﻿using EmployeeCRUDDemonstration.Models;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

using System.Web.Mvc;

namespace EmployeeCRUDDemonstration.Controllers

{

public class HomeController : Controller

{

// GET: Home

public ActionResult Index()

{

EmployeeRepository repo = new EmployeeRepository();

var list = repo.GetAllEmployees();

return View(list);

}

public ActionResult CreateEmployee()

{

return View();

}

[HttpPost]

public ActionResult CreateEmployee(Employee emp)

{

if (ModelState.IsValid)

{

EmployeeRepository repo = new EmployeeRepository();

bool isCreated = false;

string errorMessageIfAny = "";

try

{

isCreated = repo.Create(emp);

}

catch(Exception ex)

{

errorMessageIfAny = ex.Message;

}

if (isCreated)

{

TempData["ResultMessage"] = "New Employee is Created Successfully";

return RedirectToAction("Index");

}

else

{

ModelState.AddModelError("InsertFailed", "New Employee Insertion is failed, "+ errorMessageIfAny);

}

}

return View();

}

public ActionResult DeleteEmployee(int Id)

{

EmployeeRepository repo = new EmployeeRepository();

bool isDeleted = repo.Delete(Id);

if (isDeleted)

{

TempData["ResultMessage"] = "Record is Deleted Successfully";

return RedirectToAction("Index");

}

else

{

return View();

}

}

public ActionResult EditEmployee(int Id)

{

EmployeeRepository repo = new EmployeeRepository();

var emp = repo.GetEmployeeById(Id);

return View(emp);

}

[HttpPost]

public ActionResult EditEmployee(Employee emp)

{

if (ModelState.IsValid)

{

EmployeeRepository repo = new EmployeeRepository();

bool isUpdated = repo.Update(emp);

if (isUpdated)

{

return RedirectToAction("Index");

}

else

{

ModelState.AddModelError("UpdateFailed", "Update Failed");

}

}

return View();

}

}

}