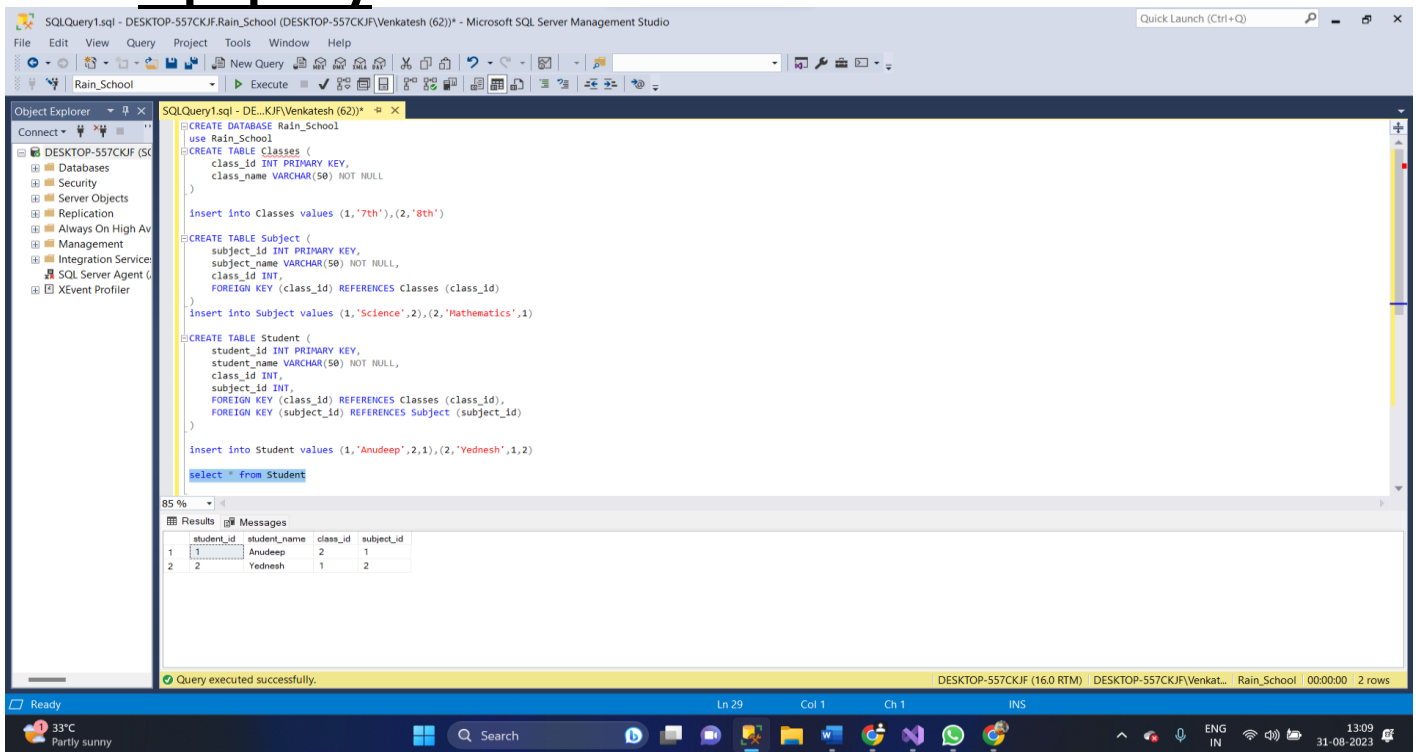
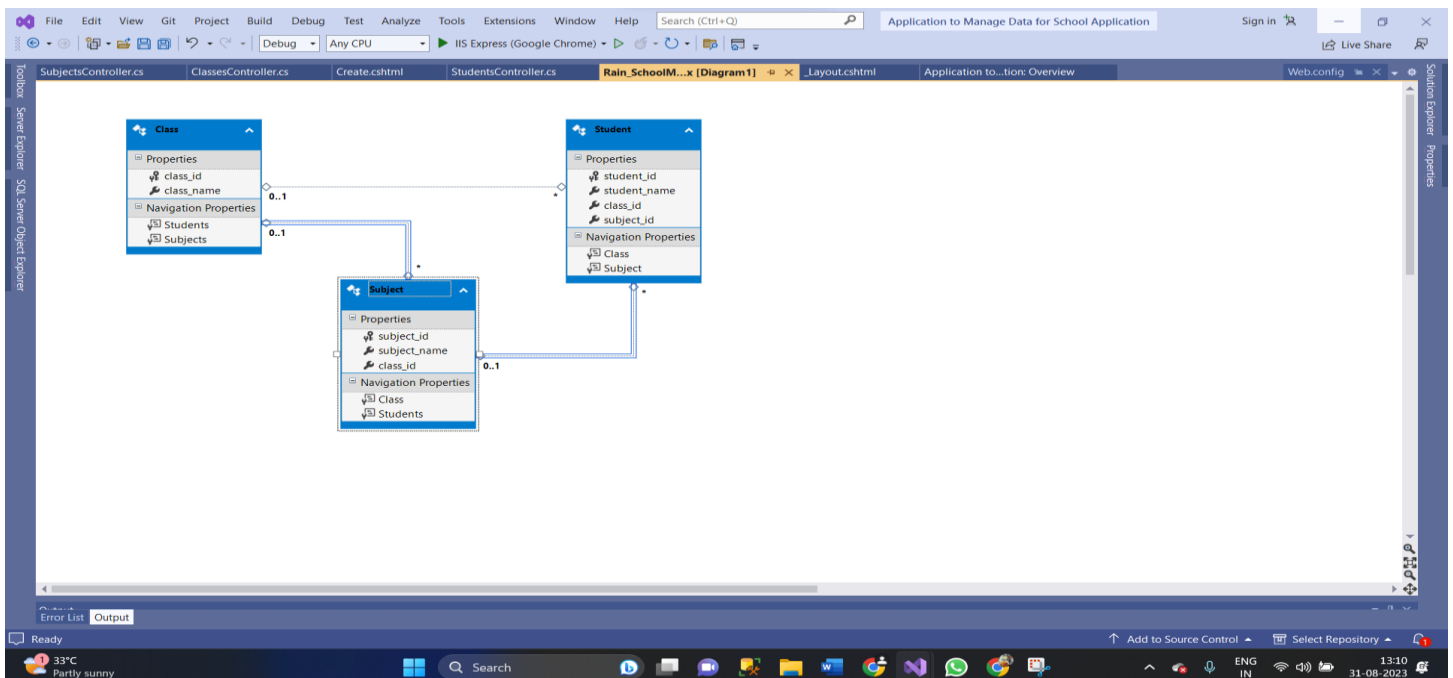


Sql-query



Visual studio 2022



ClassesController.cs → Code

```
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 Application_to_Manage_Data_for_School_Application.Models;

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

        // 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);
            }
            Class @class = db.Classes.Find(id);
            if (@class == null)
            {
                return HttpNotFound();
            }
            return View(@class);
        }

        // 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 = "class_id,class_name")] Class
@class)
        {
            if (ModelState.IsValid)
            {
                db.Classes.Add(@class);
                db.SaveChanges();
                return RedirectToAction("Index");
            }

            return View(@class);
        }

        // GET: Classes/Edit/5
        public ActionResult Edit(int? id)
        {
            if (id == null)

```

```

        {
            return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
        }
        Class @class = db.Classes.Find(id);
        if (@class == null)
        {
            return HttpNotFound();
        }
        return View(@class);
    }

    // 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 = "class_id,class_name")] Class
@class)
    {
        if (ModelState.IsValid)
        {
            db.Entry(@class).State = EntityState.Modified;
            db.SaveChanges();
            return RedirectToAction("Index");
        }
        return View(@class);
    }

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

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

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

```

SubjectController.cs ➔Code

```
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 Application_to_Manage_Data_for_School_Application.Models;

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

        // GET: Subjects
        public ActionResult Index()
        {
            var subjects = db.Subjects.Include(s => s.Class);
            return View(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()
        {
            ViewBag.class_id = new SelectList(db.Classes, "class_id",
"            "class_name");
            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 =
"subject_id,subject_name,class_id")] Subject subject)
        {
            if (ModelState.IsValid)
            {
                db.Subjects.Add(subject);
                db.SaveChanges();
                return RedirectToAction("Index");
            }
        }
    }
}
```

```

        ViewBag.class_id = new SelectList(db.Classes, "class_id",
"class_name", subject.class_id);
        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();
        }
        ViewBag.class_id = new SelectList(db.Classes, "class_id",
"class_name", subject.class_id);
        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 =
"subject_id,subject_name,class_id")] Subject subject)
    {
        if (ModelState.IsValid)
        {
            db.Entry(subject).State = EntityState.Modified;
            db.SaveChanges();
            return RedirectToAction("Index");
        }
        ViewBag.class_id = new SelectList(db.Classes, "class_id",
"class_name", subject.class_id);
        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.cs → Code

```

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 Application_to_Manage_Data_for_School_Application.Models;

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

        // GET: Students
        public ActionResult Index()
        {
            var students = db.Students.Include(s => s.Class).Include(s =>
s.Subject);
            return View(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()
        {
            ViewBag.class_id = new SelectList(db.Classes, "class_id",
"class_name");

```

```

        ViewBag.subject_id = new SelectList(db.Subjects, "subject_id",
"subject_name");
        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 =
"student_id,student_name,class_id,subject_id")] Student student)
    {
        if (ModelState.IsValid)
        {
            db.Students.Add(student);
            db.SaveChanges();
            return RedirectToAction("Index");
        }

        ViewBag.class_id = new SelectList(db.Classes, "class_id",
"class_name", student.class_id);
        ViewBag.subject_id = new SelectList(db.Subjects, "subject_id",
"subject_name", student.subject_id);
        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();
        }
        ViewBag.class_id = new SelectList(db.Classes, "class_id",
"class_name", student.class_id);
        ViewBag.subject_id = new SelectList(db.Subjects, "subject_id",
"subject_name", student.subject_id);
        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 =
"student_id,student_name,class_id,subject_id")] Student student)
    {
        if (ModelState.IsValid)
        {
            db.Entry(student).State = EntityState.Modified;
            db.SaveChanges();
            return RedirectToAction("Index");
        }
    }

```

```

        ViewBag.class_id = new SelectList(db.Classes, "class_id",
"class_name", student.class_id);
        ViewBag.subject_id = new SelectList(db.Subjects, "subject_id",
"subject_name", student.subject_id);
        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);
    }
}

```

Layout.cshtml → Code

```

<!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("About", "About", "Home", new { area =
"" }, new { @class = "nav-link" })</li>
            <li>@Html.ActionLink("Contact", "Contact", "Home", new { area
= "" }, new { @class = "nav-link" })</li>
            <li>@Html.ActionLink("Class", "Index", "Classes", new { area
= "" }, new { @class = "nav-link" })</li>
            <li>@Html.ActionLink("Subject", "Index", "Subjects", new {
area = "" }, new { @class = "nav-link" })</li>
            <li>@Html.ActionLink("Students Details", "Index", "Students",
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>

```