h1> <br />
            <p style="text-align:justify"><span style="font-size:18pt"> <b>Mô tả:</b>@Model.Mota
                </span></p> <br />
            <p style="font-size:18pt"><b>Giá bán: <span style="color:red">@Model.Giaban VNĐ </span>
                </p>
        </td>
        <td style="width:40%">
            
        </td>
    </tr>
</table>
<br />
<div id="Datmua">
    <a href="@Url.Action("ThemGiohang", "GioHang", new {id=@Model.Masach})" >
        ĐẶT MUA </a>
</div>
```

- Chạy xem kết quả: Khi click xem chi tiết 1 sản phẩm bất kỳ:

XÂY DỰNG TRANG GIỎ HÀNG ĐỂ ĐẶT HÀNG TRÊN WEBSITE

- Xem lại cấu trúc các Table có liên quan đến Giỏ hàng trong CSDL:

SACH *			
	Column Name	Data Type	Allow Nulls
🔑	Masach	int	<input type="checkbox"/>
	Tensach	nvarchar(100)	<input type="checkbox"/>
	Giaban	decimal(18, 0)	<input checked="" type="checkbox"/>
	Mota	nvarchar(MAX)	<input checked="" type="checkbox"/>
	Anhbia	varchar(50)	<input checked="" type="checkbox"/>
	Ngaycapnhat	datetime	<input checked="" type="checkbox"/>
	Soluongton	int	<input checked="" type="checkbox"/>
	MaCD	int	<input checked="" type="checkbox"/>
	MaNXB	int	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

DONDATHANG *			
	Column Name	Data Type	Allow Nulls
🔑	MaDonHang	int	<input type="checkbox"/>
	Dathanhtoan	bit	<input checked="" type="checkbox"/>
	Tinhtranggiaohang	bit	<input checked="" type="checkbox"/>
	Ngaydat	datetime	<input checked="" type="checkbox"/>
	Ngaygiao	datetime	<input checked="" type="checkbox"/>
	MaKH	int	<input checked="" type="checkbox"/>

CHITIETDONTHANG *			
	Column Name	Data Type	Allow Nulls
🔑	MaDonHang	int	<input type="checkbox"/>
🔑	Masach	int	<input type="checkbox"/>
	Soluong	int	<input checked="" type="checkbox"/>
	Dongia	decimal(18, 0)	<input checked="" type="checkbox"/>

- Tạo Class **Giohang** trong **Models**:
 - ✓ Khai báo các thuộc tính cho Giỏ hàng.
 - ✓ Constructor khởi tạo Giỏ hàng, với tham số truyền vào là Mã sách.

```
using BookStore.Models;
namespace BookStore.Controllers
{
    public class Giohang
    {
        QLBanSachDB db = new QLBanSachDB();
        public int iMasach { set; get; }
        public string sTensach { set; get; }
        public string sAnhbia { set; get; }
        public Double dDongia { set; get; }
        public int iSoluong { set; get; }
        public Double dThanhtien
        {
            get { return iSoluong * dDongia; }
        }
        //Khởi tạo giỏ hàng theo mã sách được truyền vào với số lượng mặc định là 1
        public Giohang(int Masach)
        {
            iMasach = Masach;
            SACH sach = db.SACHes.Single(n => n.Masach == iMasach);
            sTensach = sach.Tensach;
            sAnhbia = sach.Anhbia;
            dDongia = double.Parse(sach.Giaban.ToString());
            iSoluong = 1;
        }
    }
}
```

- Tạo mới Controller: **GiohangController**

- ✓ Cập nhật phương thức Lấy giỏ hàng từ **Session[« Giohang »]** nếu có, nếu không sẽ khởi tạo giỏ hàng rỗng.

```
using BookStore.Models;
namespace BookStore.Controllers
{
    public class GiohangController : Controller
    {
        QLBanSachDB db = new QLBanSachDB();
        //Lấy giỏ hàng
        public List<Giohang> Laygiohang()
        {
            List<Giohang> lstGiohang = Session["Giohang"] as List<Giohang>;
            if (lstGiohang==null)
            {
                //Nếu giỏ hàng chưa tồn tại thì khởi tạo listGiohang
                lstGiohang = new List<Giohang>();
                Session["Giohang"] = lstGiohang;
            }
            return lstGiohang;
        }
    }
}
```

- ✓ Cập nhật phương thức **ThemGiohang()**

```
//Thêm hàng vào giỏ
public ActionResult ThemGiohang(int iMasach, string strURL)
{
    //Lấy ra Session giỏ hàng
    List<Giohang> lstGiohang = Laygiohang();
    //Kiểm tra sách này tồn tại trong Session["Giohang"] chưa?
    Giohang sanpham = lstGiohang.Find(n => n.iMasach == iMasach);
    if (sanpham==null)
    {
        sanpham = new Giohang(iMasach);
        lstGiohang.Add(sanpham);
        return Redirect(strURL);
    }
    else
    {
        sanpham.iSoluong++;
        return Redirect(strURL);
    }
}
```

- ✓ Cập nhật phương thức tính **tổng số lượng**

```
//Tổng số lượng
private int TongSoLuong()
{
    int iTongSoLuong = 0;
    List<Giohang> lstGiohang = Session["Giohang"] as List<Giohang>;
    if (lstGiohang != null)
    {
        iTongSoLuong = lstGiohang.Sum(n => n.iSoluong);
    }
    return iTongSoLuong;
}
```

- ✓ Cập nhật phương thức tính **Tongtien()**


```
//Tính tổng tiền
private double Tongtien()
{
    double iTongtien = 0;
    List<Giohang> lstGiohang = Session["Giohang"] as List<Giohang>;
    if (lstGiohang != null)
    {
        iTongtien = lstGiohang.Sum(n => n.dThanhtien);
    }
    return iTongtien;
}
```

- ✓ Cập nhật phương thức xử lý hiển thị Giỏ hàng : **GioHang()**

```
// Xây dựng trang giỏ hàng
public ActionResult Giohang()
{
    List<Giohang> lstGiohang = Laygiohang();
    if (lstGiohang.Count == 0)
    {
        return RedirectToAction("Index", "BookStore");
    }
    ViewBag.Tongsoluong = TongSoLuong();
    ViewBag.Tongtien = Tongtien();
    return View(lstGiohang);
}
```

- Từ Phương thức **GioHang()** xây dựng View **GioHang.cshtml** tương ứng như sau :

```
@using BookStore.Models
@model List<Giohang>
@{
    ViewBag.Title = "Giohang";
    Layout = "~/Views/Shared/_LayoutClient.cshtml";
}

<h2 style="text-align:center">THÔNG TIN GIỎ HÀNG</h2>
<table align="center" border="1" cellpadding="0" cellspacing="0" style="font-size:13pt" >
<tr style="text-align:center; font-weight:bold">
    <td>Mã sách</td>
    <td>Tên sách</td>
    <td>Ảnh bìa</td>
    <td>Số lượng</td>
    <td>Đơn giá</td>
    <td>Thành tiền</td>
</tr>
@foreach (var item in Model)
{
    <tr style="text-align:center; font-weight:bold">
        <td>@item.iMasach</td>
        <td>@item.sTensach</td>
        <td></td>
        <td>@item.iSoluong</td>
        <td>@String.Format("{0:0,0}", item.dDongia)</td>
        <td>@String.Format("{0:0,0}", item.dThanhtien)</td>
    </tr>
}
```



```
<tr style="text-align:right; font-weight:bold;color:red">
    <td colspan="3">Số lượng sách: @ViewBag.Tongsoluong</td>
    <td colspan="3">Tổng tiền: @String.Format("{0:0,0}", ViewBag.Tongtien) VNĐ </td>
</tr>
</table>
```

- Xem và cập nhật lại liên kết dẫn đến trang Giỏ hàng : Trang **Details.cshtml**

```
...
<div id="Datmua">
    <a href="@Url.Action("ThemGiohang", "GioHang", new { @iMasach = Model.Masach,
        @strURL=Request.Url.ToString() })" >
        ĐẶT MUA </a>
</div>
```

- Thực thi xem Kết quả : Từ trang **Details** : Click vào liên kết **đặt mua** (Thực hiện trên một số sản phẩm, và thực hiện lặp lại vài lần trên 1 sản phẩm.)
- Kích vào liên kết **Giỏ hàng** trên menu ngang. Kết quả: Các sản phẩm được chọn đã lưu vào Session["Giohang"]

THÔNG TIN GIỎ HÀNG

Mã sách	Tên sách	Ảnh bìa	Số lượng	Đơn giá	Thành tiền
20	Tin Học Văn Phòng - Microsoft Word 2000 (Tái Bản Lần Thứ Nhất)		3	15,600	46,800
22	Hướng Dẫn Tự Học Photoshop CS (Photoshop 8.0)		1	25,000	25,000
			Số lượng sách: 4 Tổng tiền: 71,800 VNĐ		

- Cập nhật liên kết “Chi tiết” và “Xóa” trong View **Giohang.cshtml**

```
@using BookStore.Models
@model List<Giohang>
@{
    ViewBag.Title = "Giohang";
    Layout = "~/Views/Shared/_LayoutClient.cshtml";
}
<h2 style="text-align:center">THÔNG TIN GIỎ HÀNG</h2>
<table align="center" border="1" cellpadding="0" cellspacing="0" style="font-size:13pt" >
    <tr style="text-align:center; font-weight:bold">
        <td>Mã sách</td>
        <td>Tên sách</td>
        <td>Ảnh bìa</td>
        <td>Số lượng</td>
        <td>Đơn giá</td>
        <td>Thành tiền</td>
        <td width="50px"></td>
        <td width="50px"></td>
    </tr>
    @foreach (var item in Model)
    {
        <tr style="text-align:center; font-weight:bold">
            <td>@item.iMasach</td>
            <td>@item.sTensach</td>
```

```

<td></td>
<td>@item.iSoluong</td>
<td>@String.Format("{0:0,0}", item.dDongia)</td>
<td>@String.Format("{0:0,0}", item.dThanhTien)</td>
<td><a href="@Url.Action("Details", "BookStore", new{ id = item.iMasach })">
    Chi tiết</a></td>
<td> @Html.ActionLink("Xóa", "XoaGiohang", new{ @iMaSp = item.iMasach })</td>
</tr>
}
<tr style="text-align:right; font-weight:bold;color:red">
<td colspan="4">Số lượng sách: @ViewBag.Tongsoluong</td>
<td colspan="4">Tổng tiền: @String.Format("{0:0,0}", ViewBag.Tongtien) VNĐ </td>
</tr>
</table>

```

- Click vào liên kết “Chi tiết” sẽ mở trang Details.
- Xử lý xóa giỏ hàng:
 - ✓ Cập nhật Action XoaGiohang () vào GiohangController

```

//Xóa giỏ hàng
public ActionResult XoaGiohang(int iMaSP)
{
    //Lấy giỏ hàng từ Session
    List<Giohang> lstGiohang = Laygiohang();
    //Kiểm tra sách đã có trong Session["Giohang"]
    Giohang sanpham = lstGiohang.SingleOrDefault(n => n.iMasach == iMaSP);
    //Nếu tồn tại thì cho xóa
    if (sanpham != null)
    {
        lstGiohang.RemoveAll(n => n.iMasach == iMaSP);
        return RedirectToAction("GioHang");
    }
    if (lstGiohang.Count==0)
    {
        return RedirectToAction("Index", "BookStore");
    }
    return RedirectToAction("GioHang");
}

```

- Kết quả : Click « Xóa » để xóa mặt hàng ra khỏi giỏ.
- Bổ sung tiếp chức năng “Cập nhật” dùng cho phép điều chỉnh số lượng và cập nhật lại thành tiền và tổng tiền.
 - ✓ Cập nhật GioHangController : Bổ sung Action CapnhatGiohang()

```

//Cập nhật giỏ hàng
public ActionResult CapnhatGiohang(int iMaSP, FormCollection f)
{
    //Lấy giỏ hàng từ Session
    List<Giohang> lstGiohang = Laygiohang();
    //Kiểm tra sách đã có trong Session["Giohang"]
    Giohang sanpham = lstGiohang.SingleOrDefault(n => n.iMasach == iMaSP);
    //Nếu tồn tại thì cho sửa Soluong
    if (sanpham != null)
    {
        sanpham.iSoluong = int.Parse(f["txtSoluong"].ToString());
    }
    return RedirectToAction("GioHang");
}

```

- ✓ Điều chỉnh View : Giohang.cshtml có bổ sung thêm Form chứa button “Cập nhật”

```

@using BookStore.Models
@model List<Giohang>
@{
    ViewBag.Title = "Giohang";
    Layout = "~/Views/Shared/_LayoutClient.cshtml";
}

<h2 style="text-align:center">THÔNG TIN GIỎ HÀNG</h2>
<table align="center" border="1" cellpadding="0" cellspacing="0" style="font-size:13pt" >
    <tr style="text-align:center; font-weight:bold">
        <td>Mã sách</td>
        <td>Tên sách</td>
        <td>Ảnh bìa</td>
        <td>Số lượng</td>
        <td>Đơn giá</td>
        <td>Thành tiền</td>
        <td width="50px"></td>
        <td width="50px"></td>
        <td width="50px"></td>
    </tr>
    @foreach (var item in Model)
    {
        <tr style="text-align:center; font-weight:bold">
            <td>@item.iMasach</td>
            <td>@item.sTensach</td>
            <td></td>
            @using (Html.BeginForm("Capnhatgiohang", "Giohang", new { @iMaSp = item.iMasach }))
            {
                <td>
                    <input type="number" min="1" name="txtSoluong" value="@item.iSoluong"
                        style="background-color:yellow" />
                </td>
                <td>@String.Format("{0:0,0}", item.dDongia)</td>
                <td>@String.Format("{0:0,0}", item.dThanhTien)</td>
                <td><a href="@Url.Action("Details", "BookStore", new { id = item.iMasach })">
                    Chi tiết</a></td>
                <td> @Html.ActionLink("Xóa", "XoaGiohang", new { @iMaSp = item.iMasach })</td>
                <td><input type="submit" value="Cập nhật"/></td>
            }
        </tr>
    }
    <tr style="text-align:right; font-weight:bold;color:red">
        <td colspan="4">Số lượng sách: @ViewBag.Tongsoluong</td>
        <td colspan="5">Tổng tiền: @String.Format("{0:0,0}", ViewBag.Tongtien) VNĐ </td>
    </tr>
</table>

```

- Kết quả: Khi người dùng điều chỉnh Số lượng và Click Cập nhật, hệ thống sẽ cập nhật lại số lượng, tính lại thành tiền, tổng số lượng và tổng tiền và load lại trang GioHang.
- Bổ sung tiếp chức năng “Xóa Giỏ Hàng” cho phép xóa tất cả thông tin trong Giỏ hàng.
 - ✓ Cập nhật GioHangController : Bổ sung Action XoaTatcaGiohang()

```

public ActionResult XoaTatcaGiohang()
{
    //Lấy giỏ hàng từ Session
    List<Giohang> lstGiohang = Laygiohang();
    lstGiohang.Clear();
    return RedirectToAction("Index", "BookStore");
}

```

- ✓ Cập nhật View: Giohang.cshtml bổ sung liên kết “Xóa Giỏ Hàng”

```
<tr style="text-align:right; font-weight:bold;color:red">
  <td colspan="4">Số lượng sách: @ViewBag.Tongsoluong</td>
  <td colspan="5">Tổng tiền: @String.Format("{0:0,0}", ViewBag.Tongtien) VNĐ </td>
</tr>
<tr style="text-align:right; font-weight:bold;color:blue">
  <td colspan="9">
    @Html.ActionLink("Xóa Giỏ Hàng", "XoaTatcaGiohang")
  </td>
</tr>
</table>
```

- Kết quả: Tại trang Giỏ Hàng khi Click “Xóa Giỏ hàng”, hệ thống sẽ xóa tất cả thông tin trong Giỏ hàng và trở về trang chủ.

THÔNG TIN GIỎ HÀNG

Mã sách	Tên sách	Ảnh bìa	Số lượng	Đơn giá	Thành tiền			
20	Tin Học Văn Phòng - Microsoft Word 2000 (Tái Bản Lần Thứ Nhất)		3	15,600	46,800	Chi tiết	Xóa	Cập Nhật
23	Tin Học Văn Phòng - Microsoft Excel 2000 (Tái Bản Lần Thứ Nhất)		1	16,800	16,800	Chi tiết	Xóa	Cập Nhật
Số lượng sách: 4				Tổng tiền: 63,600 VNĐ				
Xóa Giỏ Hàng								