

Employee Class

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

namespace WebApplication10.Models
{
    public class Employee
    {
        public int id { get; set; }
        public string name { get; set; }
        public string city { get; set; }
        public string country { get; set; }
        public string gender { get; set; }
    }
}
```

Home Controller

```
using System.Collections.Generic;
using System.Web.Mvc;
using WebApplication10.Models;

namespace WebApplication10.Controllers
{
    public class HomeController : Controller
    {
        public ActionResult Index()
        {
            return View();
        }

        public ActionResult About()
        {
            ViewBag.Message = "Your application description page.";

            return View();
        }

        public ActionResult Contact()
        {
            ViewBag.Message = "Your contact page.";

            return View();
        }

        public ActionResult ViewEmployee(Employee ep1)
        {
            List<Employee> employees = new List<Employee> { ep1 };
            return View(employees);
        }
    }
}
```

```

    }

    public ActionResult AddEmployee()
    {
        return View();
    }

    [HttpPost]
    public ActionResult AddEmployee(Employee ep)
    {
        return RedirectToAction("ViewEmployee", "Home", ep);
    }
}

```

Add Employee View

```
@model WebApplication10.Models.Employee
```

```
@{
    ViewBag.Title = "AddEmployee";
    Layout = "~/Views/Shared/_Layout.cshtml";
}
```

```
<h2>AddEmployee</h2>
```

```
@using (Html.BeginForm("AddEmployee", "Home", FormMethod.Post))
{
    @Html.AntiForgeryToken()

    <div class="form-horizontal">
        <h4>Employee</h4>
        <hr />
        @Html.ValidationSummary(true, "", new { @class = "text-danger" })
        <div class="form-group">
            @Html.LabelFor(model => model.id, htmlAttributes: new { @class = "control-label col-md-2" })
            <div class="col-md-10">
                @Html.EditorFor(model => model.id, new { htmlAttributes = new { @class = "form-control" } })
                @Html.ValidationMessageFor(model => model.id, "", new { @class = "text-danger" })
            </div>
        </div>

        <div class="form-group">
            @Html.LabelFor(model => model.name, htmlAttributes: new { @class = "control-label col-md-2" })
            <div class="col-md-10">
                @Html.EditorFor(model => model.name, new { htmlAttributes = new { @class = "form-control" } })
            </div>
        </div>
    </div>
}
```

```

        @Html.ValidationMessageFor(model => model.name, "", new { @class =
"text-danger" })
    </div>
</div>

<div class="form-group">
    @Html.LabelFor(model => model.city, htmlAttributes: new { @class =
"control-label col-md-2" })
    <div class="col-md-10">
        @Html.EditorFor(model => model.city, new { htmlAttributes = new {
@class = "form-control" } })
        @Html.ValidationMessageFor(model => model.city, "", new { @class =
"text-danger" })
    </div>
</div>

<div class="form-group">
    @Html.LabelFor(model => model.country, htmlAttributes: new { @class =
"control-label col-md-2" })
    <div class="col-md-10">
        @Html.EditorFor(model => model.country, new { htmlAttributes = new {
@class = "form-control" } })
        @Html.ValidationMessageFor(model => model.country, "", new { @class
= "text-danger" })
    </div>
</div>

<div class="form-group">
    @Html.LabelFor(model => model.gender, htmlAttributes: new { @class =
"control-label col-md-2" })
    <div class="col-md-10">
        @Html.EditorFor(model => model.gender, new { htmlAttributes = new {
@class = "form-control" } })
        @Html.ValidationMessageFor(model => model.gender, "", new { @class =
"text-danger" })
    </div>
</div>

<div class="form-group">
    <div class="col-md-offset-2 col-md-10">
        <input type="submit" value="Create" class="btn btn-default" />
    </div>
</div>
</div>
}

<div>
    @Html.ActionLink("Back to List", "ViewEmployee")
</div>

@section Scripts {
    @Scripts.Render("~/bundles/jqueryval")
}

```

View Employee View

```
@model IEnumerable<WebApplication10.Models.Employee>

@{
    ViewBag.Title = "ViewEmployee";
    Layout = "~/Views/Shared/_Layout.cshtml";
}

<h2>ViewEmployee</h2>

<p>
    @Html.ActionLink("Create New", "AddEmployee")
</p>
<table class="table">
    <tr>
        <th>
            ID
        </th>
        <th>
            Name
        </th>
        <th>
            City
        </th>
        <th>
            Country
        </th>
        <th>
            Gender
        </th>
    </tr>

    @foreach (var emp in Model)
    {
        <tr>
            <td>
                @Html.DisplayFor(modelItem => emp.id)
            </td>
            <td>
                @Html.DisplayFor(modelItem => emp.name)
            </td>
            <td>
                @Html.DisplayFor(modelItem => emp.city)
            </td>
            <td>
                @Html.DisplayFor(modelItem => emp.country)
            </td>
            <td>
                @Html.DisplayFor(modelItem => emp.gender)
            </td>
            <td>
                @Html.ActionLink("Edit", "Edit", new { id = emp.id }) |
                @Html.ActionLink("Details", "Details", new { id = emp.id }) |
                @Html.ActionLink("Delete", "Delete", new { id = emp.id })
            </td>
        </tr>
    }
</table>
```

```
    }  
  </tr>  
</table>
```

Add Employee Output

The screenshot displays a web browser window with the address `https://localhost:44324/Home/AddEmployee`. The page features a dark navigation bar with the application name and several menu items. The main content area is titled 'AddEmployee' and 'Employee'. It contains a form with the following fields and values:

Field	Value
id	01
name	Azlan Ali
city	Karachi
country	Pakistan
gender	Male

Below the form is a 'Create' button and a 'Back to List' link. The footer of the page reads '© 2024 - My ASP.NET Application'.

View Employee Output

