



YASH NAKRANI

{ MODULE - 5 }



TOPS TECHNOLOGIES

1. Explain Areas in MVC?

Ans A MVC Application can contain several MVC structures (areas). Areas are the small functional units with its own set of Model, View and Controller MVC Application can have any number of areas. Each area has its own controllers, models and views. Areas are put under separate folders called areas.

2. Explain Render Section in MVC?

Ans Render Section(String) In layout pages, renders the content of the section named name . Render Section(String, Boolean) In layout pages, renders the content of the section named name .

3. Which assembly is the MVC framework is defined?

Ans The MVC framework is defined through System.Web.Mvc assembly. This is because this is the only assembly which contains classes and interfaces that support the ASP.NET Model View Controller (MVC) framework for creating Web application.

4. Apply validations using data annotation attributes.

Ans MODEL CLASS:-

```
using System.ComponentModel.DataAnnotations;
using Xunit;
using Xunit.Sdk;

namespace p1.Models
{
    public class data
    {
        [Required(ErrorMessage = "Enter First Name")]
        [RegularExpression("[A-Za-z]+", ErrorMessage = "Only alphabets.!")]
        [Display(Name = "First Name")]
        public string F_name { get; set; }

        [Required(ErrorMessage = "Enter Last")]
        [RegularExpression("[a-zA-Z]+", ErrorMessage = "only alphabets.!")]
        [Display(Name = "Last Name")]
        public string L_name { get; set; }

        [Required(ErrorMessage = "Mobile no")]
        [RegularExpression("[0-9]+", ErrorMessage = "only numbers.!")]
        [Display(Name = "Mobile No")]
        public int mobile { get; set; }
    }
}
```

```
[Required(ErrorMessage = "Enter Email")]
[EmailAddress(ErrorMessage = "enter vaild email.!")]
[Display(Name = "Email")]
public string Email { get; set; }

[Required(ErrorMessage = "Select Gender")]
[Display(Name = "Gender")]
public string Gender { get; set; }

[Required(ErrorMessage = "Select City")]
[Display(Name = "City")]
public string City { get; set; }

[Required(ErrorMessage = "Select Hobbies")]
[Display(Name = "Hobbies")]
public string Hobbies { get; set; }

[Required(ErrorMessage = "Enter Date Of Birth")]
[Display(Name = "Date Of Birth")]
public string DOB { get; set; }

[Required(ErrorMessage = "Enter Address")]
[Display(Name = "Address")]
public string Address { get; set; }

[Required(ErrorMessage = "Uploade File")]
[Display(Name = "CV File")]
public string Files { get; set; }
}
}
```

5.What is difference between javascript and jquery?

Ans

Javascript VS jQuery	
Javascript	jQuery
Javascript is a scripting Language	jQuery is a framework. It is javascript Library
Example: ID Selector: <pre>var \$el = document.querySelector("#hello");</pre> Class Selector: <pre>Function changeBachground(color) { Document.body.style.background = color; }</pre> Onload="changeBachground ('red');"	Example; ID Selector: <pre>var\$el = document.querySelector("#hello");</pre> Class Seletor: <pre>\$('body') .css ('background', '#ccc');</pre>

6. What is Linq?

Ans Language Integrated Query proposes a compact significant and understandable syntax for operating data. Language Integrated Query is a Microsoft programming tactic and model which effectively includes formal query abilities into Microsoft .NET programming languages. It has a need for the presence of particular language extensions.

7. What is difference between Linq and Entity Framework.?

Ans Short for **Language INtegrated Query**, LINQ is set of classes added to the .NET Framework 3.5 along with language enhancements added to C# 3.0 and Visual Basic 9, the versions of the language that ship with Visual Studio 2008. LINQ adds a rich, standardized query syntax as a first-class citizen in .NET programming languages that allows developers to interact with any type of data.

Entity Framework (EF) is an object-relational mapper that enables .NET developers to work with relational data using domain-specific objects. It eliminates the need for most of the data-access code that developers usually need to write. Get itAdd this to your project and start taking advantage of this powerful O/RM.

8. Create One Registration Page for Student Registration Model Class and When Student Click on Submit Button, Details of student must be display on same page.

Fields : First Name, Last Name, Email id, Mobile, Gender, City, Address, Course (Dropdown list), Fees, Tax (Label with Text 12.50%), Total Fees.

Ans

8. Create View Model for CRUD Operation in Country, State and City Table with validation using EF.

Ans

9. What is Data Annotation Validator Attributes in MVC?

Ans Data annotation attributes are attached to the properties of the model class and enforce some validation criteria. They are capable of performing validation on the server side as well as on the client side. This article discusses the basics of using these attributes in an ASP.NET MVC application.

10. What are AJAX Helpers in MVC?

Ans AJAX Helpers are used to create AJAX enabled elements like as Ajax enabled forms and links which performs request asynchronously. AJAX Helpers are extension methods of AJAXHelper class which exist in System. Web. Mvc namespace.

11. Write a program to perform join on two lists.

Ans **CONTROLLER:-**

```
using DoubleTablesASPNETMVC5.Models.DB;  
using DoubleTablesASPNETMVC5.ViewModels.Home;  
using System.Collections.Generic;
```

```
using System.Web.Mvc;

namespace DoubleTablesASPNETMVC5.Controllers
{
    public class HomeController : Controller
    {
        // GET: Home
        public ActionResult Index()
        {
            StudentViewModels svm = new StudentViewModels();
            List<Student> students = svm.GetAllStudents();

            return View(students);
        }
    }
}
```

VIEW:-

```
@model IEnumerable<DoubleTablesASPNETMVC5.Models.DB.Student>

@{
    ViewBag.Title = "Index";
}

<h2>Index</h2>

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

@if (Model.Count() > 0)
{
    <table class="table table-condensed table-hover table-striped">
    <tr>
        <th>
            @Html.DisplayNameFor(model => model.FullName)
        </th>
        <th>
            @Html.DisplayNameFor(model => model.Email)
        </th>
        <th>
            @Html.DisplayNameFor(model => model.Gender)
        </th>
        <th>
            @Html.DisplayNameFor(model => model.Mobile)
        </th>
        <th>
```

```
        @Html.DisplayNameFor(model => model.Telephone)
    </th>
    <th>
        @Html.DisplayNameFor(model => model.Notes)
    </th>
    <th></th>
</tr>

@foreach (var item in Model)
{
    <tr>
        <td>
            @Html.DisplayFor(modelItem => item.FullName)
        </td>
        <td>
            @Html.DisplayFor(modelItem => item.Email)
        </td>
        <td>
            @Html.DisplayFor(modelItem => item.Gender)
        </td>
        <td>
            @Html.DisplayFor(modelItem => item.Mobile)
        </td>
        <td>
            @Html.DisplayFor(modelItem => item.Telephone)
        </td>
        <td>
            @Html.DisplayFor(modelItem => item.Notes)
        </td>
        <td>
            @Html.ActionLink("Edit", "Edit", new { id = item.StudentId }) |
            @Html.ActionLink("Details", "Details", new { id = item.StudentId }) |
            @Html.ActionLink("Delete", "Delete", new { id = item.StudentId })
        </td>
    </tr>
}

</table>
}
else
{
    <div class="alert alert-success">
        <p>There is currently no record in the database.</p>
    </div>
}
```

12. Perform CRUD operation using LINQ

Ans `using` System;

`using` System.Collections.Generic;

`using` System.Linq;

`using` System.Web;

`using` System.Web.UI;

`using` System.Web.UI.WebControls;

`namespace` JQueryImageRotator

{

`public` partial `class` WebForm1 : System.Web.UI.Page

 {

`protected void` Page_Load(`object` sender, EventArgs e)

 {

 }

`private void` RetrieveUserDetails()

 {

 DataClasses1DataContext dtContext = `new` DataClasses1DataContext();

 GridView1.DataSource = dtContext.UserDetails;

 GridView1.DataBind();

 }

`protected void` btnSelect_Click(`object` sender, EventArgs e)

 {

 RetrieveUserDetails();

 }


```
protected void btnInsert_Click(object sender, EventArgs e)
{
    using (DataClasses1DataContext dtContext = new DataClasses1DataContext())
    {
        UserDetail Udetails = new UserDetail {

            UserName="Shridhar",
            City="New Delhi",
            Designation="SE"
        };

        dtContext.UserDetails.InsertOnSubmit(Udetails);
        dtContext.SubmitChanges();
    }
    RetrieveUserDetails();
}
```

```
protected void btnUpdate_Click(object sender, EventArgs e)
{
    using (DataClasses1DataContext dtContext = new DataClasses1DataContext())
    {
        UserDetail Udetails = dtContext.UserDetails.SingleOrDefault(x => x.UserId == 6);

        Udetails.Designation = "Leader";
        dtContext.SubmitChanges();
    }
    RetrieveUserDetails();
}
```

```
protected void btnDelete_Click(object sender, EventArgs e)
{

```

```
using (DataClasses1DataContext dtContext = new DataClasses1DataContext())
{
    UserDetail Udetails = dtContext.UserDetails.SingleOrDefault(x => x.UserId == 5);

    dtContext.UserDetails.DeleteOnSubmit(Udetails);
    dtContext.SubmitChanges();
}
RetrieveUserDetails();
}
}
}
```

13. What are required attributes for Ajax call?

Ans

The parameters specifies one or more name/value pairs for the AJAX request. The data type expected of the server response. A function to run if the request fails. A Boolean value specifying whether a request is only successful if the response has changed since the last request.

14. Create insert/update/delete/view on same view using AJAX

Ans `$(".save").live("click", function () {`

`var id = $("#grid tbody tr").length;`

`var Student_Name = $("#Student_Name_" + id).val();`

`var Student_Branch = $("#Student_Branch_" + id).val();`

`var Student_City = $("#Student_City_" + id).val();`

`var Student_State = $("#Student_State_" + id).val();`

`if (id != "") {`

`$.ajax({`

```

        type: "GET",
        contentType: "application/json; charset=utf-8",
        url: '@Url.Action("SaveRecord", "ShowData")',
        data: { "Student_Name": Student_Name, "Student_Branch": Student_Branch, "Student_City": Student_City, "Student_State": Student_State },
        dataType: "json",
        beforeSend: function () { },
        success: function (data) {
            if (data.result == true) {
                $("#divmsg").html("Record has been saved successfully !!");
                setTimeout(function () { window.location.replace("WebGridCRUD"); }, 2000
            );
            }
            else {
                alert("There is some error");
            }
        }
    });
}
});

```

```

$(".edit").live("click", function () {
    var str = $(this).attr("id").split("_");
    id = str[1];
    var Student_Name = "#Student_Name_" + id;
    var spanStudent_Name = "#spanStudent_Name_" + id;
    var Student_Branch = "#Student_Branch_" + id;
    var spanStudent_Branch = "#spanStudent_Branch_" + id;

```

```
var Student_City = "#Student_City_" + id;
var spanStudent_City = "#spanStudent_City_" + id;
var Student_State = "#Student_State_" + id;
var spanStudent_State = "#spanStudent_State_" + id;
$(Student_Name).show();
$(spanStudent_Name).hide();
$(Student_Branch).show();
$(spanStudent_Branch).hide();
$(Student_City).show();
$(spanStudent_City).hide();
$(Student_State).show();
$(spanStudent_State).hide();
$(this).hide();
$("#Update_" + id).show();
$("#Cancel_" + id).show();
});
```

```
$(".delete").live("click", function () {
var str = $(this).attr("id").split("_");
id = str[1];
var flag = confirm('Are you sure to delete ??');
if (id != "" && flag) {
$.ajax({
type: "GET",
```

```
contentType: "application/json; charset=utf-8",
url: '@Url.Action("DeleteRecord", "ShowData")',
data: { "id": id },
dataType: "json",
beforeSend: function () { },
success: function (data) {
    if (data.result == true) {
        $("#Update_" + id).parents("tr").remove();
    }
    else {
        alert('There is some error');
    }
}
});
/
```

15. What is JsonResultType in MVC?

Ans JsonResult is one of the type of MVC action result type which returns the data back to the view or the browser in the form of JSON (JavaScript Object notation format).

16. What is the meaning of Unobtrusive JavaScript?

Ans Unobtrusive JavaScript is a general term that conveys a general set of guidelines or margins to the term REST. REST is nothing but the Representational State Transfer. We can explain Unobtrusive JavaScript as- it is not your particular JavaScript code that you generally use in your markup page.

17. What are the sub types of ActionResult?

Ans **ViewResult** - Renders a specified view to the response stream

PartialViewResult - Renders a specified partial view to the response stream

EmptyResult - An empty response is returned

RedirectResult - Performs an HTTP redirection to a specified URL

RedirectToRouteResult - Performs an HTTP redirection to a URL that is determined by the routing engine, based on given route data.

JsonResult - Serializes a given ViewData object to JSON format

JavaScriptResult - Returns a piece of JavaScript code that can be executed on the client

ContentResult - Writes content to the response stream without requiring a view

FileContentResult - Returns a file to the client

FileStreamResult - Returns a file to the client, which is provided by a Stream

FilePathResult - Returns a file to the client.

18. Get Data using FormCollection and store data into session and show on another view.

```
Ans    public ActionResult Insert(FormCollection Fc)
    {
        using (MVC_PracticeEntities objContext = new MVC_PracticeEntities())
        {
            tbl_FormCollectionUse Tbl = new tbl_FormCollectionUse();
            //
            Tbl.Name = Fc["TxtName"].ToString();
            Tbl.Mobile = Convert.ToInt64(Fc["Mobile"]);
            Tbl.EmailID = Fc["TxtEmailID"].ToString();
            Tbl.Address = Fc["TxtAddress"].ToString();
            //
            objContext.AddTotbl_FormCollectionUse(Tbl);
            //
            int i = objContext.SaveChanges();
            if (i > 0)
            {
                ViewBag.Msg = "Data Saved Suuceessfully.";
            }
        }

        //_Name=F
        return View();
    }
```

```
}  
  
}  
  
}
```

20. Bind Following Controls using Model Class Dynamically DropDownList, ListBox, CheckBoxList

Ans

21. Different way to return view in MVC?

Ans

```
return View()  
return RedirectToAction()  
return Redirect()  
return RedirectToRoute()
```

22. Create Multiple Image Upload Example, Image Display after Upload. (Use Database table for save data like image name, size, type, URL)

Ans