Experiment: 2.1

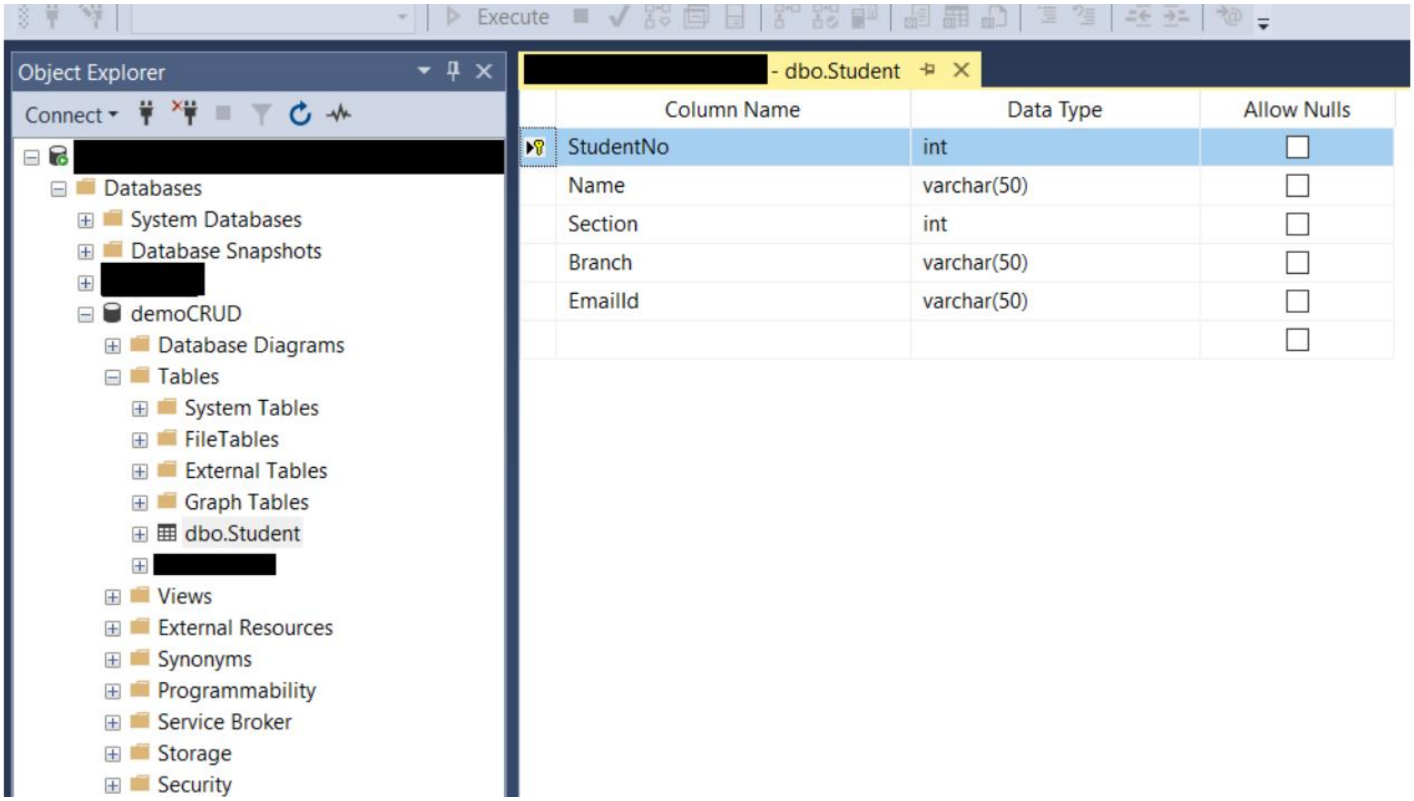
Student Name: Sandeep Kumar **UID: 20BCS4885**
Branch: BE – CSE **Section/Group: 603/A**
Semester: 6th semester **Subject: Dotnet Lab**

Aim of the practical:

Develop MVC application on CRUD Operations.

Solution:

1. **Create a Database with the following columns:** This is just a demo to make you understand the code in the article. You can create your own database according to your needs.

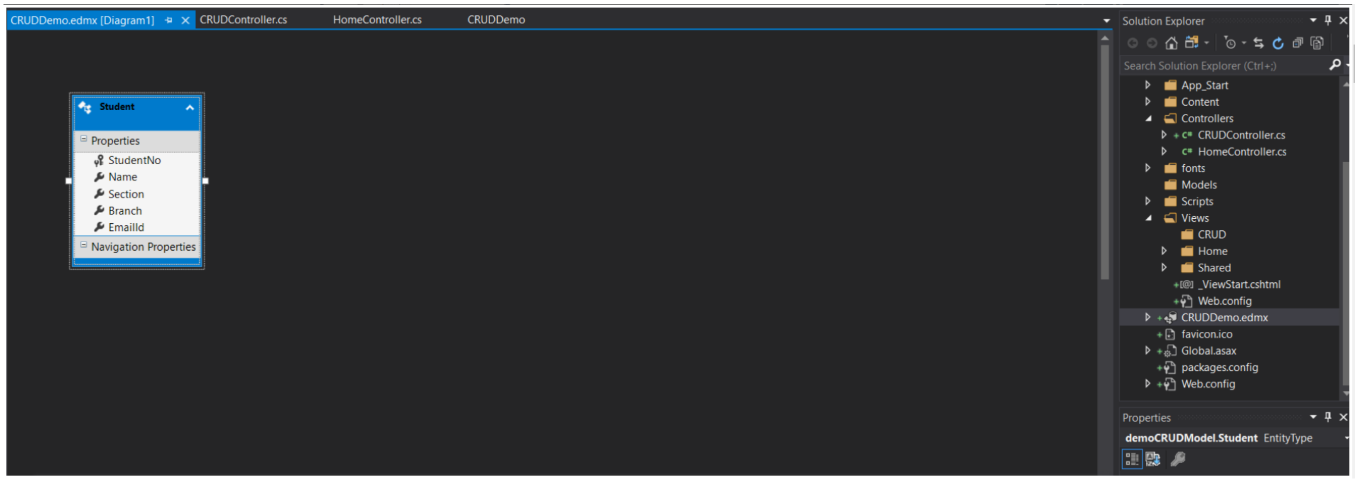


[redacted]
- Databases
  - System Databases
  - Database Snapshots
  - [redacted]
  - demoCRUD
    - Database Diagrams
    - Tables
      - System Tables
      - FileTables
      - External Tables
      - Graph Tables
      - dbo.Student
      - [redacted]
    - Views
    - External Resources
    - Synonyms
    - Programmability
    - Service Broker
    - Storage
    - Security

[redacted] - dbo.Student

Column Name	Data Type	Allow Nulls
StudentNo	int	□
Name	varchar(50)	□
Section	int	□
Branch	varchar(50)	□
EmailId	varchar(50)	□
		□

2. **Create a Project in Visual Studio:** Follow the guidelines that are given in the link provided above to create a project. After creating the project add entity data model to add connection string to your *web.config* file, to do so follow this article Add Entity Data Model to Your ASP.NET Project. The following EDMX diagram will be shown on your solution window.



ASP.NET CRUD (Create, Read, Update, Delete)

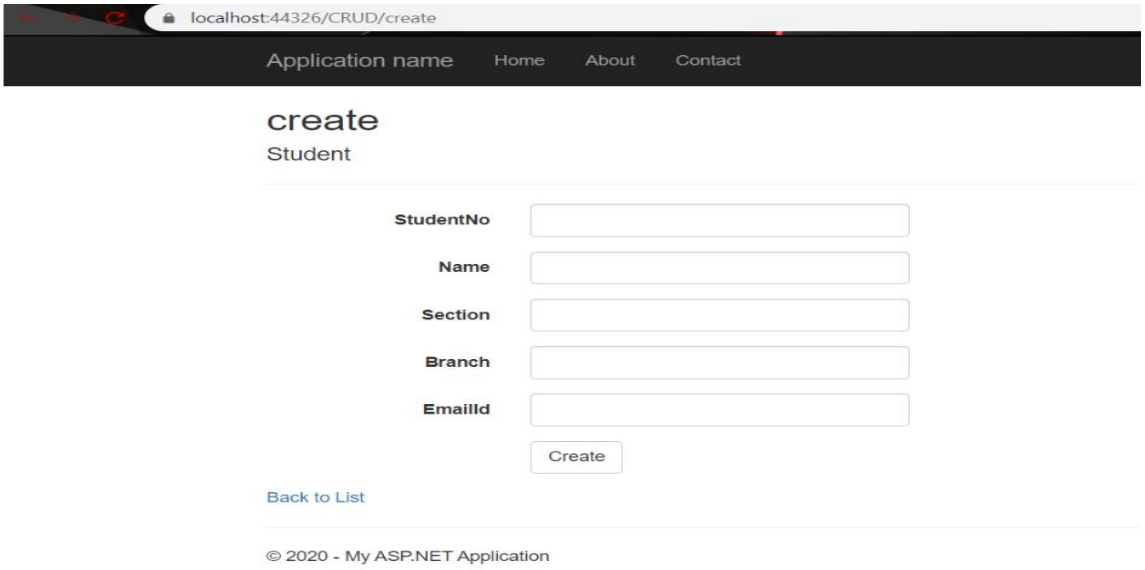
1. **Create** Now to create a new record in your database write the following code in the newly created controller.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;

namespace CRUDDemo.Controllers
{
        public class CRUDController : Controller
        {
                public ActionResult create()
                {
                        return View();
                }
                [HttpPost]
        public ActionResult create(Student model)
        {
            using(var context = new demoCRUDEntities())
            {
                context.Student.Add(model);

                context.SaveChanges();
            }
            string message = "Created the record successfully";
            ViewBag.Message = message;
            return View();
        }
    }
}
```

After this write click on the first action result and click on AddView and then select template as Create and model class as your own created model and data context class as your own created EDMX model. Then run the project and go the URL https://localhost:port_number/Controller_name/Action_Method_name

For example, https://localhost:44326/CRUD/create

Output:



localhost:44326/CRUD/create

create

Student

StudentNo

Name

Section

Branch

EmailId

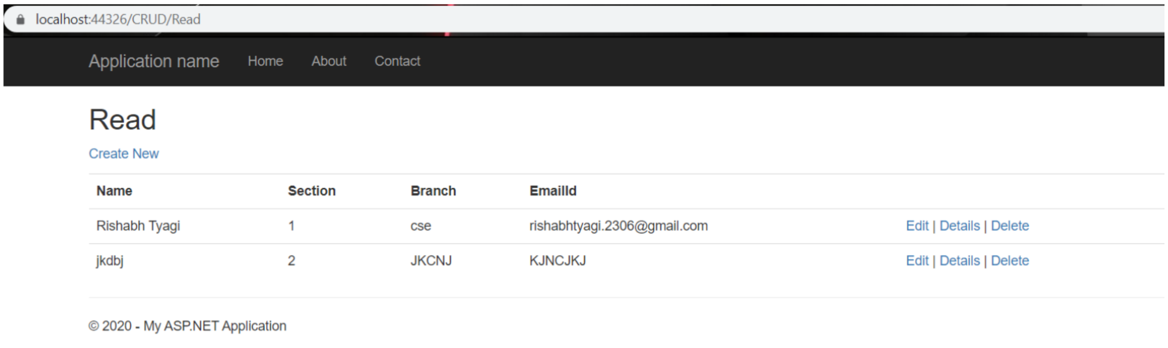
© 2020 - My ASP.NET Application

2. **Read:** Now to See the added data on your screen follow the below-given code:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;

namespace CRUDDemo.Controllers
{
        public class CRUDController : Controller {
                [HttpGet] // Set the attribute to Read
                        public ActionResult
                        Read()
                {
                        using(var context = new demoCRUDEntities())
                        {
                                var data = context.Student.ToList();
                                return View(data);
                        }
                }
        }
}
```

After this add the View but remember to change the template as List. Then run the project and go to the URL https://localhost:port_number/Controller_name/Action_Method_name
For Example https://localhost:44326/CRUD/Read

Output:



Read

Create New

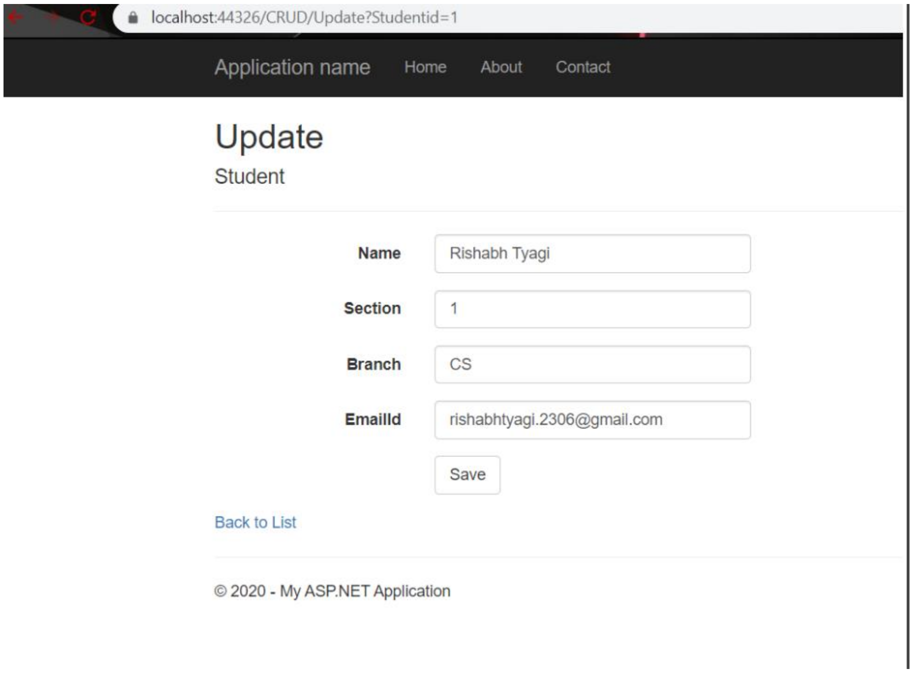
Name	Section	Branch	EmailId	
Rishabh Tyagi	1	cse	rishabhtyagi.2306@gmail.com	Edit | Details | Delete
jkdbj	2	JKCNJ	KJNCJKJ	Edit | Details | Delete

© 2020 - My ASP.NET Application

3. **Update:** Now, to update the existing record follow the code given below:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;

namespace CRUDDemo.Controllers
{
  public class CRUDController : Controller
  {
      public ActionResult Update(int Studentid)
      {
            using(var context = new demoCRUDEntities())
            {
                  var data = context.Student.Where(x => x.StudentNo ==
  Studentid).SingleOrDefault();
                  return View(data);
            }
      }
      [HttpPost]
      [ValidateAntiForgeryToken]
      public ActionResult Update(int Studentid, Student model)
      {
            using(var context = new demoCRUDEntities())
            {

                  var data = context.Student.FirstOrDefault(x => x.StudentNo ==
  Studentid);

                        if (data != null)
                  {
                        data.Name = model.Name;
                        data.Section = model.Section;
                        data.EmailId = model.EmailId;
                        data.Branch = model.Branch;
                        context.SaveChanges();

                        return RedirectToAction("Read");
                  }
                  else
                        return View();
                  }
            }
        }
  }
```

After this add view similarly as done previously but remember to change the template to Edit. Then run the project and go to the URL https://localhost:port_number/Controller_name/Action_Method_name?ID_U_want_to_edit



Update

Student

Name: Rishabh Tyagi

Section: 1

Branch: CS

EmailId: rishabhtyagi.2306@gmail.com

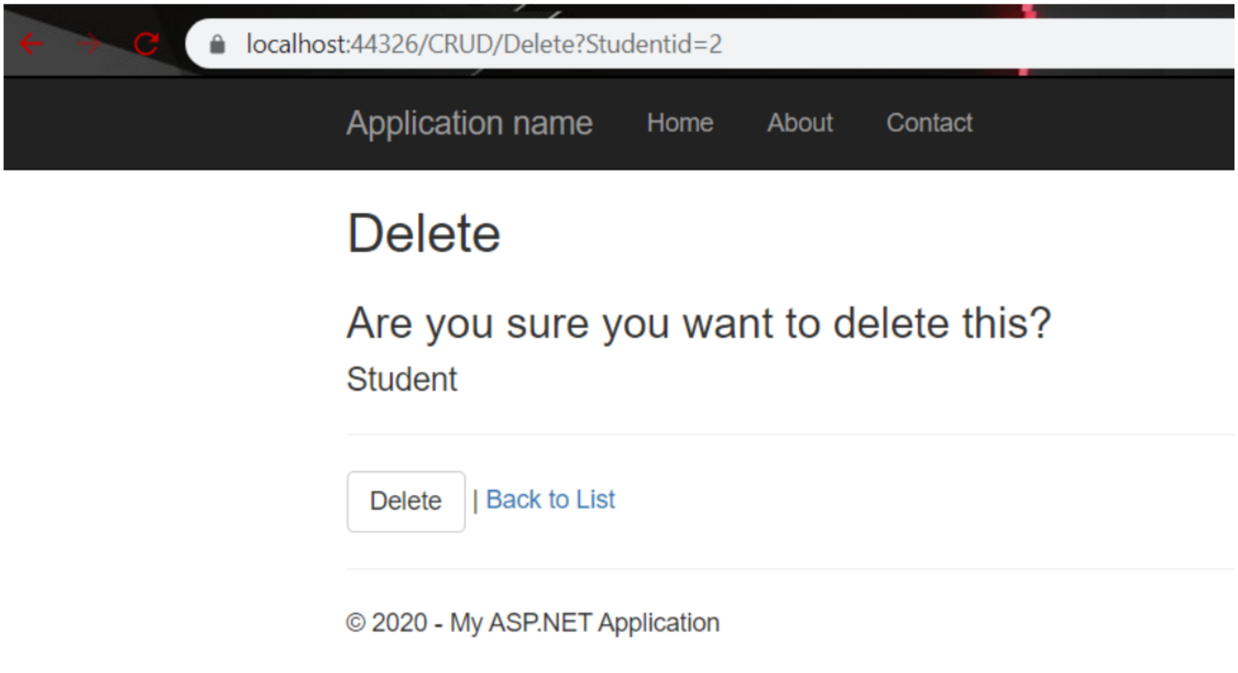
© 2020 - My ASP.NET Application

4. **Delete:** Now, to delete a record from the database follow the code given below

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;

namespace CRUDDemo.Controllers
{
        public class CRUDController : Controller {
                public ActionResult Delete()
                {
                        return View();
                }

                [HttpPost]
                [ValidateAntiForgeryToken] public ActionResult
                Delete(int Studentid)
                {
                        using(var context = new demoCRUDEntities())
                        {
                                var data = context.Student.FirstOrDefault(x = > x.StudentNo == Studentid);

                                if (data != null) {
                                        context.Student.Remove(data);
                                        context.SaveChanges();
                                        return RedirectToAction("Read");
                                }
                                else
                                        return View();
                        }
                }
        }
}
```

After this added view as done previously, but remember to change the template to Delete. Then run the project and go to the URL https://localhost:port_number/Controller_name/Action_Method_name?ID_U_want_to_Delete

For Example, https://localhost:44326/CRUD/Delete?Studentid=1



Delete

Are you sure you want to delete this?

Student

© 2020 - My ASP.NET Application