Create an MVC Application to Manage Data for School Application

DESCRIPTION

Using ASP.NET MVC project to create a web based application, which will allow an admin to do CRUD operations on Student, Subject and Classe tables in the database

Background of the problem statement:

Rainbow School is creating software for school management. After creating the database, the first step is to populate the database using a web-based application.

You must use the following:

Visual Studio 2019 Community Edition

SQL Server 17 Express Edition

Following requirements should be met:

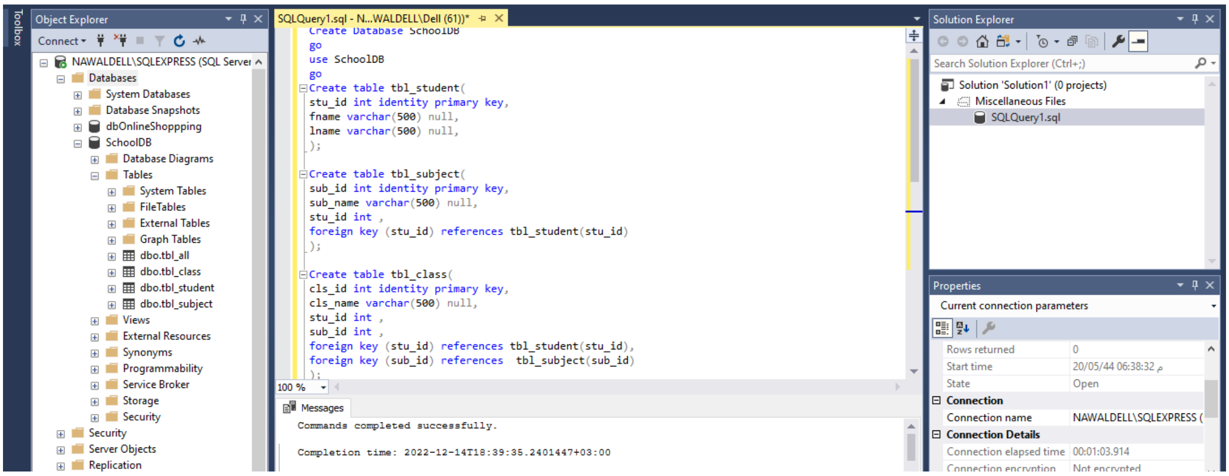
Some of the source code should be tracked on GitHub repositories. You need to document the tracked files that are ignored during the final push to the GitHub repository.

The submission of your GitHub repository link is mandatory. In order to track your task, you need to share the link of the repository in the document.

The step-by-step process involved in completing this task should be documented.

Source Code

1- DB

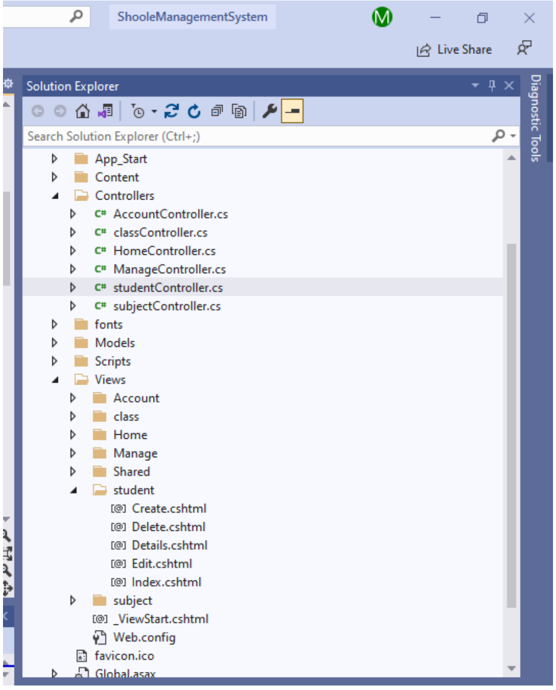


```sql
Create Database SchoolDB
go
use SchoolDB
go
Create table tbl_student(
stu_id int identity primary key,
fname varchar(500) null,
lname varchar(500) null,
);

Create table tbl_subject(
sub_id int identity primary key,
sub_name varchar(500) null,
stu_id int ,
foreign key (stu_id) references tbl_student(stu_id)
);

Create table tbl_class(
cls_id int identity primary key,
cls_name varchar(500) null,
stu_id int ,
sub_id int ,
foreign key (stu_id) references tbl_student(stu_id),
foreign key (sub_id) references  tbl_subject(sub_id)
);
```

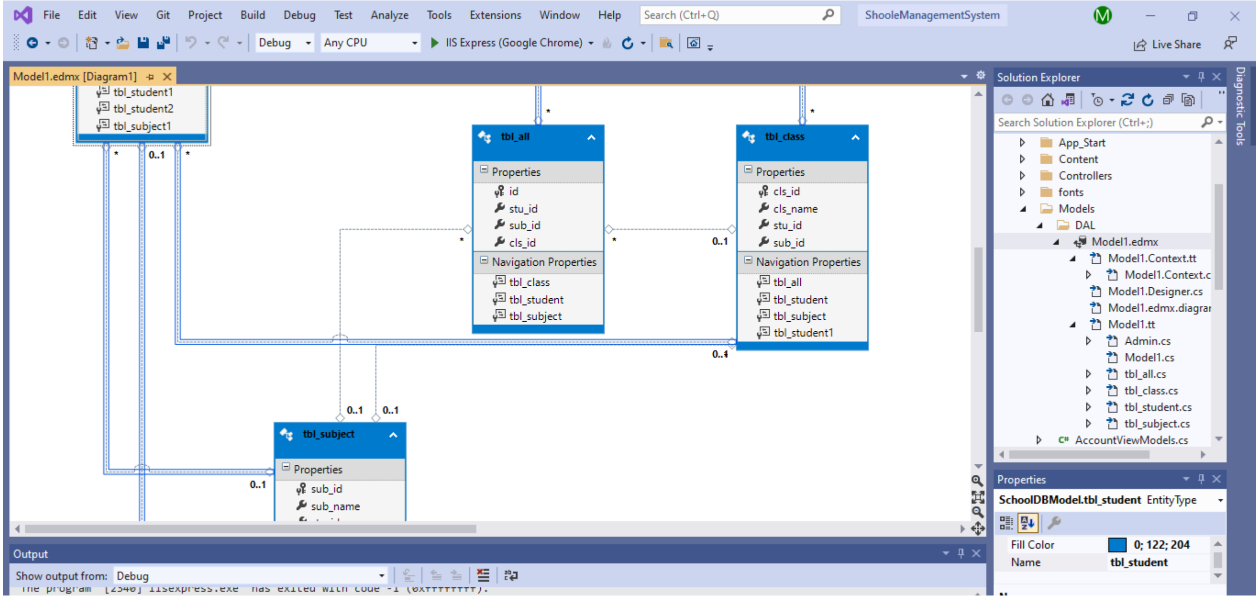
```
Commands completed successfully.

Completion time: 2022-12-14T18:39:35.2401447+03:00
```

Rows returned 0
Start time 20/05/44 06:38:32 م
State Open
Connection
Connection name NAWALDELL\SQLEXPRESS (
Connection Details
Connection elapsed time 00:01:03.914
Connection encryption Not encrypted

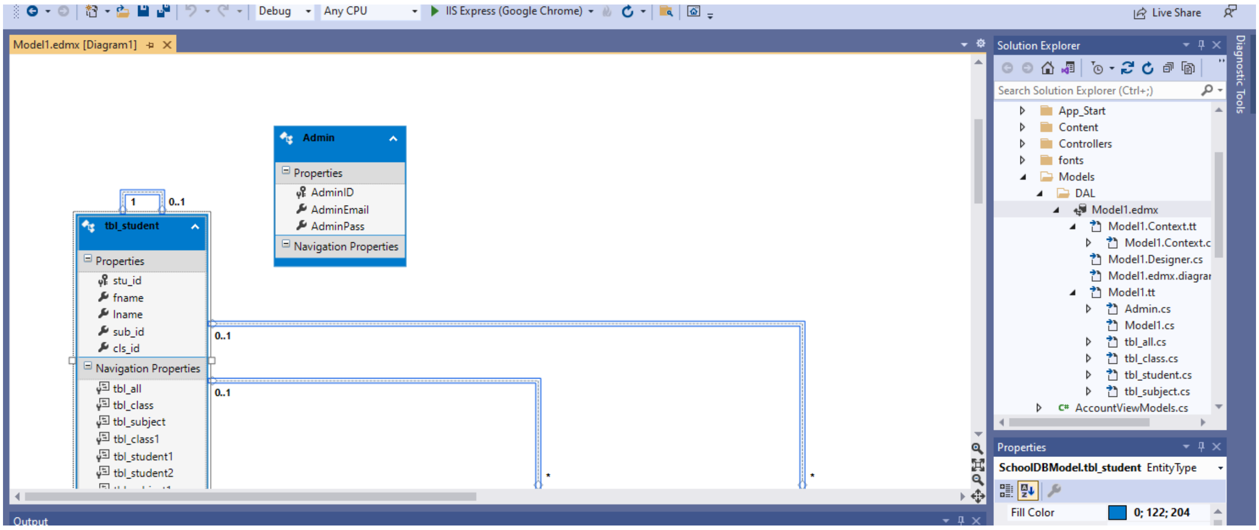
2- The files of Project



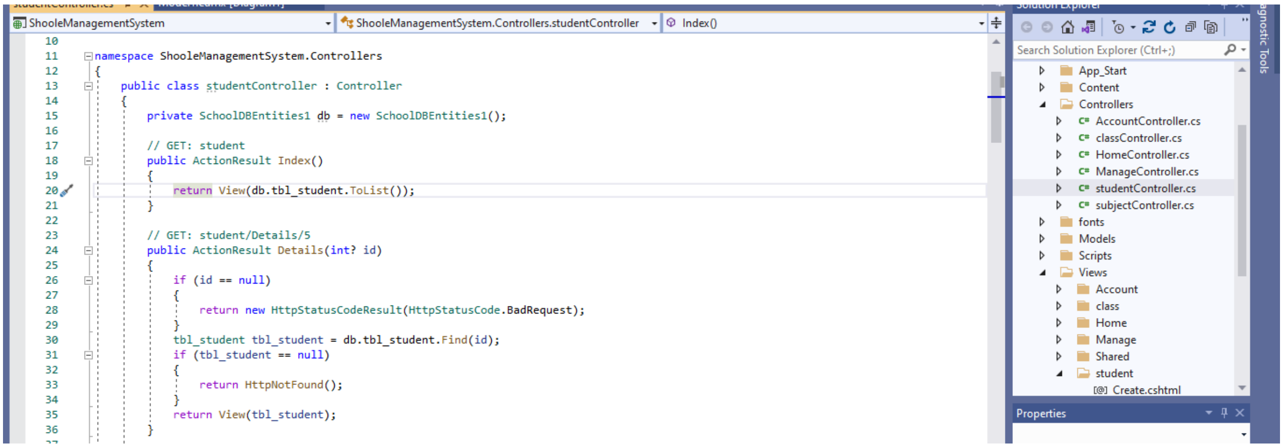
3- DB diagram



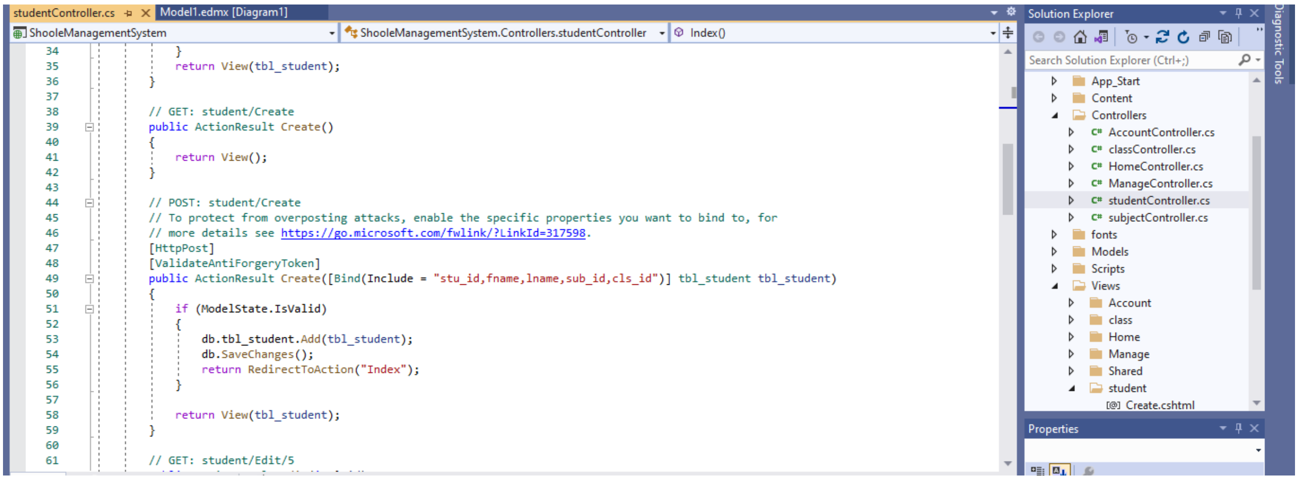
```
The program '[2540] iisexpress.exe' has exited with code -1 (0xffffffff).
```



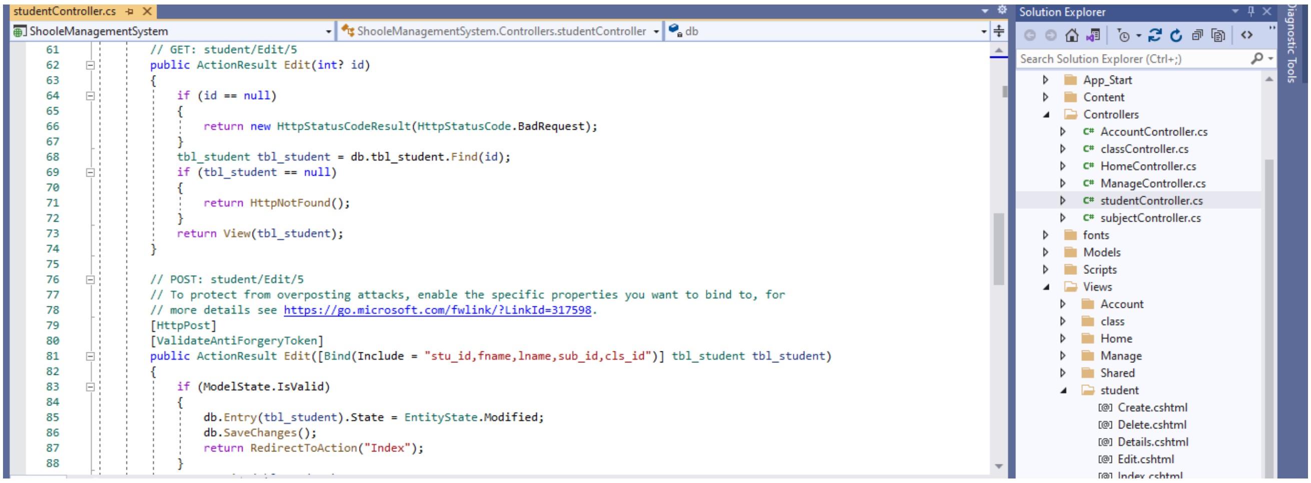
4- Student CDUD – controller



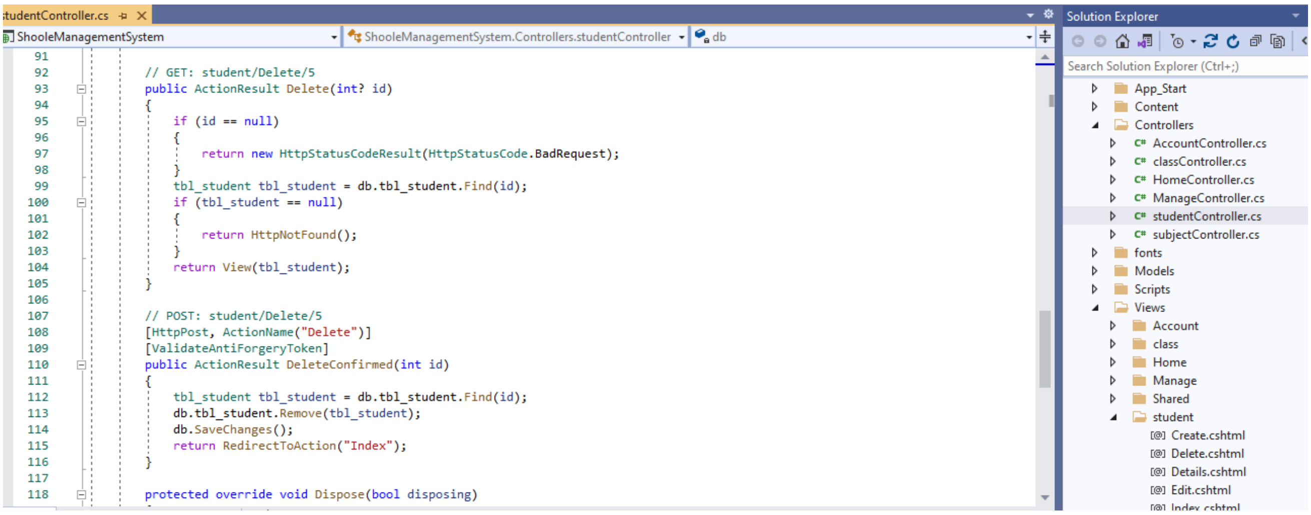
```
namespace ShooleManagementSystem.Controllers
{
    public class studentController : Controller
    {
        private SchoolDBEntities1 db = new SchoolDBEntities1();

        // GET: student
        public ActionResult Index()
        {
            return View(db.tbl_student.ToList());
        }

        // GET: student/Details/5
        public ActionResult Details(int? id)
        {
            if (id == null)
            {
                return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
            }
            tbl_student tbl_student = db.tbl_student.Find(id);
            if (tbl_student == null)
            {
                return HttpNotFound();
            }
            return View(tbl_student);
        }
```



```
            }
            return View(tbl_student);
        }

        // GET: student/Create
        public ActionResult Create()
        {
            return View();
        }

        // POST: student/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 = "stu_id,fname,lname,sub_id,cls_id")] tbl_student tbl_student)
        {
            if (ModelState.IsValid)
            {
                db.tbl_student.Add(tbl_student);
                db.SaveChanges();
                return RedirectToAction("Index");
            }

            return View(tbl_student);
        }

        // GET: student/Edit/5
```

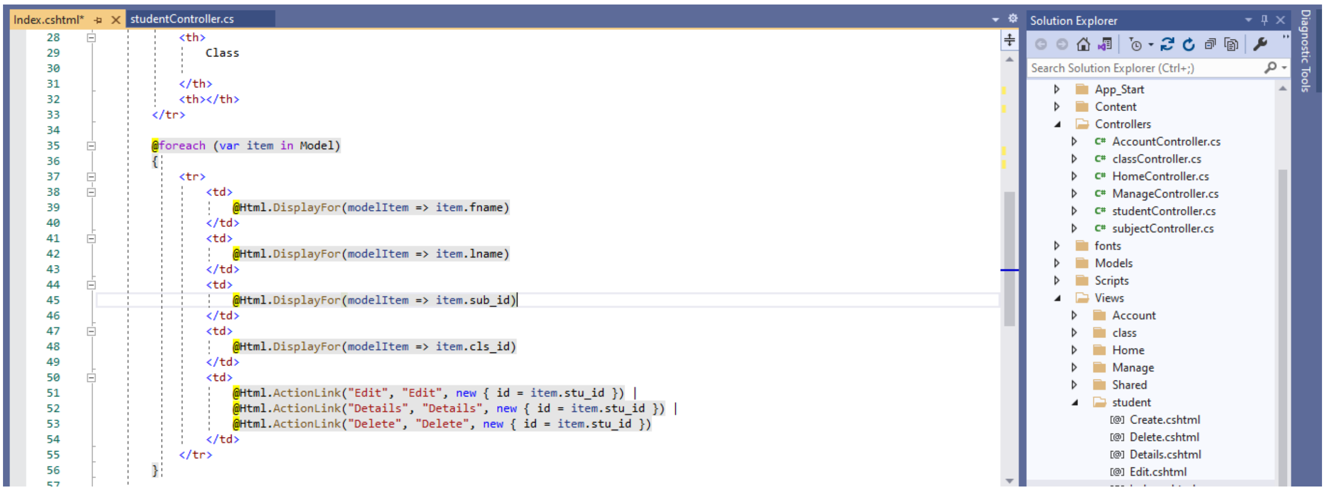


```
        // GET: student/Edit/5
        public ActionResult Edit(int? id)
        {
            if (id == null)
            {
                return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
            }
            tbl_student tbl_student = db.tbl_student.Find(id);
            if (tbl_student == null)
            {
                return HttpNotFound();
            }
            return View(tbl_student);
        }

        // POST: student/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 = "stu_id,fname,lname,sub_id,cls_id")] tbl_student tbl_student)
        {
            if (ModelState.IsValid)
            {
                db.Entry(tbl_student).State = EntityState.Modified;
                db.SaveChanges();
                return RedirectToAction("Index");
            }
```



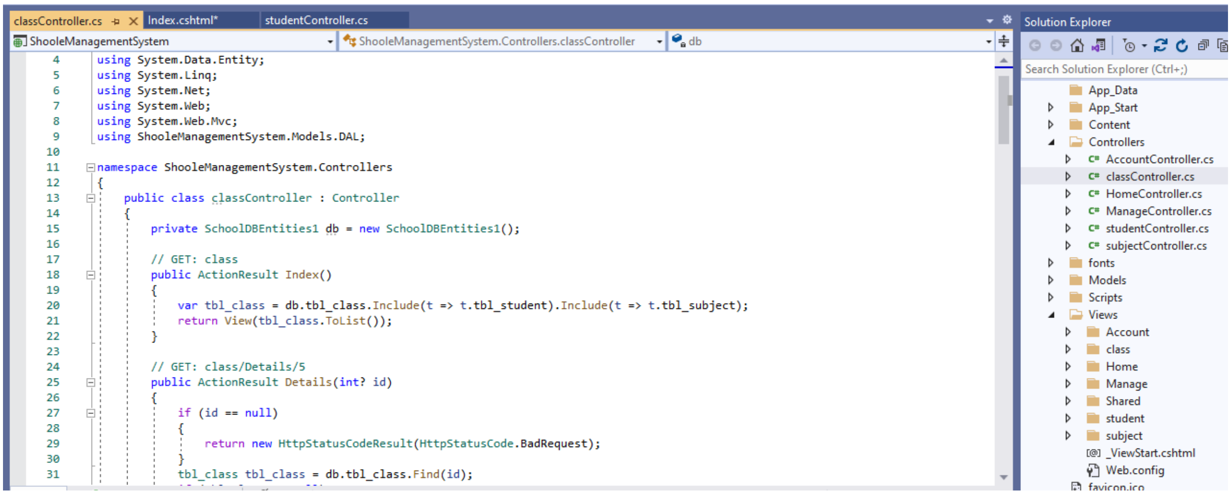
```

        // GET: student/Delete/5
        public ActionResult Delete(int? id)
        {
            if (id == null)
            {
                return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
            }
            tbl_student tbl_student = db.tbl_student.Find(id);
            if (tbl_student == null)
            {
                return HttpNotFound();
            }
            return View(tbl_student);
        }

        // POST: student/Delete/5
        [HttpPost, ActionName("Delete")]
        [ValidateAntiForgeryToken]
        public ActionResult DeleteConfirmed(int id)
        {
            tbl_student tbl_student = db.tbl_student.Find(id);
            db.tbl_student.Remove(tbl_student);
            db.SaveChanges();
            return RedirectToAction("Index");
        }

        protected override void Dispose(bool disposing)
```

Student – index page



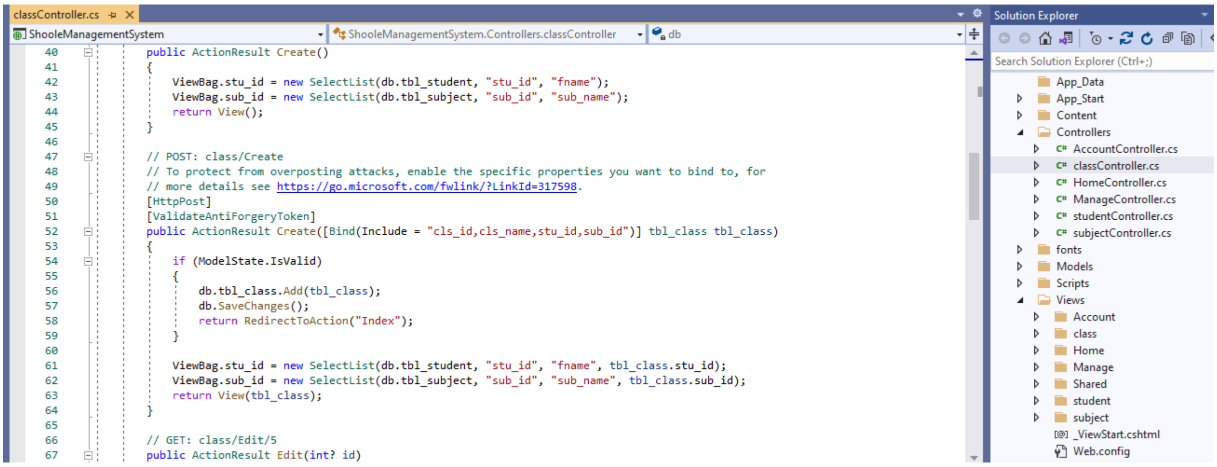
```
        <th>
            Class

        </th>
        <th></th>
    </tr>

@foreach (var item in Model)
{
    <tr>
        <td>
            @Html.DisplayFor(modelItem => item.fname)
        </td>
        <td>
            @Html.DisplayFor(modelItem => item.lname)
        </td>
        <td>
            @Html.DisplayFor(modelItem => item.sub_id)
        </td>
        <td>
            @Html.DisplayFor(modelItem => item.cls_id)
        </td>
        <td>
            @Html.ActionLink("Edit", "Edit", new { id = item.stu_id }) |
            @Html.ActionLink("Details", "Details", new { id = item.stu_id }) |
            @Html.ActionLink("Delete", "Delete", new { id = item.stu_id })
        </td>
    </tr>
}
```

5- Class – CRUD Controller



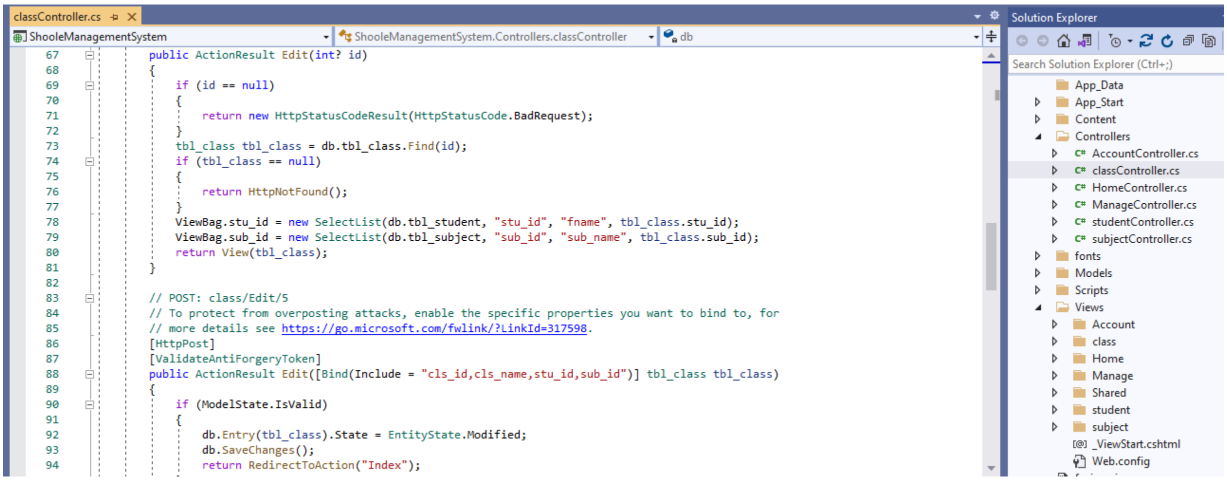
```
using System.Data.Entity;
using System.Linq;
using System.Net;
using System.Web;
using System.Web.Mvc;
using ShooleManagementSystem.Models.DAL;

namespace ShooleManagementSystem.Controllers
{
    public class classController : Controller
    {
        private SchoolDBEntities1 db = new SchoolDBEntities1();

        // GET: class
        public ActionResult Index()
        {
            var tbl_class = db.tbl_class.Include(t => t.tbl_student).Include(t => t.tbl_subject);
            return View(tbl_class.ToList());
        }

        // GET: class/Details/5
        public ActionResult Details(int? id)
        {
            if (id == null)
            {
                return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
            }
            tbl_class tbl_class = db.tbl_class.Find(id);
```

Class – create view



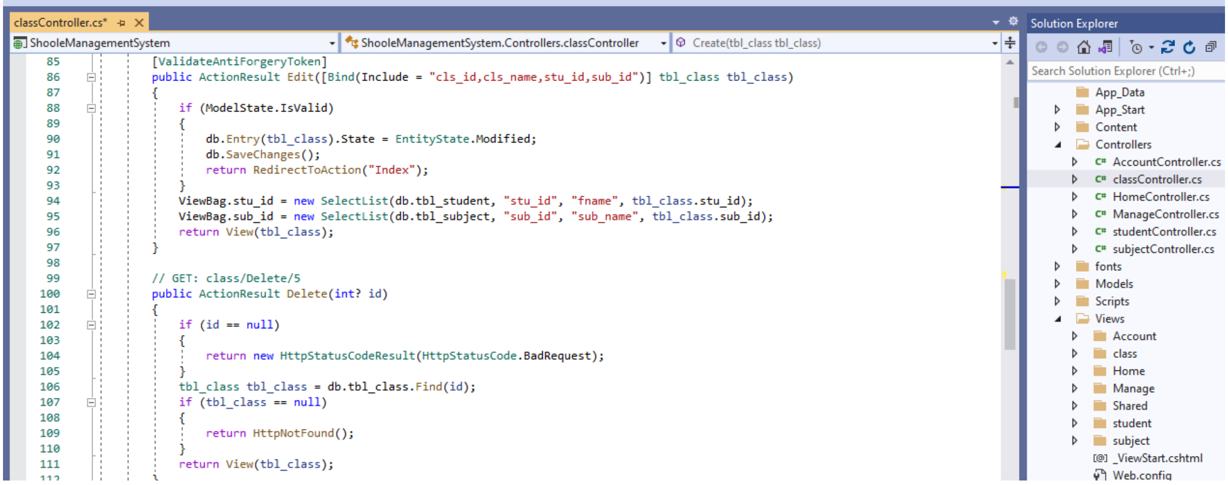
```
        public ActionResult Create()
        {
            ViewBag.stu_id = new SelectList(db.tbl_student, "stu_id", "fname");
            ViewBag.sub_id = new SelectList(db.tbl_subject, "sub_id", "sub_name");
            return View();
        }

        // POST: class/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 = "cls_id,cls_name,stu_id,sub_id")] tbl_class tbl_class)
        {
            if (ModelState.IsValid)
            {
                db.tbl_class.Add(tbl_class);
                db.SaveChanges();
                return RedirectToAction("Index");
            }

            ViewBag.stu_id = new SelectList(db.tbl_student, "stu_id", "fname", tbl_class.stu_id);
            ViewBag.sub_id = new SelectList(db.tbl_subject, "sub_id", "sub_name", tbl_class.sub_id);
            return View(tbl_class);
        }

        // GET: class/Edit/5
        public ActionResult Edit(int? id)
```

Class – Edit View

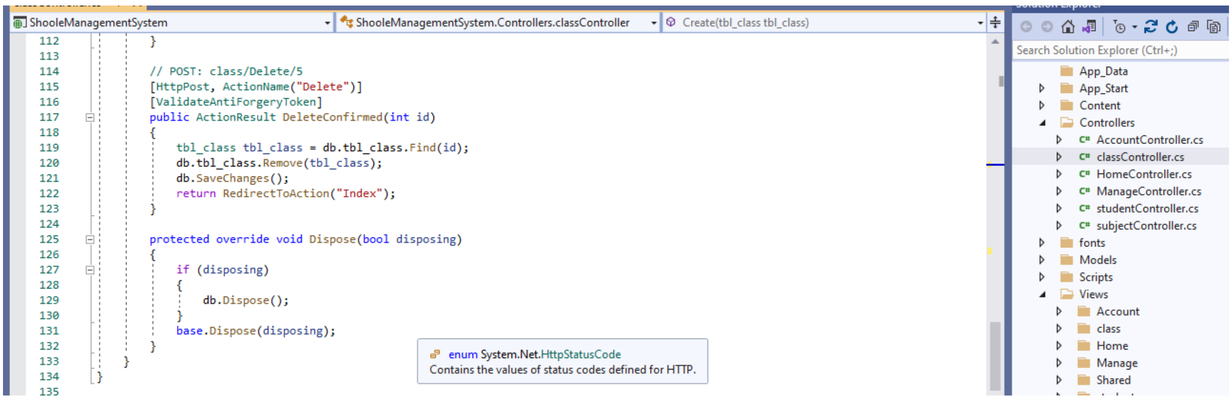


```
        public ActionResult Edit(int? id)
        {
            if (id == null)
            {
                return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
            }
            tbl_class tbl_class = db.tbl_class.Find(id);
            if (tbl_class == null)
            {
                return HttpNotFound();
            }
            ViewBag.stu_id = new SelectList(db.tbl_student, "stu_id", "fname", tbl_class.stu_id);
            ViewBag.sub_id = new SelectList(db.tbl_subject, "sub_id", "sub_name", tbl_class.sub_id);
            return View(tbl_class);
        }

        // POST: class/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 = "cls_id,cls_name,stu_id,sub_id")] tbl_class tbl_class)
        {
            if (ModelState.IsValid)
            {
                db.Entry(tbl_class).State = EntityState.Modified;
                db.SaveChanges();
                return RedirectToAction("Index");
```

Class – delete view



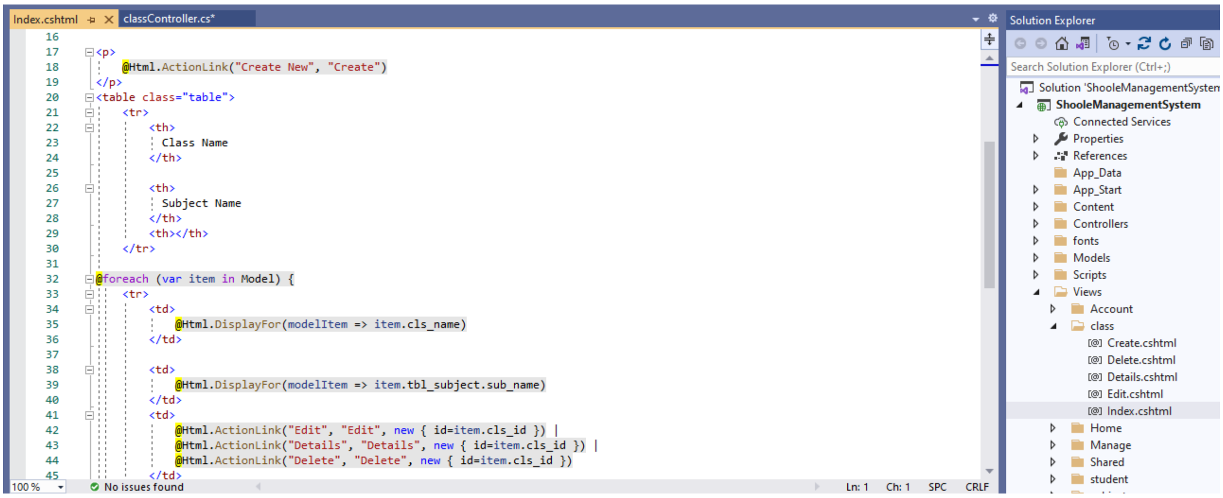
```
        [ValidateAntiForgeryToken]
        public ActionResult Edit([Bind(Include = "cls_id,cls_name,stu_id,sub_id")] tbl_class tbl_class)
        {
            if (ModelState.IsValid)
            {
                db.Entry(tbl_class).State = EntityState.Modified;
                db.SaveChanges();
                return RedirectToAction("Index");
            }
            ViewBag.stu_id = new SelectList(db.tbl_student, "stu_id", "fname", tbl_class.stu_id);
            ViewBag.sub_id = new SelectList(db.tbl_subject, "sub_id", "sub_name", tbl_class.sub_id);
            return View(tbl_class);
        }

        // GET: class/Delete/5
        public ActionResult Delete(int? id)
        {
            if (id == null)
            {
                return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
            }
            tbl_class tbl_class = db.tbl_class.Find(id);
            if (tbl_class == null)
            {
                return HttpNotFound();
            }
            return View(tbl_class);
```



```
        }

        // POST: class/Delete/5
        [HttpPost, ActionName("Delete")]
        [ValidateAntiForgeryToken]
        public ActionResult DeleteConfirmed(int id)
        {
            tbl_class tbl_class = db.tbl_class.Find(id);
            db.tbl_class.Remove(tbl_class);
            db.SaveChanges();
            return RedirectToAction("Index");
        }

        protected override void Dispose(bool disposing)
        {
            if (disposing)
            {
                db.Dispose();
            }
            base.Dispose(disposing);
        }
    }
}
```

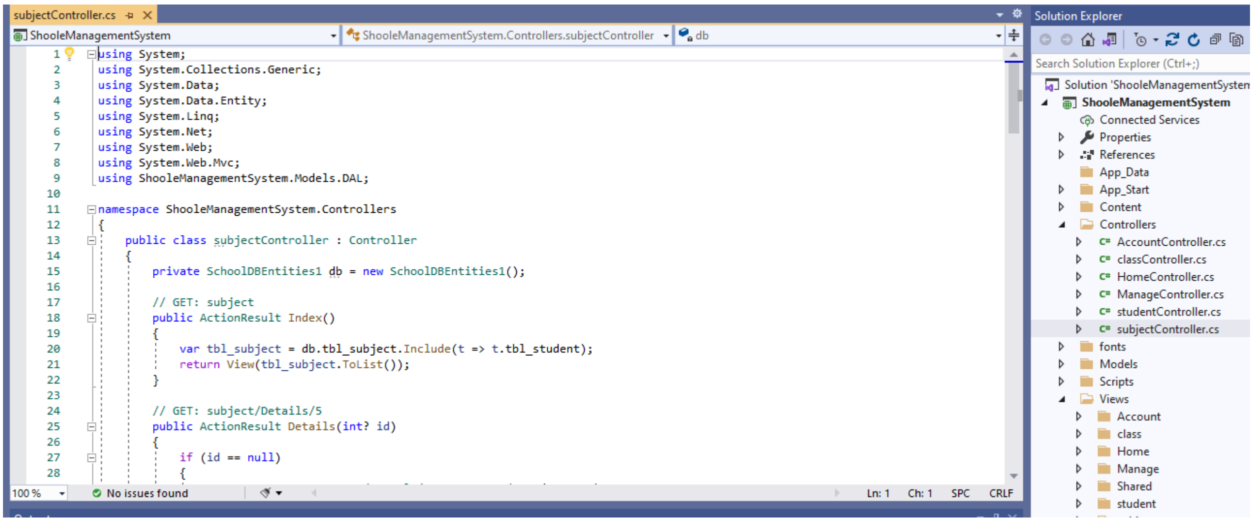
enum System.Net.HttpStatusCode
Contains the values of status codes defined for HTTP.

Class – index page



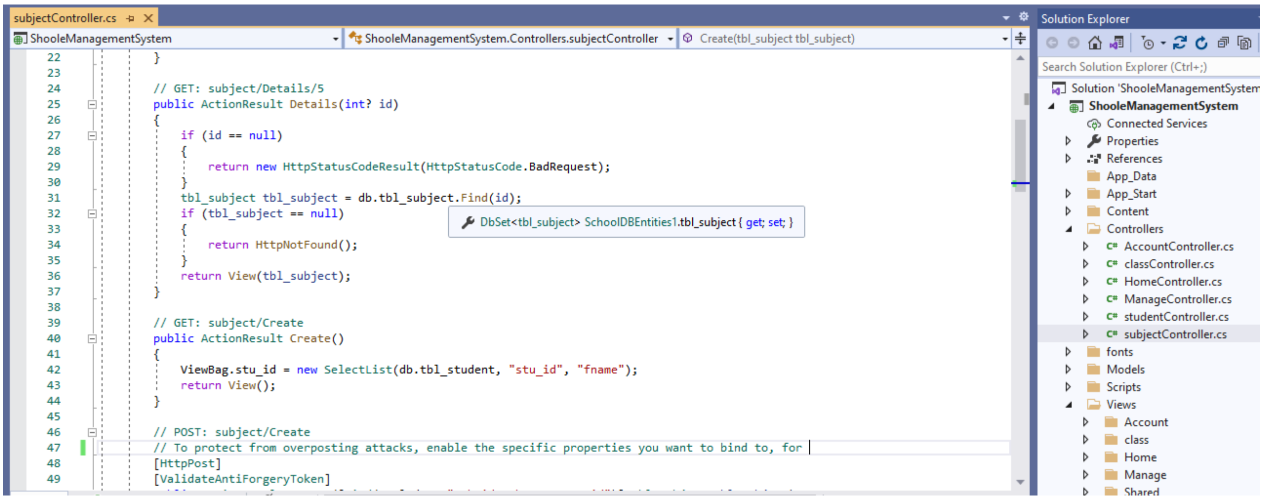
```

<p>
    @Html.ActionLink("Create New", "Create")
</p>
<table class="table">
    <tr>
        <th>
            Class Name
        </th>

        <th>
            Subject Name
        </th>
        <th></th>
    </tr>

@foreach (var item in Model) {
    <tr>
        <td>
            @Html.DisplayFor(modelItem => item.cls_name)
        </td>

        <td>
            @Html.DisplayFor(modelItem => item.tbl_subject.sub_name)
        </td>
        <td>
            @Html.ActionLink("Edit", "Edit", new { id=item.cls_id }) |
            @Html.ActionLink("Details", "Details", new { id=item.cls_id }) |
            @Html.ActionLink("Delete", "Delete", new { id=item.cls_id })
        </td>
```

6- Subject – CRUD operations -- Controller



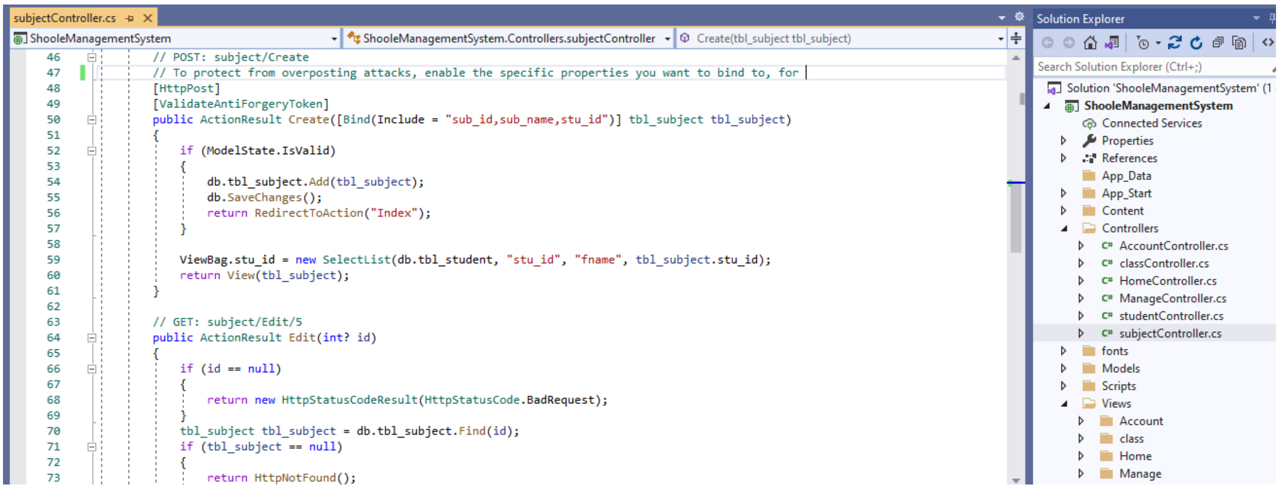
```
using System;
using System.Collections.Generic;
using System.Data;
using System.Data.Entity;
using System.Linq;
using System.Net;
using System.Web;
using System.Web.Mvc;
using ShooleManagementSystem.Models.DAL;

namespace ShooleManagementSystem.Controllers
{
    public class subjectController : Controller
    {
        private SchoolDBEntities1 db = new SchoolDBEntities1();

        // GET: subject
        public ActionResult Index()
        {
            var tbl_subject = db.tbl_subject.Include(t => t.tbl_student);
            return View(tbl_subject.ToList());
        }

        // GET: subject/Details/5
        public ActionResult Details(int? id)
        {
            if (id == null)
            {
```

Subject – details & Create views



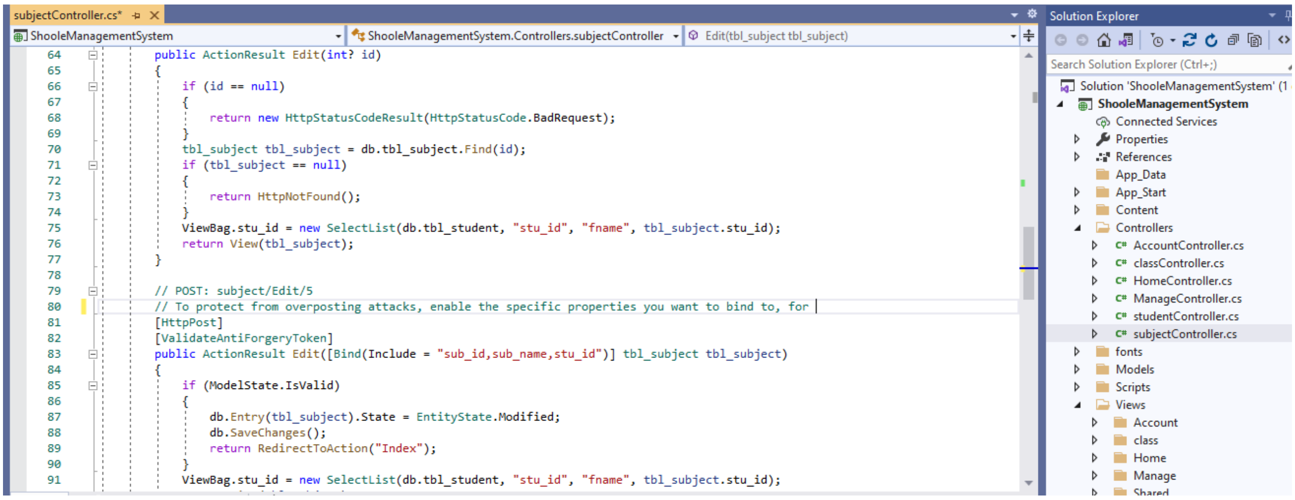
```
        }

        // GET: subject/Details/5
        public ActionResult Details(int? id)
        {
            if (id == null)
            {
                return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
            }
            tbl_subject tbl_subject = db.tbl_subject.Find(id);
            if (tbl_subject == null)
            {
                return HttpNotFound();
            }
            return View(tbl_subject);
        }

        // GET: subject/Create
        public ActionResult Create()
        {
            ViewBag.stu_id = new SelectList(db.tbl_student, "stu_id", "fname");
            return View();
        }

        // POST: subject/Create
        // To protect from overposting attacks, enable the specific properties you want to bind to, for
        [HttpPost]
        [ValidateAntiForgeryToken]
```

DbSet<tbl_subject> SchoolDBEntities1.tbl_subject { get; set; }



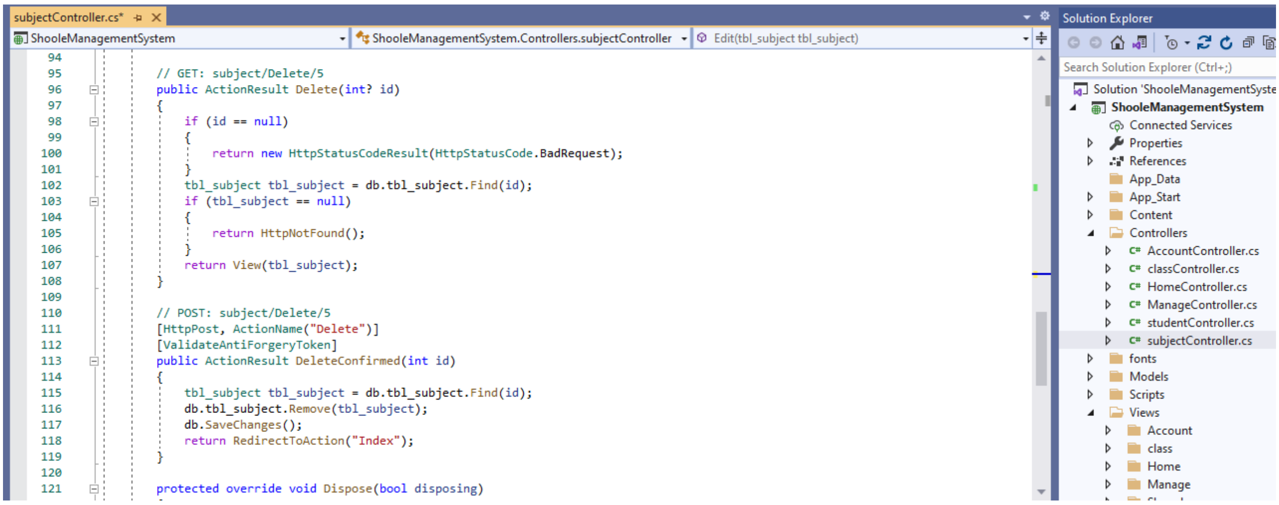
```
        // POST: subject/Create
        // To protect from overposting attacks, enable the specific properties you want to bind to, for
        [HttpPost]
        [ValidateAntiForgeryToken]
        public ActionResult Create([Bind(Include = "sub_id,sub_name,stu_id")] tbl_subject tbl_subject)
        {
            if (ModelState.IsValid)
            {
                db.tbl_subject.Add(tbl_subject);
                db.SaveChanges();
                return RedirectToAction("Index");
            }

            ViewBag.stu_id = new SelectList(db.tbl_student, "stu_id", "fname", tbl_subject.stu_id);
            return View(tbl_subject);
        }

        // GET: subject/Edit/5
        public ActionResult Edit(int? id)
        {
            if (id == null)
            {
                return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
            }
            tbl_subject tbl_subject = db.tbl_subject.Find(id);
            if (tbl_subject == null)
            {
                return HttpNotFound();
```

Subject – Edit view

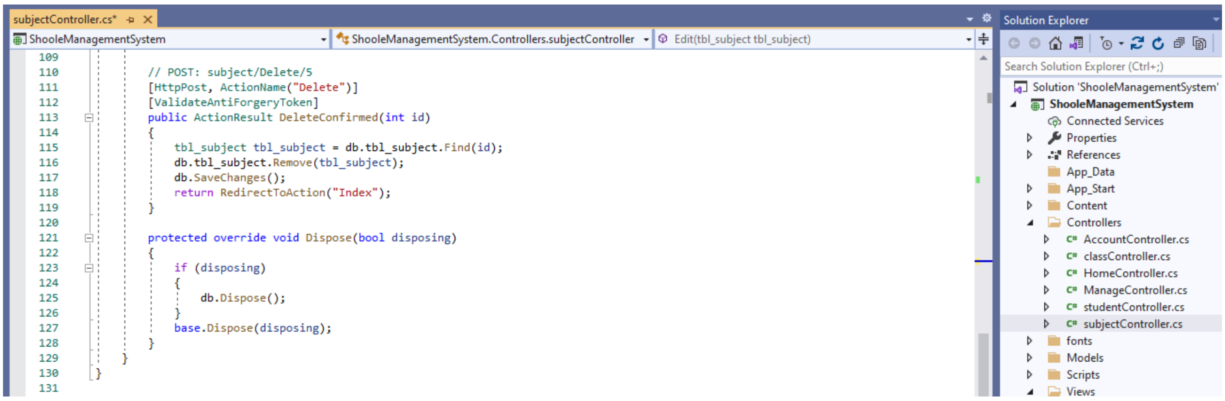


```
        public ActionResult Edit(int? id)
        {
            if (id == null)
            {
                return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
            }
            tbl_subject tbl_subject = db.tbl_subject.Find(id);
            if (tbl_subject == null)
            {
                return HttpNotFound();
            }
            ViewBag.stu_id = new SelectList(db.tbl_student, "stu_id", "fname", tbl_subject.stu_id);
            return View(tbl_subject);
        }

        // POST: subject/Edit/5
        // To protect from overposting attacks, enable the specific properties you want to bind to, for
        [HttpPost]
        [ValidateAntiForgeryToken]
        public ActionResult Edit([Bind(Include = "sub_id,sub_name,stu_id")] tbl_subject tbl_subject)
        {
            if (ModelState.IsValid)
            {
                db.Entry(tbl_subject).State = EntityState.Modified;
                db.SaveChanges();
                return RedirectToAction("Index");
            }
            ViewBag.stu_id = new SelectList(db.tbl_student, "stu_id", "fname", tbl_subject.stu_id);
```

Subject – delete view

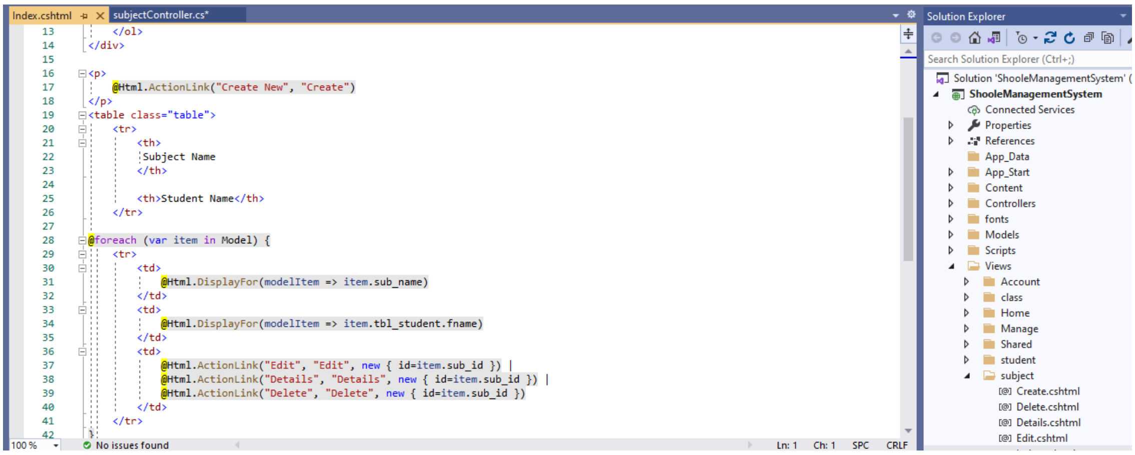


```

        // GET: subject/Delete/5
        public ActionResult Delete(int? id)
        {
            if (id == null)
            {
                return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
            }
            tbl_subject tbl_subject = db.tbl_subject.Find(id);
            if (tbl_subject == null)
            {
                return HttpNotFound();
            }
            return View(tbl_subject);
        }

        // POST: subject/Delete/5
        [HttpPost, ActionName("Delete")]
        [ValidateAntiForgeryToken]
        public ActionResult DeleteConfirmed(int id)
        {
            tbl_subject tbl_subject = db.tbl_subject.Find(id);
            db.tbl_subject.Remove(tbl_subject);
            db.SaveChanges();
            return RedirectToAction("Index");
        }

        protected override void Dispose(bool disposing)
```



```

        // POST: subject/Delete/5
        [HttpPost, ActionName("Delete")]
        [ValidateAntiForgeryToken]
        public ActionResult DeleteConfirmed(int id)
        {
            tbl_subject tbl_subject = db.tbl_subject.Find(id);
            db.tbl_subject.Remove(tbl_subject);
            db.SaveChanges();
            return RedirectToAction("Index");
        }

        protected override void Dispose(bool disposing)
        {
            if (disposing)
            {
                db.Dispose();
            }
            base.Dispose(disposing);
        }
    }
}

```

Subject – index page



```
    </ol>
</div>

<p>
    @Html.ActionLink("Create New", "Create")
</p>
<table class="table">
    <tr>
        <th>
            Subject Name
        </th>

        <th>Student Name</th>
    </tr>

@foreach (var item in Model) {
    <tr>
        <td>
            @Html.DisplayFor(modelItem => item.sub_name)
        </td>
        <td>
            @Html.DisplayFor(modelItem => item.tbl_student.fname)
        </td>
        <td>
            @Html.ActionLink("Edit", "Edit", new { id=item.sub_id }) |
            @Html.ActionLink("Details", "Details", new { id=item.sub_id }) |
            @Html.ActionLink("Delete", "Delete", new { id=item.sub_id })
        </td>
    </tr>
}
```