

Student:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.ComponentModel.DataAnnotations;
using System.ComponentModel.DataAnnotations.Schema;

namespace LVC.Models //DO NOT change the namespace name
{
    //Fill your code here
    public class Student{
        [Key]
        [DatabaseGenerated(DatabaseGeneratedOption.Identity)]
        public int ID{get;set;}

        [Required(ErrorMessage="User Name is Mandatory")]
        [StringLength(30,ErrorMessage="Invalid User Name")]
        public string Name{get;set;}

        [Required(ErrorMessage="Email is Mandatory")]
        [RegularExpression(@"[a-z0-9._%+-]+@[a-z0-9.-]+\.[a-z]{2,4}",ErrorMessage="Invalid Email Id")]
        public string Email{get;set;}

        [Required(ErrorMessage="Mobile is Mandatory")]
        [RegularExpression(@"[0-9]{10}",ErrorMessage="Invalid Mobile No")]
        public string Mobile{get;set;}

        public string Remarks{get;set;}
    }
}
```

---

StudentContext:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Data.Entity;
using System.Data.Entity.Infrastructure;
namespace LVC.Models //DO NOT change the namespace name
{
    public class StudentContext : DbContext //DO NOT change the class name
    {
        public StudentContext():base("name=LVCDB") //DO NOT change the connection
        name
        {
        }
        //Fill your code here
        public DbSet<Student> Students{get;set;}
    }
}
```

---

StudentController:

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;
using LVC.Models;

//Do not change the namespace name
namespace LVC.Controllers
{
    public class StudentController : Controller //Do not change the class name
    {
        public StudentContext db;
        public ActionResult Index()
        {
            //Fill your code here
            db=new StudentContext();
            List<Student> student=db.Students.ToList<Student>();
            return View(student);
        }
        public ActionResult Create()
        {
            return View();
        }

        [HttpPost]
        public ActionResult Create(Student student){
            db=new StudentContext();
            db.Students.Add(student);
            if(db.SaveChanges(>0){
                ViewBag.Message="Registration completed successfully";
            }
            else{
                ViewBag.Message="Error while registration, please contact the LVC
admin";
            }
            return View(student);
        }

        //Implement your code here

    }
}

```

---

\_Layout:

```

<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>View All Registrations</title>
    <link href="~/Content/Site.css" rel="stylesheet" type="text/css" />
    <link href="~/Content/bootstrap.min.css" rel="stylesheet" type="text/css" />
    <script src="~/Scripts/modernizr-2.8.3.js"></script>
</head>

```

```

<body>
    <div class="navbar navbar-inverse navbar-fixed-top">
        <div class="container">
            <div class="navbar-header">
                <button type="button" class="navbar-toggle" data-toggle="collapse"
data-target=".navbar-collapse">
                    <span class="icon-bar"></span>
                    <span class="icon-bar"></span>
                    <span class="icon-bar"></span>
                </button>
                @Html.ActionLink("Live Virtual Class", "Index", "Home", new { area
= "" }, new { @class = "navbar-brand" })
            </div>
            <div class="navbar-collapse collapse">
                <ul class="nav navbar-nav"></ul>
            </div>
        </div>
    </div>

    <div class="container body-content">
        @RenderBody()
        <hr />
        <footer>
            <p>&copy; @DateTime.Now.Year - LVC</p>
        </footer>
    </div>

    <script src="~/Scripts/jquery-3.3.1.min.js"></script>
    <script src="~/Scripts/bootstrap.min.js"></script>
</body>
</html>

```

-----

\_ViewStart:

```

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

```

-----

Create:

```
@model LVC.Models.Student
```

```

@{
    ViewBag.Title = "Register";
}
<h2>Register User</h2>

```

```

@using (Html.BeginForm("Create", "Student", FormMethod.Post))
{
    @Html.AntiForgeryToken()
    @Html.LabelFor(m=>m.Name)
    @Html.TextBoxFor(m=>m.Name, new {id="Name"})
    @Html.LabelFor(m=>m.Email)
    @Html.TextBoxFor(m=>m.Email, new {id="Email"})
    @Html.LabelFor(m=>m.Mobile)
    @Html.TextBoxFor(m=>m.Mobile, new {id="Mobile"})
}

```

```

        @Html.LabelFor(m=>m.Remarks)
        @Html.TextBoxFor(m=>m.Remarks, new {id="Remarks"})
        <input type="submit" id="Submit" value="Register" />
    }
    <span id="message">
        @ViewBag.Message
    </span>

<div>
    @Html.ActionLink("View All Registration", "Index", "Student")
</div>

<script src="~/Scripts/jquery-3.3.1.min.js"></script>
<script src="~/Scripts/jquery.validate.min.js"></script>
<script src="~/Scripts/jquery.validate.unobtrusive.min.js"></script>
-----

```

Index:

```

@model IEnumerable<LVC.Models.Student>

@{
    ViewBag.Title = "View All Registration";
}
<html>
<body>
<h2>User List</h2>

<p>
    @Html.ActionLink("Register", "Create", "Student")
</p>

<table border="1">
    <tr>
        <th>@Html.DisplayNameFor(m=>m.ID)</th>
        <th>@Html.DisplayNameFor(m=>m.Name)</th>
        <th>@Html.DisplayNameFor(m=>m.Email)</th>
        <th>@Html.DisplayNameFor(m=>m.Mobile)</th>
        <th>@Html.DisplayNameFor(m=>m.Remarks)</th>
    </tr>
    @foreach(var s in Model){
        <tr>
            <td>@Html.DisplayFor(m=>s.ID)</td>
            <td>@Html.DisplayFor(m=>s.Name)</td>
            <td>@Html.DisplayFor(m=>s.Email)</td>
            <td>@Html.DisplayFor(m=>s.Mobile)</td>
            <td>@Html.DisplayFor(m=>s.Remarks)</td>
        </tr>
    }
</table>
</body>
</html>

```