```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Threading.Tasks;
using Microsoft.AspNetCore.Http;
using Microsoft.AspNetCore.Mvc;
using Microsoft.EntityFrameworkCore;
using SimpilearnP4.Data;
using SimpilearnP4.Models;

namespace SimpilearnP4.Controllers
{
    [Route("api/[controller]")]
    [ApiController]
    public class RainBowSchoolsController : ControllerBase
    {
        private readonly SimpilearnDbContext _context;

        public RainBowSchoolsController(SimpilearnDbContext context)
        {
            _context = context;
        }

        // GET: api/RainBowSchools
        [HttpGet]
        public async Task<ActionResult<IEnumerable<RainBowSchool>>>
GetRainBowSchool()
        {
          if (_context.RainBowSchool == null)
          {
              return NotFound();
          }
            return await _context.RainBowSchool.ToListAsync();
        }

        // GET: api/RainBowSchools/5
        [HttpGet("{id}")]
        public async Task<ActionResult<RainBowSchool>> GetRainBowSchool(int id)
        {
          if (_context.RainBowSchool == null)
          {
              return NotFound();
          }
            var rainBowSchool = await _context.RainBowSchool.FindAsync(id);

            if (rainBowSchool == null)
            {
                return NotFound();
            }

            return rainBowSchool;
        }

        // PUT: api/RainBowSchools/5
        // To protect from overposting attacks, see
https://go.microsoft.com/fwlink/?linkid=2123754
        [HttpPut("{id}")]
        public async Task<IActionResult> PutRainBowSchool(int id, RainBowSchool
rainBowSchool)
        {
            if (id != rainBowSchool.Id)
            {
                return BadRequest();
            }

            _context.Entry(rainBowSchool).State = EntityState.Modified;

            try
            {
                await _context.SaveChangesAsync();
            }
            catch (DbUpdateConcurrencyException)
            {
                if (!RainBowSchoolExists(id))
                {
                    return NotFound();
                }
                else
                {
                    throw;
                }
            }

            return NoContent();
        }

        // POST: api/RainBowSchools
        // To protect from overposting attacks, see
https://go.microsoft.com/fwlink/?linkid=2123754
        [HttpPost]
        public async Task<ActionResult<RainBowSchool>>
PostRainBowSchool(RainBowSchool rainBowSchool)
        {
          if (_context.RainBowSchool == null)
          {
              return Problem("Entity set 'SimpilearnDbContext.RainBowSchool'  is
null.");
          }
            _context.RainBowSchool.Add(rainBowSchool);
            await _context.SaveChangesAsync();

            return CreatedAtAction("GetRainBowSchool", new { id = rainBowSchool.Id
}, rainBowSchool);
        }

        // DELETE: api/RainBowSchools/5
        [HttpDelete("{id}")]
        public async Task<IActionResult> DeleteRainBowSchool(int id)
        {
            if (_context.RainBowSchool == null)
            {
                return NotFound();
            }
            var rainBowSchool = await _context.RainBowSchool.FindAsync(id);
            if (rainBowSchool == null)
            {
                return NotFound();
            }

            _context.RainBowSchool.Remove(rainBowSchool);
            await _context.SaveChangesAsync();

            return NoContent();
        }

        private bool RainBowSchoolExists(int id)
        {
            return (_context.RainBowSchool?.Any(e => e.Id ==
id)).GetValueOrDefault();
        }
    }
}
```

```
namespace SimpilearnP4.Models
{
    public class RainBowSchool
    {
        public int Id { get; set; }
        public string StudentName { get; set; }
        public string Subject { get; set; }
        public double SubjectMarks { get; set; }
    }
}
```

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Threading.Tasks;
using Microsoft.EntityFrameworkCore;
using SimpilearnP4.Models;

namespace SimpilearnP4.Data
{
    public class SimpilearnDbContext : DbContext
    {
        public SimpilearnDbContext (DbContextOptions<SimpilearnDbContext> options)
            : base(options)
        {
        }

        public DbSet<SimpilearnP4.Models.RainBowSchool> RainBowSchool { get; set; }
= default!;
    }
}
```

```
// <auto-generated />
using Microsoft.EntityFrameworkCore;
using Microsoft.EntityFrameworkCore.Infrastructure;
using Microsoft.EntityFrameworkCore.Metadata;
using Microsoft.EntityFrameworkCore.Storage.ValueConversion;
using SimpilearnP4.Data;

#nullable disable

namespace SimpilearnP4.Migrations
{
    [DbContext(typeof(SimpilearnDbContext))]
    partial class SimpilearnDbContextModelSnapshot : ModelSnapshot
    {
        protected override void BuildModel(ModelBuilder modelBuilder)
        {
#pragma warning disable 612, 618
            modelBuilder
                .HasAnnotation("ProductVersion", "7.0.10")
                .HasAnnotation("Relational:MaxIdentifierLength", 128);

            SqlServerModelBuilderExtensions.UseIdentityColumns(modelBuilder);

            modelBuilder.Entity("SimpilearnP4.Models.RainBowSchool", b =>
                {
                    b.Property<int>("Id")
                        .ValueGeneratedOnAdd()
                        .HasColumnType("int");


SqlServerPropertyBuilderExtensions.UseIdentityColumn(b.Property<int>("Id"));

                    b.Property<string>("StudentName")
                        .IsRequired()
                        .HasColumnType("nvarchar(max)");

                    b.Property<string>("Subject")
                        .IsRequired()
                        .HasColumnType("nvarchar(max)");

                    b.Property<double>("SubjectMarks")
                        .HasColumnType("float");

                    b.HasKey("Id");

                    b.ToTable("RainBowSchool");
                });
#pragma warning restore 612, 618
        }
    }
}
```