using System;

using System.Collections.Generic;

using System.Linq;

using System.Threading.Tasks;

using EntityFrameDotNetCoreDemo.Models;

using Microsoft.AspNetCore.Mvc;

namespace EntityFrameDotNetCoreDemo.Controllers

{

public class StudentController : Controller

{

StudentDbContext db;

public StudentController(StudentDbContext studentDbContext)

{

db = studentDbContext;

}

public IActionResult Index()

{

if(TempData["msg"] != null)

ViewBag.msg = TempData["msg"].ToString();

return View(db.Students.ToList());

}

public IActionResult Create()

{

Student student = new Student();

return View(student);

}

[HttpPost]

public IActionResult Create(Student student)

{

if (student != null)

{

db.Students.Add(student);

db.SaveChanges();

//ViewData["msg"] = "Record Inserted";

TempData["msg"] = "Record Inserted";

}

//return View();

return RedirectToAction("Index");

}

public IActionResult Delete(int id)

{

Student student = db.Students.FirstOrDefault(x => x.Id == id);

if(student!=null)

{ return View(student);

}

else

{

ViewBag.msg = "Record not found";

return View();

}

}

[HttpPost]

public IActionResult Delete(int id, Student student)

{

db.Students.Remove(student);

db.SaveChanges();

TempData["msg"] = "Record deleted";

return RedirectToAction("Index");

}

public IActionResult Display(int id)

{

Student student = db.Students.FirstOrDefault(x => x.Id == id);

if (student != null)

{

return View(student);

}

else

{

ViewBag.msg = "Record not found";

return View();

}

}

public IActionResult Edit(int id)

{

Student student = db.Students.FirstOrDefault(x => x.Id == id);

if (student != null)

{

return View(student);

}

else

{

ViewBag.msg = "Record not found";

return View();

}

}

[HttpPost]

public IActionResult Edit(int id, Student student)

{

foreach(Student temp in db.Students)

{

if(temp.Id == id)

{

temp.Name = student.Name;

temp.Address = student.Address;

temp.Marks = student.Marks;

break;

}

}

db.SaveChanges();

TempData["msg"] = "Record updated";

return RedirectToAction("Index");

}

}

}

Index View

@model IEnumerable<EntityFrameDotNetCoreDemo.Models.Student>

@{

ViewData["Title"] = "Index";

}

<h1>Index</h1>

<p>

<a **asp-action**="Create">Create New</a>

</p>

@{ if (ViewBag.msg != null)

{

@ViewBag.msg

}

}

<table class="table">

<thead>

<tr>

<th>

@Html.DisplayNameFor(model => model.Name)

</th>

<th>

@Html.DisplayNameFor(model => model.Address)

</th>

<th>

@Html.DisplayNameFor(model => model.Marks)

</th>

<th></th>

</tr>

</thead>

<tbody>

@foreach (var item in Model) {

<tr>

<td>

@Html.DisplayFor(modelItem => item.Name)

</td>

<td>

@Html.DisplayFor(modelItem => item.Address)

</td>

<td>

@Html.DisplayFor(modelItem => item.Marks)

</td>

<td>

<a **asp-action**="Edit" **asp-route-id**="@item.Id">Edit</a> |

<a **asp-action**="Details" **asp-route-id**="@item.Id">Details</a> |

<a **asp-action**="Delete" **asp-route-id**="@item.Id">Delete</a>

</td>

</tr>

}

</tbody>

</table>

Create

@model EntityFrameDotNetCoreDemo.Models.Student

@{

ViewData["Title"] = "Create";

}

<h1>Create</h1>

<h4>Student</h4>

<hr />

@ViewBag.msg

<div class="row">

<div class="col-md-4">

<form **asp-action**="Create">

<div **asp-validation-summary**="ModelOnly" class="text-danger"></div>

<div class="form-group">

<label **asp-for**="Name" class="control-label"></label>

<input **asp-for**="Name" class="form-control" />

<span **asp-validation-for**="Name" class="text-danger"></span>

</div>

<div class="form-group">

<label **asp-for**="Address" class="control-label"></label>

<input **asp-for**="Address" class="form-control" />

<span **asp-validation-for**="Address" class="text-danger"></span>

</div>

<div class="form-group">

<label **asp-for**="Marks" class="control-label"></label>

<input **asp-for**="Marks" class="form-control" />

<span **asp-validation-for**="Marks" class="text-danger"></span>

</div>

<div class="form-group">

<input type="submit" value="Create" class="btn btn-primary" />

</div>

</form>

</div>

</div>

<div>

<a **asp-action**="Index">Back to List</a>

</div>

@section Scripts {

@{await Html.RenderPartialAsync("\_ValidationScriptsPartial");}

}

Display

@model EntityFrameDotNetCoreDemo.Models.Student

@{

ViewData["Title"] = "Display";

}

<h1>Display</h1>

@{ if (ViewBag.msg != null)

{

<h1> @ViewBag.msg </h1>

}

else

{

<div>

<h4>Student</h4>

<hr />

<dl class="row">

<dt class = "col-sm-2">

@Html.DisplayNameFor(model => model.Name)

</dt>

<dd class = "col-sm-10">

@Html.DisplayFor(model => model.Name)

</dd>

<dt class = "col-sm-2">

@Html.DisplayNameFor(model => model.Address)

</dt>

<dd class = "col-sm-10">

@Html.DisplayFor(model => model.Address)

</dd>

<dt class = "col-sm-2">

@Html.DisplayNameFor(model => model.Marks)

</dt>

<dd class = "col-sm-10">

@Html.DisplayFor(model => model.Marks)

</dd>

</dl>

</div>

<div>

<a **asp-action**="Edit" **asp-route-id**="@Model.Id">Edit</a> |

<a **asp-action**="Index">Back to List</a>

</div>

}

}

Edit

@model EntityFrameDotNetCoreDemo.Models.Student

@{

ViewData["Title"] = "Edit";

}

@{ if (ViewBag.msg != null)

{

<h1> @ViewBag.msg </h1>

}

else

{

<h1>Edit</h1>

<h4>Student</h4>

<hr />

<div class="row">

<div class="col-md-4">

<form **asp-action**="Edit">

<div **asp-validation-summary**="ModelOnly" class="text-danger"></div>

<input **type**="hidden" **asp-for**="Id" />

<div class="form-group">

<label **asp-for**="Name" class="control-label"></label>

<input **asp-for**="Name" class="form-control" />

<span **asp-validation-for**="Name" class="text-danger"></span>

</div>

<div class="form-group">

<label **asp-for**="Address" class="control-label"></label>

<input **asp-for**="Address" class="form-control" />

<span **asp-validation-for**="Address" class="text-danger"></span>

</div>

<div class="form-group">

<label **asp-for**="Marks" class="control-label"></label>

<input **asp-for**="Marks" class="form-control" />

<span **asp-validation-for**="Marks" class="text-danger"></span>

</div>

<div class="form-group">

<input type="submit" value="Save" class="btn btn-primary" />

</div>

</form>

</div>

</div>

<div>

<a **asp-action**="Index">Back to List</a>

</div>

@section Scripts {

@{await Html.RenderPartialAsync("\_ValidationScriptsPartial");}

}

}

}

Delete

@model EntityFrameDotNetCoreDemo.Models.Student

@{

ViewData["Title"] = "Delete";

}

@{ if (ViewBag.msg != null)

{

<h1> @ViewBag.msg </h1>

}

else

{

<h1>Delete</h1>

<h3>Are you sure you want to delete this?</h3>

<div>

<h4>Student</h4>

<hr />

<dl class="row">

<dt class="col-sm-2">

@Html.DisplayNameFor(model => model.Name)

</dt>

<dd class="col-sm-10">

@Html.DisplayFor(model => model.Name)

</dd>

<dt class="col-sm-2">

@Html.DisplayNameFor(model => model.Address)

</dt>

<dd class="col-sm-10">

@Html.DisplayFor(model => model.Address)

</dd>

<dt class="col-sm-2">

@Html.DisplayNameFor(model => model.Marks)

</dt>

<dd class="col-sm-10">

@Html.DisplayFor(model => model.Marks)

</dd>

</dl>

<form **asp-action**="Delete">

<input **type**="hidden" **asp-for**="Id" />

<input type="submit" value="Delete" class="btn btn-danger" /> |

<a **asp-action**="Index">Back to List</a>

</form>

</div>

}

}