[Year]

ASP.net - MVC 3

BASIC DISCUSSION

MD. SAIFULLAH AL AZAD

Contents

What is MVC (Model view controller)?

MVC is architectural pattern which separates the representation and the user interaction. It’s divided in three broader sections, “Model”, “View” and “Controller”. Below is how each one of them handles the task.

- The “View” is responsible for look and feel.
- “Model” represents the real world object and provides data to the “View”.
- The “Controller” is responsible to take the end user request and load the appropriate “Model” and “View”.

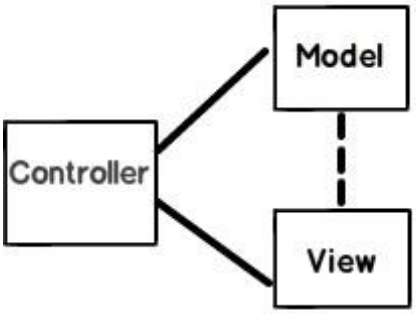


Figure: - MVC (Model view controller)

Can you explain the complete flow of MVC?

Below are the steps how control flows in MVC (Model, view and controller) architecture:-

- All end user requests are first sent to the controller.
- The controller depending on the request decides which model to load. The controller loads the model and attaches the model with the appropriate view.
- The final view is then attached with the model data and sent as a response to the end user on the browser.

Is MVC suitable for both windows and web application?

MVC architecture is suited for web application than windows. For window application MVP i.e. “Model view presenter” is more applicable.IfyouareusingWPFandSLMVVMismoresuitableduetobindings.

What are the benefits of using MVC?

There are two big benefits of MVC:-

Separation of concerns is achieved as we are moving the code behind to a separate class file. By moving the binding code to a separate class file we can reuse the code to a great extent.

Automated UI testing is possible because now the behind code (UI interaction code) has moved to a simple.NET class. This gives us opportunity to write unit tests and automate manual testing.

Is MVC different from a 3 layered architecture?

MVC is an evolution of a 3 layered traditional architecture. Many components of 3 layered architecture are part of MVC. So below is how the mapping goes.

Functionality	**3 layered / tiered architecture**	**Model view controller architecture**
Look and Feel	User interface.	View.
UI logic	User interface.	Controller
Business logic /validations	Middle layer	Model.
Request is first sent to	User interface	Controller.
Accessing data	Data access layer.	Data access layer.

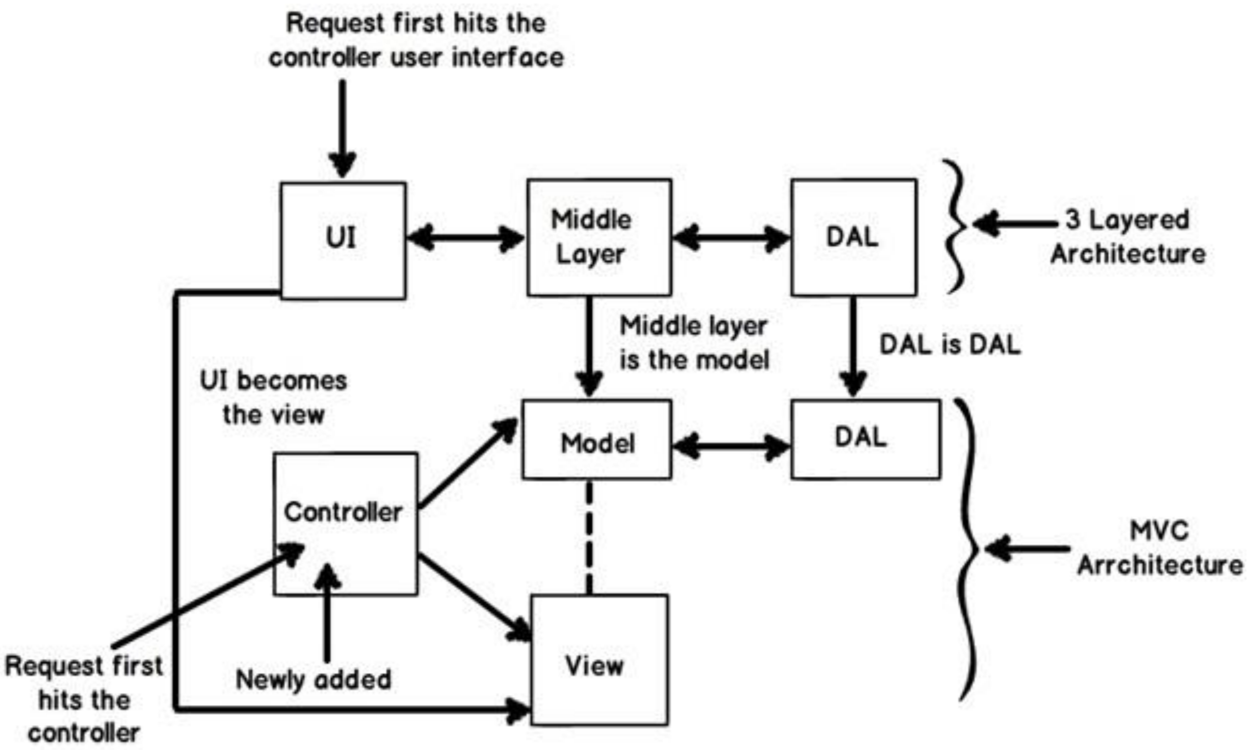


Figure: - 3 layered architecture

What is the latest version of MVC?

When this note was written, four versions where released of MVC. MVC 1 , MVC 2, MVC 3 and MVC 4. So the latest is MVC 4.

What is the difference between each version of MVC?

Below is a detail table of differences. But during interview it's difficult to talk about all of them due to time limitation. So I have highlighted important differences which you can run through before the interviewer.

MVC 2	**MVC 3**	**MVC 4**
Client-Side Validation Templated Helpers Areas Asynchronous Controllers Html.ValidationSummary Helper Method DefaultValueAttribute in Action-Method Parameters Binding Binary Data with Model Binders DataAnnotations Attributes Model-Validator Providers New RequireHttpsAttribute Action Filter Templated Helpers Display Model-Level Errors	**Razor** Readymade project templates **HTML 5 enabled templatesSupport for Multiple View EnginesJavaScript and Ajax** Model Validation Improvements	**ASP.NET Web A** Refreshed and mod templatesNew mob **Many new feature apps** Enhanced support f methods

What are routing in MVC?

Routing helps you to define a URL structure and map the URL with the controller.

For instance let’s say we want that when any user types “http://localhost/View/ViewCustomer/”, it goes to the “Customer” Controller and invokes “DisplayCustomer” action. This is defined by adding an entry in to the “routes” collection using the “maproute” function. Below is the under lined code which shows how the URL structure and mapping with controller and action is defined.

⊟Collapse | Copy Code

```
routes.MapRoute(
               "View", // Route name
               "View/ViewCustomer/{id}", // URL with parameters
               new { controller = "Customer", action = "DisplayCustomer",
id = UrlParameter.Optional }); // Parameter defaults
```

Where is the route mapping code written?

The route mapping code is written in the “global.asax” file.

Can we map multiple URL’s to the same action?

Yes , you can , you just need to make two entries with different key names and specify the same controller and action.

How can we navigate from one view to other view using hyperlink?

By using “ActionLink” method as shown in the below code. The below code will create a simple URL which help to navigate to the “Home” controller and invoke the “GotoHome” action.

⊟Collapse | Copy Code

```
<%= Html.ActionLink("Home","Gotohome") %>
```

How can we restrict MVC actions to be invoked only by GET or POST?

We can decorate the MVC action by “HttpGet” or “HttpPost” attribute to restrict the type of HTTP calls. For instance you can see in the below code snippet the “DisplayCustomer” action can only be invoked by “HttpGet”. If we try to make Http post on “DisplayCustomer” it will throw an error.

⊟Collapse | Copy Code

```
[HttpGet]
        public ViewResult DisplayCustomer(int id)
        {
            Customer objCustomer = Customers[id];
            return View("DisplayCustomer",objCustomer);
        }
```

How can we maintain session in MVC?

Sessions can be maintained in MVC by 3 ways tempdata ,viewdata and viewbag.

What is the difference between tempdata ,viewdata and viewbag?

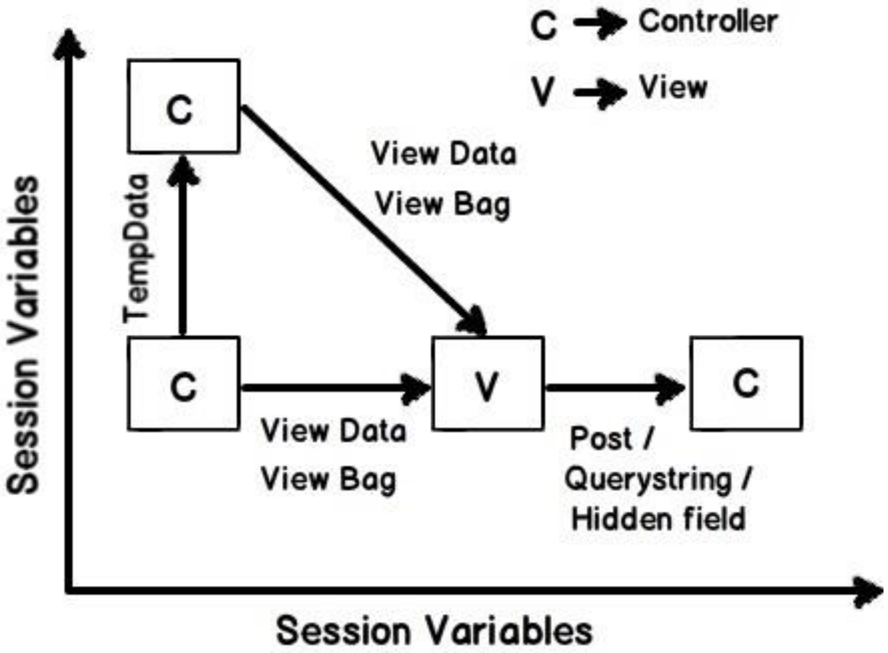


Figure:- difference between tempdata , viewdata and viewbag

Temp data: -Helps to maintain data when you move from one controller to other controller or from one action to other action. In other words when you redirect,“tempdata” helps to maintain data between those redirects. It internally uses session variables.

View data: - Helps to maintain data when you move from controller to view.

View Bag: - It’s a dynamic wrapper around view data. When you use “Viewbag” type casting is not required. It uses the dynamic keyword internally.

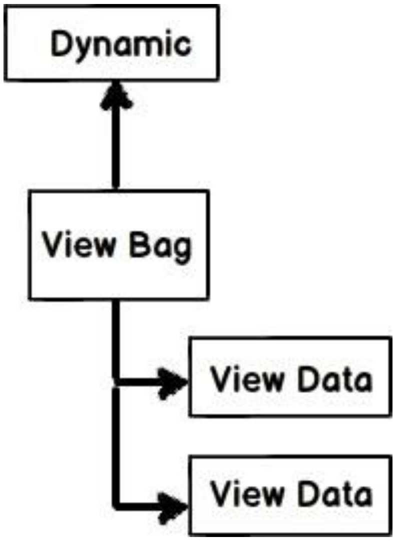


Figure:-dynamic keyword

Session variables: - By using session variables we can maintain data from any entity to any entity.

Hidden fields and HTML controls: - Helps to maintain data from UI to controller only. So you can send data from HTML controls or hidden fields to the controller using POST or GET HTTP methods.

Below is a summary table which shows different mechanism of persistence.

Maintains data between	**ViewData/ViewBag**	**TempData**	**Hidden fields**	**Session**
Controller to Controller	No	Yes	No	Yes
Controller to View	Yes	No	No	Yes
View to Controller	No	No	Yes	Yes

What are partial views in MVC?

Partial view is a reusable view (like a user control) which can be embedded inside other view. For example let’s say all your pages of your site have a standard structure with left menu, header and footer as shown in the image below.

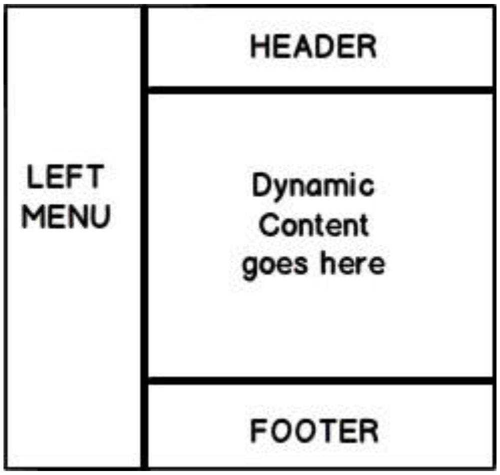
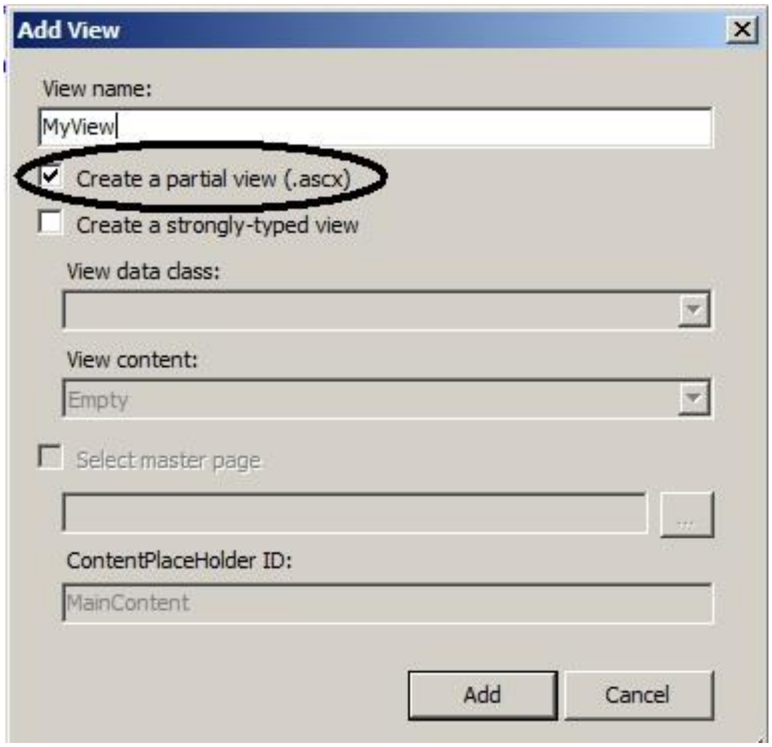


Figure:- partial views in MVC

For every page you would like to reuse the left menu, header and footer controls. So you can go and create partial views for each of these items and then you call that partial view in the main view.

How did you create partial view and consume the same?

When you add a view to your project you need to check the “Create partial view” check box.



Add View

View name: MyView

☑ Create a partial view (.ascx)

□ Create a strongly-typed view

View data class:

View content: Empty

□ Select master page

ContentPlaceHolder ID: MainContent

Figure:-createpartialview

Once the partial view is created you can then call the partial view in the main view using "Html.RenderPartial" method as shown in the below code snippet.

⊟Collapse | Copy Code

```
<body>
<div>
<% Html.RenderPartial("MyView"); %>
</div>
</body>
```

How can we do validations in MVC?

One of the easy ways of doing validation in MVC is by using data annotations. Data annotations are nothing but attributes which you can be applied on the model properties. For example in the below code snippet we have a simple "customer" class with a property "customercode".

This"CustomerCode" property is tagged with a "Required" data annotation attribute. In other words if this model is not provided customer code it will not accept the same.

⊟Collapse | Copy Code

```
public class Customer
{
        [Required(ErrorMessage="Customer code is required")]
        public string CustomerCode
        {
            set;
            get;
        }
}
```

In order to display the validation error message we need to use "ValidateMessageFor" method which belongs to the "Html" helper class.

⊟Collapse | Copy Code

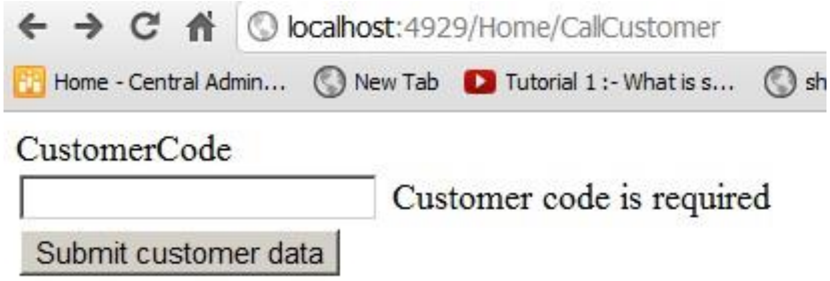
```
<% using (Html.BeginForm("PostCustomer", "Home", FormMethod.Post))
{ %>
<%=Html.TextBoxFor(m => m.CustomerCode)%>
<%=Html.ValidationMessageFor(m => m.CustomerCode)%>
<input type="submit" value="Submit customer data" />
<%}%>
```

Later in the controller we can check if the model is proper or not by using "ModelState.IsValid" property and accordingly we can take actions.

⊟Collapse | Copy Code

```
public ActionResult PostCustomer(Customer obj)
{
if (ModelState.IsValid)
{
                obj.Save();
                return View("Thanks");
}
else
{
                    return View("Customer");
}
}
```

Below is a simple view of how the error message is displayed on the view.



CustomerCode

Customer code is required

Figure:- validations in MVC

Can we display all errors in one go?

Yes we can, use “ValidationSummary” method from HTML helper class.

⊟Collapse | Copy Code

```
<%= Html.ValidationSummary() %>
```

What are the other data annotation attributes for validation in MVC?

If you want to check string length, you can use “StringLength”.

⊟Collapse | Copy Code

```
[StringLength(160)]
public string FirstName { get; set; }
```

In case you want to use regular expression, you can use “RegularExpression” attribute.

⊟Collapse | Copy Code

```
[RegularExpression(@"[A-Za-z0-9._%+-]+@[A-Za-z0-9.-]+\.[A-Za-z]{2,4}")]public
string Email { get; set; }
```

If you want to check whether the numbers are in range, you can use the “Range” attribute.

⊟Collapse | Copy Code

```
[Range(10,25)]public int Age { get; set; }
```

Some time you would like to compare value of one field with other field, we can use the “Compare” attribute.

⊟Collapse | Copy Code

```
public string Password { get; set; }[Compare("Password")]public string
ConfirmPass { get; set; }
```

In case you want to get a particular error message , you can use the “Errors” collection.

⊟Collapse | Copy Code

```
var ErrMessage = ModelState["Email"].Errors[0].ErrorMessage;
```

If you have created the model object yourself you can explicitly call “TryUpdateModel” in your controller to check if the object is valid or not.

⊟Collapse | Copy Code

```
TryUpdateModel(NewCustomer);
```

In case you want add errors in the controller you can use “AddModelError” function.

⊟Collapse | Copy Code

```
ModelState.AddModelError("FirstName", "This is my server-side error.");
```

How can we enable data annotation validation on client side?

It’s a two-step process first reference the necessary jquery files.

⊟Collapse | Copy Code

```
<script src="<%= Url.Content("~/Scripts/jquery-1.5.1.js") %>"
type="text/javascript"></script>
<script src="<%= Url.Content("~/Scripts/jquery.validate.js") %>"
type="text/javascript"></script>
<script src="<%= Url.Content("~/Scripts/jquery.validate.unobtrusive.js") %>"
type="text/javascript"></script>
```

Second step is to call “EnableClientValidation” method.

⊟Collapse | Copy Code

```
<% Html.EnableClientValidation(); %>
```

What is razor in MVC?

It’s a light weight view engine. Till MVC we had only one view type i.e.ASPX, Razor was introduced in MVC 3.

Why razor when we already had ASPX?

Razor is clean, lightweight and syntaxes are easy as compared to ASPX. For example in ASPX to display simple time we need to write.

⊟Collapse | Copy Code

```
<%=DateTime.Now%>
```

In Razor it's just one line of code.

⊟Collapse | Copy Code

```
@DateTime.Now
```

So which is a better fit Razor or ASPX?

As per Microsoft razor is more preferred because it's light weight and has simple syntaxes.

How can you do authentication and authorization in MVC?

You can use windows or forms authentication for MVC.

How to implement windows authentication for MVC?

For windows authentication you need to go and modify the "web.config" file and set authentication mode to windows.

⊟Collapse | Copy Code

```
<authentication mode="Windows"/>
<authorization>
<deny users="?"/>
</authorization>
```

Then in the controller or on the action you can use the "Authorize" attribute which specifies which users have access to these controllers and actions. Below is the code snippet for the same. Now only the users specified in the controller and action can access the same.

⊟Collapse | Copy Code

```
[Authorize(Users= @"WIN-3LI600MWLQN\Administrator")]
    public class StartController : Controller
    {
        //
        // GET: /Start/
        [Authorize(Users = @"WIN-3LI600MWLQN\Administrator")]
        public ActionResult Index()
        {
            return View("MyView");
        }
    }
```

How do you implement forms authentication in MVC?

Forms authentication is implemented the same way as we do in ASP.NET. So the first step is to set authentication mode equal to forms. The "loginUrl" points to a controller here rather than page.

⊟Collapse | Copy Code

```
<authentication mode="Forms">
<forms loginUrl="~/Home/Login"  timeout="2880"/>
</authentication>
```

We also need to create a controller where we will check the user is proper or not. If the user is proper we will set the cookie value.

⊟Collapse | Copy Code

```
public ActionResult Login()
{
if ((Request.Form["txtUserName"] == "Shiv") && (Request.Form["txtPassword"]
== "Shiv@123"))
{
            FormsAuthentication.SetAuthCookie("Shiv",true);
            return View("About");
}
else
{
            return View("Index");
}
}
```

All the other actions need to be attributed with “Authorize” attribute so that any unauthorized user if he makes a call to these controllers it will redirect to the controller (in this case the controller is “Login”) which will do authentication.

⊟Collapse | Copy Code

```
[Authorize]
PublicActionResult Default()
{
return View();
}
[Authorize]
publicActionResult About()
{
return View();
}
```

How to implement Ajax in MVC?

You can implement Ajax in two ways in MVC: -

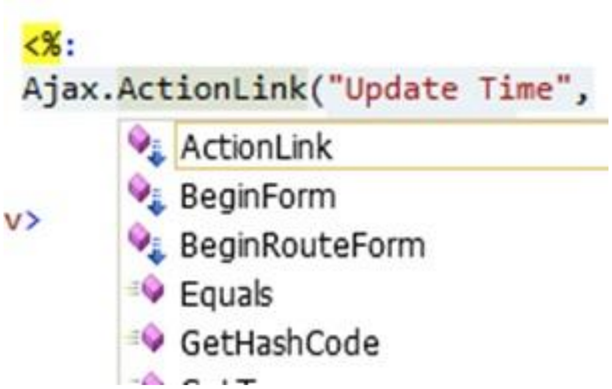
- Ajax libraries
- Jquery

Below is a simple sample of how to implement Ajax by using “Ajax” helper library. In the below code you can see we have a simple form which is created by using “Ajax.BeginForm” syntax. This form calls a controller action called as “getCustomer”. So now the submit action click will be an asynchronous ajax call.

⊟Collapse | Copy Code

```
<script language="javascript">
function OnSuccess(data1)
{
// Do something here
}
</script>
<div>
<%
        var AjaxOpt = new AjaxOptions{OnSuccess="OnSuccess"};
    %>
<% using (Ajax.BeginForm("getCustomer","MyAjax",AjaxOpt)) { %>
<input id="txtCustomerCode" type="text" /><br />
<input id="txtCustomerName" type="text" /><br />
<input id="Submit2" type="submit" value="submit"/></div>
<%} %>
```

In case you want to make ajax calls on hyperlink clicks you can use "Ajax.ActionLink" function as shown in the below code.



```
<%:
Ajax.ActionLink("Update Time",
v>
```

Figure:- implement Ajax in MVC

So if you want to create Ajax asynchronous hyperlink by name "GetDate" which calls the "GetDate" function on the controller , below is the code for the same. Once the controller responds this data is displayed in the HTML DIV tag by name "DateDiv".

⊟Collapse | Copy Code

```
<span id="DateDiv" />
<%:
Ajax.ActionLink("Get Date","GetDate",
new AjaxOptions {UpdateTargetId = "DateDiv" })
%>
```

Below is the controller code. You can see how "GetDate" function has a pause of 10 seconds.

⊟Collapse | Copy Code

```
public class Default1Controller : Controller
{
       public string GetDate()
```

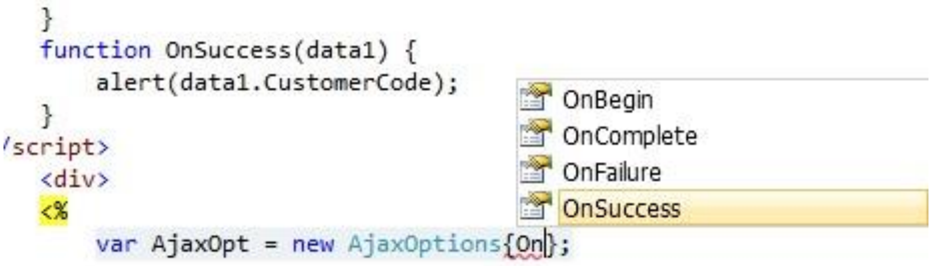
```
        {
            Thread.Sleep(10000);
            return DateTime.Now.ToString();
        }
}
```

The second way of making Ajax call in MVC is by using Jquery. In the below code you can see we are making an ajax POST call to a URL "/MyAjax/getCustomer". This is done by using "$.post". All this logic is put in to a function called as "GetData" and you can make a call to the "GetData" function on a button or a hyper link click event as you want.

⊟Collapse | Copy Code

```
function GetData()
    {
        var url = "/MyAjax/getCustomer";
        $.post(url, function (data)
        {
            $("#txtCustomerCode").val(data.CustomerCode);
            $("#txtCustomerName").val(data.CustomerName);
        }
        )
    }
```

What kind of events can be tracked in AJAX?



```
  }
  function OnSuccess(data1) {
      alert(data1.CustomerCode);
  }
/script>
  <div>
  <%
      var AjaxOpt = new AjaxOptions{On};
```

Figure:- tracked in AJAX

What is the difference between "ActionResult" and "ViewResult"?

"ActionResult" is an abstract class while "ViewResult" derives from "ActionResult" class. "ActionResult" has several derived classes like "ViewResult" ,"JsonResult" , "FileStreamResult" and so on.

"ActionResult" can be used to exploit polymorphism and dynamism. So if you are returning different types of view dynamically "ActionResult" is the best thing. For example in the below code snippet you can see we have a simple action called as "DynamicView". Depending on the flag ("IsHtmlView") it will either return "ViewResult" or "JsonResult".

⊟Collapse | Copy Code

```
public ActionResult DynamicView()
```

```
{
   if (IsHtmlView)
     return View(); // returns simple ViewResult
   else
     return Json(); // returns JsonResult view
}
```

What are the different types of results in MVC?

⊟Collapse | Copy Code

Note: *-It’s difficult to remember all the 12 types. But some important ones you can remember for the interview are “ActionResult”, “ViewResult” and “JsonResult”. Below is a detailed list for your interest.*

There 12 kinds of results in MVC, at the top is “ActionResult”class which is a base class that canhave11subtypes’sas listed below: -

1. ViewResult - Renders a specified view to the response stream
2. PartialViewResult - Renders a specified partial view to the response stream
3. EmptyResult - An empty response is returned
4. RedirectResult - Performs an HTTP redirection to a specified URL
5. RedirectToRouteResult - Performs an HTTP redirection to a URL that is determined by the routing engine, based on given route data
6. JsonResult - Serializes a given ViewData object to JSON format
7. JavaScriptResult - Returns a piece of JavaScript code that can be executed on the client
8. ContentResult - Writes content to the response stream without requiring a view
9. FileContentResult - Returns a file to the client
10. FileStreamResult - Returns a file to the client, which is provided by a Stream
11. FilePathResult - Returns a file to the client

What are “ActionFilters”in MVC?

“ActionFilters” helps you to perform logic while MVC action is executing or after a MVC action has executed.

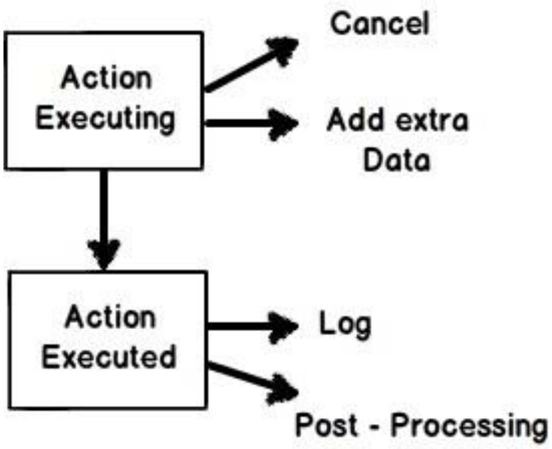


Figure:- “ActionFilters”in MVC

Action filters are useful in the following scenarios:-

1. Implement post-processinglogicbeforethe action happens.
2. Cancel a current execution.
3. Inspect the returned value.
4. Provide extra data to the action.

You can create action filters by two ways:-

- Inline action filter.
- Creating an “ActionFilter” attribute.

To create a inline action attribute we need to implement “IActionFilter” interface.The “IActionFilter” interface has two methods “OnActionExecuted” and “OnActionExecuting”. We can implement pre-processing logic or cancellation logic in these methods.

⊟Collapse | Copy Code

```
public class Default1Controller : Controller , IActionFilter
    {
        public ActionResult Index(Customer obj)
        {
            return View(obj);
        }
        void IActionFilter.OnActionExecuted(ActionExecutedContext 
filterContext)
        {
            Trace.WriteLine("Action Executed");
        }
        void IActionFilter.OnActionExecuting(ActionExecutingContext 
filterContext)
        {
            Trace.WriteLine("Action is executing");
        }
    }
```

The problem with inline action attribute is that it cannot be reused across controllers. So we can convert the inline action filter to an action filter attribute. To create an action filter attribute we need to inherit from “ActionFilterAttribute” and implement “IActionFilter” interface as shown in the below code.

⊟Collapse | Copy Code

```
public class MyActionAttribute : ActionFilterAttribute , IActionFilter
{
void IActionFilter.OnActionExecuted(ActionExecutedContext filterContext)
{
     Trace.WriteLine("Action Executed");
}
void IActionFilter.OnActionExecuting(ActionExecutingContext filterContext)
{
      Trace.WriteLine("Action executing");
}
}
```

Later we can decorate the controllers on which we want the action attribute to execute. You can see in the below code I have decorated the “Default1Controller” with “MyActionAttribute” class which was created in the previous code.

⊟Collapse | Copy Code

```
[MyActionAttribute]
public class Default1Controller : Controller
{
 public ActionResult Index(Customer obj)
 {
 return View(obj);
 }
}
```

Can we create our custom view engine using MVC?

Yes, we can create our own custom view engine in MVC. To create our own custom view engine we need to follow 3 steps:-

Let’ say we want to create a custom view engine where in the user can type a command like “<DateTime>” and it should display the current date and time.

Step 1:- We need to create a class which implements “IView” interface. In this class we should write the logic of how the view will be rendered in the “render” function. Below is a simple code snippet for the same.

⊟Collapse | Copy Code

```
public class MyCustomView : IView
    {
        private string _FolderPath; // Define where  our views are stored
        public string FolderPath
        {
            get { return _FolderPath; }
            set { _FolderPath = value; }
        }

        public void Render(ViewContext viewContext, System.IO.TextWriter 
writer)
        {
           // Parsing logic <dateTime>
            // read the view file
            string strFileData = File.ReadAllText(_FolderPath);
            // we need to and replace <datetime> datetime.now value
            string strFinal = strFileData.Replace("<DateTime>", 
DateTime.Now.ToString());
            // this replaced data has to sent for display
            writer.Write(strFinal);
        }
    }
```

Step 2 :-We need to create a class which inherits from “VirtualPathProviderViewEngine” and in this class we need to provide the folder path and the extension of the view name. For instance for razor the extension is “cshtml” , for aspx the view extension is “.aspx” , so in the same way for our custom view we need to provide an extension. Below is how the code looks like. You can see the “ViewLocationFormats” is set to the “Views” folder and the extension is “.myview”.

⊟Collapse | Copy Code

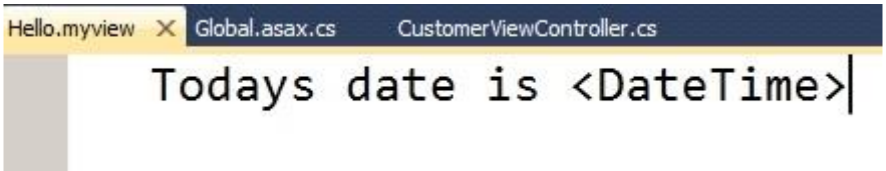
```
public class MyViewEngineProvider : VirtualPathProviderViewEngine
    {
        // We will create the object of Mycustome view
        public MyViewEngineProvider() // constructor
        {
            // Define the location of the View file
            this.ViewLocationFormats = new string[] {
"~/Views/{1}/{0}.myview", "~/Views/Shared/{0}.myview" }; //location and
extension of our views
        }
        protected override IView CreateView(ControllerContext
controllerContext, string viewPath, string masterPath)
        {
            var physicalpath =
controllerContext.HttpContext.Server.MapPath(viewPath);
            MyCustomView obj = new MyCustomView(); // Custom view engine
class
            obj.FolderPath = physicalpath; // set the path where the views
will be stored
            return obj; // returned this view paresing logic so that it can
be registered in the view engine collection
        }
        protected override IView CreatePartialView(ControllerContext
controllerContext, string partialPath)
        {
            var physicalpath =
controllerContext.HttpContext.Server.MapPath(partialPath);
            MyCustomView obj = new MyCustomView(); // Custom view engine
class
            obj.FolderPath = physicalpath; // set the path where the views
will be stored
            return obj; // returned this view paresing logic so that it can
be registered in the view engine collection
        }
    }
```

Step 3:- We need to register the view in the custom view collection. The best place to register the custom view engine in the “ViewEngines” collection is the “global.asax” file. Below is the code snippet for the same.

⊟Collapse | Copy Code

```
protected void Application_Start()
 {
            // Step3 :-  register this object in the view engine collection
            ViewEngines.Engines.Add(new MyViewEngineProvider());
<span class="Apple-tab-span" style="white-space: pre; ">      </span>…..
}
```

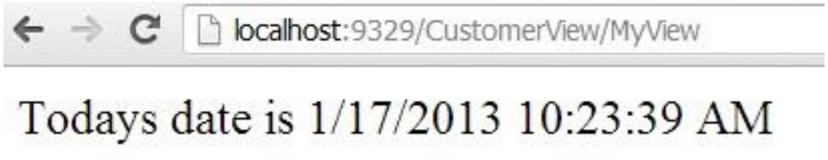
Below is a simple output of the custom view written using the commands defined at the top.



```
Todays date is <DateTime>
```

Figure:-customviewengineusingMVC

If you invoke this view you should see the following output.



Todays date is 1/17/2013 10:23:39 AM

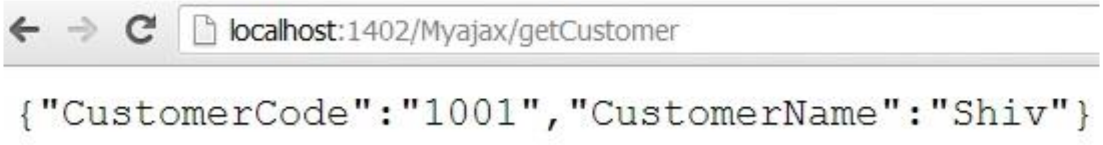
How to send result back in JSON format in MVC?

In MVC we have “JsonResult” class by which we can return back data in JSON format. Below is a simple sample code which returns back “Customer” object in JSON format using “JsonResult”.

⊟Collapse | Copy Code

```
public JsonResult getCustomer()
{
Customer obj = new Customer();
obj.CustomerCode = "1001";
obj.CustomerName = "Shiv";
 return Json(obj,JsonRequestBehavior.AllowGet);
}
```

Below is the JSON output of the above code if you invoke the action via the browser.



```
{"CustomerCode":"1001","CustomerName":"Shiv"}
```

What is “WebAPI”?

HTTP is the most used protocol.For past many years browser was the most preferred client by which we can consume data exposed over HTTP. But as years passed by client variety started spreading out. We had demand to consume data on HTTP from clients like mobile,javascripts,windows application etc.

For satisfying the broad range of client “REST” was the proposed approach. You can read more about “REST” from WCF chapter.

“WebAPI” is the technology by which you can expose data over HTTP following REST principles.

But WCF SOAP also does the same thing, so how does “WebAPI” differ?

	SOAP	**WEB API**
Size	Heavy weight because of complicated WSDL structure.	Light weight, only the necessary informat transferred.
Protocol	Independent of protocols.	Only for HTTP protocol
Formats	To parse SOAP message, the client needs to understand WSDL format. Writing custom code for parsing WSDL is a heavy duty task. If your client is smart enough to create proxy objects like how we have in .NET (add reference) then SOAP is easier to consume and call.	Output of “WebAPI” are simple string message,JSON,Simple XML format etc. S parsing logic for the same in very easy.
Principles	SOAP follows WS-* specification.	WEB API follows REST principles. (Plea REST in WCF chapter).

With WCF also you can implement REST,So why "WebAPI"?

WCF was brought in to implement SOA, never the intention was to implement REST."WebAPI'" is built from scratch and the only goal is to create HTTP services using REST. Due to the one point focus for creating “REST” service “WebAPI” is more preferred.

How to implement “WebAPI” in MVC?

Below are the steps to implement "webAPI" :-

Step1:-Create the project using the "WebAPI" template.

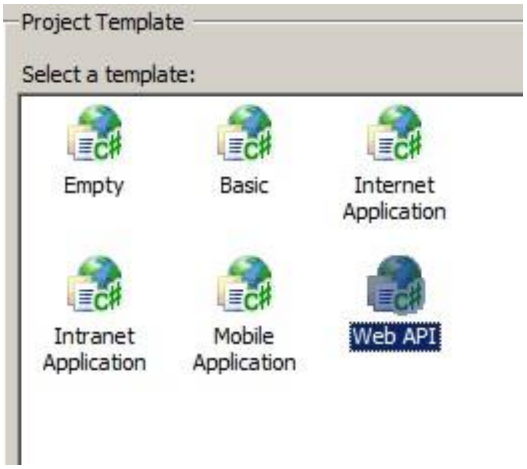


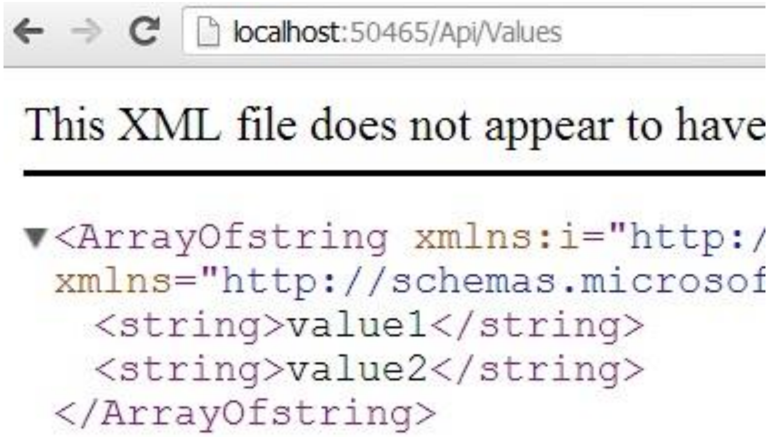
Figure:- implement "WebAPI" in MVC

Step 2:- Once you have created the project you will notice that the controller now inherits from "ApiController" and you can now implement "post","get","put" and "delete" methods of HTTP protocol.

⊟Collapse | Copy Code

```
public class ValuesController : ApiController
    {
        // GET api/values
        public IEnumerable<string> Get()
        {
            return new string[] { "value1", "value2" };
        }
        // GET api/values/5
        public string Get(int id)
        {
            return "value";
        }
        // POST api/values
        public void Post([FromBody]string value)
        {
        }
        // PUT api/values/5
        public void Put(int id, [FromBody]string value)
        {
        }
        // DELETE api/values/5
        public void Delete(int id)
        {
        }
    }
```

Step 3:-If you make a HTTP GET call you should get the below results.



This XML file does not appear to have

```
▼<ArrayOfstring xmlns:i="http:/
 xmlns="http://schemas.microsof
   <string>value1</string>
   <string>value2</string>
 </ArrayOfstring>
```

Session 2:

What are the 3 main components of an ASP.NET MVC application?

1. M - Model
2. V - View
3. C - Controller

In which assembly is the MVC framework defined?

System.Web.Mvc

Is it possible to combine ASP.NET webforms and ASP.MVC and develop a single web application?

Yes, it is possible to combine ASP.NET webforms and ASP.MVC and develop a single web application.

What does Model, View and Controller represent in an MVC application?

Model: Model represents the application data domain. In short the applications business logic is contained with in the model.

View: Views represent the user interface, with which the end users interact. In short the all the user interface logic is contained with in the UI.

Controller: Controller is the component that responds to user actions. Based on the user actions, the respective controller, work with the model, and selects a view to render that displays the user interface. The user input logic is contained with in the controller.

What is the greatest advantage of using asp.net mvc over asp.net webforms?

It is difficult to unit test UI with webforms, where views in mvc can be very easily unit tested.

Which approach provides better support for test driven development - ASP.NET MVC or ASP.NET Webforms?

ASP.NET MVC

What are the advantages of ASP.NET MVC?

1. Extensive support for TDD. With asp.net MVC, views can also be very easily unit tested.
2. Complex applications can be easily managed
3. Seperation of concerns. Different aspects of the application can be divided into Model, View and Controller.
4. ASP.NET MVC views are light weight, as they donot use viewstate.

Is it possible to unit test an MVC application without running the controllers in an ASP.NET process?

Yes, all the features in an asp.net MVC application are interface based and hence mocking is much easier. So, we don't have to run the controllers in an ASP.NET process for unit testing.

Is it possible to share a view across multiple controllers?

Yes, put the view into the shared folder. This will automatically make the view available across multiple controllers.

What is the role of a controller in an MVC application?

The controller responds to user interactions, with the application, by selecting the action method to execute and alse selecting the view to render.

Where are the routing rules defined in an asp.net MVC application?

In Application_Start event in Global.asax

Name a few different return types of a controller action method?

The following are just a few return types of a controller action method. In general an action method can return an instance of a any class that derives from ActionResult class.

1. ViewResult
2. JavaScriptResult
3. RedirectResult
4. ContentResult
5. JsonResult

What is the significance of NonActionAttribute?

In general, all public methods of a controller class are treated as action methods. If you want prevent this default behaviour, just decorate the public method with NonActionAttribute.

What is the significance of ASP.NET routing?

ASP.NET MVC uses ASP.NET routing, to map incoming browser requests to controller action methods. ASP.NET Routing makes use of route table. Route table is created when your web application first starts. The route table is present in the Global.asax file.

What are the 3 segments of the default route, that is present in an ASP.NET MVC application?

1st Segment - Controller Name
2nd Segment - Action Method Name
3rd Segment - Parameter that is passed to the action method

Example: http://pragimtech.com/Customer/Details/5
Controller Name = Customer
Action Method Name = Details
Parameter Id = 5

ASP.NET MVC application, makes use of settings at 2 places for routing to work correctly. What are these 2 places?

1. Web.Config File : ASP.NET routing has to be enabled here.
2. Global.asax File : The Route table is created in the application Start event handler, of the Global.asax file.

What is the adavantage of using ASP.NET routing?

In an ASP.NET web application that does not make use of routing, an incoming browser request should map to a physical file. If the file does not exist, we get page not found error.

An ASP.NET web application that does make use of routing, makes use of URLs that do not have to map to specific files in a Web site. Because the URL does not have to map to a file, you can use URLs that are descriptive of the user's action and therefore are more easily understood by users.

What are the 3 things that are needed to specify a route?

1. URL Pattern - You can include placeholders in a URL pattern so that variable data can be passed to the request handler without requiring a query string.
2. Handler - The handler can be a physical file such as an .aspx file or a controller class.
3. Name for the Route - Name is optional.

Is the following route definition a valid route definition?

{controller}{action}/{id}
No, the above definition is not a valid route definition, because there is no literal value or delimiter between the placeholders. Therefore, routing cannot determine where to separate the value for the controller placeholder from the value for the action placeholder.

What is the use of the following default route?

{resource}.axd/{*pathInfo}
This route definition, prevent requests for the Web resource files such as WebResource.axd or ScriptResource.axd from being passed to a controller.

What is the difference between adding routes, to a webforms application and to

an mvc application?

To add routes to a webforms application, we use MapPageRoute() method of the RouteCollection class, where as to add routes to an MVC application we use MapRoute() method.

How do you handle variable number of segments in a route definition?

Use a route with a catch-all parameter. An example is shown below. * is referred to as catch-all parameter.
controller/{action}/{*parametervalues}

What are the 2 ways of adding constraints to a route?

1. Use regular expressions
2. Use an object that implements IRouteConstraint interface

Give 2 examples for scenarios when routing is not applied?

1. A Physical File is Found that Matches the URL Pattern - This default behaviour can be overriden by setting the RouteExistingFiles property of the RouteCollection object to true.
2. Routing Is Explicitly Disabled for a URL Pattern - Use the RouteCollection.Ignore() method to prevent routing from handling certain requests.

What is the use of action filters in an MVC application?

Action Filters allow us to add pre-action and post-action behavior to controller action methods.

If I have multiple filters impleted, what is the order in which these filters get executed?

1. Authorization filters
2. Action filters
3. Response filters
4. Exception filters

What are the different types of filters, in an asp.net mvc application?

1. Authorization filters
2. Action filters
3. Result filters
4. Exception filters

Give an example for Authorization filters in an asp.net mvc application?

1. RequireHttpsAttribute
2. AuthorizeAttribute

Which filter executes first in an asp.net mvc application?

Authorization filter

What are the levels at which filters can be applied in an asp.net mvc application?

1. Action Method
2. Controller
3. Application

[b]Is it possible to create a custom filter?[/b]

Yes

What filters are executed in the end?

Exception Filters

Is it possible to cancel filter execution?

Yes

What type of filter does OutputCacheAttribute class represents?

Result Filter

What are the 2 popular asp.net mvc view engines?

1. Razor
2. .aspx

What symbol would you use to denote, the start of a code block in razor views?

@

What symbol would you use to denote, the start of a code block in aspx views?

<%= %>

In razor syntax, what is the escape sequence character for @ symbol?

The escape sequence character for @ symbol, is another @ symbol

When using razor views, do you have to take any special steps to proctect your asp.net mvc application from cross site scripting (XSS) attacks?

No, by default content emitted using a @ block is automatically HTML encoded to protect from cross site scripting (XSS) attacks.

When using aspx view engine, to have a consistent look and feel, across all pages of the application, we can make use of asp.net master pages. What is asp.net master pages equivalent, when using razor views?

To have a consistent look and feel when using razor views, we can make use of layout pages. Layout pages, reside in the shared folder, and are named as _Layout.cshtml

What are sections?

Layout pages, can define sections, which can then be overriden by specific views making use of the layout. Defining and overriding sections is optional.

What are the file extensions for razor views?

1. .cshtml - If the programming lanugaue is C#
2. .vbhtml - If the programming lanugaue is VB

How do you specify comments using razor syntax?

Razor syntax makes use of @* to indicate the begining of a comment and *@ to indicate the end. An example is shown below.
@* This is a Comment *@