LAB EXAM

MS.Net Technologies

Name:-Soham Banage.

Total Marks: 20 Time: 2 hours

1. Create a menu driven C# application to interact with the database.

You have to insert data into "Students" table and retrieve a list of all the students from the Students

table and display their names, roll numbers, and their marks. In the above question, perform update

and delete operations as well.

---------- Ans-----

```
using Microsoft.Data.SqlClient;
using System;
using System.Data;
using static System.Net.Mime.MediaTypeNames;
using static System.Runtime.InteropServices.JavaScript.JSType;

namespace DotNetLab
{
    internal class Program
    {
        static void Main(string[] args)
        {
            bool exit = false;
            while (!exit)
            {
                Console.WriteLine("1. Update students");
                Console.WriteLine("2. Delete students");
                Console.WriteLine("3. All students");


                Console.WriteLine("0. Exit");
                Console.Write("Enter your choice: ");

                int choice;
                if (int.TryParse(Console.ReadLine(), out choice))
                {
                    switch (choice)
                    {

                        case 1:
                            Update();
                            break;
                        case 2:
                            Delete();
                            break;
                        case 3:
                            StuDataRetrival();
                            break;
                        case 0:
```

```
                            exit = true;
                            break;
                        default:
                            Console.WriteLine("Invalid choice. Please try
again.");
                            break;
                    }
                }
                else
                {
                    Console.WriteLine("Invalid choice. Please try again.");
                }

                Console.WriteLine();
            }
        }
        static void Update()
        {
            SqlConnection cn = new SqlConnection();
            cn.ConnectionString = @"Data Source = (localdb)\ProjectModels;
Initial Catalog = SBjune23; Integrated Security = True; Connect Timeout = 30";
            cn.Open();
            try
            {
                Student student = new Student();

                Console.Write("Enter rollno where you have to update : ");
                student.rollno = int.Parse(Console.ReadLine());

                Console.Write("Enter Student Name: ");
                student.name = Console.ReadLine();

                Console.Write("Enter students marks: ");
                student.marks = int.Parse(Console.ReadLine());


                SqlCommand cmd = new SqlCommand();
                cmd.Connection = cn;
                cmd.CommandType = CommandType.Text;
                cmd.CommandText = "Update Student Set name = @name ,marks =
@marks Where rollno = @rollno";

                cmd.Parameters.AddWithValue("@rollno", student.rollno);
                cmd.Parameters.AddWithValue("@name", student.name);
                cmd.Parameters.AddWithValue("@marks", student.marks);

                cmd.ExecuteNonQuery();
                Console.WriteLine("Updation Done ");
            }
            catch (Exception ex)
            {
                Console.WriteLine(ex.Message);
            }
            finally { cn.Close(); }
        }
        static void Delete()
        {
            SqlConnection cn = new SqlConnection();
            cn.ConnectionString = @"Data Source = (localdb)\ProjectModels;
Initial Catalog = SBjune23; Integrated Security = True; Connect Timeout = 30";
            cn.Open();
            try
            {
```

```
                Student student = new Student();

                Console.Write("Enter student rollno whose data to be deleted  :
");
                student.rollno = int.Parse(Console.ReadLine());



                SqlCommand cmd = new SqlCommand();
                cmd.Connection = cn;
                cmd.CommandType = CommandType.Text;
                cmd.CommandText = "Delete from Student where rollno = @rollno ";
                cmd.Parameters.AddWithValue("@rollno", student.rollno);


                cmd.ExecuteNonQuery();
                Console.WriteLine("Deletion Done");

            }
            catch (Exception ex)
            {
                Console.WriteLine(ex.Message);
            }
            finally { cn.Close(); }
        }
        public static void StuDataRetrival()
        {
            SqlConnection cn = new SqlConnection();
            cn.ConnectionString = @"Data Source = (localdb)\ProjectModels;
Initial Catalog = SBjune23; Integrated Security = True; Connect Timeout = 30";
            cn.Open();

            try
            {
                SqlCommand cmd = new SqlCommand();
                cmd.Connection = cn;
                cmd.CommandType = CommandType.Text;
                cmd.CommandText = "select * from Student";

                SqlDataReader dr = cmd.ExecuteReader();

                while (dr.Read())
                {
                    Console.Write(dr["name"]+" ;");
                    Console.Write(dr["marks"]);
                    Console.WriteLine();
                }
                dr.Close();

            }
            catch (Exception ex)
            {
                Console.WriteLine(ex.Message);
            }
            finally { cn.Close(); }

        }

    }
        public class Student
        {
            public int rollno
            {
```

```
            get; set;
        }
        public string? name
        {
            get; set;
        }
        public int marks
        {
            get; set;
        }

    }
}
```

All Students Data

```
1. Update students
2. Delete students
3. All students
0. Exit
Enter your choice: 3
Soham ;89
Bharat ;95
Ritu ;92
```

Update Data

```
1. Update students
2. Delete students
3. All students
0. Exit
Enter your choice: 1
Enter rollno where you have to update : 2
Enter Student Name: Lalit
Enter students marks: 86
Updation Done
```

```
1. Update students
2. Delete students
3. All students
0. Exit
Enter your choice: 3
Soham ;89
Lalit ;86
Ritu ;92
```

Delete data

```
1. Update students
2. Delete students
3. All students
0. Exit
Enter your choice: 2
Enter student rollno whose data to be deleted  : 3
Deletion Done
```

```
1. Update students
2. Delete students
3. All students
0. Exit
Enter your choice: 3
Soham ;89
Lalit ;86
```

2. Create a model class named Car. Add an Index by using model binding, perform create, read update

and Delete operations for details such as CarId, Name, bhp, cc, fuel tank and fuelEconomy.

CAR CLASS

```
using Microsoft.Data.SqlClient;
using ModleBindingLab.Models;
using System.Data;

namespace ModleBindingLab.Models
{
    public class Car
    {
        public int CarId
        {
            get; set;
        }
        public string Name
        {
            get;set;
        }
        public int Bhp
        {
            get;set;
        }
        public int CC
        {
            get;set;
        }
        public int Fueltank
        {
            get;set;
        }
        public string FuelEconomy
        {
            get;set;
        }
        public static List<Car> GetAllCars()
        {
            List<Car> carlst = new List<Car>();

            SqlConnection cn = new SqlConnection();
            cn.ConnectionString = @"Data Source = (localdb)\ProjectModels;
Initial Catalog = SBjune23; Integrated Security = True; Connect Timeout = 30";
```

```
            cn.Open();

            try
            {
                SqlCommand cmd = new SqlCommand();
                cmd.Connection = cn;
                cmd.CommandType = CommandType.Text;
                cmd.CommandText = "SELECT * FROM Car";

                SqlDataReader dr = cmd.ExecuteReader();

                while (dr.Read())
                {
                    Car cr = new Car();
                    {
                        cr.CarId = Convert.ToInt32(dr["CarId"]);
                        cr.Name = dr["Name"].ToString();
                        cr.Bhp = Convert.ToInt32(dr["Bhp"]);
                        cr.CC = Convert.ToInt32(dr["CC"]);
                        cr.Fueltank = Convert.ToInt32(dr["Bhp"]);
                        cr.FuelEconomy = dr["Name"].ToString();


                    };

                    carlst.Add(cr);

                }

                foreach (Car c in carlst)
                {
                    Console.WriteLine($"CarId: {c.CarId}");
                    Console.WriteLine($"Name: {c.Name}");
                    Console.WriteLine($"Bhp: {c.Bhp}");
                    Console.WriteLine($"CC: {c.CC}");
                    Console.WriteLine($"Fueltank: {c.Fueltank}");
                    Console.WriteLine($"FuelEconomy: {c.FuelEconomy}");

                    Console.WriteLine();
                }


                dr.Close();

            }
            catch (Exception ex)
            {
                Console.WriteLine(ex.Message);
            }
            finally
            {
                cn.Close();
            }

            return carlst;
        }
        public static Car GetSingleCar(int CarId)
        {
            Car cr = null;
            SqlConnection cn = new SqlConnection();
            cn.ConnectionString = @"Data Source = (localdb)\ProjectModels;
Initial Catalog = SBjune23; Integrated Security = True; Connect Timeout = 30";
            cn.Open();
```

```
            try
            {
                SqlCommand cmd = new SqlCommand();
                cmd.Connection = cn;
                cmd.CommandType = CommandType.Text;

                cmd.CommandText = "Select * from Car Where CarId = @CarId";
                cmd.Parameters.AddWithValue("@CarId", CarId);

                SqlDataReader dr = cmd.ExecuteReader();
                while (dr.Read())
                {
                    cr = new Car();
                    {

                        Console.WriteLine(cr.CarId =
Convert.ToInt32(dr["CarId"]));
                        Console.WriteLine(cr.Name = dr["Name"].ToString());
                        Console.WriteLine(cr.Bhp = Convert.ToInt32(dr["Bhp"]));
                        Console.WriteLine(cr.CC = Convert.ToInt32(dr["CC"]));
                        Console.WriteLine(cr.Fueltank =
Convert.ToInt32(dr["Fueltank"]));
                        Console.WriteLine(cr.FuelEconomy =
Convert.ToString(dr["FuelEconomy"]));


                    };
                }
            }
            catch (Exception ex)
            {
                Console.WriteLine(ex.Message);
            }
            finally { cn.Close(); }
            return cr;
        }
        public static void Update(Car obj)
        {
            SqlConnection cn = new SqlConnection();
            cn.ConnectionString = @"Data Source = (localdb)\ProjectModels;
Initial Catalog = SBjune23; Integrated Security = True; Connect Timeout = 30";
            cn.Open();

            try
            {
                SqlCommand cmd = new SqlCommand();
                cmd.Connection = cn;
                cmd.CommandType = CommandType.Text;
                cmd.CommandText = "Update Car Set Name = @Name,Bhp = @Bhp,CC =
@CC,Fueltank =@Fueltank,FuelEconomy = @FuelEconomy Where EmpNo = @EmpNo";
                cmd.Parameters.AddWithValue("@CarId", obj.CarId);
                cmd.Parameters.AddWithValue("@Name", obj.Name);
                cmd.Parameters.AddWithValue("@Bhp", obj.Bhp);
                cmd.Parameters.AddWithValue("@CC", obj.CC);
                cmd.Parameters.AddWithValue("@Fueltank", obj.Fueltank);
                cmd.Parameters.AddWithValue("@FuelEconomy", obj.FuelEconomy);

                cmd.ExecuteNonQuery();

            }
            catch (Exception ex)
            {
                Console.WriteLine(ex.Message); ;
```

```
            }
            finally { cn.Close(); }
        }
        public static void DeleteCar(int CarId)
        {
            SqlConnection cn = new SqlConnection();
            cn.ConnectionString = @"Data Source = (localdb)\ProjectModels;
Initial Catalog = SBjune23; Integrated Security = True; Connect Timeout = 30";
            cn.Open();
            try
            {
                SqlCommand cmd = new SqlCommand();
                cmd.Connection = cn;
                cmd.CommandType = CommandType.Text;
                cmd.CommandText = "Delete from Car where CarId = @CarId ";
                cmd.Parameters.AddWithValue("@CarId", CarId);

                cmd.ExecuteNonQuery();

            }
            catch (Exception ex)
            {
                Console.WriteLine(ex.Message);
            }
            finally { cn.Close(); }
        }

    }

}
```

CAR CONTROLLER

```
using Microsoft.AspNetCore.Http;
using Microsoft.AspNetCore.Mvc;
using ModleBindingLab.Models;

namespace ModleBindingLab.Controllers
{
    public class CarController1 : Controller
    {
        // GET: CarController1
        public ActionResult Index()
        {
            List<Car> lstcar = Car.GetAllCars();
            return View(lstcar);
        }

        // GET: CarController1/Details/5
        public ActionResult Details(int id)
        {
            return View();
        }

        // GET: CarController1/Create
        public ActionResult Create()
        {
            return View();
        }

        // POST: CarController1/Create
        [HttpPost]
        [ValidateAntiForgeryToken]
        public ActionResult Create(IFormCollection collection)
```

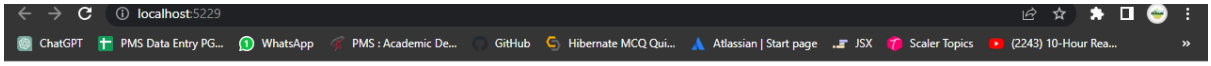
```
        {
            try
            {
                return RedirectToAction(nameof(Index));
            }
            catch
            {
                return View();
            }
        }

        // GET: CarController1/Edit/5
        public ActionResult Edit(int id)
        {
            Car obj = Car.GetSingleCar(id);
            return View(obj);
        }

        // POST: CarController1/Edit/5
        [HttpPost]
        [ValidateAntiForgeryToken]
        public ActionResult Edit(Car cr, IFormCollection collection)
        {
            try
            {
                Car.Update(cr);
                return RedirectToAction(nameof(Index));
            }
            catch
            {
                return View();
            }
        }

        // GET: CarController1/Delete/5
        public ActionResult Delete(int id)
        {
            Car obj = Car.GetSingleCar(id);
            return View(obj);
        }

        // POST: CarController1/Delete/5
        [HttpPost]
        [ValidateAntiForgeryToken]
        public ActionResult Delete(int id, Car obj)
        {
            try
            {
                Car.DeleteCar(id);
                return RedirectToAction(nameof(Index));
            }
            catch
            {
                return View();
            }
        }
    }
}
```



ModleBindingLab Home Privacy

Index

Create New

CarId	Name	Bhp	CC	Fueltank	FuelEconomy	
1	Tata	125	1500	125	Tata	Edit | Details | Delete

ModleBindingLab Home Privacy

Edit

Car

CarId	1
Name	Tata
Bhp	125
CC	1500
Fueltank	1
FuelEconomy	Avg

Edit | Back to List

ModleBindingLab Home Privacy

Delete

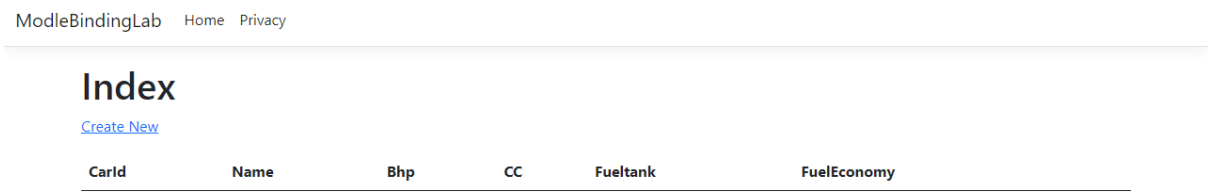
Are you sure you want to delete this?

Car

CarId	1
Name	Tata
Bhp	125
CC	1500
Fueltank	1
FuelEconomy	Avg

Delete | Back to List

AFTER DELETE



Index

Create New

CarId	Name	Bhp	CC	Fueltank	FuelEconomy