Controller:

using ShopDB.Models;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.Mvc;

using System.Data.Entity;

using ShopDB.Models.Vm;

using System.IO;

using System.Net;

namespace ShopDB.Controllers

{

public class UserController : Controller

{

ShopDBEntities db = new ShopDBEntities();

// GET: User

public ActionResult Index()

{

var user = db.Users.Include(d=>d.Details.Select(p=>p.Product)).OrderByDescending(u=>u.UserId).ToList();

return View(user);

}

[HttpGet]

public ActionResult Create()

{

return View();

}

[HttpPost]

public ActionResult Create(Vmodel vmodel, int[]productId)

{

if(ModelState.IsValid)

{

User user = new User()

{

UserName = vmodel.UserName,

Age = vmodel.Age,

UserDate = vmodel.UserDate,

IsAudil=vmodel.IsAudil,

};

HttpPostedFileBase file = vmodel.ImageFile;

if (file != null)

{

string fileName = Path.Combine("/image/" + DateTime.Now.Ticks.ToString() + Path.GetExtension(file.FileName));

file.SaveAs(Server.MapPath(fileName));

user.ImagePath = fileName;

}

foreach (var p in productId)

{

var product = db.Products.FirstOrDefault(f => f.ProductId == p);

Detail detail = new Detail()

{

User = user,

UserId = user.UserId,

ProductId = product.ProductId,

ProductName = product.ProductName,

};

db.Details.Add(detail);

}

db.SaveChanges();

return RedirectToAction("Index");

}

return View(vmodel);

}

public ActionResult AddProduct(int? id)

{

ViewBag.product = new SelectList(db.Products, "productId", "ProductName", (id != null) ? id.ToString() : "");

return PartialView("\_addProduct");

}

[HttpGet]

public ActionResult Edit(int? id)

{

User user = db.Users.Find(id);

Vmodel vmodel = new Vmodel()

{

UserId = user.UserId,

UserName = user.UserName,

Age = user.Age,

UserDate = user.UserDate,

IsAudil = user.IsAudil,

ImagePath = user.ImagePath,

};

return View(vmodel);

}

[HttpPost]

public ActionResult Edit(Vmodel vmodel, int[] productId)

{

if (ModelState.IsValid)

{

var user = db.Users.Find(vmodel.UserId);

user.UserName = vmodel.UserName;

user.Age = vmodel.Age;

user.UserDate = vmodel.UserDate;

user.IsAudil = vmodel.IsAudil;

HttpPostedFileBase file = vmodel.ImageFile;

if (file != null)

{

string fileName = Path.Combine("/image/" + DateTime.Now.Ticks.ToString() + Path.GetExtension(file.FileName));

file.SaveAs(Server.MapPath(fileName));

user.ImagePath = fileName;

}

else

{

user.ImagePath = user.ImagePath;

}

var dp = db.Details.Where(u => u.UserId == user.UserId).ToList();

foreach(var d in dp)

{

db.Details.Remove(d);

}

foreach (var p in productId)

{

var product = db.Products.FirstOrDefault(f => f.ProductId == p);

Detail detail = new Detail()

{

User = user,

UserId = user.UserId,

ProductId = product.ProductId,

ProductName = product.ProductName,

};

db.Details.Add(detail);

}

db.SaveChanges();

return RedirectToAction("Index");

}

return View(vmodel);

}

public ActionResult Delete(int ?id)

{

if(id != null)

{

var user = db.Users.FirstOrDefault(u => u.UserId == id);

var detail = db.Details.Where(u => u.UserId == id).ToList();

foreach (var d in detail)

{

db.Details.Remove(d);

}

db.Users.Remove(user);

db.SaveChanges();

return RedirectToAction("Index");

}

return View("Index");

}

}

}

Partisial Page:

@model ShopDB.Models.Product

<div id="remove-product">

<div>

@Html.DropDownListFor(p=>p.ProductId,ViewBag.product as SelectList,"--selete product")

</div>

<div>

@Html.ActionLink("remove", "",null, new {id="removebtn"})

</div>

</div>

Index:

@model IEnumerable<ShopDB.Models.User>

@{

ViewBag.Title = "Index";

}

<h2>Index</h2>

<div>

@Html.ActionLink("Create","Create")

</div>

<table>

<thead>

<tr>

<th>

@Html.DisplayNameFor(u => u.UserId)

</th>

<th>

@Html.DisplayNameFor(u => u.ImagePath)

</th>

<th>

@Html.DisplayNameFor(u => u.UserName)

</th>

<th>

@Html.DisplayNameFor(u => u.Age)

</th>

<th>

@Html.DisplayNameFor(u => u.UserDate)

</th>

<th>

@Html.DisplayNameFor(u => u.IsAudil)

</th>

</tr>

</thead>

<tbody>

@foreach (var user in Model)

{

<tr>

<td>

@Html.DisplayFor(u => user.UserId)

</td>

<td>

<img src="@user.ImagePath" width="40" />

</td>

<td>

@Html.DisplayFor(u => user.UserName)

</td>

<td>

@Html.DisplayFor(u => user.Age)

</td>

<td>

@Html.DisplayFor(u => user.UserDate)

</td>

<td>

@Html.DisplayFor(u => user.IsAudil)

</td>

<td>

@Html.ActionLink("Edit", "Edit", new { id = user.UserId })

</td>

<td>

@Html.ActionLink("Delete", "Delete", new { id = user.UserId })

</td>

</tr>

<tr>

<td class="5">

<table>

<thead>

<tr>

<th>

@Html.Label("ProductId")

</th>

<th>

@Html.Label("ProductName")

</th>

</tr>

</thead>

<tbody>

@foreach (var p in user.Details)

{

<tr>

<td>

@Html.DisplayFor(u => p.ProductId)

</td>

<td>

@Html.DisplayFor(u => p.ProductName)

</td>

</tr>

}

</tbody>

</table>

</td>

</tr>

}

</tbody>

</table>

**Create Page:**

@model ShopDB.Models.Vm.Vmodel

@{

ViewBag.Title = "Create";

}

<h2>Create</h2>

<div>

@using (Html.BeginForm("Create", "User", FormMethod.Post, new { enctype = "multipart/form-data" }))

{

<div>

@Html.Label("UserName")

@Html.TextBoxFor(m => m.UserName)

</div>

<div>

@Html.Label("Image")

@Html.TextBoxFor(m => m.ImageFile, null, new { Type = "file" })

</div>

<div>

@Html.Label("Age")

@Html.TextBoxFor(m => m.Age, null, new { type = "number" })

</div>

<div>

@Html.Label("Date")

@Html.TextBoxFor(m => m.UserDate, null, new { Type = "date" })

</div>

<div>

@Html.Label("IsAdult")

@Html.EditorFor(m => m.IsAudil)

</div>

<div>

<div>

@Html.ActionLink("add More", "#", null, new { id = "addmore" })

</div>

<div id="container">

@Html.Action("addproduct", "User")

</div>

</div>

<div>

<input type="submit" value="Create" />

</div>

}

<div>

@Html.ActionLink("Back to List", "Index")

</div>

</div>

Edit Page:

@model ShopDB.Models.Vm.Vmodel

@{

ViewBag.Title = "Edit";

}

@using (Html.BeginForm("Edit", "User", FormMethod.Post, new { enctype = "multipart/form-data" }))

{

<div>

<h2>Edit Users</h2>

@Html.HiddenFor(u => u.UserId)

<div>

@Html.Label("UserName")

@Html.TextBoxFor(m => m.UserName)

</div>

<div>

@Html.Label("Image")

@Html.TextBoxFor(m => m.ImageFile, null, new { Type = "file" })

<img src="@Model.ImagePath" width="40" />

</div>

<div>

@Html.Label("Age")

@Html.TextBoxFor(m => m.Age, null, new { type = "number" })

</div>

<div>

@Html.Label("Date")

@Html.TextBox( "UserDate",Model.UserDate?.ToString("yyyy-MM-dd"), null, new { Type = "date" })

</div>

<div>

@Html.Label("IsAdult")

@Html.EditorFor(m => m.IsAudil)

</div>

<div>

<div>

@Html.ActionLink("add More", "#", null, new { id = "addmore" })

</div>

<div id="container">

@Html.Action("addproduct", "User")

</div>

</div>

<div>

<input type="submit" value="Update Data" />

</div>

</div>

}

<div>

@Html.ActionLink("Back to List", "Index")

</div>

View Model:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

namespace ShopDB.Models.Vm

{

public class Vmodel

{

public Vmodel()

{

this.Products = new List<Product>();

}

public int UserId { get; set; }

public string UserName { get; set; }

public string Age { get; set; }

public Nullable<System.DateTime> UserDate { get; set; }

public Nullable<bool> IsAudil { get; set; }

public string ImagePath { get; set; }

public HttpPostedFileBase ImageFile { get; set; }

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

}

}

Script Page:

<script type="text/javascript">

$(document).ready(() => {

$("#addmore").click((e) => {

e.preventDefault();

$.ajax({

url: "/User/addproduct",

type: "get",

success: (data) => {

$("#container").append(data);

},

error: () => {

alert("user not added");

}

})

})

$(document).on("click", "#removebtn", (e) => {

e.preventDefault();

$(e.currentTarget).closest("#remove-product").remove();

})

})

</script>