1. Explain MVC (Model-View-Controller) in general?
Ans: MVC (Model-View-Controller) is an architectural software pattern that basically decouples various components of a web application. By using MVC pattern, we can develop applications that are more flexible to changes without affecting the other components of our application.
"Model", is basically domain data.
"View", is user interface to render domain data.
"Controller", translates user actions into appropriate operations performed on model.

2. What is ASP.NET MVC?
Ans: ASP.NET MVC is a web development framework from Microsoft that is based on MVC (Model-View-Controller) architectural design pattern. Microsoft has streamlined the development of MVC based applications using ASP.NET MVC framework.

3. Difference between ASP.NET MVC and ASP.NET WebForms?
Ans: ASP.NET Web Forms uses Page controller pattern approach for rendering layout, whereas ASP.NET MVC uses Front controller approach. In case of Page controller approach, every page has its own controller i.e. code-behind file that processes the request. On the other hand, in ASP.NET MVC, a common controller for all pages processes the requests.
Follow the link for the difference between the ASP.NET MVC and ASP.NET WebForms.

4. What are the Core features of ASP.NET MVC?
Ans: Core features of ASP.NET MVC framework are:
Clear separation of application concerns (Presentation and Business Logic)
An extensible and pluggable framework
Extensive support for ASP.NET Routing
Support for existing ASP.NET features
Follow for detailed understanding of above mentioned core features.

5. Can you please explain the request flow in ASP.NET MVC framework?
Ans: Request flow for ASP.NET MVC framework is as follows:
Request hits the controller coming from client.
Controller plays its role and decides which model to use in order to serve the request.
Further passing that model to view which then transforms the model and generate an appropriate response that is rendered to client.

ASP.NET MVC Request Flow

6. What is Routing in ASP.NET MVC?
Ans: In case of a typical ASP.NET application, incoming requests are mapped to physical files such as .aspx file. ASP.NET MVC framework uses friendly URLs that more easily describe user's action but not mapped to physical files. ASP.NET MVC framework uses a routing engine, that maps URLs to controller classes. We can define routing rules for the engine, so that it can map incoming request URLs to appropriate controller.Practically, when a user types a URL in a browser window for an ASP.NET MVC application and presses "go" button, routing engine uses routing rules that are defined in Global.asax file in order to parse the URL and find out the path of corresponding controller.
You can find ASP.NET MVC Routing further details here.

7. What is the difference between ViewData, ViewBag and TempData?
Ans: In order to pass data from controller to view and in next subsequent request, ASP.NET MVC framework provides different options i.e. ViewData, ViewBag and TempData.
Both ViewBag and ViewData are used to to communicate between controller and corresponding view.But this communication is only for server call, it becomes null if redirect occurs. So, in short, its a mechanism to maintain state between controller and corresponding view.

ViewData is a dictionary object while ViewBag is a dynamic property (a new C# 4.0 feature).
ViewData being a dictionary object is accessible using strings as keys and also requires typecasting for complex types.On the other hand, ViewBag doesn't have typecasting and null checks.
TempData is also a dictionary object that stays for the time of an HTTP Request. So, Tempdata can be used to maintain data between redirects i.e from one controller to the other controller.
You can easily find examples for implementation of ViewBag, ViewData and TempData here.

8. What are Action Methods in ASP.NET MVC?
Ans: As I already explained about request flow in ASP.NET MVC framework that request coming from client hits controller first. Actually MVC application determines the corresponding controller by using routing rules defined in Global.asax. And controllers have specific methods for each user actions. Each request coming to controller is for a specific Action Method. The following code example, "ShowBooks" is an example of an Action Method.

```
public ViewResult ShowBooks(int id)
{
        var computerBook = db.Books.Where(p => P.BookID == id).First();
        return View(computerBook);
 }
```

9. Explain the role of Model in ASP.NET MVC?
Ans: One of the core feature of ASP.NET MVC is that it separates the input and UI logic from business logic. Role of Model in ASP.NET MVC is to contain all application logic including validation, business and data access logic except view i.e. input and controller i.e UI logic. Model is normally responsible for accessing data from some persistent medium like database and manipulate it, so you can expect that interviewer can ask questions on database access topics here along with ASP.NET MVC Interview Questions.

10. What are Action Filters in ASP.NET MVC?
Ans: If we need to apply some specific logic before or after action methods, we use action filters. We can apply these action filters to a controller or a specific controller action. Action filters are basically custom classes that provide a mean for adding pre-action or post-action behavior to controller actions. For example: Authorize filter can be used to restrict access to a specific user or a role.
OutputCache filter can cache the output of a controller action for a specific duration.

ASP.NET MVC is an amazing framework for developing applications. Above mentioned ASP.NET MVC interview questions must be prepared before appearing for a MVC interview.

ASP.NET Web API

Q: What is ASP.NET Web API?
Ans: ASP.NET Web API is a framework that simplifies building HTTP services for broader range of clients (including browsers as well as mobile devices) on top of .NET Framework. Using ASP.NET Web API we can create non-SOAP based services like plain XML or JSON strings etc. with many other advantages including:

Create resource-oriented services using the full features of HTTP.
Exposing services to a variety of clients easily like browsers or mobile devices etc.
What are the advantages of using ASP.NET Web API?

Using ASP.NET Web API has a number of advantages, but core of the advantages are:
It works the HTTP way using standard HTTP verbs like GET, POST, PUT, DELETE etc for all CRUD operations.
Complete support for routing.

Response generated in JSON or XML format using MediaTypeFormatter.
It has the ability to be hosted in IIS as well as self-host outside of IIS.
Supports Model binding and Validation.
Support for OData.

Q: What new features are introduced in ASP.NET Web API 2.0?
Ans:
More new features introduced in ASP.NET Web API framework v2.0 are as follows:
Attribute Routing
External Authentication
CORS (Cross-Origin Resource Sharing)
OWIN (Open Web Interface for .NET) Self Hosting
IHttpActionResult
Web API OData
You can follow a good Web API new feature details on Top 5 New Features in ASP.NET Web API 2 here.

Q: WCF Vs ASP.NET Web API?
Ans: Actually, Windows Communication Foundation is designed to exchange standard SOAP-based messages using variety of transport protocols like HTTP, TCP, NamedPipes or MSMQ etc. On the other hand, ASP.NET API is a framework for building non-SOAP based services over HTTP only. For more details, please follow here.

Q: Is it true that ASP.NET Web API has replaced WCF?

It's a misconception that ASP.NET Web API has replaced WCF. It's another way of building non-SOAP based services, for example, plain XML or JSON string etc.
Yes, it has some added advantages like utilizing full features of HTTP and reaching more clients such as mobile devices etc.
But WCF is still a good choice for following scenarios:
If we intended to use transport other than HTTP e.g. TCP, UDP or Named Pipes.
Messag Queuing scenario using MSMQ.
One-way communication or Duplex communication
A good understanding for WCF(Windows Communication Foundation), please follow WCF Tutorial.

Q: MVC Vs ASP.NET Web API?
Ans: As in previous ASP.NET Web API Interview Questions, we discussed that purpose of Web API framework is to generate HTTP services that reaches more clients by generating data in raw format, for example, plain XML or JSON string. So, ASP.NET Web API creates simple HTTP services that renders raw data. On the other hand, ASP.NET MVC framework is used to develop web applications that generates Views as well as data. ASP.NET MVC facilitates in rendering HTML easy.

Q: How to return View from ASP.NET Web API method?
Ans: (A tricky Interview Question) No, we can't return view from ASP.NET Web API Method. As we discussed in earlier interview question about difference between ASP.NET MVC and Web API that ASP.NET Web API creates HTTP services that renders raw data. Although, it's quite possible in ASP.NET MVC application.

Q: How to restrict access to Web API method to specific HTTP Verb?

Ans: Attribute programming plays it's role here. We can easily restrict access to an ASP.NET Web API method to be called using a specific HTTP method. For example, we may required in a scenario to restrict access to a Web API method through HTTP POST only as follows:
[HttpPost] public void UpdateStudent(Student aStudent) {
StudentRepository.AddStudent(aStudent); }

Q: Can we use Web API with ASP.NET Web Form?

Ans: Yes, ASP.NET Web API is bundled with ASP.NET MVC framework but still it can be used with ASP.NET Web Form. It can be done in three simple steps as follows:

Q: Create a Web API Controller.
Ans: Add a routing table to Application_Start method of Global.asax.
Make a jQuery AJAX Call to Web API method and get data.
jQuery call to Web API for all CRUD (Create, Retrieve, Update, Delete) operations can be found here.

Q: How we can provide an alias name for ASP.NET Web API action?
Ans: We can provide an alias name for ASP.NET Web API action same as in case of ASP.NET MVC by using "ActionName" attribute as follows:
[HttpPost] [ActionName("SaveStudentInfo")] public void UpdateStudent(Student aStudent) { StudentRepository.AddStudent(aStudent); }
In this ASP.NET Tutorial, we covered most important Interview Questions on ASP.NET Web API framework. Hopefully, it will be helpful for Web API developer Interview but along with these questions, do the practical implementation as much as you can. In Practical guide to ASP.NET Web API, you can find a good step by step approach for understanding and implementing ASP.NET Web API services.

C#

1. What is the base class in .net from which all the classes are derived from?
ANS:
System.Object

2. What is the difference between method overriding and method overloading?
ANS: In method overriding, we change the method definition in the derived class that changes the method behavior. Method overloading is creating a method with the same name within the same class having different signatures.

3. What are the different ways a method can be overloaded?
ANS: Methods can be overloaded using different data types for parameter, different order of parameters, and different number of parameters.

4. Why can't you specify the accessibility modifier for methods inside the interface?
ANS: In an interface, we have virtual methods that do not have method definition. All the methods are there to be overridden in the derived class. That's why they all are public.

5. How can we set class to be inherited, but prevent the method from being over-ridden?
ANS: Declare the class as public and make the method sealed to prevent it from being overridden.

6. What happens if the inherited interfaces have conflicting method names?
ANS: Implement is up to you as the method is inside your own class. There might be problem when the methods from different interfaces expect different data, but as far as compiler cares you're okay.

7. What is the difference between a Struct and a Class?
ANS: Structs are value-type variables and classes are reference types. Structs stored on the stack, causes additional overhead but faster retrieval. Structs cannot be inherited.

8. How to use nullable types in .Net?
ANS: Value types can take either their normal values or a null value. Such types are called nullable types.
Int? someID = null;
If(someID.HasVAlue)
{

}

9. How we can create an array with non-default values?
ANS: We can create an array with non-default values using Enumerable.Repeat.

10. What is difference between is and as operators in c#?
ANS: "is" operator is used to check the compatibility of an object with a given type and it returns the result as Boolean.
"as" operator is used for casting of object to a type or a class.

11. What's a multicast delegate?
ANS: A delegate having multiple handlers assigned to it is called multicast delegate. Each handler is assigned to a method.

12. What are indexers in C# .NET?
ANS: Indexers are known as smart arrays in C#. It allows the instances of a class to be indexed in the same way as array.
Eg:
public int this[int index]   // Indexer declaration

13. What is difference between the "throw" and "throw ex" in .NET?
ANS: "Throw" statement preserves original error stack whereas "throw ex" have the stack trace from their throw point. It is always advised to use "throw" because it provides more accurate error information.

14. What are C# attributes and its significance?
ANS: C# provides developers a way to define declarative tags on certain entities eg. Class, method etc. are called attributes. The attribute's information can be retrieved at runtime using Reflection.

15. How to implement singleton design pattern in C#?
ANS: In singleton pattern, a class can only have one instance and provides access point to it globally.
Eg:
Public sealed class Singleton
{
  Private static readonly Singleton _instance = new Singleton();
}

16. What is the difference between directcast and ctype?
ANS: DirectCast is used to convert the type of an object that requires the run-time type to be the same as the specified type in DirectCast. Ctype is used for conversion where the conversion is defined between the expression and the type.

17. Is C# code is managed or unmanaged code?
ANS: C# is managed code because Common language runtime can compile C# code to Intermediate language.

18. What are the namespaces used in C#.NET?
ANS: Namespace is a logical grouping of class.
using System;
using System.Collections.Generic;
using System.Windows.Forms;

19. What are the characteristics of C#?
ANS: There are several characteristics of C# are:
Simple
Type safe
Flexible
Object oriented
Compatible
Consistent
Interoperable
Modern

20. What are the different categories of inheritance?
ANS: Single inheritance : Contains one base class and one derived class.

Hierarchical inheritance : Contains one base class and multiple derived classes of the same base class.
Multilevel inheritance : Contains a class derived from a derived class.
Multiple inheritance : Contains several base classes and a derived class.
Hybrid inheritance : is nothing but a any combination of single,multiple and inheritance inheritance. Multiple inheritance class is not possible as it give assembly level error when we use the same name for method. Multiple inheritance in interface is possible.

21. What are the basic concepts of object oriented programming?
ANS: It is necessary to understand some of the concepts used extensively in object oriented programming.These include
Objects
Classes
Data abstraction and encapsulation
Inheritance
Polymorphism
Message passing.

22. Can you inherit multiple interfaces?
ANS: Yes. Multiple interfaces may be inherited in C#.

23. What is inheritance?
ANS: Inheritance is deriving the new class from the already existing one.

24. Define scope?
ANS: Scope refers to the region of code in which a variable may be accessed.

25. What is the difference between public, static and void?
ANS: public: The keyword public is an access modifier that tells the C# compiler that the Main method is accessible by anyone.
static: The keyword static declares that the Main method is a global one and can be called without creating an instance of the class. The compiler stores the address of the method as the entry point and uses this information to begin execution before any objects are created.
void : The keyword void is a type modifier that states that the Main method does not return any value.

26. What are the modifiers in C#?
ANS: Abstract
Sealed
Virtual
Const
Event
Extern
Override
Readonly
Static
New

27. What are the types of access modifiers in C#?
ANS: Access modifiers in C# are:
public
protect
private
internal
internal protect

28. What is boxing and unboxing?
ANS: Implicit conversion of value type to reference type of a variable is known as BOXING, for example integer to object type conversion.
Conversion of reference type variable back to value type is called as UnBoxing.

29. What is object?
ANS: An object is an instance of a class. An object is created by using operator new. A class that creates an object in memory will contain the information about the values and behaviors (or methods) of that specific object.

30. Where are the types of arrays in C#?
ANS:
Single-Dimensional
Multidimensional

Jagged arrays

JavaScript

1. JavaScript vs. Jscript
Ans: Both JavaScript and Jscript are almost similar. Java script was developed by Netscape. Microsoft implemented its own scripting language and named it as Jscript.....
Read answer

2. What is the difference between Client side JavaScript and Server side JavaScript.
Ans: Client side java script comprises the basic language and predefined objects which are relevant to running java script in a browser. The client side java script is embedded directly by in the HTML pages.....
Read answer

3. Where are cookies actually stored on the hard disk?
Ans: The storage of cookies on the hard disk depends on OS and the browser. The Netscape navigator on Windows, the file cookies.txt contains all the cookies. The path is :.....
Read answer

4. What is the difference between a web-garden and a web-farm?
Ans: Web garden is a web hosting system. It is a setup of multi processors in single server. Web farm is a web hosting system. It is a multi-server scenario

5. What is the difference between SessionState and ViewState?
Ans: The values of controls of a particular page of the client browser is persisted by ViewState at the time of post back operation is done. If the user requests another page, the data of previous page is no longer available.....
Read answer

6. How to Accessing Elements using javascript?
Ans: The elements of JavaScript are accessed by their names. By default the browser is accessed by the element 'windows' and the page by 'document'. The corresponding element has user defined names for forms and its elements.....
Read answer

7. What is the difference between undefined value and null value?
Ans: Undefined value: A value that is not defined and has no keyword is known as undefined value. For example in the declaration, int number; the number has undefined value.....
Read answer

8. How to set the cursor to wait in JavaScript?
Ans: The cursor can set to wait in JavaScript by using the property 'cursor' property. The following example illustrates the usage.

9. What is decodeURI(), encodeURI() in JavaScript?
Ans: To send the characters that can not be specified in a URL should be converted into their equivalent hex encoding. To perform this task the methods encodeURI() and decodeURI() are used.

10. Methods GET vs. POST in HTML forms.
Ans: Encoding form data into URL is needed by the GET method. The form data is to be appeared within the message body , by the POST method. By specification, GET is used basically for retrieving data where as POST is used for data storing, data updating, ordering a product or even e-mailing.

11. What does the EnableViewStateMac setting in an aspx page do?
Ans: EnableViewStateMac setting is a security measure in ASP.Net. It ensures the view state for a page not to tamper. To to so " EnableViewStateMac=true "is used.

12. What are windows object and navigator object in JavaScript?
Ans: Windows object is top level object in Java script. It contains several other objects such as, document, history, location, name, menu bar etc., in itself. Window object is the global object for Java script that is written at client-side.

13. How to detect the operating system on the client machine in JavaScript?
Ans: The navigator.appVersion string should be used to find the name of the operating system on the client machine.

14. How to set a HTML document's background color in JavaScript?
Ans: Using document object the back ground color can be changed by JavaScript.

15. How do you assign object properties in JavaScript?

Ans: Java script object properties are assigned like assigning a value to a variable. For example, the title property of document object can be assigned as follows:
document.title="Welcome to the world of Javascripting";

16. What is JavaScript?
Ans: JavaScript is a scripting language most often used for client-side web development.

17. What boolean operators does JavaScript support?
Ans: ==, !=, < , >, <=, >=.

18. Is a javascript script faster than an ASP script?
Ans: JSP is faster then ASP as the script is run on the client side.

19. What is == operator?
Ans: The '==' operator is a boolean comparison operator that returns true if the variables.

20. What is negative infinity?
Ans: It's a number that is obtained by dividing a negative number by zero. (in JSP).

21. What's relationship between JavaScript and ECMAScript?
Ans: JavaScript is a scripting language most often used for client-side web development.

22. What does isNaN function do?
Ans: The isNaN function determines if the value is a number or not and depending upon the result, it returns true or false.

23. How to read and write a file using javascript?
Ans: There are two ways to do it: 1. Using JavaScript extensions (runs from JavaScript Editor), or 2. Using a web page and ActiveX objects (Internet Explorer only).

24. How do you create a new object in JavaScript?
Ans: In order to generate dynamic content, JSP provides for creating, modifying and interacting with Java objects. The implicit objects like page, config, request, etc are called so because their availability in the in JSP page is automatic.

25. How to create arrays in JavaScript?
Ans: Although you can create the arrays using 'new' (var myArray = new myArray[10];), it is recommended that you create it in the following way: var myArray = [];

26. What are the different actions that are performed using JavaScript?

Ans: JavaScript allow the computer to be more secure by putting the privacy policies and disable the unauthorized access to the files. It allows many actions to be performed like Setting the browser's preferences and receiving the request from the servers for the client computer. The settings that are used gets

saved on the client side on the features like actions buttons, appearance and printing.
It allows easy launching of an application on the client computer with interactive data to be communicated between the server and the client.
It allows reading and writing of files or directories on the client or server side.
It allows easy capturing of the data that is live streamed from server to the client machine for retransmission.
It also allows to safe and secure the application from the outside world.

27. How can JavaScript language be separated from objects?

Ans: JavaScript treats and creates applications for the scripting to make the browser's compatible for use. The language is separated from the objects as it allows the syntax to change the environment. It is a language that keeps the page element in the HTML document. JavaScript allows the elements of the page to remain in sync with the document objects. The language is used to create objects that are connected to page elements and other elements in a language. The separation allows the concept of development and effort to be shared with each factor. The JavaScript language allows dynamic data to be presented using the weakly typed language. It also support any action to be taken to support user interface and graphics.

28. What is the use of DOM?

Ans: DOM is also known as Document Object Model which is used to develop a model with documents or web pages containing objects like elements, links, etc. These objects can be manipulated or certain actions like add, delete or change of an element can be performed using this document object model. Through this change in attributes can be done to get all the list of all the elements in the document. The DOM model responds to API calls that result in documented level of DOM recommendation. It is used to support additional behavior on the web page and use of API give an extensible advantage over other models existing. DOM codes are reused to meet the requirement of the real world and to make all program interoperable.

29. What is the importance of <SCRIPT> tag?

Ans: JavaScript is used inside <SCRIPT> tag in HTML document. The tags that are provided provides the necessary information like alret to the browser for the program to begin interpreting all the text between the tags. The <script> tag uses JavaScript interpreter to handle the libraries that are written or the code of the program. JavaScript is a case sensitive language and the tags are used to tell the browser that if it is JavaScript enabled to use the text written in between the <Script> and </Script> tags. The example is given as:

```
<HTML>
<HEAD>
<!--
<SCRIPT> // Starting of the scripting language
//Your code here
</SCRIPT> --> // End of the scripting language
</HEAD>
<BODY>
// your code here
</BODY>
</HTML>
```

30. Why JavaScript is called as Script for all browsers?

Ans: JavaScript is written after <SCRIPT> tag and it is surrounded in between the <!-- your code --> tags this is also known as comment tag. JavaScript interpreter treats the tag such that it treats all the lines in the comments as script lines. The JavaScript comment starts with // inside the <SCRIPT> tag. The script is contained inside <HTML> tag that contains a comment tag. The browser that is non-

compatible with JavaScripts ignore the lines and move on, but compatible browsers always treats it as a script and execute it. The browser treat the written lines between the comment tag as normal lines and just thinking of the comment ignores it. Some browsers just treat the <!-- comment --> as a comment only and ignores whatever is written inside it.

31. What are the requirements of Web application using JavaScript?

Ans: There are lots of application that require certain things when the user uses a JavaScript like
Data entry validation: this tell that if the field of the form is filled out then during the processing of the server the client side can interact with it.
Serverless CGIs: this describes the processes that are not used with JavaScript but programmed as CGI on server, it gives low performace due to more interaction between the applicatioin and the user.
Dynamic HTML interactivity: it allows dynamic position of the data without using any other scripting language.
CGI prototyping: allows more reduction of time to access the user interface before implementing the application of the CGI.

32. What are the different objects used in JavaScripts?

Ans: JavaScripts uses a hierarchical structure that applies to the objects in a document. There are some objects that show the relationship of one object to another using the language.
Window object: this is the topmost object in the hierarchy. It represent the content area of browser window that consists of HTML documents. Each frame is also a window that has some actions inside it.
Document object: This object gets loaded in a window and consists of objects of different kind in the model. It consists of the content that will be written in the script.
Form object: Form objects are used for more interaction with the users. It represents the form elements inside <FORM>...</FORM> tag.

33. Why is object naming important to use in JavaScript?

Ans: Object naming is used to create script references to objects while assigning names to every
scriptable object in the HTML code. The browsers that are script compatible looks for the optional tags and attributes that enables the assigning of a unique name to each object. The example is:

<form name=”dataEntry” method=get>
<input type=”text” name=”entry”>
<frame src=”info.html” name=”main”>
The names act as a nametags through which the elements can be easily identified and easily located by the browsers. The references made for each object includes the object hierarchy from the top down. References are used to include names of each object that are coming in the window object. The object naming conventions are easy way to loacte the objects and the linking between them can be done more comfortably.

34. What are the methods involved in JavaScript?

Ans: Method is an informative that gets performed over an action that is related to the object. Method either performs on some object or affect any part of the the script or a document. Object can have as many number of methods that have associations with other objects. There is a method that is used by the JavaScript statement that includes a reference to an object this is given as:
document.orderForm.submit()
document.orderForm.entry.select()
These are the functions which perform the dynamic interaction with the user. The first statement execute the element when pressed

submit button to send a form to a server. These two statements are related to only the form. The scripts that are invoked will have the write of the document as well and will be written as:

```
document.write("Give the version " +
navigator.appVersion)
document.write(" of <B>" +
navigator.appName + "</B>.")
```

35. Explain with an example the use of event handlers in JavaScript.

Ans: The events in JavaScript are the actions in a document that result from user activity. The actions are like clicking on a button or typing typing a character in the form. JavaScript object in a document that receives events of different kinds. To handle the events that are taking place requires an even handler that can handle the execution of the events. Event acts like an added attribute that is entered in object's HTML. The attribute is consisting of event name, sign like (=), instructions. The following code shows the event handler as:

```
<HTML>
<BODY>
<FORM>
<INPUT TYPE="button" VALUE="Hello"
onClick="window.alert ('HELLO WORLD')">
</FORM>
</BODY>
</HTML>
```