**ABOUT ASP.NET MVC**

The ASP.Net MVC is the framework provided by Microsoft that lets you develop the applications that follows the principles of Model-View-Controller (MVC) design pattern. The .Net programmers new to MVC thinks that it is similar to WebForms Model (Normal ASP.Net), but it is far different from the WebForms programming.   
This article will tell you how to quick learn the basics of MVC along with some frequently asked interview questions and answers on ASP.Net MVC

**1. What is ASP.Net MVC**

The ASP.Net MVC is the framework provided by Microsoft to achieve     separation of concerns that leads to easy maintainability of the     application.

Model is supposed to handle data related activity

View deals with user interface related work

Controller is meant for managing the application flow by communicating between Model and View.

**2. Why to use ASP.Net MVC**

The strength of MVC (i.e. ASP.Net MVC) listed below will answer this question

* MVC reduces the dependency between the components; this makes your code more testable.
* MVC does not recommend use of server controls, hence the processing time required to generate HTML response is drastically reduced.
* The integration of java script libraries like jQuery, Microsoft MVC becomes easy as compared to Webforms approach.

**3. What do you mean by Razor**

The Razor is the new View engine introduced in MVC 3.0.

The View engine is responsible for processing the view files [e.g. .aspx, .cshtml] in order to generate HTML response.  
The previous versions of MVC were dependent on ASPX view engine.

**4. Can we use ASPX view engine in latest versions of MVC**

Yes. The Recommended way is to prefer Razor View

**5. What are the benefits of Razor View?**

* The syntax for server side code is simplified
* The length of code is drastically reduced
* Razor syntax is easy to learn and reduces the complexity

**6. What is the extension of Razor View file?**

.cshtml (for c#) and .vbhtml (for vb)

**7. How to create a Controller in MVC**

Create a simple class and extend it from Controller class. The bare minimum requirement for a class to become a controller is to inherit it from ControllerBase is the class that is required to inherit to create the controller but Controller class inherits from ControllerBase.

**8. How to create an Action method in MVC**

Add a simple method inside a controller class with ActionResult return type.

**9. How to access a view on the server**

The browser generates the request in which the information like Controller name, Action Name and Parameters are provided, when server receives this URL it resolves the Name of Controller and Action, after that it calls the specified action with provided parameters. Action normally does some processing and returns the ViewResult by specifying the view name (blank name searches according to naming conventions).

**10. What is the default Form method (i.e. GET, POST) for an action method**

GET. To change this you can add an action level attribute e.g [HttpPost]  
  
 

**11. What is a Filter in MVC?**

When user (browser) sends a request to server an action method of a controller gets invoked; sometimes you may require executing a custom code before or after action method gets invoked, this custom code is called as Filter.

**12. What are the different types of Filters in MVC?**

a. Authorization filter

b. Action filter

c. Result filter

d. Exception filter

[Do not forget the order mentioned above as filters gets executed as per above mentioned sequence]

**13. Explain the use of Filter with an example?**

Suppose you are working on a MVC application where URL is sent in an encrypted format instead of a plain text, once encrypted URL is received by server it will ignore action parameters because of URL encryption.

To solve this issue you can create global action filter by overriding OnActionExecuting method of controller class, in this you can extract the action parameters from the encrypted URL and these parameters can be set on filterContext to send plain text parameters to the actions.

**14. What is a HTML helper?**

A HTML helper is a method that returns string; return string usually is the HTML tag. The standard HTML helpers (e.g. Html.BeginForm(),Html.TextBox())available in MVC are lightweight as it does not rely on event model or view state as that of in ASP.Net server controls.  
  
  
**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](http://www.dotnetobject.com/expage.php?exurl=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 ascatch-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 \*@