**Source Code for Create an MVC Application to Manage Data for School Application**

**Classess.cs:**

using System;

using System.Collections.Generic;

using System.ComponentModel.DataAnnotations.Schema;

using System.ComponentModel.DataAnnotations;

using System.Linq;

using System.Web;

namespace WenAppPracticeSec\_4.Models

{

[Table("ClTable")]

public class Classes

{

[Key]

public int ClassId { get; set; }

[Required]

public string ClassName { get; set; }

public virtual Student Student { get; set; }

}

}

**Student.cs:**

using System;

using System.Collections.Generic;

using System.ComponentModel.DataAnnotations;

using System.ComponentModel.DataAnnotations.Schema;

using System.Linq;

using System.Web;

namespace WenAppPracticeSec\_4.Models

{

[Table("StTable")]

public class Student

{

[Key]

public int StudentId { get; set; }

[Required]

public string FirstName { get; set; }

[Required]

public string LastName { get; set; }

[Required]

public DateTime BirthDate { get; set; }

public int ClassId { get; set; }

public virtual ICollection<Classes> Classes { get; set; }

public virtual ICollection<Subject> Subject { get; set; }

}

}

**Subject.cs**

using System;

using System.Collections.Generic;

using System.ComponentModel.DataAnnotations.Schema;

using System.ComponentModel.DataAnnotations;

using System.Linq;

using System.Web;

namespace WenAppPracticeSec\_4.Models

{

[Table("SubTable")]

public class Subject

{

[Key]

public int SubjectId { get; set; }

[Required]

public string SubjectName { get; set; }

public virtual Student Student { get; set; }

}

}

**ClassessControllers:**

using System;

using System.Collections.Generic;

using System.Data;

using System.Data.Entity;

using System.Linq;

using System.Net;

using System.Web;

using System.Web.Mvc;

using WenAppPracticeSec\_4.Data;

using WenAppPracticeSec\_4.Models;

namespace WenAppPracticeSec\_4.Controllers

{

public class ClassesController : Controller

{

private SchoolDbContext db = new SchoolDbContext();

// GET: Classes

public ActionResult Index()

{

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

}

// GET: Classes/Details/5

public ActionResult Details(int? id)

{

if (id == null)

{

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

}

Classes classes = db.Classes.Find(id);

if (classes == null)

{

return HttpNotFound();

}

return View(classes);

}

// GET: Classes/Create

public ActionResult Create()

{

return View();

}

// POST: Classes/Create

// To protect from overposting attacks, enable the specific properties you want to bind to, for

// more details see https://go.microsoft.com/fwlink/?LinkId=317598.

[HttpPost]

[ValidateAntiForgeryToken]

public ActionResult Create([Bind(Include = "ClassId,ClassName")] Classes classes)

{

if (ModelState.IsValid)

{

db.Classes.Add(classes);

db.SaveChanges();

return RedirectToAction("Index");

}

return View(classes);

}

// GET: Classes/Edit/5

public ActionResult Edit(int? id)

{

if (id == null)

{

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

}

Classes classes = db.Classes.Find(id);

if (classes == null)

{

return HttpNotFound();

}

return View(classes);

}

// POST: Classes/Edit/5

// To protect from overposting attacks, enable the specific properties you want to bind to, for

// more details see https://go.microsoft.com/fwlink/?LinkId=317598.

[HttpPost]

[ValidateAntiForgeryToken]

public ActionResult Edit([Bind(Include = "ClassId,ClassName")] Classes classes)

{

if (ModelState.IsValid)

{

db.Entry(classes).State = EntityState.Modified;

db.SaveChanges();

return RedirectToAction("Index");

}

return View(classes);

}

// GET: Classes/Delete/5

public ActionResult Delete(int? id)

{

if (id == null)

{

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

}

Classes classes = db.Classes.Find(id);

if (classes == null)

{

return HttpNotFound();

}

return View(classes);

}

// POST: Classes/Delete/5

[HttpPost, ActionName("Delete")]

[ValidateAntiForgeryToken]

public ActionResult DeleteConfirmed(int id)

{

Classes classes = db.Classes.Find(id);

db.Classes.Remove(classes);

db.SaveChanges();

return RedirectToAction("Index");

}

protected override void Dispose(bool disposing)

{

if (disposing)

{

db.Dispose();

}

base.Dispose(disposing);

}

}

}

**SubjectsController:**

using System;

using System.Collections.Generic;

using System.Data;

using System.Data.Entity;

using System.Linq;

using System.Net;

using System.Web;

using System.Web.Mvc;

using WenAppPracticeSec\_4.Data;

using WenAppPracticeSec\_4.Models;

namespace WenAppPracticeSec\_4.Controllers

{

public class SubjectsController : Controller

{

private SchoolDbContext db = new SchoolDbContext();

// GET: Subjects

public ActionResult Index()

{

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

}

// GET: Subjects/Details/5

public ActionResult Details(int? id)

{

if (id == null)

{

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

}

Subject subject = db.Subjects.Find(id);

if (subject == null)

{

return HttpNotFound();

}

return View(subject);

}

// GET: Subjects/Create

public ActionResult Create()

{

return View();

}

// POST: Subjects/Create

// To protect from overposting attacks, enable the specific properties you want to bind to, for

// more details see https://go.microsoft.com/fwlink/?LinkId=317598.

[HttpPost]

[ValidateAntiForgeryToken]

public ActionResult Create([Bind(Include = "SubjectId,SubjectName")] Subject subject)

{

if (ModelState.IsValid)

{

db.Subjects.Add(subject);

db.SaveChanges();

return RedirectToAction("Index");

}

return View(subject);

}

// GET: Subjects/Edit/5

public ActionResult Edit(int? id)

{

if (id == null)

{

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

}

Subject subject = db.Subjects.Find(id);

if (subject == null)

{

return HttpNotFound();

}

return View(subject);

}

// POST: Subjects/Edit/5

// To protect from overposting attacks, enable the specific properties you want to bind to, for

// more details see https://go.microsoft.com/fwlink/?LinkId=317598.

[HttpPost]

[ValidateAntiForgeryToken]

public ActionResult Edit([Bind(Include = "SubjectId,SubjectName")] Subject subject)

{

if (ModelState.IsValid)

{

db.Entry(subject).State = EntityState.Modified;

db.SaveChanges();

return RedirectToAction("Index");

}

return View(subject);

}

// GET: Subjects/Delete/5

public ActionResult Delete(int? id)

{

if (id == null)

{

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

}

Subject subject = db.Subjects.Find(id);

if (subject == null)

{

return HttpNotFound();

}

return View(subject);

}

// POST: Subjects/Delete/5

[HttpPost, ActionName("Delete")]

[ValidateAntiForgeryToken]

public ActionResult DeleteConfirmed(int id)

{

Subject subject = db.Subjects.Find(id);

db.Subjects.Remove(subject);

db.SaveChanges();

return RedirectToAction("Index");

}

protected override void Dispose(bool disposing)

{

if (disposing)

{

db.Dispose();

}

base.Dispose(disposing);

}

}

}

**StudentController:**

using System;

using System.Collections.Generic;

using System.Data;

using System.Data.Entity;

using System.Linq;

using System.Net;

using System.Web;

using System.Web.Mvc;

using WenAppPracticeSec\_4.Data;

using WenAppPracticeSec\_4.Models;

namespace WenAppPracticeSec\_4.Controllers

{

public class StudentsController : Controller

{

private SchoolDbContext db = new SchoolDbContext();

// GET: Students

public ActionResult Index()

{

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

}

// GET: Students/Details/5

public ActionResult Details(int? id)

{

if (id == null)

{

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

}

Student student = db.Students.Find(id);

if (student == null)

{

return HttpNotFound();

}

return View(student);

}

// GET: Students/Create

public ActionResult Create()

{

return View();

}

// POST: Students/Create

// To protect from overposting attacks, enable the specific properties you want to bind to, for

// more details see https://go.microsoft.com/fwlink/?LinkId=317598.

[HttpPost]

[ValidateAntiForgeryToken]

public ActionResult Create([Bind(Include = "StudentId,FirstName,LastName,BirthDate,ClassId")] Student student)

{

if (ModelState.IsValid)

{

db.Students.Add(student);

db.SaveChanges();

return RedirectToAction("Index");

}

return View(student);

}

// GET: Students/Edit/5

public ActionResult Edit(int? id)

{

if (id == null)

{

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

}

Student student = db.Students.Find(id);

if (student == null)

{

return HttpNotFound();

}

return View(student);

}

// POST: Students/Edit/5

// To protect from overposting attacks, enable the specific properties you want to bind to, for

// more details see https://go.microsoft.com/fwlink/?LinkId=317598.

[HttpPost]

[ValidateAntiForgeryToken]

public ActionResult Edit([Bind(Include = "StudentId,FirstName,LastName,BirthDate,ClassId")] Student student)

{

if (ModelState.IsValid)

{

db.Entry(student).State = EntityState.Modified;

db.SaveChanges();

return RedirectToAction("Index");

}

return View(student);

}

// GET: Students/Delete/5

public ActionResult Delete(int? id)

{

if (id == null)

{

return new HttpStatusCodeResult(HttpStatusCode.BadRequest);

}

Student student = db.Students.Find(id);

if (student == null)

{

return HttpNotFound();

}

return View(student);

}

// POST: Students/Delete/5

[HttpPost, ActionName("Delete")]

[ValidateAntiForgeryToken]

public ActionResult DeleteConfirmed(int id)

{

Student student = db.Students.Find(id);

db.Students.Remove(student);

db.SaveChanges();

return RedirectToAction("Index");

}

protected override void Dispose(bool disposing)

{

if (disposing)

{

db.Dispose();

}

base.Dispose(disposing);

}

}

}

**SchoolDbContext:**

using System;

using System.Collections.Generic;

using System.Data.Entity;

using System.Linq;

using System.Web;

namespace WenAppPracticeSec\_4.Data

{

public class SchoolDbContext : DbContext

{

// You can add custom code to this file. Changes will not be overwritten.

//

// If you want Entity Framework to drop and regenerate your database

// automatically whenever you change your model schema, please use data migrations.

// For more information refer to the documentation:

// http://msdn.microsoft.com/en-us/data/jj591621.aspx

public SchoolDbContext() : base("name=SchoolDbContext")

{

}

public System.Data.Entity.DbSet<WenAppPracticeSec\_4.Models.Classes> Classes { get; set; }

public System.Data.Entity.DbSet<WenAppPracticeSec\_4.Models.Student> Students { get; set; }

public System.Data.Entity.DbSet<WenAppPracticeSec\_4.Models.Subject> Subjects { get; set; }

}

}

**LayOut.cshtml:**

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8" />

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>@ViewBag.Title - My ASP.NET Application</title>

@Styles.Render("~/Content/css")

@Scripts.Render("~/bundles/modernizr")

</head>

<body>

<nav class="navbar navbar-expand-sm navbar-toggleable-sm navbar-dark bg-dark">

<div class="container">

@Html.ActionLink("Application name", "Index", "Home", new { area = "" }, new { @class = "navbar-brand" })

<button type="button" class="navbar-toggler" data-bs-toggle="collapse" data-bs-target=".navbar-collapse" title="Toggle navigation" aria-controls="navbarSupportedContent"

aria-expanded="false" aria-label="Toggle navigation">

<span class="navbar-toggler-icon"></span>

</button>

<div class="collapse navbar-collapse d-sm-inline-flex justify-content-between">

<ul class="navbar-nav flex-grow-1">

<li>@Html.ActionLink("Home", "Index", "Home", new { area = "" }, new { @class = "nav-link" })</li>

<li>@Html.ActionLink("Classes", "Index", "Classes", new { area = "" }, new { @class = "nav-link" })</li>

<li>@Html.ActionLink("Student", "Index", "Students", new { area = "" }, new { @class = "nav-link" })</li>

<li>@Html.ActionLink("Subject", "Index", "Subjects", new { area = "" }, new { @class = "nav-link" })</li>

<li>@Html.ActionLink("About", "About", "Home", new { area = "" }, new { @class = "nav-link" })</li>

<li>@Html.ActionLink("Contact", "Contact", "Home", new { area = "" }, new { @class = "nav-link" })</li>

</ul>

</div>

</div>

</nav>

<div class="container body-content">

@RenderBody()

<hr />

<footer>

<p>&copy; @DateTime.Now.Year - My ASP.NET Application</p>

</footer>

</div>

@Scripts.Render("~/bundles/jquery")

@Scripts.Render("~/bundles/bootstrap")

@RenderSection("scripts", required: false)

</body>

</html>