Student.cs

```
using System;
using System.Collections.Generic;
using System.ComponentModel.DataAnnotations;
using System.ComponentModel.DataAnnotations.Schema;
using System.Linq;
using System.Web;

namespace WebApplication1.Models
{
    [Table("Student")]
    public class Student
    {
        [Key]
        public int Id { get; set; }
        [Required]
        public string Name { get; set; }
        [Required]

        public DateTime BirthDate { get; set; }
        public string FatherName { get; set; }
        [Required]
        public double MobileNumber { get; set;}
        public virtual ICollection<Classes> classes { get; set; }
        public virtual ICollection<Subject> Subject { get; set; }

    }
}
```

Classes.cs

```
using System;
using System.Collections.Generic;
using System.ComponentModel.DataAnnotations;
using System.ComponentModel.DataAnnotations.Schema;
using System.Linq;
using System.Web;

namespace WebApplication1.Models
{
    [Table("Table")]
    public class Classes
    {
        [Key]
        public int CId { get; set; }
        [Required]
        public string Class { get; set; }
        [Required]
        public string Section { get; set; }
        public int StudentId { get; set; }
        public virtual Student Student { get; set; }

    }
}
```

Subject.cs

```
using System;
using System.Collections.Generic;
using System.ComponentModel.DataAnnotations;
using System.ComponentModel.DataAnnotations.Schema;
using System.Linq;
using System.Web;

namespace WebApplication1.Models
{
    [Table("Subject")]
    public class Subject
    {
        [Key]
        public int SId { get; set; }
        [Required]
        public string SubjectName {  get; set; }
        [Required]
        public int Score { get; set; }

        public int StudentName { get; set; }
        public virtual Student Student { get; set; }

    }
}
```

StudentsController.cs

```
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 WebApplication1.Data;
using WebApplication1.Models;

namespace WebApplication1.Controllers
{
    public class StudentsController : Controller
    {
        private WebAppliDbContext db = new WebAppliDbContext();

        // 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 =
"Id,Name,BirthDate,FatherName,MobileNumber")] 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 = "Id,Name,BirthDate,FatherName,MobileNumber")]
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);
        }
    }
}
```

ClassesController.cs

```
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 WebApplication1.Data;
using WebApplication1.Models;

namespace WebApplication1.Controllers
{
    public class ClassesController : Controller
    {
        private WebAppliDbContext db = new WebAppliDbContext();

        // GET: Classes
        public ActionResult Index()
        {
            var classes = db.Classes.Include(c => c.Student);
            return View(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()
        {
            ViewBag.StudentId = new SelectList(db.Students, "Id", "Name");
            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 = "CId,Class,Section,StudentId")] Classes
classes)
        {
            if (ModelState.IsValid)
            {
                db.Classes.Add(classes);
                db.SaveChanges();
                return RedirectToAction("Index");
            }

            ViewBag.StudentId = new SelectList(db.Students, "Id", "Name", classes.StudentId);
            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();
            }
            ViewBag.StudentId = new SelectList(db.Students, "Id", "Name", classes.StudentId);
            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 = "CId,Class,Section,StudentId")] Classes
classes)
        {
            if (ModelState.IsValid)
            {
                db.Entry(classes).State = EntityState.Modified;
                db.SaveChanges();
                return RedirectToAction("Index");
            }
            ViewBag.StudentId = new SelectList(db.Students, "Id", "Name", classes.StudentId);
            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);
        }
    }
}
```

SubjectsControllers.cs

```
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 WebApplication1.Data;
using WebApplication1.Models;

namespace WebApplication1.Controllers
{
    public class SubjectsController : Controller
    {
        private WebAppliDbContext db = new WebAppliDbContext();

        // 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 = "SId,SubjectName,Score,StudentName")]
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 = "SId,SubjectName,Score,StudentName")] 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);
        }
    }
}
```