WEB TECHNOLOGY DOTNET ASP AND ADO

1. Program to Show Database Connectivity.

```
using MySql.Data.MySqlClient;
public class DatabaseConnectivity
    public static void Main()
        string connectionString =
'server=localhost;user=root;password=root;database=cplus";
        using (MySqlConnection conn = new MySqlConnection(connectionString))
            try
                conn.Open();
                Console.WriteLine("Connected to the database!");
                string query = "SELECT EmpNo, EmpName, EmpDesignation,
EmpSalary FROM Employees";
                MySqlCommand cmd = new MySqlCommand(query, conn);
                using (MySqlDataReader reader = cmd.ExecuteReader())
                    while (reader.Read())
                        Console.WriteLine($"EmpNo: {reader["EmpNo"]}, Name:
{reader["EmpName"]}, Designation: {reader["EmpDesignation"]}, Salary:
{reader["EmpSalary"]}");
            catch (Exception ex)
                Console.WriteLine("Error: " + ex.Message);
```

```
    PS C:\Users\hp\OneDrive\Desktop\C-SharpPrograms> dotnet run
        Connected to the database!
        EmpNo: 1, Name: Kavita, Designation: South Delhi, Salary: 1.00
    PS C:\Users\hp\OneDrive\Desktop\C-SharpPrograms> []
```

2. Program to show data Insertion into the database.

Table Objects:

```
namespace CSharpASPandADO.Models
{
    public class Employee
    {
        public int EmpNo { get; set; }
        public string EmpName { get; set; }
        public string EmpDesignation { get; set; }
        public decimal EmpSalary { get; set; }
}
```

Service Layer:

```
using MySql.Data.MySqlClient;
using CSharpASPandADO.Models;
public class EmployeeService
    private readonly IConfiguration _configuration;
    private readonly string _connectionString;
    public EmployeeService(IConfiguration configuration)
        _configuration = configuration;
        connectionString =
 configuration.GetConnectionString("DefaultConnection");
    public void AddEmployee(Employee employee)
        using (MySqlConnection conn = new MySqlConnection(_connectionString))
            string query = "INSERT INTO Employees (EmpNo, EmpName,
EmpDesignation, EmpSalary) VALUES (@EmpNo, @EmpName, @EmpDesignation,
@EmpSalary)";
            MySqlCommand cmd = new MySqlCommand(query, conn);
            cmd.Parameters.AddWithValue("@EmpNo", employee.EmpNo);
            cmd.Parameters.AddWithValue("@EmpName", employee.EmpName);
            cmd.Parameters.AddWithValue("@EmpDesignation",
employee.EmpDesignation);
            cmd.Parameters.AddWithValue("@EmpSalary", employee.EmpSalary);
            conn.Open();
```

```
cmd.ExecuteNonQuery();
}
}
```

Controller Layer:

```
using Microsoft.AspNetCore.Mvc;
using CSharpASPandADO.Models;
namespace CSharpASPandADO.Controllers
    public class EmployeeController : Controller
        private readonly EmployeeService _employeeService;
        public EmployeeController(EmployeeService employeeService)
           _employeeService = employeeService;
        // GET: Display the employee form
        [HttpGet]
        public IActionResult Index()
            return View();
        // POST: Handle form submission
        [HttpPost]
        public IActionResult Index(Employee employee)
        {
            if (ModelState.IsValid)
                _employeeService.AddEmployee(employee);
                ViewBag.Message = "Employee added successfully.";
                // Clear form inputs after successful submission
                ModelState.Clear();
                return View(); // Stay on the same view to show the message
            return View(employee); // Re-display form with validation errors
```

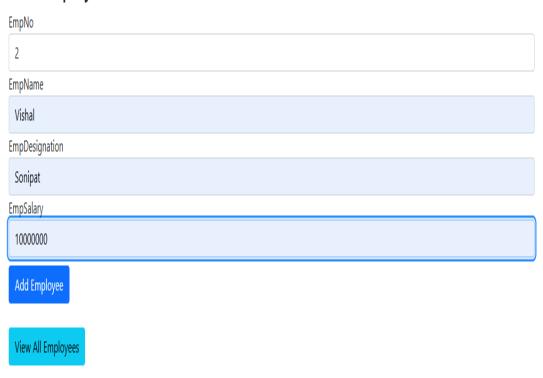
Index.chtml:

```
@model CSharpASPandADO.Models.Employee
@{
   ViewData["Title"] = "Add Employee";
<h2>Add Employee</h2>
<!-- Display validation summary -->
@Html.ValidationSummary(true, "", new { @class = "text-danger" })
<form asp-action="Index" method="post">
    <div>
        <label asp-for="EmpNo"></label>
        <input asp-for="EmpNo" class="form-control" />
        <span asp-validation-for="EmpNo" class="text-danger"></span>
    </div>
    <div>
        <label asp-for="EmpName"></label>
        <input asp-for="EmpName" class="form-control" />
        <span asp-validation-for="EmpName" class="text-danger"></span>
    </div>
    <div>
        <label asp-for="EmpDesignation"></label>
        <input asp-for="EmpDesignation" class="form-control" />
        <span asp-validation-for="EmpDesignation" class="text-danger"></span>
    </div>
    <div>
        <label asp-for="EmpSalary"></label>
        <input asp-for="EmpSalary" class="form-control" />
        <span asp-validation-for="EmpSalary" class="text-danger"></span>
    </div>
    <button type="submit" class="btn btn-primary mt-2">Add Employee</button>
</form>
<a asp-action="EmployeeInformation" class="btn btn-info">View All
Employees</a>
<!-- Success message -->
@if (ViewBag.Message != null)
@ViewBag.Message
```

```
@section Scripts {
    @{await Html.RenderPartialAsync("_ValidationScriptsPartial");}
}
```

CSharpASPandADO Home Privacy

Add Employee



Employee added successfully.

3. Program to show Update feature into the database.

Service Layer Method:

```
public void UpdateEmployee(Employee employee)
{
    using (MySqlConnection conn = new MySqlConnection(_connectionString))
    {
        string query = "UPDATE Employees SET EmpName=@EmpName,
        EmpDesignation=@EmpDesignation, EmpSalary=@EmpSalary WHERE EmpNo=@EmpNo";
            MySqlCommand cmd = new MySqlCommand(query, conn);
            cmd.Parameters.AddWithValue("@EmpNo", employee.EmpNo);
            cmd.Parameters.AddWithValue("@EmpName", employee.EmpName);
            cmd.Parameters.AddWithValue("@EmpDesignation",
            employee.EmpDesignation);
            cmd.Parameters.AddWithValue("@EmpSalary", employee.EmpSalary);

            conn.Open();
            cmd.ExecuteNonQuery();
        }
}
```

Controller Layer Method:

```
public IActionResult UpdateEmployee(int id)
{
    var emp = _employeeService.GetEmployeeById(id);
    return View(emp);
}

// POST: Edit employee
[HttpPost]
public IActionResult UpdateEmployee(Employee employee)
{
    _employeeService.UpdateEmployee(employee);
    return RedirectToAction("EmployeeInformation");
}
```

UpdateEmployee.cshtml:

CSharpASPandADO Home Privacy

Update Employee EmpName Vishal Kaushik EmpDesignation Sonipat EmpSalary 2000000

```
MySQL localhost:33060+ ssl cplus SQL > select * from employees;

EmpNo | EmpName | EmpDesignation | EmpSalary |

1 | Kavita | South Delhi | 1.00 |

2 | Vishal Kaushik | Sonipat | 2000000.00 |

Prows in set (0.0006 sec)

MySQL localhost:33060+ ssl cplus SQL > ____
```

4. Program to show Delete feature into the Database.

Service Layer Method;

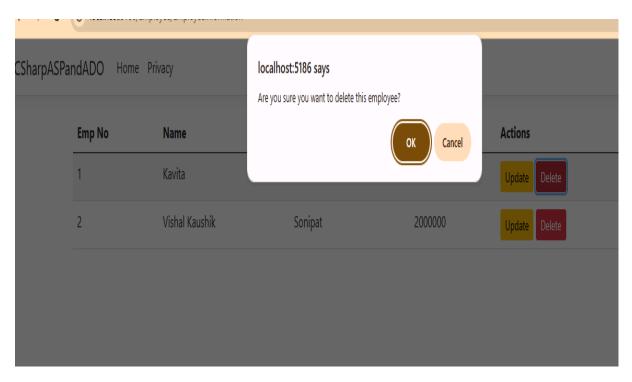
```
// Delete Employee
public void DeleteEmployee(int empNo)
{
    using (MySqlConnection conn = new MySqlConnection(_connectionString))
    {
        string query = "DELETE FROM Employees WHERE EmpNo=@EmpNo";
        MySqlCommand cmd = new MySqlCommand(query, conn);
        cmd.Parameters.AddWithValue("@EmpNo", empNo);

        conn.Open();
        cmd.ExecuteNonQuery();
    }
}
```

Controller Layer Method:

```
// GET: Delete employee
public IActionResult Delete(int id)
{
    _employeeService.DeleteEmployee(id);
    return RedirectToAction("EmployeeInformation");
}
```

Output:



```
      MySQL
      localhost:33060+ ssl
      cplus
      SQL
      > select * from employees;

      +----+
      EmpNo | EmpName | EmpDesignation | EmpSalary |

      |
      2 | Vishal Kaushik | Sonipat | 2000000.00 |

      |
      1 row in set (0.0006 sec)

      MySQL
      localhost:33060+ ssl
      cplus
      SQL
```

5. Program to show details of all Employee(Select feature) from the database.

Service Layer Method:

```
public List<Employee> GetAllEmployees()
    List<Employee> employeeList = new List<Employee>();
    using (MySqlConnection con = new
MySqlConnection(_configuration.GetConnectionString("DefaultConnection")))
        string query = "SELECT EmpNo, EmpName, EmpDesignation, EmpSalary FROM
Employees";
        MySqlCommand cmd = new MySqlCommand(query, con);
        con.Open();
        MySqlDataReader reader = cmd.ExecuteReader();
        while (reader.Read())
            Employee emp = new Employee
                EmpNo = Convert.ToInt32(reader["EmpNo"]),
                EmpName = reader["EmpName"].ToString(),
                EmpDesignation = reader["EmpDesignation"].ToString(),
                EmpSalary = Convert.ToInt32(reader["EmpSalary"])
            };
            employeeList.Add(emp);
        reader.Close();
    return employeeList;
```

Controller Layer Method:

```
public IActionResult EmployeeInformation()
{
    var employees = _employeeService.GetAllEmployees();
    return View(employees);
}
```

CSharpASPandADO Home Privacy

Emp No	Name	Designation	Salary	Actions
1	Ankur	Sonipat	2222222	Update Delete
2	Vishal Kaushik	Sonipat	2000000	Update Delete
3	Raman	Karnal	20000	Update Delete
4	Ankit	Gurgoan	250000	Update Delete

```
      MySQL
      localhost:33060+ ssl
      cplus
      SQL
      > select * from employees;

      | EmpNo | EmpName
      | EmpDesignation | EmpSalary |

      | 1 | Ankur
      | Sonipat | 2222222.00 |

      | 2 | Vishal Kaushik | Sonipat | 2000000.00 |
      | 20000000 |

      | 3 | Raman | Karnal | 20000.00 |
      | 250000.00 |

      | 4 | Ankit | Gurgoan | 250000.00 |
      | 4 | rows in set (0.0006 sec)

      MySQL
      localhost:33060+ ssl
      cplus
      SQL
      > __
```