

Buổi 3: KẾT NỐI CƠ SỞ DỮ LIỆU - LINQ TO SQL

A. TẠO CƠ SỞ DỮ LIỆU

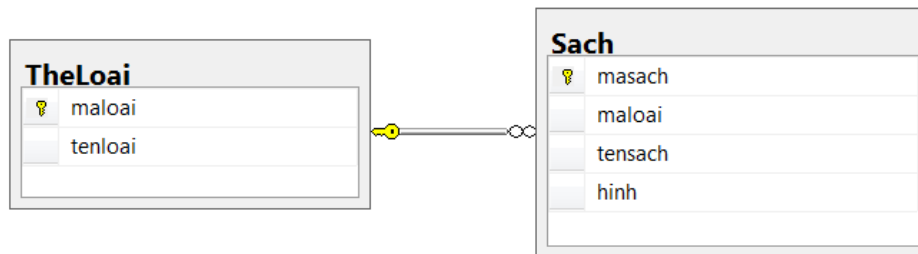
1. Tạo cơ sở dữ liệu trong SQL Server: **NHASACH**
2. Bảng **TheLoai**

| | Column Name | Data Type | Allow Nulls |
|----|-------------|--------------|-------------------------------------|
| PK | maloai | int | <input type="checkbox"/> |
| | tenloai | nvarchar(30) | <input checked="" type="checkbox"/> |

3. Bảng **Sach**

| | Column Name | Data Type | Allow Nulls |
|----|-------------|--------------|-------------------------------------|
| PK | masach | int | <input type="checkbox"/> |
| | maloai | int | <input checked="" type="checkbox"/> |
| | tensach | nvarchar(50) | <input checked="" type="checkbox"/> |
| | hinh | varchar(50) | <input checked="" type="checkbox"/> |

4. Database Diagram



5. Script

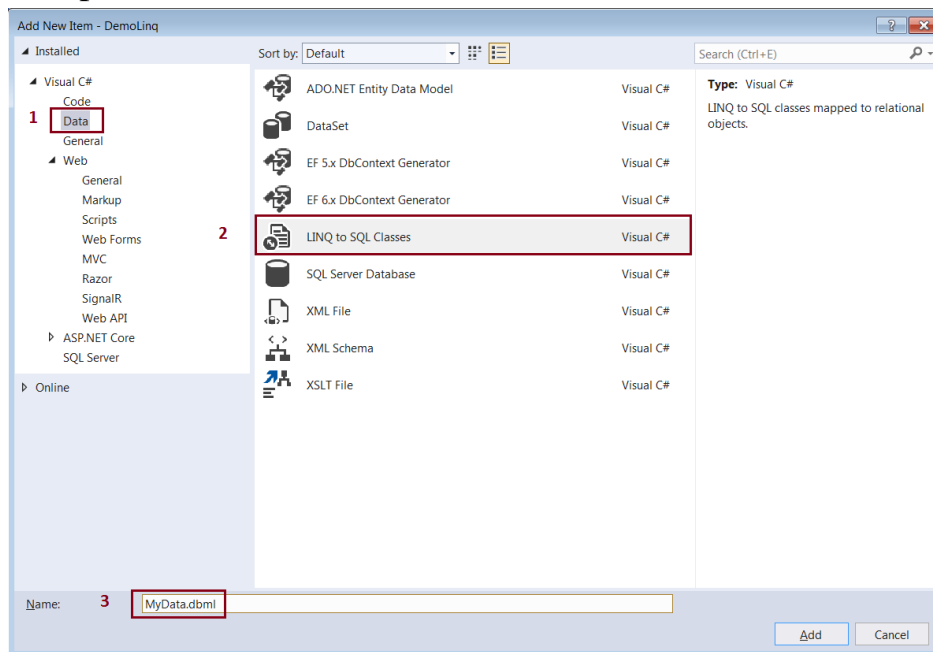
```
create database NhaSach
use NhaSach
create table TheLoai
(
    maloai int identity primary key,
    tenloai nvarchar(30)
)
create table Sach
(
    masach int identity primary key,
    maloai int references TheLoai(maloai),
    tensach nvarchar(50),
    hinh varchar(50)
)
-----
insert into TheLoai(tenloai) values(N'Công nghệ thông tin')
insert into TheLoai(tenloai) values(N'Kinh tế')
insert into TheLoai(tenloai) values(N'Mỹ thuật')
-----
insert into Sach(maloai,tensach,hinh) values(1,N'Lập trình
Java','/Content/images/book1.jpg')
insert into Sach(maloai,tensach,hinh) values(2,N'Quản trị kinh doanh quốc
tế','/Content/images/book2.jpg')
insert into Sach(maloai,tensach,hinh) values(3,N'Kỹ thuật tô
màu','/Content/images/book3.jpg')
```

B. HƯỚNG DẪN THỰC HIỆN

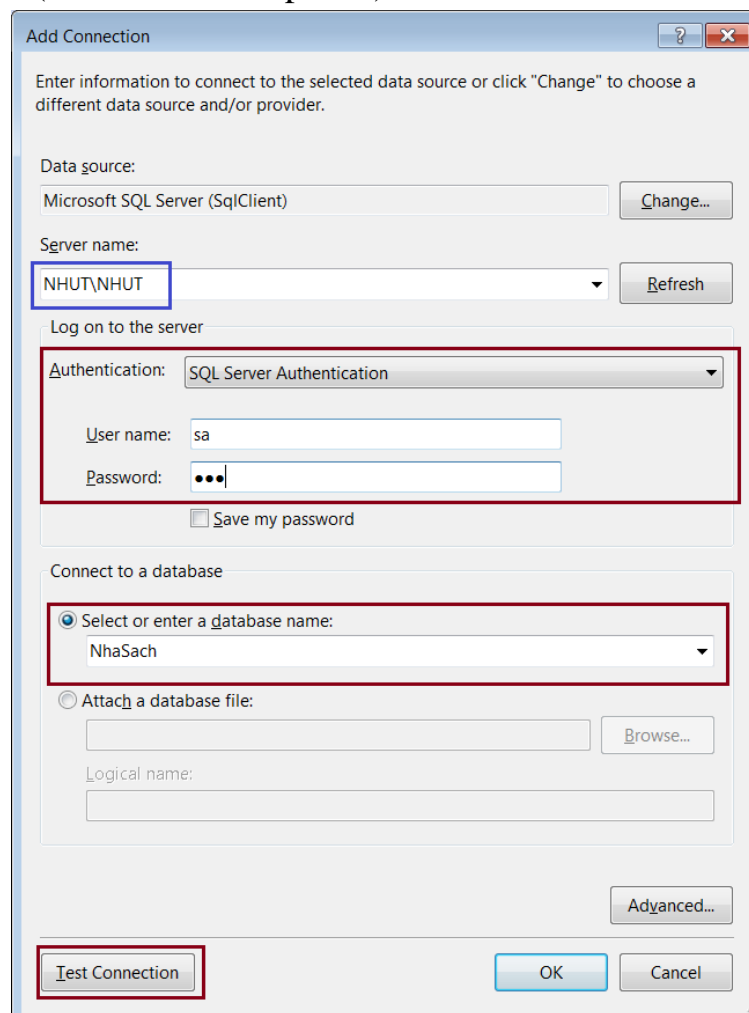
1. Tạo project ASP.NET MCV

2. Tạo kết nối Linq trong project

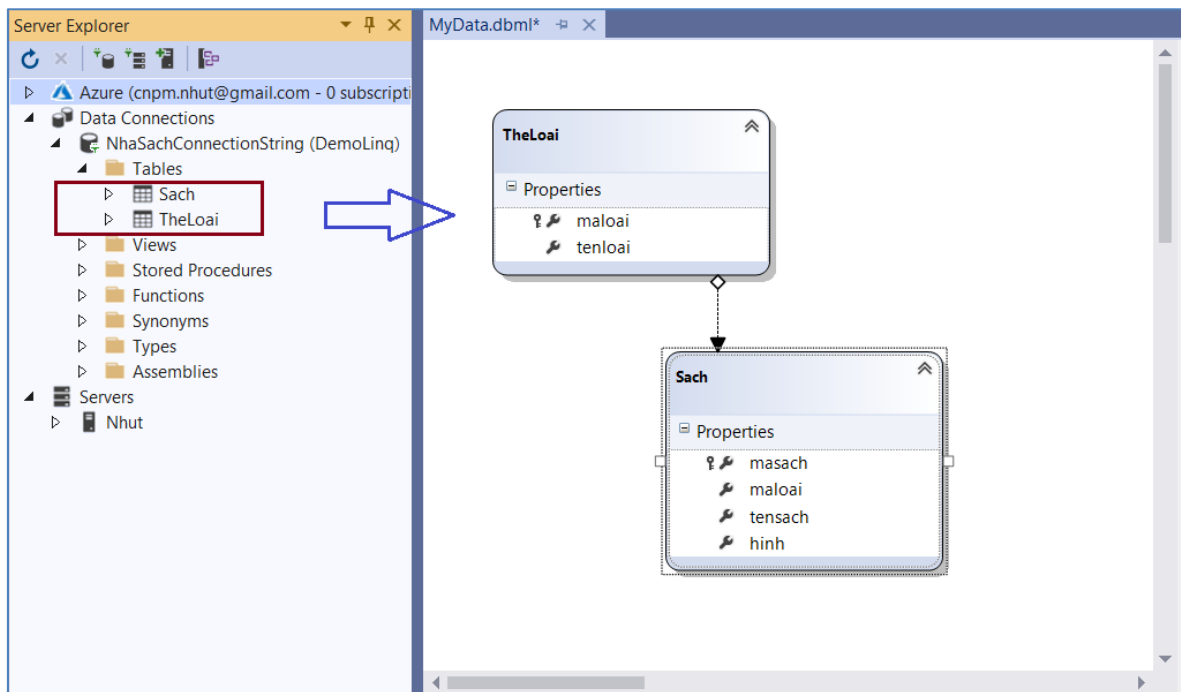
- Click phải vào Models → Add → New Item ...



- Mở Server Explorer (View/Server Explorer) → Connect to Database

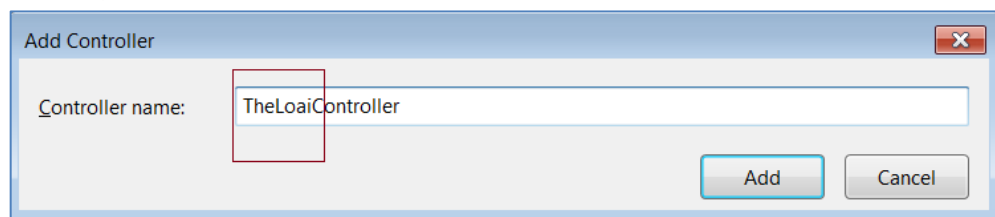


- Drag chuột kéo các bảng vào vùng **MyData.dbml**



3. Tạo Controller: TheLoaiController

- Click phải vào Controllers → Add → Controller... → Đặt tên (tên của lớp xử lý theo sau bằng Controller)



- Thêm code vào file TheLoaiController.cs

```
using DemoLinq.Models;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;

namespace DemoLinq.Controllers
{
    0 references
    public class TheLoaiController : Controller
    {
        // GET: TheLoai
        MyDataDataContext data = new MyDataDataContext();
        0 references
        public ActionResult Index()
        {
            var all_theloai = from tt in data.TheLoais select tt;
            return View(all_theloai);
        }
    }
}
```

Annotations in the code:

- Tên Model**: Points to `MyDataDataContext` in the line `MyDataDataContext data = new MyDataDataContext();`
- Tên Table trong Model**: Points to `TheLoais` in the line `var all_theloai = from tt in data.TheLoais select tt;`

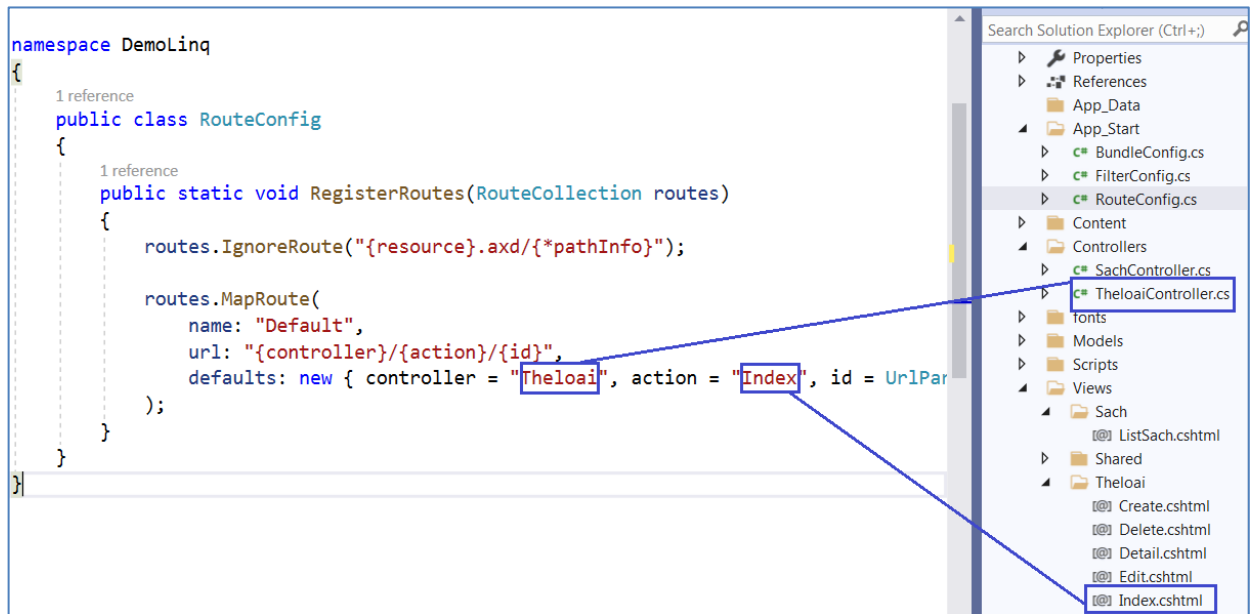
- Tạo View cho trang Index

The screenshot shows the 'Add View' dialog box. The 'View name' field contains 'Index'. The 'Template' dropdown menu is set to 'Empty (without model)'. The 'Model class' dropdown is empty. In the 'Options' section, the checkbox 'Use a layout page:' is checked, and the text box next to it contains '~/_Views/Shared/_Layout.cshtml'. There are 'Add' and 'Cancel' buttons at the bottom right.

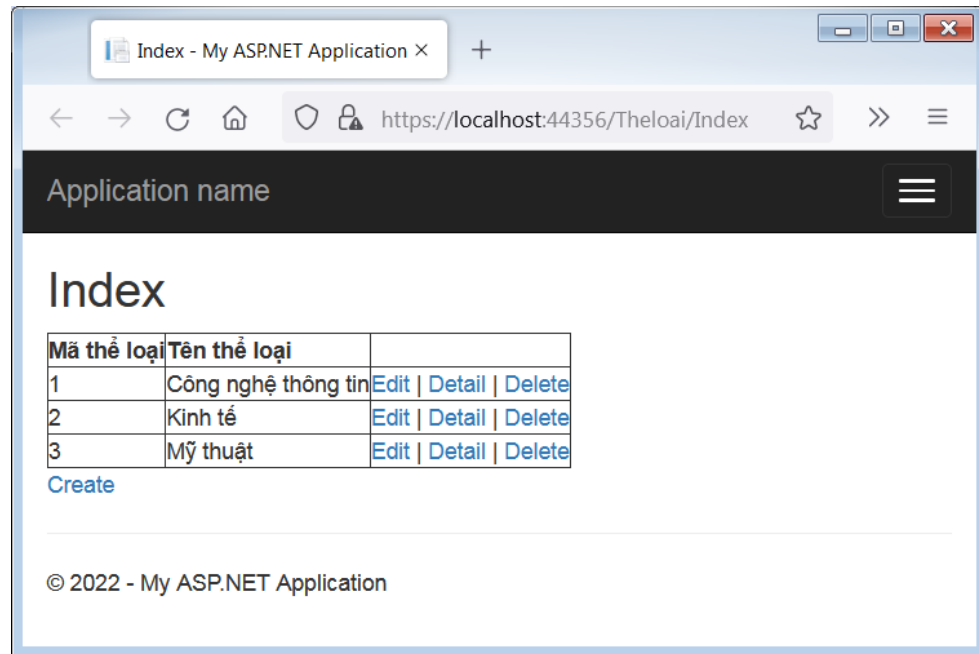
- Layout trang *Index.cshtml*

```
@model IEnumerable<DemoLinq.Models.TheLoai>
@{
    ViewBag.Title = "Index";
    Layout = "~/Views/Shared/_Layout.cshtml";
}
<h2>Index</h2>
<table border="1px">
    <tr>
        <th>Mã thể loại</th>
        <th>Tên thể loại </th>
        <th></th>
    </tr>
    @foreach (var item in Model)
    {
        <tr>
            <td>@item.maloai</td>
            <td>@item.tenloai</td>
            <td>
                @Html.ActionLink("Edit", "Edit", new { id = item.maloai }) |
                @Html.ActionLink("Detail", "Detail", new { id = item.maloai }) |
                @Html.ActionLink("Delete", "Delete", new { id = item.maloai })
            </td>
        </tr>
    }
</table>
@Html.ActionLink("Create", "Create", "TheLoai")
```

- Thay đổi trang hiển thị mặc định trong **App_Start / RouteConfig.cs**



- Kết quả thực thi trang `Index.cshtml`



4. Code xử lý các chức năng Detail, Create, Edit, Delete

- Code trong TheloaiController.cs

```
using DemoLinq.Models;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;

namespace DemoLinq.Controllers
{
    public class TheloaiController : Controller
    {
        // GET: Theloai
        MyDataDataContext data = new MyDataDataContext();
        //-----Index-TheLoai-----
        public ActionResult Index()
        {
            var all_theloai = from tt in data.TheLoais select tt;
            return View(all_theloai);
        }
        //-----Detail-----
        public ActionResult Detail(int id)
        {
            var D_theloai = data.TheLoais.Where(m => m.maloai == id).First();
            return View(D_theloai);
        }
        //-----Create-----
        public ActionResult Create()
        {
            return View();
        }
        [HttpPost]
        public ActionResult Create(FormCollection collection, TheLoai tl)
        {
            var ten = collection["tenloai"];
            if (string.IsNullOrEmpty(ten))
            {
                ViewData["Error"] = "Don't empty!";
            }
            else
            {
                tl.tenloai = ten;
                data.TheLoais.InsertOnSubmit(tl);
                data.SubmitChanges();
                return RedirectToAction("Index");
            }
            return this.Create();
        }
        //-----Edit-----
        public ActionResult Edit(int id)
        {
            var E_category = data.TheLoais.First(m => m.maloai == id);
            return View(E_category);
        }
        [HttpPost]
        public ActionResult Edit(int id, FormCollection collection)
        {

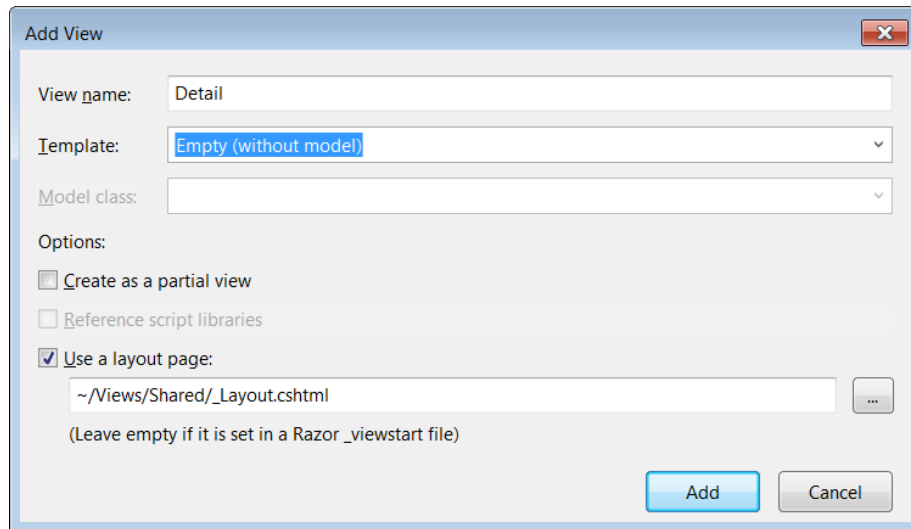
```

```

        var theloai = data.TheLoais.First(m => m.maloai == id);
        var E_tenloai = collection["tenloai"];
        theloai.maloai = id;
        if (string.IsNullOrEmpty(E_tenloai))
        {
            ViewData["Error"] = "Don't empty!";
        }
        else
        {
            theloai.tenloai = E_tenloai;
            UpdateModel(theloai);
            data.SubmitChanges();
            return RedirectToAction("Index");
        }
        return this.Edit(id);
    }
    //-----Delete-----
    public ActionResult Delete(int id)
    {
        var D_theloai = data.TheLoais.First(m => m.maloai == id);
        return View(D_theloai);
    }
    [HttpPost]
    public ActionResult Delete(int id, FormCollection collection)
    {
        var D_theloai = data.TheLoais.Where(m => m.maloai == id).First();
        data.TheLoais.DeleteOnSubmit(D_theloai);
        data.SubmitChanges();
        return RedirectToAction("Index");
    }
}
}

```

- Tạo trang Detail.cshtml

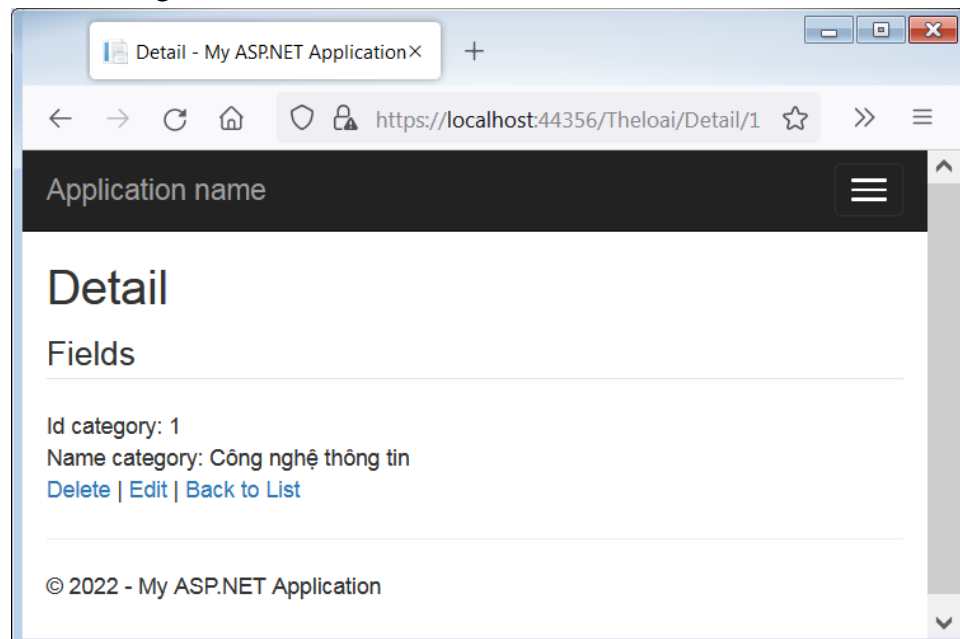


- Layout cho trang Detail.cshtml

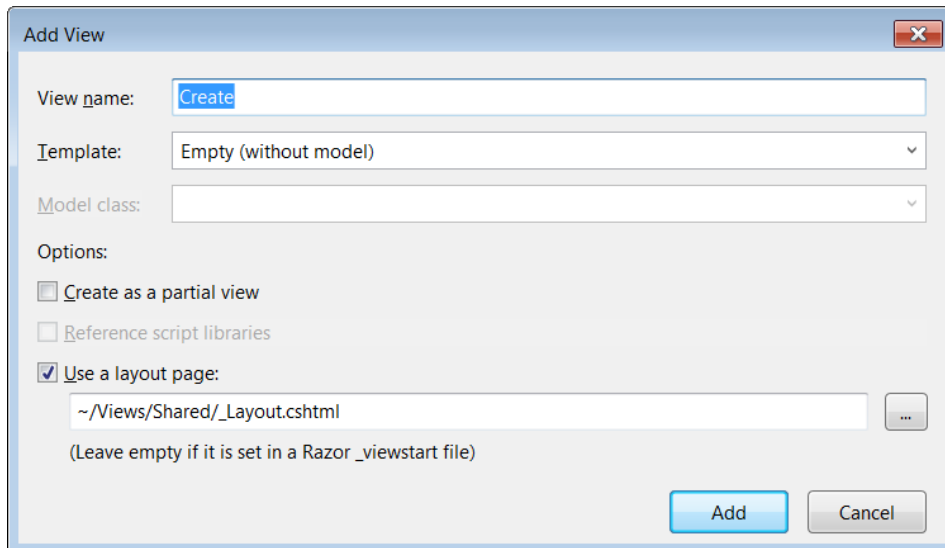
```
@{
    ViewBag.Title = "Detail";
    Layout = "~/Views/Shared/_Layout.cshtml";
}

<h2>Detail</h2>
<fieldset>
    <div class="display-label">Mã loại: @Model.maloai</div>
    <div class="display-label">Tên loại: @Model.tenloai</div>
</fieldset>
@using (Html.BeginForm())
{
    <p>
        @Html.ActionLink("Delete", "Delete", new { id = Model.maloai }) |
        @Html.ActionLink("Edit", "Edit", new { id = Model.maloai }) |
        @Html.ActionLink("Back to List", "Index")
    </p>
}
```

- Kết quả thực thi trang Detail



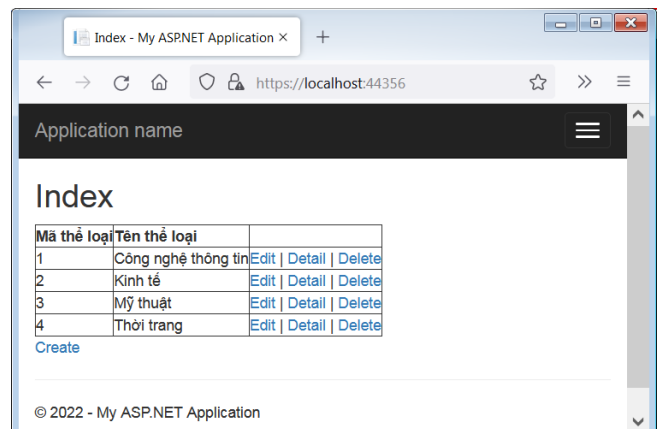
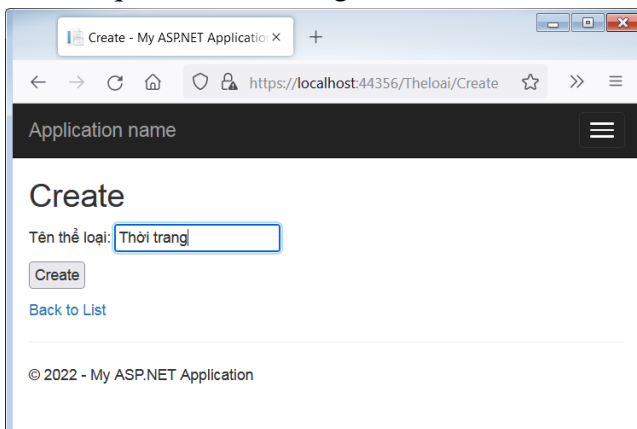
- Tạo trang Create.cshtml



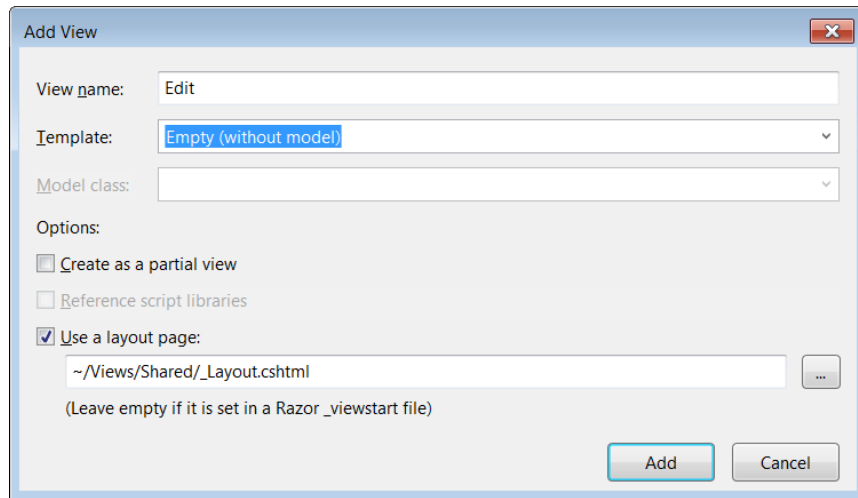
- Layout cho trang Create.cshtml

```
@{
    ViewBag.Title = "Create";
    Layout = "~/Views/Shared/_Layout.cshtml";
}
<h2>Create</h2>
@using (Html.BeginForm()) {
    @Html.ValidationSummary(true)
    <fieldset>
        <p>
            Tên thẻ loại:
            @Html.TextBox("tenloai")<font color="red">@ViewData["Error"]</font>
        </p>
        <p>
            <input type="submit" value="Create" />
        </p>
    </fieldset>
}
<div>
    @Html.ActionLink("Back to List", "Index")
</div>
```

- Kết quả thực thi trang Create.cshtml



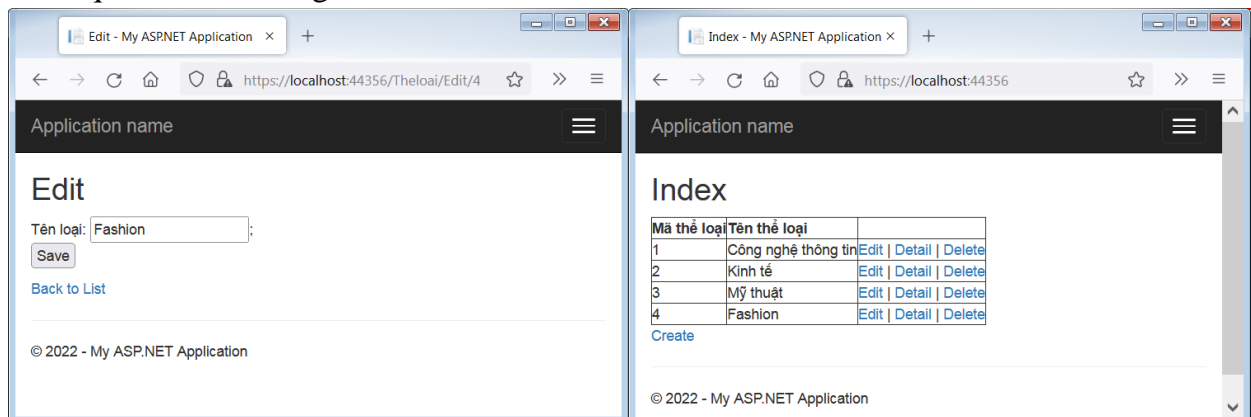
- Tạo trang Edit.cshtml



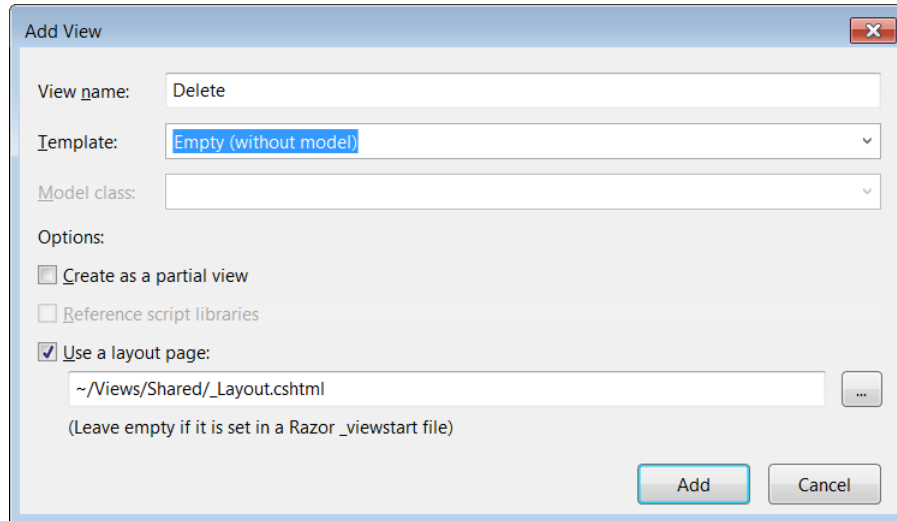
- Layout cho trang Edit.cshtml

```
@{
    ViewBag.Title = "Edit";
    Layout = "~/Views/Shared/_Layout.cshtml";
}
<h2>Edit</h2>
@using (Html.BeginForm())
{
    @Html.ValidationSummary(true)
    <fieldset>
        <div class="editor-field">
            Tên loại:
            @Html.TextBox("tenloai");
            @ViewData["Error"]
        </div>
        <p>
            <input type="submit" value="Save" />
        </p>
    </fieldset>
}
<div>
    @Html.ActionLink("Back to List", "Index")
</div>
```

- Kết quả thực thi trang Edit.cshtml



- Tạo trang Delete.cshtml



- Layout cho trang Delete.cshtml

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

<h2>Delete</h2>
<fieldset>
    <div class="display-label">Mã loại: @Html.Encode(Model.maloai)</div>
    <div class="display-label">Tên loại: @Html.Encode(Model.tenloai)</div>
</fieldset>
@using (Html.BeginForm())
{
    <p>
        <input type="submit" value="Delete" />
        @Html.ActionLink("Back to List", "Index")
    </p>
}
```

- Kết quả thực thi trang Delete.cshtml

