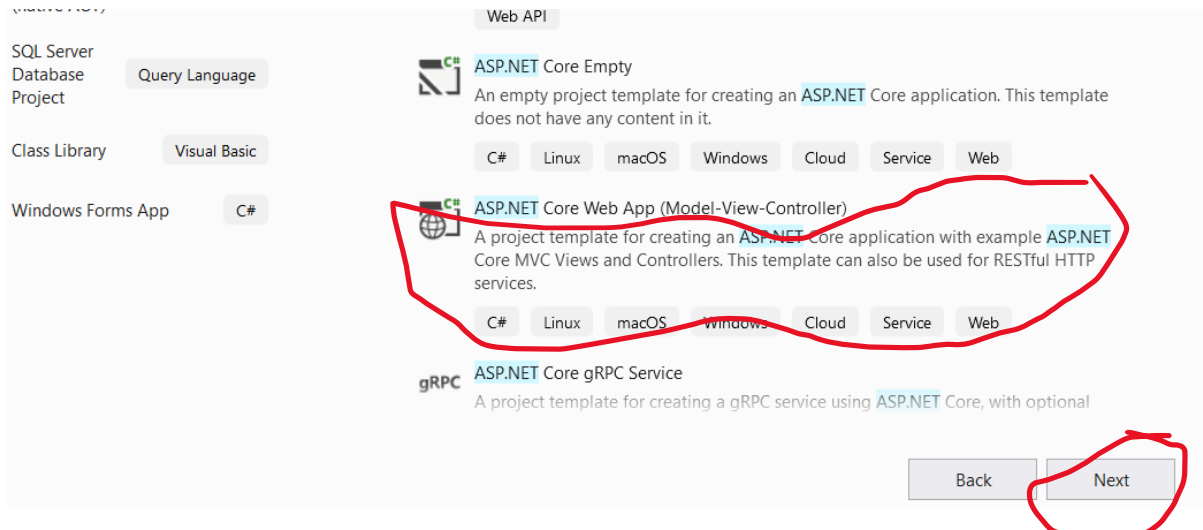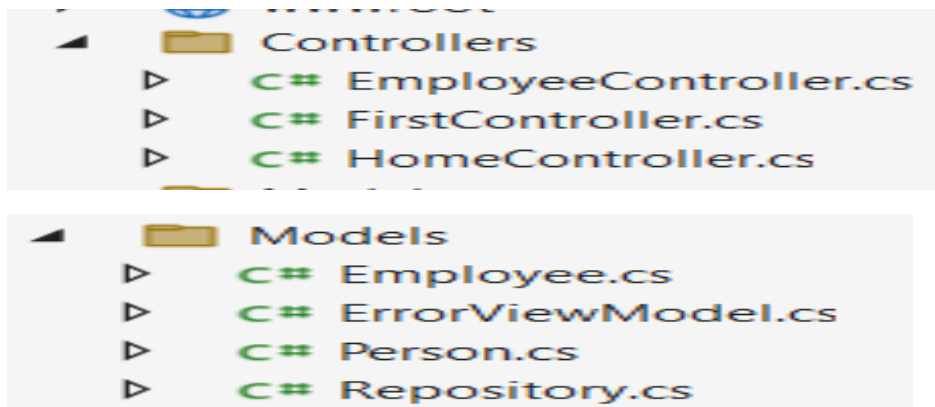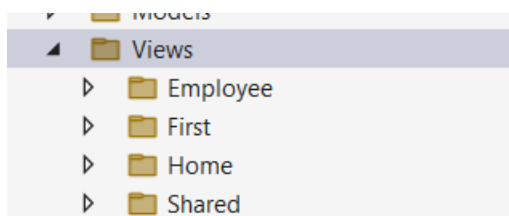## EPAM-LAB12

Develop a web application to perform CRUD operations on EMP details using ASP.NET MVC.

In Controllers folder create three files and in models 4 files as shown in the below fig:

· **In Views folder create 3 other folders**

Controllers/EmplyeeController.cs

```csharp
1   using FirstApp.Models;
2   using Microsoft.AspNetCore.Mvc;
3   namespace FirstApp.Controllers{
        0 references
4       public class EmployeeController : Controller {
            0 references
5           public IActionResult Index()       {
6               return View(Repository.AllEmpoyees);
7           }
            0 references
8           public IActionResult Create()       {
9               return View();
10          }
11          [HttpPost]
            0 references
12          public IActionResult Create(Employee employee)       {
13              if (ModelState.IsValid)            {
14                  Repository.Create(employee);
15                  return View("Thanks", employee);
16              }
17              else
18                  return View();
19          }
            0 references
20          public IActionResult Update(string empname)        {
21              Employee employee = Repository.AllEmpoyees.Where(e => e.Name == empname).FirstOrDefault();
22              return View(employee);
```

```csharp
22              return View(employee);
23          }
24          [HttpPost]
            0 references
25          public IActionResult Update(Employee employee, string empname){
26              if (ModelState.IsValid)            {
27                  Repository.AllEmpoyees.Where(e => e.Name == empname).FirstOrDefault().Age = employee.Age;
28                  Repository.AllEmpoyees.Where(e => e.Name == empname).FirstOrDefault().Salary = employee.Salary;
29                  Repository.AllEmpoyees.Where(e => e.Name == empname).FirstOrDefault().Department = employee.Department;
30                  Repository.AllEmpoyees.Where(e => e.Name == empname).FirstOrDefault().Sex = employee.Sex;
31                  Repository.AllEmpoyees.Where(e => e.Name == empname).FirstOrDefault().Name = employee.Name;
32
33                  return RedirectToAction("Index"); }
34              else
35                  return View(); }
36          [HttpPost]
            0 references
37          public IActionResult Delete(string empname){
38              Employee employee = Repository.AllEmpoyees.Where(e => e.Name == empname).FirstOrDefault();
39              Repository.Delete(employee);
40              return RedirectToAction("Index");
41          }
42      }
43  }
```

Controllers/FirstController.cs

```csharp
1   using FirstApp.Models;
2   using Microsoft.AspNetCore.Mvc;
3   namespace FirstApp.Controllers{
        0 references
4       public class FirstController : Controller {
            0 references
5           public string Index()   {
6               return "Hello World";
7           }
            0 references
8           public IActionResult Hello()     {
9               ViewBag.Message = "Hello World";
10              return View();
11          }
            0 references
12          public IActionResult Info()         {
13              Person person = new Person();
14              person.Name = "Bhavana";
15              person.Age = 18;
16              person.Location = "India";
17              return View(person);
18          }
19      }
20  }
```

Controllers/HomeController.cs

```csharp
using FirstApp.Models;
using Microsoft.AspNetCore.Mvc;
using System.Diagnostics;
namespace FirstApp.Controllers{
    // 3 references
    public class HomeController : Controller {
        private readonly ILogger<HomeController> _logger;
        // 0 references
        public HomeController(ILogger<HomeController> logger)      {
            _logger = logger;
        }
        // 0 references
        public IActionResult Index()      {
            return View();
        }
        // 0 references
        public IActionResult Privacy()      {
            return View();
        }
        [ResponseCache(Duration = 0, Location = ResponseCacheLocation.None, NoStore = true)]
        // 0 references
        public IActionResult Error()
        {
            return View(new ErrorViewModel { RequestId = Activity.Current?.Id ?? HttpContext.TraceIdentifier });
        }
    }
}
```
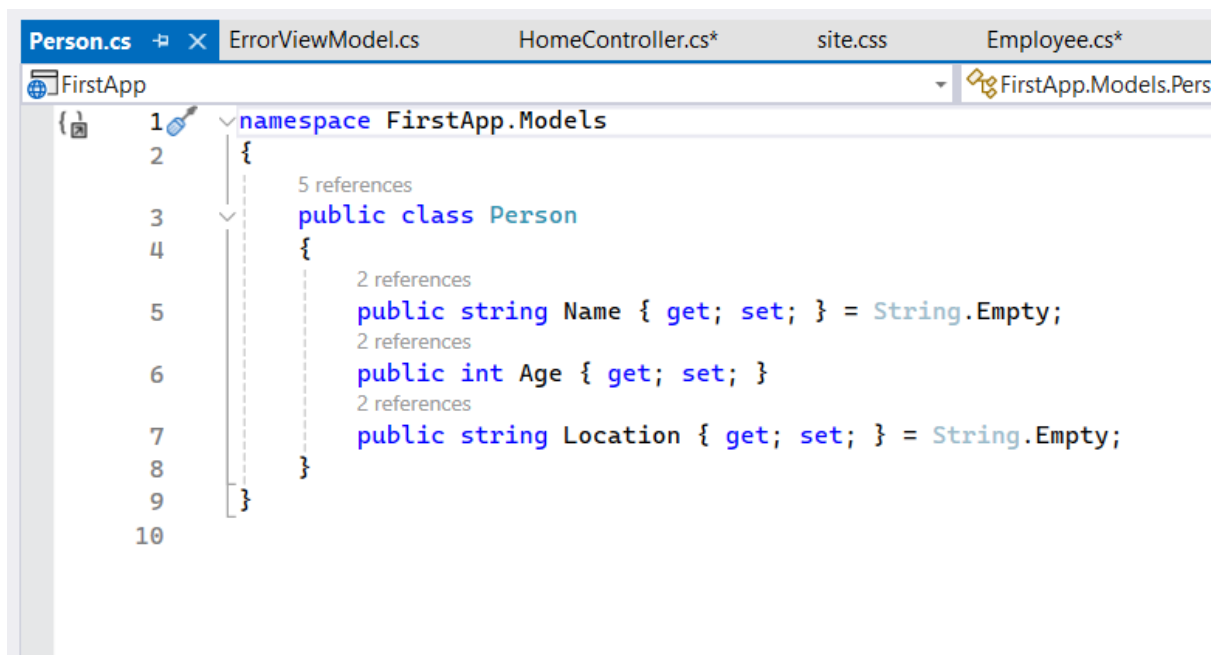
Models/EmployeeModel.cs

```csharp
using System.ComponentModel.DataAnnotations;
namespace FirstApp.Models
{
    // 22 references
    public class Employee
    {
        [Required(ErrorMessage = "Please enter your name")]
        // 19 references
        public string Name { get; set; }

        [Range(16, 60, ErrorMessage = "Ages 16-60 only")]
        // 9 references
        public int Age { get; set; }

        [RegularExpression(@"\d+(\.\d{1,2})?", ErrorMessage = "Invalid, enter like # or #.##")]
        // 9 references
        public decimal Salary { get; set; }

        // 9 references
        public string Department { get; set; }

        [RegularExpression(@"^[MF]+$", ErrorMessage = "Select any one option")]
        // 11 references
        public Char Sex { get; set; }
    }
}
```
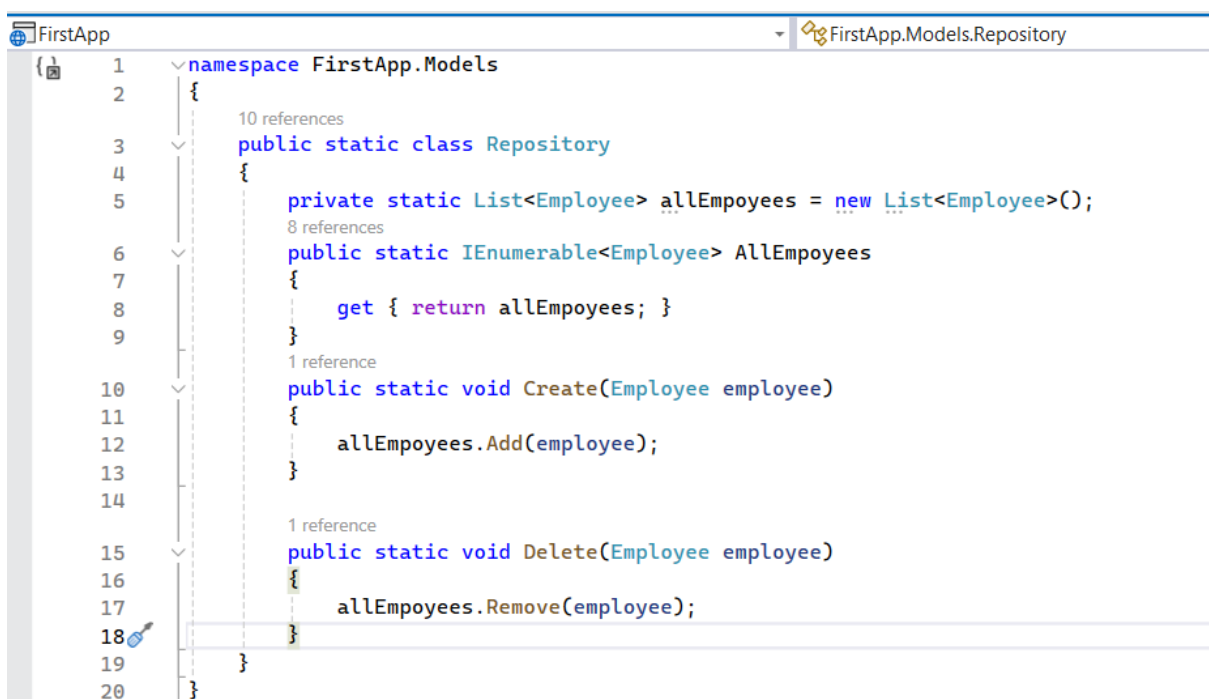
Models/ ErrorViewModel.cs

```csharp
namespace FirstApp.Models
{
    // 4 references
    public class ErrorViewModel
    {
        // 3 references
        public string? RequestId { get; set; }

        // 1 reference
        public bool ShowRequestId => !string.IsNullOrEmpty(RequestId);
    }
}
```

Models/Person.cs

```csharp
namespace FirstApp.Models
{
    5 references
    public class Person
    {
        2 references
        public string Name { get; set; } = String.Empty;
        2 references
        public int Age { get; set; }
        2 references
        public string Location { get; set; } = String.Empty;
    }
}
```

Models/Repository.cs

```csharp
namespace FirstApp.Models
{
    10 references
    public static class Repository
    {
        private static List<Employee> allEmpoyees = new List<Employee>();
        8 references
        public static IEnumerable<Employee> AllEmpoyees
        {
            get { return allEmpoyees; }
        }
        1 reference
        public static void Create(Employee employee)
        {
            allEmpoyees.Add(employee);
        }

        1 reference
        public static void Delete(Employee employee)
        {
            allEmpoyees.Remove(employee);
        }
    }
}
```

Views/Empolyee/create.cshtml

```html
@model Employee
@{
    ViewData["Title"] = "Create";
}
<a asp-action="Index">View All Employees</a>
<h2>Create</h2>
<form method="post">
    <p>
        <label asp-for="Name">Your Name:</label>
        <input asp-for="Name" />
        <span asp-validation-for="Name"></span>
    </p>
    <p>
        <label asp-for="Age">Your Age:</label>
        <input asp-for="Age" />
        <span asp-validation-for="Age"></span>
    </p>
    <p>
        <label asp-for="Salary">Your Salary:</label>
        <input asp-for="Salary" />
        <span asp-validation-for="Salary"></span>
    </p>
    <p>
        <label asp-for="Department">Your Department:</label>
        <select asp-for="Department">
            <option value="Development">Development</option>
            <option value="HR">HR</option>
            <option value="Research">Research</option>
        </select>
        <span asp-validation-for="Department"></span>
    </p>
    <p>
        <label asp-for="Sex">Your Sex:</label>
        <input type="radio" asp-for="Sex" value="M" />Male
        <input type="radio" asp-for="Sex" value="F" />Female
        <span asp-validation-for="Sex"></span>
    </p>
    <p><button type="submit">Submit</button></p>
</form>
```

Views/Empolyee/index.cshtml

```
C# FirstApp

     1      @model IEnumerable<Employee>
     2      @{
     3          ViewData["Title"] = "Index";
     4      }
     5      <a asp-action="Create">Create an Employees</a>
     6
     7      <h2>Here is the list of Employees</h2>
     8      <table style="width:50%">
     9          <thead>
    10              <tr>
    11                  <th>Name</th>
    12                  <th>Age</th>
    13                  <th>Salary</th>
    14                  <th>Department</th>
    15                  <th>Sex</th>
    16                  <th></th>
    17                  <th></th>
    18              </tr>
    19          </thead>
    20          <tbody>
    21              @foreach (Employee e in Model)
    22              {
    23                  <tr>
    24                      <td>@e.Name</td>
    25                      <td>@e.Age</td>
    26                      <td>@e.Salary</td>
```

```
    28                      <td>@e.Sex</td>
    29                      <td><a asp-action="Update" asp-route-empname="@e.Name">Update</a></td>
    30                      <td>
    31                          <form asp-route-empname="@e.Name" asp-action="Delete" method="post">
    32                              <button type="submit">Delete</button>
    33                          </form>
    34                      </td>
    35                  </tr>
    36              }
    37          </tbody>
    38      </table>
```

Views/Empolyee/Thanks.cshtml

```
IrstApp

     1      @model Employee
     2      @{
     3          ViewData["Title"] = "Thanks";
     4      }
     5
     6      <a asp-action="Index">View All Employees</a>
     7      <h2>Thanks, @Model.Name</h2>
```

Views/Empolyee/Update.cshtml

```razor
@model Employee
@{
    ViewData["Title"] = "Update";
}
<a asp-action="Index">View All Employees</a>
<h2>Update</h2>
<form method="post">
    <p>
        <label asp-for="Name">Your Name:</label>
        <input asp-for="Name" />
        <span asp-validation-for="Name"></span>
    </p>
    <p>
        <label asp-for="Age">Your Age:</label>
        <input asp-for="Age" />
        <span asp-validation-for="Age"></span>
    </p>
    <p>
        <label asp-for="Salary">Your Salary:</label>
        <input asp-for="Salary" />
        <span asp-validation-for="Salary"></span>
    </p>
    <p>
        <label asp-for="Department">Your Department:</label>
        <select asp-for="Department">
            <option value="Development">Development</option>
            <span asp-validation-for="Department"></span>
    </p>
    <p>
        <label asp-for="Sex">Your Sex:</label>
        <input type="radio" asp-for="Sex" value="M" />Male
        <input type="radio" asp-for="Sex" value="F" />Female
        <span asp-validation-for="Sex"></span>
    </p>
    <p><button type="submit">Submit</button></p>
</form>
```

Program.cs

```csharp
var builder = WebApplication.CreateBuilder(args);

// Add services to the container.
builder.Services.AddControllersWithViews();

var app = builder.Build();

// Configure the HTTP request pipeline.
if (!app.Environment.IsDevelopment())
{
    app.UseExceptionHandler("/Home/Error");
    // The default HSTS value is 30 days. You may want to change this for
    app.UseHsts();
}

app.UseHttpsRedirection();
app.UseStaticFiles();

app.UseRouting();

app.UseAuthorization();

app.MapControllerRoute(
    name: "default",
    pattern: "{controller=Home}/{action=Index}/{id?}");
```

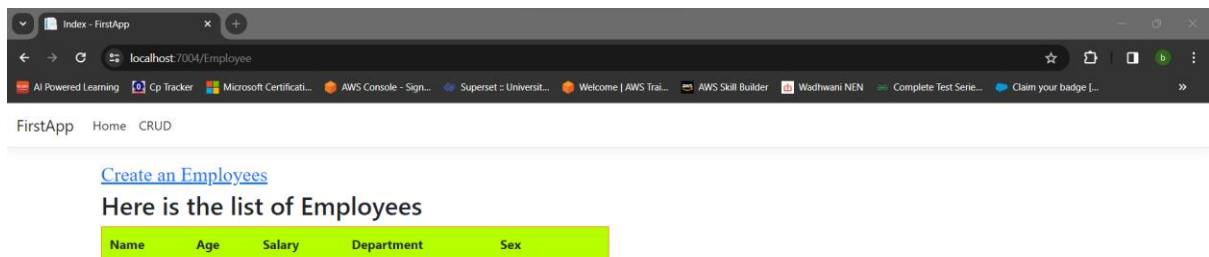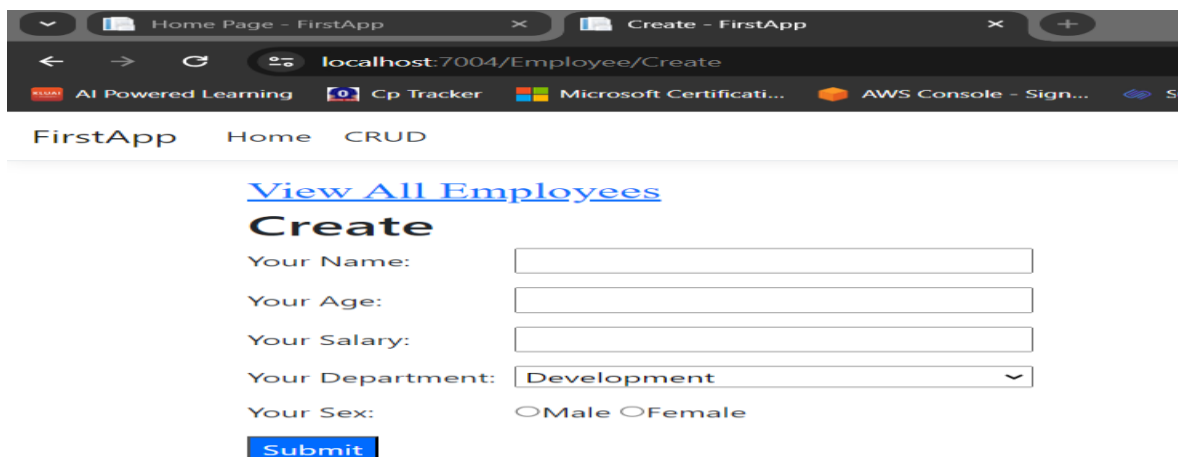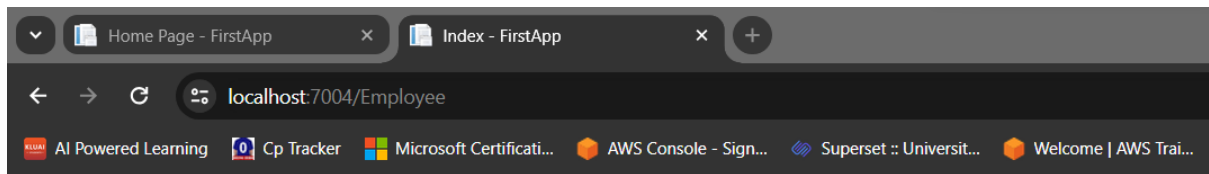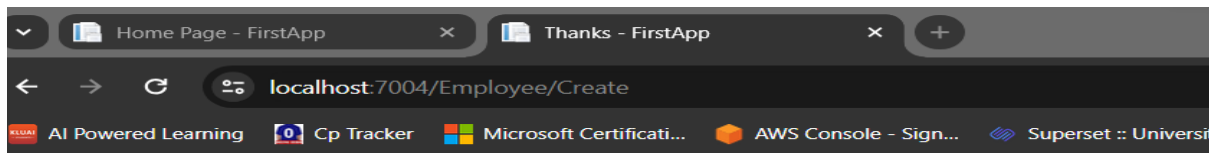**OUTPUT:**

Home page:
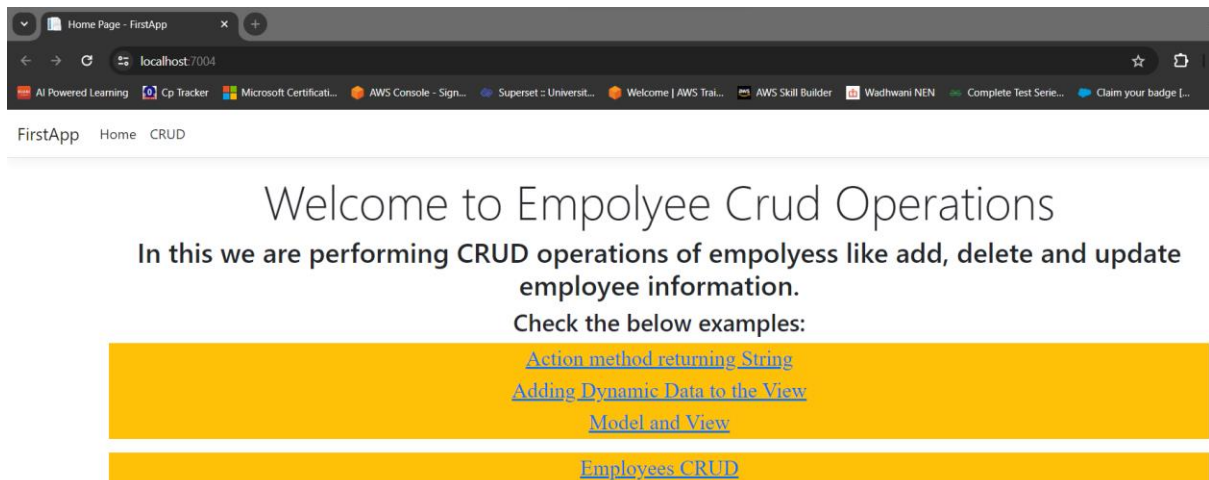
Employee List page

Employee inserting page

After inserting Employees

FirstApp   Home   CRUD

Create an Employees

Here is the list of Employees

Name	Age	Salary	Department	Sex		
Mallampati	30	50000	Development	F	Update	Delete

Updating Employee information

FirstApp   Home   CRUD

View All Employees

Update

Your Name: Bhavana

Your Age: 30

Your Salary: 50000.00

Your Department: Development

Your Sex: ○Male ●Female

Submit

After update:

After inserting any Employee, the output is shown below

After deleting Employee