Employee

21 February 2023 21:03

```
using Microsoft.AspNetCore.Mvc;
using Microsoft.Data.SqlClient;
using Microsoft.DotNet.Scaffolding.Shared.Messaging;
using System.ComponentModel.DataAnnotations;
using System.Data;
namespace EmployeeSite.Models
{
    public class Employee
    {
        [Required]
        public int EmpId { get; set; }
        [StringLength(20)]
        [MinLength(3)]
        public string Name { get; set; }
        [DataType(DataType.Currency)]
        public decimal Salary { get; set; }
        [Phone]
        public string Phone { get; set; }
        [EmailAddress]
        public string Email { get; set; }
        [Required]
        public int DeptNo { get; set; }


        public static void InsertEmployee(Employee employee)
        {
            SqlConnection cn = new SqlConnection();
            //Data Source=(localdb)\MSSQLLocalDB;Initial
Catalog=EmployeeSiteDB;Integrated Security=True;Connect Timeout=
30;Encrypt=False;TrustServerCertificate=False;ApplicationIntent=ReadWrite;MultiSu
bnetFailover=False
            cn.ConnectionString = "Data Source=(localdb)\\MSSQLLocalDB;Initial
Catalog=EmployeeSiteDB;Integrated Security=True;";
            try
            {
                cn.Open();
               // SqlCommand cmd = new SqlCommand();
                //cmd.Connection = cn;
                SqlCommand cmd = cn.CreateCommand();
                cmd.CommandType = CommandType.Text;
                cmd.CommandText = "insert into Employees values (@EmpId, @Name,
@Salary, @Phone , @Email, @DeptNo )";
                cmd.Parameters.AddWithValue("@EmpId", employee.EmpId);
                cmd.Parameters.AddWithValue("@Name", employee.Name);
                cmd.Parameters.AddWithValue("@Salary", employee.Salary);
                cmd.Parameters.AddWithValue("@Phone", employee.Phone);
                cmd.Parameters.AddWithValue("@Email", employee.Email);
                cmd.Parameters.AddWithValue("@DeptNo", employee.DeptNo);
                cmd.ExecuteNonQuery();
            }
            catch (Exception ex)
```

```
            {
                //throw new Exception("")
            }
            finally
            {
                cn.Close();
            }
        }
        public static Employee GetSingleEmployee(int id)
        {
            SqlConnection cn = new SqlConnection();

            //Data Source=(localdb)\MSSQLLocalDB;Initial
Catalog=EmployeeSiteDB;Integrated Security=True;Connect Timeout=
30;Encrypt=False;TrustServerCertificate=False;ApplicationIntent=ReadWrite;MultiSu
bnetFailover=False
            cn.ConnectionString = "Data Source=(localdb)\\MSSQLLocalDB;Initial
Catalog=EmployeeSiteDB;Integrated Security=True;";
            try
            {
                cn.Open();
                SqlCommand cmd = new SqlCommand();
                cmd.Connection = cn;
                cmd.CommandType = CommandType.Text;
                cmd.CommandText = "select * from Employees where EmpId=@EmpId";
                cmd.Parameters.AddWithValue("@EmpId", id);
                SqlDataReader dr = cmd.ExecuteReader();
                if (dr.Read())
                {
                    return new Employee { EmpId = (int)dr[0], Name = (string)
dr[1], Salary = (decimal)dr[2], Phone = (string)dr[3], Email = (string)dr[4],
DeptNo = (int)dr[5] };
                    //o.EmpId = dr.GetInt32(0);
                    //o.Name = dr.GetString(1);
                    //o.Salary = dr.GetDecimal(2);
                    //o.Phone = dr.GetString(3);
                    //o.Email = dr.GetString(4);
                    //o.DeptNo = dr.GetInt32(5);

                }
                else
                {
                    throw new ExpClass("Exception Occurred");
                }
                dr.Close();
            }
            catch (Exception ex)
            {
                return null;
            }
            finally
            {
                cn.Close();
            }

        }
        public static List<Employee> GetAllEmployee()
        {
```

```
            SqlConnection cn = new SqlConnection();
            List<Employee> list = new List<Employee>();
            //Data Source=(localdb)\MSSQLLocalDB;Initial
Catalog=EmployeeSiteDB;Integrated Security=True;Connect Timeout=
30;Encrypt=False;TrustServerCertificate=False;ApplicationIntent=ReadWrite;MultiSu
bnetFailover=False
            cn.ConnectionString = "Data Source=(localdb)\\MSSQLLocalDB;Initial
Catalog=EmployeeSiteDB;Integrated Security=True;";
            try
            {
                cn.Open();
                SqlCommand cmd = new SqlCommand();
                cmd.Connection = cn;
                cmd.CommandType = CommandType.Text;
                cmd.CommandText = "select * from Employees ";
                SqlDataReader dr = cmd.ExecuteReader();
                while (dr.Read())
                {
                    //Employee o = new Employee();
                    //o.EmpId = dr.GetInt32(0);
                    //o.Name = dr.GetString(1);
                    //o.Salary = dr.GetDecimal(2);
                    //o.Phone = dr.GetInt64(3);
                    //o.Email = dr.GetString(4);
                    //o.DeptNo = dr.GetInt32(5);
                    //list.Add(o);
                    list.Add(new Employee { EmpId =(int) dr[0], Name = (string)
dr[1], Salary = (decimal)dr[2], Phone =(string)dr[3], Email = (string)dr[4],
DeptNo = (int)dr[5] });
                }
                dr.Close();
            }
            catch (Exception ex)
            {
                //throw new Exception("")
            }
            finally
            {
                cn.Close();
            }
            return list;
        }
        public static void UpdateEmployee(Employee employee)
        {
            SqlConnection cn = new SqlConnection();
            //Data Source=(localdb)\MSSQLLocalDB;Initial
Catalog=EmployeeSiteDB;Integrated Security=True;Connect Timeout=
30;Encrypt=False;TrustServerCertificate=False;ApplicationIntent=ReadWrite;MultiSu
bnetFailover=False
            cn.ConnectionString = "Data Source=(localdb)\\MSSQLLocalDB;Initial
Catalog=EmployeeSiteDB;Integrated Security=True;";
            try
            {
                cn.Open();
                SqlCommand cmd = new SqlCommand();
                cmd.Connection = cn;
                cmd.CommandType = CommandType.Text;
                cmd.CommandText = "update Employees set Name=@Name,
```

```
Salary=@Salary, Phone=@Phone, Email=@Email, DeptNo=@DeptNo where EmpId=@EmpId";
                cmd.Parameters.AddWithValue("@EmpId", employee.EmpId);
                cmd.Parameters.AddWithValue("@Name", employee.Name);
                cmd.Parameters.AddWithValue("@Salary", employee.Salary);
                cmd.Parameters.AddWithValue("@Phone", employee.Phone);
                cmd.Parameters.AddWithValue("@Email", employee.Email);
                cmd.Parameters.AddWithValue("@DeptNo", employee.DeptNo);
                cmd.ExecuteNonQuery();
            }
            catch (Exception e)
            {
            }
            finally
            {
                cn.Close();
            }
        }
        public static void DeleteEmployee(int id)
        {
            SqlConnection cn = new SqlConnection();
            //Data Source=(localdb)\MSSQLLocalDB;Initial
Catalog=EmployeeSiteDB;Integrated Security=True;Connect Timeout=
30;Encrypt=False;TrustServerCertificate=False;ApplicationIntent=ReadWrite;MultiSu
bnetFailover=False
            cn.ConnectionString = "Data Source=(localdb)\\MSSQLLocalDB;Initial
Catalog=EmployeeSiteDB;Integrated Security=True;";
            try
            {
                cn.Open();
                SqlCommand cmd = new SqlCommand();
                cmd.Connection = cn;
                cmd.CommandType = CommandType.Text;
                cmd.CommandText = "delete from Employees where EmpId=@EmpId";
                cmd.Parameters.AddWithValue("@EmpId", id);
                cmd.ExecuteNonQuery();
            }
            catch (Exception e)
            {
            }
            finally
            {
                cn.Close();
            }
        }

    }
    public class ExpClass : Exception
    {
        public ExpClass(string msg) : base(msg)
        {
        }
    }
}
```