**JAVASCRIPT ----**

1. What is JavaScript?

JavaScript is a client-side as well as server side scripting language that can be inserted into HTML pages and is understood by web browsers. JavaScript is also an Object based Programming language

2. Enumerate the differences between Java and JavaScript?

Java is a complete programming language. In contrast, JavaScript is a coded program that can be introduced to HTML pages. These two languages are not at all inter-dependent and are designed for the different intent. Java is an object - oriented programming (OOPS) or structured programming language like C++ or C whereas JavaScript is a client-side scripting language.

3. What are JavaScript Data Types?

Following are the JavaScript Data types:

* Number
* String
* Boolean
* Object
* Undefined

4. What is the use of isNaN function?

isNan function returns true if the argument is not a number otherwise it is false.

5. Between JavaScript and an ASP script, which is faster?

JavaScript is faster. JavaScript is a client-side language and thus it does not need the assistance of the web server to execute. On the other hand, ASP is a server-side language and hence is always slower than JavaScript. Javascript now is also a server side language (nodejs).

6. What is negative infinity?

Negative Infinity is a number in JavaScript which can be derived by dividing negative number by zero.

7. Is it possible to break JavaScript Code into several lines?

Breaking within a string statement can be done by the use of a backslash, '\', at the end of the first line

Example:

document.write("This is \a program");

And if you change to a new line when not within a string statement, then javaScript ignores break in line.

Example:

var x=1, y=2,

z=

x+y;

The above code is perfectly fine, though not advisable as it hampers debugging.

8. Which company developed JavaScript?

Netscape is the software company who developed JavaScript.

9. What are undeclared and undefined variables?

Undeclared variables are those that do not exist in a program and are not declared. If the program tries to read the value of an undeclared variable, then a runtime error is encountered.

Undefined variables are those that are declared in the program but have not been given any value. If the program tries to read the value of an undefined variable, an undefined value is returned.

10. Write the code for adding new elements dynamically?

<html>

<head>

<title>t1</title>

<script type="text/javascript">

function addNode() { var newP = document.createElement("p");

var textNode = document.createTextNode(" This is a new text node");

newP.appendChild(textNode); document.getElementById("firstP").appendChild(newP); }

</script> </head>

<body> <p id="firstP">firstP<p> </body>

</html>

11. What are global variables? How are these variable declared and what are the problems associated with using them?

Global variables are those that are available throughout the length of the code, that is, these have no scope. The var keyword is used to declare a local variable or object. If the var keyword is omitted, a global variable is declared.

Example:

// Declare a global globalVariable = "Test";

The problems that are faced by using global variables are the clash of variable names of local and global scope. Also, it is difficult to debug and test the code that relies on global variables.

12. What is a prompt box?

A prompt box is a box which allows the user to enter input by providing a text box. Label and box will be provided to enter the text or number.

13. What is 'this' keyword in JavaScript?

'This' keyword refers to the object from where it was called.

14. Explain the working of timers in JavaScript? Also elucidate the drawbacks of using the timer, if any?

Timers are used to execute a piece of code at a set time or also to repeat the code in a given interval of time. This is done by using the functions setTimeout, setInterval and clearInterval.

The setTimeout(function, delay) function is used to start a timer that calls a particular function after the mentioned delay. The setInterval(function, delay) function is used to repeatedly execute the given function in the mentioned delay and only halts when cancelled. The clearInterval(id)function instructs the timer to stop.

Timers are operated within a single thread, and thus events might queue up, waiting to be executed.

15. Which symbol is used for comments in Javascript?

// for Single line comments and

/\* Multi

Line

Comment

\*/

16. What is the difference between ViewState and SessionState?

'ViewState' is specific to a page in a session.

'SessionState' is specific to user specific data that can be accessed across all pages in the web application.

17. What is === operator?

=== is called as strict equality operator which returns true when the two operands are having the same value without any type conversion.

18. Explain how can you submit a form using JavaScript?

To submit a form using JavaScript use document.form[0].submit();

document.form[0].submit();

19. Does JavaScript support automatic type conversion?

Yes JavaScript does support automatic type conversion, it is the common way of type conversion used by JavaScript developers

20. How can the style/class of an element be changed?

It can be done in the following way:

document.getElementById("myText").style.fontSize = "20?;

or

document.getElementById("myText").className = "anyclass";

21. Explain how to read and write a file using JavaScript?

There are two ways to read and write a file using JavaScript

* Using JavaScript extensions
* Using a web page and Active X objects

22. What are all the looping structures in JavaScript?

Following are looping structures in Javascript:

* For
* While
* do-while loops

23. What is called Variable typing in Javascript?

Variable typing is used to assign a number to a variable and the same variable can be assigned to a string.

Example

i = 10;

i = "string";

This is called variable typing.

24. How can you convert the string of any base to integer in JavaScript?

The parseInt() function is used to convert numbers between different bases. parseInt() takes the string to be converted as its first parameter, and the second parameter is the base of the given string.

In order to convert 4F (of base 16) to integer, the code used will be -

parseInt ("4F", 16);

25. Explain the difference between "==" and "==="?

"==" checks only for equality in value whereas "===" is a stricter equality test and returns false if either the value or the type of the two variables are different.

26. What would be the result of 3+2+"7"?

Since 3 and 2 are integers, they will be added numerically. And since 7 is a string, its concatenation will be done. So the result would be 57.

27. Explain how to detect the operating system on the client machine?

In order to detect the operating system on the client machine, the navigator.platform string (property) should be used.

28. What do mean by NULL in Javascript?

The NULL value is used to represent no value or no object. It implies no object or null string, no valid boolean value, no number and no array object.

29. What is the function of delete operator?

The delete keyword is used to delete the property as well as its value.

Example

var student= {age:20, batch:"ABC"};

delete student.age;

30. What is an undefined value in JavaScript?

Undefined value means the

* Variable used in the code doesn't exist
* Variable is not assigned to any value
* Property doesn't exist

31. What are all the types of Pop up boxes available in JavaScript?

* Alert
* Confirm and
* Prompt

32. What is the use of Void(0)?

Void(0) is used to prevent the page from refreshing and parameter "zero" is passed while calling.

Void(0) is used to call another method without refreshing the page.

33. How can a page be forced to load another page in JavaScript?

The following code has to be inserted to achieve the desired effect:

<script language="JavaScript" type="text/javascript" >

<!-- location.href="http://newhost/newpath/newfile.html"; //--></script>

34. What is the data type of variables of in JavaScript?

All variables in the JavaScript are object data types.

35. What is the difference between an alert box and a confirmation box?

An alert box displays only one button which is the OK button.

But a Confirmation box displays two buttons namely OK and cancel.

36. What are escape characters?

Escape characters (Backslash) is used when working with special characters like single quotes, double quotes, apostrophes and ampersands. Place backslash before the characters to make it display.

Example:

document.write "I m a "good" boy"

document.write "I m a \"good\" boy"

37. What are JavaScript Cookies?

Cookies are the small test files stored in a computer and it gets created when the user visits the websites to store information that they need. Example could be User Name details and shopping cart information from the previous visits.

38. Explain what is pop()method in JavaScript?

The pop() method is similar as the shift() method but the difference is that the Shift method works at the start of the array. Also the pop() method take the last element off of the given array and returns it. The array on which is called is then altered.

Example:

var cloths = ["Shirt", "Pant", "TShirt"];

cloths.pop();

//Now cloth becomes Shirt,Pant

39. Whether JavaScript has concept level scope?

No. JavaScript does not have concept level scope. The variable declared inside the function has scope inside the function.

40. Mention what is the disadvantage of using innerHTML in JavaScript?

If you use innerHTML in JavaScript the disadvantage is

* Content is replaced everywhere
* We cannot use like "appending to innerHTML"
* Even if you use +=like "innerHTML = innerHTML + 'html'" still the old content is replaced by html
* The entire innerHTML content is re-parsed and build into elements, therefore its much slower
* The innerHTML does not provide validation and therefore we can potentially insert valid and broken HTML in the document and break it

41. What is break and continue statements?

Break statement exits from the current loop.

Continue statement continues with next statement of the loop.

42. What are the two basic groups of dataypes in JavaScript?

They are as –

* Primitive
* Reference types.

Primitive types are number and Boolean data types. Reference types are more complex types like strings and dates.

43. How generic objects can be created?

Generic objects can be created as:

var I = new object();

44. What is the use of type of operator?

'Typeof' is an operator which is used to return a string description of the type of a variable.

45. Which keywords are used to handle exceptions?

Try… Catch---finally is used to handle exceptions in the JavaScript

Try{

Code

}

Catch(exp){

Code to throw an exception

}

Finally{

Code runs either it finishes successfully or after catch

}

46. Which keyword is used to print the text in the screen?

document.write("Welcome") is used to print the text – Welcome in the screen.

47. What is the use of blur function?

Blur function is used to remove the focus from the specified object.

48. What is variable typing?

Variable typing is used to assign a number to a variable and then assign string to the same variable. Example is as follows:

i= 8;

i="john";

49. How to find operating system in the client machine using JavaScript?

The 'Navigator.appversion' is used to find the name of the operating system in the client machine.

50. What are the different types of errors in JavaScript?

There are three types of errors:

* Load time errors: Errors which come up when loading a web page like improper syntax errors are known as Load time errors and it generates the errors dynamically.
* Run time errors: Errors that come due to misuse of the command inside the HTML language.
* Logical Errors: These are the errors that occur due to the bad logic performed on a function which is having different operation.

51. What is the use of Push method in JavaScript?

The push method is used to add or append one or more elements to the end of an Array. Using this method, we can append multiple elements by passing multiple arguments

52. What is unshift method in JavaScript?

Unshift method is like push method which works at the beginning of the array. This method is used to prepend one or more elements to the beginning of the array.

53. What is the difference between JavaScript and Jscript?

Both are almost similar. JavaScript is developed by Netscape and Jscript was developed by Microsoft .

54. How are object properties assigned?

Properties are assigned to objects in the following way -

obj["class"] = 12;

or

obj.class = 12;

55. What is the 'Strict' mode in JavaScript and how can it be enabled?

Strict Mode adds certain compulsions to JavaScript. Under the strict mode, JavaScript shows errors for a piece of codes, which did not show an error before, but might be problematic and potentially unsafe. Strict mode also solves some mistakes that hamper the JavaScript engines to work efficiently.

Strict mode can be enabled by adding the string literal "use strict" above the file. This can be illustrated by the given example:

function myfunction() {

"use strict";

var v = "This is a strict mode function";

}

56. What is the way to get the status of a CheckBox?

The status can be acquired as follows -

alert(document.getElementById('checkbox1').checked);

If the CheckBox will be checked, this alert will return TRUE.

57. How can the OS of the client machine be detected?

The navigator.appVersion string can be used to detect the operating system on the client machine.

58. Explain window.onload and onDocumentReady?

The onload function is not run until all the information on the page is loaded. This leads to a substantial delay before any code is executed.

onDocumentReady loads the code just after the DOM is loaded. This allows early manipulation of the code.

59. How will you explain closures in JavaScript? When are they used?

Closure is a locally declared variable related to a function which stays in memory when the function has returned.

For example:

function greet(message) {

console.log(message);

}

function greeter(name, age) {

return name + " says howdy!! He is " + age + " years old";

}

// Generate the message

var message = greeter("James", 23);

// Pass it explicitly to greet

greet(message);

This function can be better represented by using closures

function greeter(name, age) {

var message = name + " says howdy!! He is " + age + " years old";

return function greet() {

console.log(message);

};

}

// Generate the closure

var JamesGreeter = greeter("James", 23);

// Use the closure

JamesGreeter();

60. How can a value be appended to an array?

A value can be appended to an array in the given manner -

arr[arr.length] = value;

61. Explain the for-in loop?

The for-in loop is used to loop through the properties of an object.

The syntax for the for-in loop is -

for (variable name in object){

statement or block to execute

}

In each repetition, one property from the object is associated to the variable name, and the loop is continued till all the properties of the object are depleted.

62. Describe the properties of an anonymous function in JavaScript?

A function that is declared without any named identifier is known as an anonymous function. In general, an anonymous function is inaccessible after its declaration.

Anonymous function declaration -

var anon = function() {

alert('I am anonymous');

};

anon();

63. What is the difference between .call() and .apply()?

The function .call() and .apply() are very similar in their usage except a little difference. .call() is used when the number of the function's arguments are known to the programmer, as they have to be mentioned as arguments in the call statement. On the other hand, .apply() is used when the number is not known. The function .apply() expects the argument to be an array.

The basic difference between .call() and .apply() is in the way arguments are passed to the function. Their usage can be illustrated by the given example.

var someObject = {

myProperty : 'Foo',

myMethod : function(prefix, postfix) {

alert(prefix + this.myProperty + postfix);

}

};

someObject.myMethod('<', '>'); // alerts '<Foo>'

var someOtherObject = {

myProperty : 'Bar'

};

someObject.myMethod.call(someOtherObject, '<', '>'); // alerts '<Bar>'

someObject.myMethod.apply(someOtherObject, ['<', '>']); // alerts '<Bar>'

64. Define event bubbling?

JavaScript allows DOM elements to be nested inside each other. In such a case, if the handler of the child is clicked, the handler of parent will also work as if it were clicked too.

65. Is JavaScript case sensitive? Give an example?

Yes, JavaScript is case sensitive. For example, a function parseInt is not same as the function Parseint.

66. What boolean operators can be used in JavaScript?

The 'And' Operator (&&), 'Or' Operator (||) and the 'Not' Operator (!) can be used in JavaScript.

\*Operators are without the parenthesis.

67. How can a particular frame be targeted, from a hyperlink, in JavaScript?

This can be done by including the name of the required frame in the hyperlink using the 'target' attribute.

<a href="/newpage.htm" target="newframe">>New Page</a>

68. What is the role of break and continue statements?

Break statement is used to come out of the current loop while the continue statement continues the current loop with a new recurrence.

69. Write the point of difference between web-garden and a web-farm?

Both web-garden and web-farm are web hosting systems. The only difference is that web-garden is a setup that includes many processors in a single server while web-farm is a larger setup that uses more than one server.

70. How are object properties assigned?

Assigning properties to objects is done in the same way as a value is assigned to a variable. For example, a form object's action value is assigned as 'submit' in the following manner - Document.form.action="submit"

71. What is the method for reading and writing a file in JavaScript?

This can be done by Using JavaScript extensions (runs from JavaScript Editor), example for opening of a file -

fh = fopen(getScriptPath(), 0);

72. How are DOM utilized in JavaScript?

DOM stands for Document Object Model and is responsible for how various objects in a document interact with each other. DOM is required for developing web pages, which includes objects like paragraph, links, etc. These objects can be operated to include actions like add or delete. DOM is also required to add extra capabilities to a web page. On top of that, the use of API gives an advantage over other existing models.

73. How are event handlers utilized in JavaScript?

Events are the actions that result from activities, such as clicking a link or filling a form, by the user. An event handler is required to manage proper execution of all these events. Event handlers are an extra attribute of the object. This attribute includes event's name and the action taken if the event takes place.

74. Explain the role of deferred scripts in JavaScript?

By default, the parsing of the HTML code, during page loading, is paused until the script has not stopped executing. It means, if the server is slow or the script is particularly heavy, then the webpage is displayed with a delay. While using Deferred, scripts delays execution of the script till the time HTML parser is running. This reduces the loading time of web pages and they get displayed faster.

75. What are the various functional components in JavaScript?

The different functional components in JavaScript are-

First-class functions: Functions in JavaScript are utilized as first class objects. This usually means that these functions can be passed as arguments to other functions, returned as values from other functions, assigned to variables or can also be stored in data structures.

Nested functions: The functions, which are defined inside other functions, are called Nested functions. They are called 'everytime' the main function is invoked.

76. Write about the errors shown in JavaScript?

JavaScript gives a message if it encounters an error. The recognized errors are -

* Load-time errors: The errors shown at the time of the page loading are counted under Load-time errors. These errors are encountered by the use of improper syntax, and thus are detected while the page is getting loaded.
* Run-time errors: This is the error that comes up while the program is running. It is caused by illegal operations, for example, division of a number by zero, or trying to access a non-existent area of the memory.
* Logic errors: It is caused by the use of syntactically correct code, which does not fulfill the required task. For example, an infinite loop.

77. What are Screen objects?

Screen objects are used to read the information from the client's screen. The properties of screen objects are -

* AvailHeight: Gives the height of client's screen
* AvailWidth: Gives the width of client's screen.
* ColorDepth: Gives the bit depth of images on the client's screen
* Height: Gives the total height of the client's screen, including the taskbar
* Width: Gives the total width of the client's screen, including the taskbar

78. Explain the unshift() method ?

This method is functional at the starting of the array, unlike the push(). It adds the desired number of elements to the top of an array. For example -

var name = [ "john" ];

name.unshift( "charlie" );

name.unshift( "joseph", "Jane" );

console.log(name);

The output is shown below:

[" joseph "," Jane ", " charlie ", " john "]

79. Define unescape() and escape() functions?

The escape () function is responsible for coding a string so as to make the transfer of the information from one computer to the other, across a network.

For Example:

<script>

document.write(escape("Hello? How are you!"));

</script>

Output: Hello%3F%20How%20are%20you%21

The unescape() function is very important as it decodes the coded string.

It works in the following way. For example:

<script>

document.write(unescape("Hello%3F%20How%20are%20you%21"));

</script>

Output: Hello? How are you!

80. What are the decodeURI() and encodeURI()?

EncodeURl() is used to convert URL into their hex coding. And DecodeURI() is used to convert the encoded URL back to normal.

<script>

var uri="my test.asp?name=ståle&car=saab";

document.write(encodeURI(uri)+ "<br>");

document.write(decodeURI(uri));

</script>

Output -

my%20test.asp?name=st%C3%A5le&car=saab

my test.asp?name=ståle&car=saab

81. Why it is not advised to use innerHTML in JavaScript?

innerHTML content is refreshed every time and thus is slower. There is no scope for validation in innerHTML and, therefore, it is easier to insert rouge code in the document and, thus, make the web page unstable.

82. What does the following statement declares?

var myArray = [[[]]];

It declares a three dimensional array.

83. How are JavaScript and ECMA Script related?

ECMA Script are like rules and guideline while Javascript is a scripting language used for web development.

84. What is namespacing in JavaScript and how is it used?

Namespacing is used for grouping the desired functions, variables etc. under a unique name. It is a name that has been attached to the desired functions, objects and properties. This improves modularity in the coding and enables code reuse.

85. How can JavaScript codes be hidden from old browsers that don't support JavaScript?

For hiding JavaScript codes from old browsers:

Add "<!--" without the quotes in the code just after the <script> tag.

Add "//-->" without the quotes in the code just before the <script> tag.

Old browsers will now treat this JavaScript code as a long HTML comment. While, a browser that supports JavaScript, will take the "<!--" and "//-->" as one-line comments.

**Beginner Level JavaScript Interview Questions**

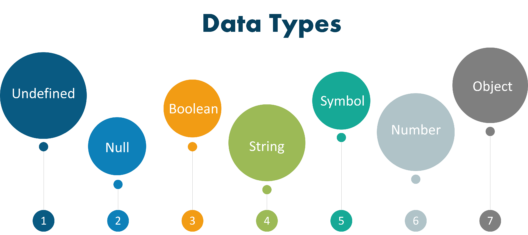
**Q1. What is the difference between Java & JavaScript?**

|  |  |
| --- | --- |
| **Java** | **JavaScript** |
| Java is an OOP programming language. | JavaScript is an OOP scripting language. |
| It creates applications that run in a virtual machine or browser. | The code is run on a browser only. |
| Java code needs to be compiled. | JavaScript code are all in the form of text. |

**Q2. What is JavaScript?**

[JavaScript](https://www.edureka.co/blog/what-is-javascript/) is a **lightweight**, **interpreted** programming language with object-oriented capabilities that allows you to build interactivity into otherwise static HTML pages. The general-purpose core of the language has been embedded in Netscape, Internet Explorer, and other web browsers.

**Q3. What are the data types supported by JavaScript?**



The **data types** supported by JavaScript are:

* Undefined
* Null
* Boolean
* String
* Symbol
* Number
* Object

**Q4. What are the features of JavaScript?**



Following are the **features** of JavaScript:

* It is a **lightweight, interpreted** programming language.
* It is designed for creating **network-centric** applications.
* It is complementary to and **integrated** with Java.
* It is an **open** and **cross-platform** scripting language.

**Q5. Is JavaScript a case-sensitive language?**

Yes, JavaScript is a**case sensitive** language.  The language keywords, variables, function names, and any other identifiers must always be typed with a consistent capitalization of letters.

**Q6. What are the advantages of JavaScript?**

Following are the **advantages** of using JavaScript −

* **Less server interaction** − You can validate user input before sending the page off to the server. This saves server traffic, which means less load on your server.
* **Immediate feedback to the visitors** − They don’t have to wait for a page reload to see if they have forgotten to enter something.
* **Increased interactivity** − You can create interfaces that react when the user hovers over them with a mouse or activates them via the keyboard.
* **Richer interfaces** − You can use JavaScript to include such items as drag-and-drop components and sliders to give a Rich Interface to your site visitors.

**Q7. How can you create an object in JavaScript?**

JavaScript supports **Object** concept very well. You can create an object using the **object literal** as follows −

|  |  |
| --- | --- |
| 1  2  3  4 | var emp = {  name: "Daniel",  age: 23  }; |

**Q8. How can you create an Array in JavaScript?**

You can define arrays using the **array literal** as follows-

|  |  |
| --- | --- |
| 1  2 | var x = [];  var y = [1, 2, 3, 4, 5]; |

**Q9. What is a name function in JavaScript & how to define it?**

A named function declares a name as soon as it is defined. It can be defined using **function**keyword as :

|  |  |
| --- | --- |
| 1  2  3 | function named(){  // write code here  } |

**Q10. Can you assign an anonymous function to a variable and pass it as an argument to another function?**

Yes! An anonymous function can be assigned to a variable. It can also be passed as an argument to another function.

**Q11. What is argument objects in JavaScript & how to get the type of arguments passed to a function?**

JavaScript variable arguments represents the **arguments** that are passed to a function. Using **typeof** operator, we can get the type of arguments passed to a function. For example −

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | function func(x){  console.log(typeof x, arguments.length);  }  func(); //==> "undefined", 0  func(7); //==> "number", 7  func("1", "2", "3"); //==> "string", 3 |

**Q12. What are the scopes of a variable in JavaScript?**

The scope of a variable is the **region** of your program in which it is **defined**. JavaScript variable will have only two scopes.  
• **Global Variables** − A global variable has global scope which means it is visible everywhere in your JavaScript code.  
•**Local Variables** − A local variable will be visible only within a function where it is defined. Function parameters are always local to that function.

**Q13. What is the purpose of ‘This’ operator in JavaScript?**

The JavaScript **this** keyword refers to the object it belongs to. This has different values depending on where it is used. In a method, this refers to the owner object and in a function, this refers to the global object.

**Q14. What is Callback?**

A **callback** is a plain JavaScript function passed to some method as an argument or option. It is a function that is to be **executed** after another function has finished executing, hence the name ‘**call back**‘. In JavaScript, functions are objects. Because of this, functions can take functions as arguments, and can be returned by other functions.



**Q15. What is Closure? Give an example.**

**Closures** are created whenever a variable that is defined outside the **current scope** is accessed from within some inner scope. It gives you access to an outer function’s scope from an inner function. In JavaScript, closures are created every time a function is created. To use a closure, simply define a function inside another function and expose it.

**Q16. Name some of the built-in methods and the values returned by them.**

|  |  |
| --- | --- |
| **Built-in Method** | **Values** |
| **CharAt()** | It returns the character at the specified index. |
| **Concat()** | It returns the character at the specified index. |
| **forEach()** | It calls a function for each element in the array. |
| **indexOf()** | It returns the index within the calling String object of the first occurrence of the specified value. |
| **length()** | It returns the length of the string. |
| **pop()** | It removes the last element from an array and returns that element. |
| **push()** | It adds one or more elements to the end of an array and returns the new length of the array. |
| **reverse()** | It reverses the order of the elements of an array. |

**Q17. What are the variable naming conventions in JavaScript?**

The following **rules** are to be followed while **naming variables** in JavaScript:

1. You should not use any of the JavaScript **reserved keyword** as variable name. For example, break or boolean variable names are not valid.
2. JavaScript variable names should not start with a **numeral** (0-9). They must begin with a letter or the underscore character. For example, 123name is an invalid variable name but \_123name or name123 is a valid one.
3. JavaScript variable names are **case sensitive**. For example, Test and test are two different variables.

**Q18. How does TypeOf Operator work?**

The **typeof** operator is used to get the data type of its operand. The operand can be either a **literal**or a **data structure** such as a variable, a function, or an object. It is a **unary** operator that is placed before its single operand, which can be of any type. Its value is a string indicating the data type of the operand.

**Q19. How to create a cookie using JavaScript?**

The simplest way to create a cookie is to assign a string value to the **document.cookie** object, which looks like this-

**Syntax :**

|  |  |
| --- | --- |
| 1 | document.cookie = "key1 = value1; key2 = value2; expires = date"; |

**Q20. How to read a cookie using JavaScript?**

Reading a cookie is just as simple as writing one, because the value of the document.cookie object is the cookie. So you can use this string whenever you want to access the cookie.

* The **document.cookie** string will keep a list of name = value pairs separated by semicolons, where name is the name of a cookie and value is its string value.
* You can use strings’ **split()** function to break the string into key and values.

**Q21. How to delete a cookie using JavaScript?**

If you want to delete a cookie so that subsequent attempts to read the cookie return nothing, you just need to set the expiration date to a time in the past. You should define the cookie path to ensure that you delete the right cookie. Some browsers will not let you delete a cookie if you don’t specify the path.

Now let’s move on to the next section of JavaScript interview questions.

**Intermediate Level JavaScript Interview Questions**

**Q22. What is the difference between Attributes and Property?**

**Attributes-**  provide more details on an element like id, type, value etc.

**Property-**  is the value assigned to the property like type=”text”, value=’Name’ etc.

**Q23. List out the different ways an HTML element can be accessed in a JavaScript code.**

Here are the list of ways an HTML element can be accessed in a Javascript code:  
(i) **getElementById(‘idname’):** Gets an element by its ID name  
(ii) **getElementsByClass(‘classname’):** Gets all the elements that have the given classname.  
(iii) **getElementsByTagName(‘tagname’):** Gets all the elements that have the given tag name.  
(iv) **querySelector():** This function takes css style selector and returns the first selected element.

**Q24. In how many ways a JavaScript code can be involved in an HTML file?**

There are 3 different ways in which a JavaScript code can be involved in an HTML file:

* **Inline**
* **Internal**
* **External**

An **inline** function is a JavaScript function, which is assigned to a variable created at runtime. You can differentiate between Inline Functions and Anonymous since an inline function is assigned to a variable and can be easily reused. When you need a JavaScript for a function, you can either have the script **integrated** in the page you are working on, or you can have it placed in a **separate** file that you call, when needed. This is the difference between an **internal**script and an **external**script.

**Q25. What are the ways to define a variable in JavaScript?**

The three possible ways of defining a variable in JavaScript are:

* **Var** – The JavaScript variables statement is used to declare a variable and, optionally, we can initialize the value of that variable. Example: var a =10; Variable declarations are processed before the execution of the code.
* **Const** – The idea of const functions is not allow them to modify the object on which they are called. When a function is declared as const, it can be called on any type of object.
* **Let** – It is a signal that the variable may be reassigned, such as a counter in a loop, or a value swap in an algorithm. It also signals that the variable will be used only in the block it’s defined in.

**Q26. What is a Typed language?**

Typed Language is in which the values are associated with**values** and not with **variables**. It is of two types:

* **Dynamically:** in this, the variable can hold multiple types; like in JS a variable can take number, chars.
* **Statically:** in this, the variable can hold only one type, like in Java a variable declared of string can take only set of characters and nothing else.

**Q27. What is the difference between Local storage & Session storage?**



**Local Storage** – The data is not sent back to the server for every HTTP request (HTML, images, JavaScript, CSS, etc) – reducing the amount of traffic between client and server. It will stay until it is manually cleared through settings or program.

**Session Storage** – It is similar to local storage; the only difference is while data stored in local storage has no expiration time, data stored in session storage gets cleared when the page session ends. Session Storage will leave when the browser is closed.

**Q28. What is the difference between the operators ‘==‘ & ‘===‘?**

The main difference between “==” and “===” operator is that formerly compares variable by making **type correction** e.g. if you compare a number with a string with numeric literal, == allows that, but === doesn’t allow that, because it not only checks the value but also type of two variable, if two variables are not of the same type “===” return false, while “==” return true.

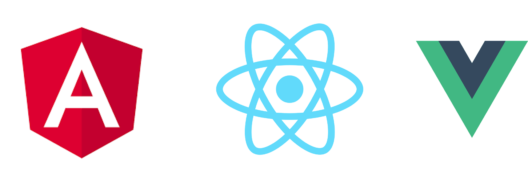
**Q29. What is the difference between null & undefined?**

Undefined means a variable has been **declared** but has not yet been **assigned** a value. On the other hand, null is an assignment value. It can be assigned to a variable as a representation of no value. Also, undefined and null are two distinct types: undefined is a type itself (undefined) while null is an object.

**Q30. What is the difference between undeclared & undefined?**

Undeclared variables are those that do not **exist** in a program and are not declared. If the program tries to read the value of an undeclared variable, then a **runtime error** is encountered. Undefined variables are those that are declared in the program but have not been given any value. If the program tries to read the value of an undefined variable, an undefined value is returned.

**Q31. Name some of the JavaScript Frameworks**

A [JavaScript framework](https://www.edureka.co/blog/top-10-javascript-frameworks/) is an application framework written in JavaScript. It differs from a JavaScript library in its control flow. There are many JavaScript Frameworks available but some of the most commonly used frameworks are:

* [Angular](https://www.edureka.co/angular-training)
* [React](https://www.edureka.co/blog/reactjs-tutorial)
* Vue

**Q32. What is the difference between window & document in JavaScript?**

|  |  |
| --- | --- |
| **Window** | **Document** |
| JavaScript window is a global object which holds variables, functions, history, location. | The document also comes under the window and can be considered as the property of the window. |

**Q33. What is the difference between innerHTML & innerText?**

**innerHTML** – It will process an HTML tag if found in a string

**innerText** – It will not process an HTML tag if found in a string

**Q34. What is an event bubbling in JavaScript?**

Event bubbling is a way of **event propagation** in the HTML DOM API, when an event occurs in an element inside another element, and both elements have registered a handle for that event. With bubbling, the event is first captured and handled by the **innermost** element and then propagated to outer elements. The execution starts from that event and goes to its parent element. Then the execution passes to its parent element and so on till the body element.

**Q35. What is NaN in JavaScript?**

**NaN** is a short form of **Not a Number.** Since NaN always compares unequal to any number, including NaN, it is usually used to indicate an error condition for a function that should return a valid number. When a string or something else is being **converted** into a **number** and that cannot be done, then we get to see NaN.

**Q36. How do JavaScript primitive/object types passed in functions?**

One of the differences between the two is that Primitive Data Types are passed By Value and Objects are passed By Reference.

* **By Value** means creating a COPY of the original. Picture it like twins: they are born exactly the same, but the first twin doesn’t lose a leg when the second twin loses his in the war.
* **By Reference** means creating an ALIAS to the original. When your Mom calls you “Pumpkin Pie” although your name is Margaret, this doesn’t suddenly give birth to a clone of yourself: you are still one, but you can be called by these two very different names.

**Q37. How can you convert the string of any base to integer in JavaScript?**

The **parseInt()** function is used to convert numbers between different bases. It takes the string to be converted as its first parameter, and the second parameter is the base of the given string.

For example-

|  |  |
| --- | --- |
| 1 | parseInt("4F", 16) |

**Q38. What would be the result of 2+5+”3″?**

Since 2 and 5 are integers, they will be added numerically. And since 3 is a string, its concatenation will be done. So the result would be 73. The ” ” makes all the difference here and represents 3 as a string and not a number.

**Q39. What are Exports & Imports?**

Imports and exports help us to write modular JavaScript code. Using Imports and exports we can split our code into multiple files. For example-

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13 | //------ lib.js ------</span>  export const sqrt = Math.sqrt;</span>  export function square(x) {</span>  return x \* x;</span>  }  export function diag(x, y) {  return sqrt(square(x) + square(y));  }    //------ main.js ------</span>   { square, diag } from 'lib';  console.log(square(5)); // 25  console.log(diag(4, 3)); // 5 |

Now with this, we have reached the final section of JavaScript Interview Questions.

**Advanced Level JavaScript Interview Questions**

**Q40. What is the ‘Strict’ mode in JavaScript and how can it be enabled?**

Strict mode is a way to introduce better error-checking into your code.

* When you use strict mode, you cannot use implicitly declared variables, or assign a value to a read-only property, or add a property to an object that is not extensible.
* You can enable strict mode by adding “use strict” at the beginning of a file, a program, or a function.

**Q41. What is a prompt box in JavaScript?**

A prompt box is a box which allows the user to enter input by providing a **text box**. The prompt() method displays a dialog box that prompts the visitor for input. A prompt box is often used if you want the user to input a value before entering a page. When a prompt box pops up, the user will have to click either “OK” or “Cancel” to proceed after entering an input value.

**Q42. What will be the output of the code below?**

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | var Y = 1;  if (function F(){})  {  y += Typeof F;</span>  }  console.log(y); |

The output would be 1undefined. The if condition statement evaluates using eval, so eval(function f(){}) returns function f(){} (which is true). Therefore, inside the if statement, executing typeof f returns undefined because the if statement code executes at run time, and the statement inside the if condition is evaluated during run time.

**Q43. What is the difference between Call & Apply?**

The **call()** method calls a function with a given this value and arguments provided individually.

**Syntax-**

|  |  |
| --- | --- |
| 1 | fun.call(thisArg[, arg1[, arg2[, ...]]]) |

The **apply()** method calls a function with a given this value, and arguments provided as an array.

**Syntax-**

|  |  |
| --- | --- |
| 1 | fun.apply(thisArg, [argsArray]) |

**Q44. How to empty an Array in JavaScript?**

There are a number of methods you can use to **empty** an **array**:

**Method 1 –**

|  |  |
| --- | --- |
| 1 | arrayList = [] |

Above code will set the variable arrayList to a new empty array. This is recommended if you don’t have references to the original array arrayList anywhere else, because it will actually create a new, empty array. You should be careful with this method of emptying the array, because if you have referenced this array from another variable, then the original reference array will remain unchanged.

**Method 2 –**

|  |  |
| --- | --- |
| 1 | arrayList.length = 0; |

The code above will clear the existing array by setting its length to 0. This way of emptying the array also updates all the reference variables that point to the original array. Therefore, this method is useful when you want to update all reference variables pointing to arrayList.

**Method 3 –**

|  |  |
| --- | --- |
| 1 | arrayList.splice(0, arrayList.length); |

The implementation above will also work perfectly. This way of emptying the array will also update all the references to the original array.

**Method 4 –**

|  |  |
| --- | --- |
| 1  2  3  4 | while(arrayList.length)  {  arrayList.pop();  } |

The implementation above can also empty arrays, but it is usually not recommended to use this method often.

**Q45. What will be the output of the following code?**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7 | var Output = (function(x)  {  Delete X;  return X;  }  )(0);  console.log(output); |

The output would be 0. The delete operator is used to delete properties from an object. Here x is not an object but a local variable. delete operators don’t affect local variables.

**Q46. What will be the output of the following code?**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | var X = { Foo : 1};  var Output = (function()  {  delete X.foo;  return X.foo;  }  )();  console.log(output); |

The output would be undefined. The delete operator is used to delete the property of an object. Here, x is an object which has the property foo, and as it is a self-invoking function, we will delete the foo property from object x. After doing so, when we try to reference a deleted property foo, the result is undefined.

**Q47. What will be the output of the following code?**

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | var Employee =  {  company: 'xyz'  }  var Emp1 = Object.create(employee);  delete Emp1.company Console.log(emp1.company); |

The output would be xyz. Here, emp1 object has company as its prototype property. The delete operator doesn’t delete prototype property. emp1 object doesn’t have company as its own property. However, we can delete the company property directly from the Employee object using delete Employee.company.

**Q48. What will be the output of the code below?**

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | //nfe (named function expression)  var Foo = Function Bar()  {  return 7;  };  typeof Bar(); |

The output would be Reference Error. A function definition can have only one reference variable as its function name.

**Q49. What is the reason for wrapping the entire content of a JavaScript source file in a function book?**

This is an increasingly common practice, employed by many popular JavaScript libraries. This technique creates a closure around the entire contents of the file which, perhaps most importantly, creates a private namespace and thereby helps avoid potential name clashes between different JavaScript modules and libraries.  
Another feature of this technique is to allow for an easy alias for a global variable. This is often used in jQuery plugins.

**Q50. What are escape characters in JavaScript?**

JavaScript escape characters enable you to write special characters without breaking your application. Escape characters (Backslash) is used when working with special characters like single quotes, double quotes, apostrophes and ampersands. Place backslash before the characters to make it display.

**For example-**

|  |  |
| --- | --- |
| 1  2 | document.write "I am a "good" boy"  document.write "I am a "good" boy" |

1) What is JavaScript?

**JavaScript** is *a scripting language*. It is different from Java language. It is object-based, lightweight, cross-platform translated language. It is widely used for client-side validation. The JavaScript Translator (embedded in the browser) is responsible for translating the JavaScript code for the web browser.[More details.](https://www.javatpoint.com/javascript-tutorial)

2) List some features of JavaScript.

Some of the features of JavaScript are:

* Lightweight
* Interpreted programming language
* Good for the applications which are network-centric
* Complementary to Java
* Complementary to HTML
* Open source

Cross-platform

3) List some of the advantages of JavaScript.

Some of the advantages of JavaScript are:

* Server interaction is less
* Feedback to the visitors is immediate
* Interactivity is high
* Interfaces are richer

4) List some of the disadvantages of JavaScript.

Some of the disadvantages of JavaScript are:

* No support for multithreading
* No support for multiprocessing
* Reading and writing of files is not allowed
* No support for networking applications.

5) Define a named function in JavaScript.

The function which has named at the time of definition is called a named function. For example

1. function msg()
2. {
3. document.writeln("Named Function");
4. }
5. msg();

6) Name the types of functions

The types of function are:

* Named - These type of functions contains name at the time of definition. For Example:
  1. function display()
  2. {
  3. document.writeln("Named Function");
  4. }
  5. display();
* Anonymous - These type of functions doesn't contain any name. They are declared dynamically at runtime.
  1. var display=function()
  2. {
  3. document.writeln("Anonymous Function");
  4. }
  5. display();

7) Define anonymous function

It is a function that has no name. These functions are declared dynamically at runtime using the function operator instead of the function declaration. The function operator is more flexible than a function declaration. It can be easily used in the place of an expression. For example:

1. var display=function()
2. {
3. alert("Anonymous Function is invoked");
4. }
5. display();

8) Can an anonymous function be assigned to a variable?

Yes, you can assign an anonymous function to a variable.

9) In JavaScript what is an argument object?

The variables of JavaScript represent the arguments that are passed to a function.

10) Define closure.

In JavaScript, we need closures when a variable which is defined outside the scope in reference is accessed from some inner scope.

1. var num = 10;
2. function sum()
3. {
4. document.writeln(num+num);
5. }
6. sum();

11) If we want to return the character from a specific index which method is used?

The JavaScript string charAt() method is used to find out a char value present at the specified index. The index number starts from 0 and goes to n-1, where n is the length of the string. The index value can't be a negative, greater than or equal to the length of the string. For example:

1. var str="Javatpoint";
2. document.writeln(str.charAt(4));

12) What is the difference between JavaScript and JScript?

Netscape provided the JavaScript language. Microsoft changed the name and called it JScript to avoid the trademark issue. In other words, you can say JScript is the same as JavaScript, but Microsoft provides it.

13) How to write a hello world example of JavaScript?

A simple example of JavaScript hello world is given below. You need to place it inside the body tag of HTML.

1. **<script** type="text/javascript"**>**
2. document.write("JavaScript Hello World!");
3. **</script>**

[More details.](https://www.javatpoint.com/javascript-example)

14) How to use external JavaScript file?

I am assuming that js file name is message.js, place the following script tag inside the head tag.

1. **<script** type="text/javascript" src="message.js"**></script>**

[More details.](https://www.javatpoint.com/external-javascript-file)

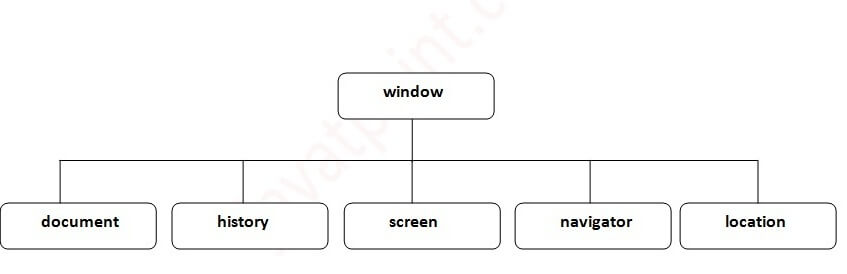
15) Is JavaScript case sensitive language?

Yes, JavaScript is a case sensitive language. For example:

1. Var msg = "JavaScript is a case-sensitive language"; //Here, var should be used to declare a variable
2. function display()
3. {
4. document.writeln(msg); // It will not display the result.
5. }
6. display();

16) What is BOM?

**BOM** stands for *Browser Object Model*. It provides interaction with the browser. The default object of a browser is a window. So, you can call all the functions of the window by specifying the window or directly. The window object provides various properties like document, history, screen, navigator, location, innerHeight, innerWidth,

[More Details: Browser Object Model](https://www.javatpoint.com/browser-object-model)

17) What is DOM? What is the use of document object?

**DOM** stands for *Document Object Model*. A document object represents the HTML document. It can be used to access and change the content of HTML.

[More Details: Document Object Model](https://www.javatpoint.com/document-object-model)

18) What is the use of window object?

The window object is created automatically by the browser that represents a window of a browser. It is not an object of JavaScript. It is a browser object.

The window object is used to display the popup dialog box. Let's see with description.

|  |  |
| --- | --- |
| **Method** | **Description** |
| alert() | displays the alert box containing the message with ok button. |
| confirm() | displays the confirm dialog box containing the message with ok and cancel button. |
| prompt() | displays a dialog box to get input from the user. |
| open() | opens the new window. |
| close() | closes the current window. |
| setTimeout() | performs the action after specified time like calling function, evaluating expressions. |

[More details.](https://www.javatpoint.com/window-object)

19) What is the use of history object?

The history object of a browser can be used to switch to history pages such as back and forward from the current page or another page. There are three methods of history object.

1. history.back() - It loads the previous page.
2. history.forward() - It loads the next page.
3. history.go(number) - The number may be positive for forward, negative for backward. It loads the given page number.

[More details.](https://www.javatpoint.com/javascript-history-object)

20) How to write a comment in JavaScript?

There are two types of comments in JavaScript.

1. Single Line Comment: It is represented by // (double forward slash)
2. Multi-Line Comment: Slash represents it with asterisk symbol as /\* write comment here \*/

[More details.](https://www.javatpoint.com/javascript-comment)

21) How to create a function in JavaScript?

To create a function in JavaScript, follow the following syntax.

1. function function\_name(){
2. //function body
3. }

[More details.](https://www.javatpoint.com/javascript-function)

22) What are the JavaScript data types?

There are two types of data types in JavaScript:

1. Primitive Data Types - The primitive data types are as follows:

|  |  |
| --- | --- |
| **Data Type** | **Description** |
| String | represents a sequence of characters, e.g., "hello" |
| Number | represents numeric values, e.g., 100 |
| Boolean | represents boolean value either false or true |
| Undefined | represents an undefined value |
| Null | represents null, i.e., no value at all |

1. Non-primitive Data Types - The non-primitive data types are as follows:

|  |  |
| --- | --- |
| **Data Type** | **Description** |
| Object | represents an instance through which we can access members |
| Array | represents a group of similar values |
| RegExp | represents regular expression |

[More details.](https://www.javatpoint.com/javascript-data-types)

23) What is the difference between == and ===?

The == operator checks equality only whereas === checks equality, and data type, i.e., a value must be of the same type.

24) How to write HTML code dynamically using JavaScript?

The innerHTML property is used to write the HTML code using JavaScript dynamically. Let's see a simple example:

1. document.getElementById('mylocation').innerHTML="<h2>This is heading using JavaScript</h2>";

[More details.](https://www.javatpoint.com/javascript-innerHTML)

25) How to write normal text code using JavaScript dynamically?

The innerText property is used to write the simple text using JavaScript dynamically. Let's see a simple example:

1. document.getElementById('mylocation').innerText="This is text using JavaScript";

[More details.](https://www.javatpoint.com/javascript-innerText)

26) How to create objects in JavaScript?

There are 3 ways to create an object in JavaScript.

1. By object literal
2. By creating an instance of Object
3. By Object Constructor

Let's see a simple code to create an object using object literal.

1. emp={id:102,name:"Rahul Kumar",salary:50000}

[More details.](https://www.javatpoint.com/javascript-objects)

27) How to create an array in JavaScript?

There are 3 ways to create an array in JavaScript.

1. By array literal
2. By creating an instance of Array
3. By using an Array constructor

Let's see a simple code to create an array using object literal.

1. var emp=["Shyam","Vimal","Ratan"];

[More details.](https://www.javatpoint.com/javascript-array)

28) What does the isNaN() function?

The isNan() function returns true if the variable value is not a number. For example:

1. function number(num) {
2. if (isNaN(num)) {
3. return "Not a Number";
4. }
5. return "Number";
6. }
7. console.log(number('1000F'));
8. // expected output: "Not a Number"
10. console.log(number('1000'));
11. // expected output: "Number"

29) What is the output of 10+20+"30" in JavaScript?

3030 because 10+20 will be 30. If there is numeric value before and after +, it treats as binary + (arithmetic operator).

1. function display()
2. {
3. document.writeln(10+20+"30");
4. }
5. display();

30) What is the output of "10"+20+30 in JavaScript?

102030 because after a string all the + will be treated as string concatenation operator (not binary +).

1. function display()
2. {
3. document.writeln("10"+20+30);
4. }
5. display();

31) Difference between Client side JavaScript and Server side JavaScript?

**Client-side JavaScript** comprises the basic language and predefined objects which are relevant to running JavaScript in a browser. The client-side JavaScript is embedded directly by in the HTML pages. The browser interprets this script at runtime.

**Server-side JavaScript** also resembles client-side JavaScript. It has a relevant JavaScript which is to run in a server. The server-side JavaScript are deployed only after compilation.

32) In which location cookies are stored on the hard disk?

The storage of cookies on the hard disk depends on the OS and the browser.

The Netscape Navigator on Windows uses a cookies.txt file that contains all the cookies. The path is c:\Program Files\Netscape\Users\username\cookies.txt

The Internet Explorer stores the cookies on a file username@website.txt. The path is: c:\Windows\Cookies\username@Website.txt.

33) What is the real name of JavaScript?

The original name was **Mocha**, a name chosen by Marc Andreessen, founder of Netscape. In September of 1995, the name was changed to LiveScript. In December 1995, after receiving a trademark license from Sun, the name JavaScript was adopted.

34) What is the difference between undefined value and null value?

**Undefined value:** A value that is not defined and has no keyword is known as undefined value. For example:

1. int number;//Here, a number has an undefined value.

**Null value:** A value that is explicitly specified by the keyword "null" is known as a null value. For example:

1. String str=null;//Here, str has a null value.

35) How to set the cursor to wait in JavaScript?

The cursor can be set to wait in JavaScript by using the property "cursor". The following example illustrates the usage:

1. **<script>**
2. window.document.body.style.cursor = "wait";
3. **</script>**

36) What is this [[[]]]?

This is a three-dimensional array.

1. var myArray = [[[]]];

37) Are Java and JavaScript same?

No, Java and JavaScript are the two different languages. Java is a robust, secured and object-oriented programming language whereas JavaScript is a client-side scripting language with some limitations.

38) What is negative infinity?

Negative Infinity is a number in JavaScript which can be derived by dividing the negative number by zero. For example:

1. var num=-5;
2. function display()
3. {
4. document.writeln(num/0);
5. }
6. display();
7. //expected output: -Infinity

39) What is the difference between View state and Session state?

"View state" is specific to a page in a session whereas "Session state" is specific to a user or browser that can be accessed across all pages in the web application.

40) What are the pop-up boxes available in JavaScript?

* Alert Box
* Confirm Box
* Prompt Box

Example of alert() in JavaScript

1. **<script** type="text/javascript"**>**
2. function msg(){
3. alert("Hello Alert Box");
4. }
5. **</script>**
6. **<input** type="button" value="click" onclick="msg()"**/>**

Example of confirm() in JavaScript

1. **<script** type="text/javascript"**>**
2. function msg(){
3. var v= confirm("Are u sure?");
4. if(v==true){
5. alert("ok");
6. }
7. else{
8. alert("cancel");
9. }
11. }
12. **</script>**
14. **<input** type="button" value="delete record" onclick="msg()"**/>**

Example of prompt() in JavaScript

1. **<script** type="text/javascript"**>**
2. function msg(){
3. var v= prompt("Who are you?");
4. alert("I am "+v);
6. }
7. **</script>**
9. **<input** type="button" value="click" onclick="msg()"**/>**

41) How can we detect OS of the client machine using JavaScript?

The **navigator.appVersion** string can be used to detect the operating system on the client machine.

42) How to submit a form using JavaScript by clicking a link?

Let's see the JavaScript code to submit the form by clicking the link.

1. **<form** name="myform" action="index.php"**>**
2. Search: **<input** type='text' name='query' **/>**
3. **<a** href="javascript: submitform()"**>**Search**</a>**
4. **</form>**
5. **<script** type="text/javascript"**>**
6. function submitform()
7. {
8. document.myform.submit();
9. }
10. **</script>**

43) Is JavaScript faster than ASP script?

Yes, because it doesn't require web server's support for execution.

44) How to change the background color of HTML document using JavaScript?

1. **<script** type="text/javascript"**>**
2. document.body.bgColor="pink";
3. **</script>**

45) How to handle exceptions in JavaScript?

By the help of try/catch block, we can handle exceptions in JavaScript. JavaScript supports try, catch, finally and throw keywords for exception handling.

46) How to validate a form in JavaScript?

1. **<script>**
2. function validateform(){
3. var name=document.myform.name.value;
4. var password=document.myform.password.value;
6. if (name==null || name==""){
7. alert("Name can't be blank");
8. return false;
9. }else if(password.length**<6**){
10. alert("Password must be at least 6 characters long.");
11. return false;
12. }
13. }
14. **</script>**
15. **<body>**
16. **<form** name="myform" method="post" action="abc.jsp" onsubmit="return validateform()" **>**
17. Name: **<input** type="text" name="name"**><br/>**
18. Password: **<input** type="password" name="password"**><br/>**
19. **<input** type="submit" value="register"**>**
20. **</form>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=jsvalidation1)

Visit here: [JavaScript form validation](https://www.javatpoint.com/javascript-form-validation).

47) How to validate email in JavaScript?

1. **<script>**
2. function validateemail()
3. {
4. var x=document.myform.email.value;
5. var atposition=x.indexOf("@");
6. var dotposition=x.lastIndexOf(".");
7. if (atposition**<1** || dotposition**<atposition**+2 || dotposition+2**>**=x.length){
8. alert("Please enter a valid e-mail address \n atpostion:"+atposition+"\n dotposition:"+dotposition);
9. return false;
10. }
11. }
12. **</script>**
13. **<body>**
14. **<form** name="myform"  method="post" action="#" onsubmit="return validateemail();"**>**
15. Email: **<input** type="text" name="email"**><br/>**
17. **<input** type="submit" value="register"**>**
18. **</form>**

[**Test it Now**](http://www.javatpoint.com/oprweb/test.jsp?filename=jsvalidation3)

Visit here: [JavaScript Email validation](_blank).

48) What is this keyword in JavaScript?

The this keyword is a reference variable that refers to the current object. For example:

1. var address=
2. {
3. company:"Javatpoint",
4. city:"Noida",
5. state:"UP",
6. fullAddress:function()
7. {
8. return this.company+" "+this.city+" "+this.state;
9. }
10. };
11. var fetch=address.fullAddress();
12. document.writeln(fetch);

49) What is the requirement of debugging in JavaScript?

JavaScript didn't show any error message in a browser. However, these mistakes can affect the output. The best practice to find out the error is to debug the code. The code can be debugged easily by using web browsers like Google Chrome, Mozilla Firebox.

To perform debugging, we can use any of the following approaches:

* Using console.log() method
* Using debugger keyword

50) What is the use of debugger keyword in JavaScript?

JavaScript debugger keyword sets the breakpoint through the code itself. The debugger stops the execution of the program at the position it is applied. Now, we can start the flow of execution manually. If an exception occurs, the execution will stop again on that particular line.. For example:

1. function display()
2. {
3. x = 10;
4. y = 15;
5. z = x + y;
6. debugger;
7. document.write(z);
8. document.write(a);
9. }
10. display();

51) What is the role of a strict mode in JavaScript?

The JavaScript strict mode is used to generates silent errors. It provides "use strict"; expression to enable the strict mode. This expression can only be placed as the first statement in a script or a function. For example:

1. "use strict";
2. x=10;
3. console.log(x);

52) What is the use of Math object in JavaScript?

The JavaScript math object provides several constants and methods to perform a mathematical operation. Unlike date object, it doesn't have constructors. For example:

1. function display()
2. {
3. document.writeln(Math.random());
4. }
5. display();

53) What is the use of a Date object in JavaScript?

The JavaScript date object can be used to get a year, month and day. You can display a timer on the webpage by the help of JavaScript date object.

1. function display()
2. {
3. var date=new Date();
4. var day=date.getDate();
5. var month=date.getMonth()+1;
6. var year=date.getFullYear();
7. document.write("**<br>**Date is: "+day+"/"+month+"/"+year);
8. }
9. display();

54) What is the use of a Number object in JavaScript?

The JavaScript number object enables you to represent a numeric value. It may be integer or floating-point. JavaScript number object follows the IEEE standard to represent the floating-point numbers.

1. function display()
2. {
3. var x=102;//integer value
4. var y=102.7;//floating point value
5. var z=13e4;//exponent value, output: 130000
6. var n=new Number(16);//integer value by number object
7. document.write(x+" "+y+" "+z+" "+n);
8. }
9. display();

55) What is the use of a Boolean object in JavaScript?

The JavaScript Boolean is an object that represents value in two states: true or false. You can create the JavaScript Boolean object by Boolean() constructor.

1. function display()
2. {
3. document.writeln(10**<20**);//true
4. document.writeln(10**<5**);//false
5. }
6. display();

56) What is the use of a TypedArray object in JavaScript?

The JavaScript TypedArray object illustrates an array like a view of an underlying binary data buffer. There is any number of different global properties, whose values are TypedArray constructors for specific element types.

1. function display()
2. {
3. var arr1= [1,2,3,4,5,6,7,8,9,10];
4. arr1.copyWithin(2) ;
5. document.write(arr1);
6. }
7. display();

57) What is the use of a Set object in JavaScript?

The JavaScript Set object is used to store the elements with unique values. The values can be of any type i.e. whether primitive values or object references. For example:

1. function display()
2. {
3. var set = new Set();
4. set.add("jQuery");
5. set.add("AngularJS");
6. set.add("Bootstrap");
7. for (let elements of set) {
8. document.writeln(elements+"**<br>**");
9. }
10. }
11. display();

58) What is the use of a WeakSet object in JavaScript?

The JavaScript WeakSet object is the type of collection that allows us to store weakly held objects. Unlike Set, the WeakSet are the collections of objects only. It doesn't contain the arbitrary values. For example:

1. function display()
2. {
3. var ws = new WeakSet();
4. var obj1={};
5. var obj2={};
6. ws.add(obj1);
7. ws.add(obj2);
8. //Let's check whether the WeakSet object contains the added object
9. document.writeln(ws.has(obj1)+"**<br>**");
10. document.writeln(ws.has(obj2));
11. }
12. display()

59) What is the use of a Map object in JavaScript?

The JavaScript Map object is used to map keys to values. It stores each element as key-value pair. It operates the elements such as search, update and delete on the basis of specified key. For example:

1. function display()
2. {
3. var map=new Map();
4. map.set(1,"jQuery");
5. map.set(2,"AngularJS");
6. map.set(3,"Bootstrap");
7. document.writeln(map.get(1)+"**<br>**");
8. document.writeln(map.get(2)+"**<br>**");
9. document.writeln(map.get(3));
10. }
11. display();

60) What is the use of a WeakMap object in JavaScript?

The JavaScript WeakMap object is a type of collection which is almost similar to Map. It stores each element as a key-value pair where keys are weakly referenced. Here, the keys are objects and the values are arbitrary values. For example:

1. function display()
2. {
3. var wm = new WeakMap();
4. var obj1 = {};
5. var obj2 = {};
6. var obj3= {};
7. wm.set(obj1, "jQuery");
8. wm.set(obj2, "AngularJS");
9. wm.set(obj3,"Bootstrap");
10. document.writeln(wm.has(obj2));
11. }
12. display();

**Q #1) What is JavaScript?**

**Answer:** JavaScript is a scripting language developed by Netscape. It can be used to program web browser or even servers. It can dynamically update the contents of the webpage, which is the beauty of JavaScript.

**Q #2) What are the advantages of using External JavaScript?**

**Answer:** Using External JavaScript in our code has many advantages as stated below.

* Separation of Code is done.
* Code Maintainability is Easy.
* Performance is better.

**Q #3) In the following Code snippet can you please predict the output or If you get an error, please explain the error?**

<!DOCTYPE html>

<html>

<body>

<h2> <strong> Sample: Software Testing Help</strong> </h2>

<p id="studentName"></p>

<script>

var studentName = "Sajeesh Sreeni"; // String 'Sajeesh Sreeni' stored in studentName

var studentName; // varaible is decalred again

document.getElementById("studentName").innerHTML =

"Redeclaring the varaible will not lose the value!.<br>"

+"Here the value in studentName is "+ studentName;

</script>

</body>

</html>

**Answer**: This code will not produce any errors.

Redeclaration of the variables is allowed in JavaScript. Hence, the value of the variable will not be lost after the execution of the statement here.

**Q #4) In the following Code snippet can you please predict the output or If you get an error; please explain the error?**

<!DOCTYPE html>

<html>

<body>

<h2> <strong> Sample: Software Testing Help</strong> </h2>

<p id="sum\_first"></p>

<p id="sum\_second"></p>

<script>

var sum\_first =50+20+' Sajeesh Sreeni ';

var sum\_second= " Sajeesh Sreeni "+50+20;

document.getElementById("sum\_first").innerHTML = "The first varaible sum is :"+sum\_first +

"<br>The second varaible sum is :"+sum\_second ;

</script>

</body>

</html>

**Answer:** This code will not show any error!

**The output of the code snippet here is:**

*The first variable sum is: 70 Sajeesh Sreeni*  
*The second variable sum is: Sajeesh Sreeni 5020*

**Q #5) What is the difference between test () and exec () methods?**

**Answer:**Both test () and exec () are RegExp expression methods.

Using test (), we will search a string for a given pattern, if it finds the matching text then it returns the Boolean value ‘true’ and else it returns ‘false’.

But in exec (), we will search a string for a given pattern, if it finds the matching text then it returns the pattern itself and else it returns ‘null’ value.

**Q #6) What are the Advantages of JavaScript?**

**Answer:**JavaScript Scripting language has many advantages as stated below.

* **Lightweight:** JavaScript is easy to implement. It has small memory footprints.
* **Interpreted:** It is an interpreted language. Instructions are executed directly.
* **Object-oriented:** JavaScript is an object-oriented language.
* **First class functions:** In JavaScript, a function can be used as a value.
* **Scripting Language:** It’s a language in which instructions are written for a run-time environment.

**Q #7) In the following Code snippet can you please predict the output or If you get an error; please explain the error?**

<!DOCTYPE html>

<html>

<body>

<h2> <strong> Sample: Software Testing Help</strong> </h2>

<p style='text-decoration:underline'>Example Const Variable </p>

<p id="display"></p>

<script>

const first\_num;

first\_num =1000;

document.getElementById("display").innerHTML = "First Number:"+ first\_num;

</script>

</body>

</html>

**Answer:**The ‘const’ variable ’first\_num’ is not initialized with a value, so the code will produce a Syntax Error.

**The output of the code snippet here is:**

*Error: Uncaught SyntaxError: Missing initializer in the const declaration*

**Q #8) Have you used any browser for debugging? If yes, how is it done?**

**Answer:**By, Pressing ‘F12’ key in the keyboard we can enable debugging in the browser. Chose the ‘Console’ tab to view the result.

In Console, we can set breakpoints and View the value in variables. All the modern browsers have a built-in debugger with them**(For Example: Chrome, Firefox, Opera, and Safari)**. This feature can be turned ON and OFF.

**Q #9) What is the use of ‘debugger’ keyword in JavaScript code?**

**Answer:** Using the ‘debugger’ keyword in the code is like using breakpoints in the debugger.

To test the code, the debugger must be enabled for the browser. If debugging is disabled for the browser, the code will not work. During debugging the code below should stop executing, before it goes to the next line.

**Q #10) What are the distinct types of Error Name Values?**

**Answer:** There are 6 types of values in ‘Error Name’ Property.

| **Error** | **Description** |
| --- | --- |
| Range Error | We will get this error if we use a number outside the range |
| Syntax Error | This error raises when we use the incorrect syntax. (Please refer Ques No: 7) |
| Reference Error | This error is thrown if used an undeclared variable Please refer Ques No: 19 |
| Eval Error | Thrown due to the error in eval(). New JavaScript version doesn’t have this error |
| Type Error | Value is outside the range of types used. Please refer Ques No :22 |
| URI Error | Due to the usage of illegal characters. |

**Q #11) What is JavaScript Hoisting?**

**Answer:** Using ‘JavaScript Hoisting’ method, when an interpreter runs the code, all the variables are hoisted to the top of the original /current scope. If you have a variable declared anywhere inside the JavaScript code then it is brought to the top.

This method is only applicable for the declaration of a variable and is not applicable for initialization of a variable. Functions are also hoisted to the top, whereas function explanations are not hoisted to the top.

Basically, where we declared the variable inside the code doesn’t matter much.

**Q #12) What is JavaScript ‘Strict Mode’?**

**Answer:**‘Strict mode’ is a restricted variant of JavaScript.

Usually, JavaScript is ‘not very strict’ in throwing errors.

But in ‘Strict mode’ it will throw all types of errors, even the silent errors. Thus, the process of debugging becomes easier. And the chances for making mistake for the developer is reduced.

**Q #13) What are the characteristics of JavaScript ‘Strict Mode’?**

**Answer: Given below are the characteristics of JavaScript ‘Strict Mode’:**

* ‘Strict Mode’ will stop developers from creating global variables.
* Developers are restricted from using duplicate parameters.
* Strict mode will restrict you from using JavaScript keyword as a variable name or function name.
* Strict mode is declared with ‘use strict’ keyword at the beginning of the script.
* All browser support strict mode.

**Q #14) What are Self Invoking Functions?**

**Answer:**They are also known as ‘Immediately Invoked Function Expressions’ or ‘Self Executing Anonymous Functions’. These functions are invoked automatically in the code, hence they are named as ‘Self Invoking Functions’.

Usually, we define a function and invoke it, but if we want to execute a function automatically where it is explained, and if we are not going to call it again, we can use anonymous functions. And these types of function have no name.

**Q #15) What is the syntax of ‘Self Invoking Function’? Give an example?**

**Answer:**

**The syntax for the Self-Invoking function:**

(function () {

return () } () ;

Here the last ‘()’ parenthesis in the syntax states that it is a function expression.

**Example of Self Invoked Functions:**

<!DOCTYPE html>

<html>

<body>

<h2> <strong> Sample: Software Testing Help</strong> </h2>

<p style='text-decoration:underline'>Example for Self-Invoking </p>

<p id="dispaly\_num"></p>

<script>

(function (){

elem = document.getElementById("dispaly\_num");

elem.innerHTML = "This function has no name.<br>It is called automatically";

}());

</script>

</body>

</html>

Here the anonymous function is automatically invoked in the code snippet.

The function is used to set the text property of <p> tag having ‘display\_num’ as Id.

**The output of the code snippet here is:**

*This function has no name.*  
*It is called automatically*

**Q #16) In the following Code snippet, can you please predict the output or If you get an error; please explain the error?**

**Answer:**

<!DOCTYPE html>

<html>

<body>

<h2> <strong> Sample : Software Testing Help</strong> </h2>

<p style='text-decoration:underline'>Example for JavaScript Hoisting </p>

<p id="dispaly\_num"></p>

<script>

first\_num = 100; // Assign value 100 to num

elem = document.getElementById("dispaly\_num");

elem.innerHTML = " Here the variable first\_num:<u> "+first\_num +"</u> is taken to the top <br>" +

"Since second variable is initialised the value is not taken to the top and it's value is "

+ "<u>"+second\_num +"</u> “;

var first\_num; // declaration only

var second\_num =200; // Initialised the variable

</script>

</body>

</html>

Please refer previous Ques no: 11, as explained there, the interpreter will take all the variables declared except initialization to the top.

As per this, ‘first\_num’ variable is taken to the top and ‘second\_num’ variable is initialized with a value, so it is not taken to the top. The is code will not throw an error. But the value of ‘second\_num’ is undefined.

**The output of the code snippet here is:**

*Here the variable first\_num: 100 is taken to the top*  
*Since the second variable is initialized the value is not taken to the top and its value is undefined*

**Q #17) If you need to hide the JavaScript code from the older browser versions, how will you perform it?**

**Answer:**In JavaScript Code, after the <script> tag add ‘<! –’ HTML tag.

This will make the browser to not execute the JavaScript code if it was being an older version of it. Also, after the end </script> tag add ‘//–>’ HTML tag.

This method will help in solving compatibility issues and UI issues to an extent.

<!DOCTYPE html>

<html>

<body>

<h2> <strong> Sample: Software Testing Help</strong> </h2>

<p id="display"></p>

<script> <!--

document.getElementById("display").innerHTML = "Here I am not using an older version of browser.<br>

So the code will work in my browser";

//-->

</script>

</body>

</html>

Here the code snippet after a <script> tag is executed in my browser as I am not using an older version of the browser.

**The output of the code snippet here is:**

*Here I am not using an older version of the browser.*  
*So the code will work in my browser*

**Q #18) In the following Code snippet can you please predict the output or If you get an error, please explain the error?**

<!DOCTYPE html>

<html>

<body>

<h2> <strong> Sample: Software Testing Help</strong> </h2>

<p style='text-decoration:underline'>Find the output </p>

<p id="display"></p>

<script>

var first\_num =500;

var result= function(){

document.getElementById("display").innerHTML = first\_num;

var first\_num =1000;

}

result();

</script>

</body>

</html>

**Answer:** Here in the code given above, the value of the ‘first\_num’ variable will not be 1000.

In JavaScript, there is no hoisting for variable initialization. The function ‘result ()’ will choose the local variable ‘first\_num’, as it is declared inside the function. Since the variable is declared after it is used, the value of ‘first\_num’ is undefined.

**The output of the code snippet here is:**

*Undefined*

**Q #19) What is the difference between ‘var’ and ‘let’ keyword?**

**Answer: The Differences are as follows:**

| **Var** | **let** |
| --- | --- |
| ’var’ keyword was introduced in JavaScript code from the beginning Stage itself. | ‘let’ keyword is introduced in 2015 only. |
| ’Var’ keyword has function scope. The variable defined with var is available anywhere within the function | A variable declared with ‘let’ keyword has a scope only with in that block. So, let has a Block Scope. |
| The variable declared with ‘var’ be hoisted | The variable declared with ‘let’ be hoisted |

**Q #20) In the following Code snippet can you please predict the output or If you get an error; please explain the error?**

<!DOCTYPE html>

<html>

<body>

<h2> <strong> Sample: Software Testing Help</strong> </h2>

<p style='text-decoration:underline'>Find the output </p>

<p id="display\_first"></p>

<p id="display\_second"></p>

<script>

if(true){

var first\_num =1000;

let second\_num=500;

}

document.getElementById("display\_first").innerHTML = "First Number:" + first\_num;

document.getElementById("display\_second").innerHTML = "Second Number:" + second\_num;

</script>

</body>

</html>

**Answer:**

**The output of the code snippet here is:**

*First*Number :*1000*

We will get *‘First*Number :*1000’* as output. There is an ‘Uncaught Reference Error’ error also. In the code snippet, the scope of ‘second\_num’ is only with in the if() block. If a developer tries to access the value outside the block, he will get a ‘Uncaught Reference error’.  
Uncaught Reference Error: second\_num is not defined.

**Q #21) What is the difference between ‘==’ and ‘===’?**

**Answer:**

Both ‘==’ and ‘===’ are comparison operators.

| **‘==’ operator** | **‘===’ operator** |
| --- | --- |
| It is known as ‘Type Converting Operator’ | It is known as ‘Strict Equality Operator’ |
| It compares Value, do not compare type | It compares both value and type. |

**Q #22) What is the difference between ‘let’ and ‘const’?**

**Answer :**

| **let** | **const** |
| --- | --- |
| using ‘let’ we can change the value of variable any number of times | using ‘const’, after the first assignment of the value we cannot redefine the value again |
| Consider the code  { let first\_num =1;  first\_num=2; document. write (first\_num); } Here the code will give an output, since the change in value of first\_num is possible. | Consider the code  { const second\_num =1;  second\_num=2; document. write (second\_num); } Here the code will produce an error, since the ‘second\_num’ is assigned with a second value. |

**Q #23) In the following Code snippet can you please predict the output or If you get an error; please explain the error?**

<!DOCTYPE html>

<html>

<body>

<h2> <strong> Sample: Software Testing Help</strong> </h2>

<p style='text-decoration:underline'>Example of 'Const' Keyword </p>

<p id="display\_first"></p>

<p id="display\_second"></p>

<script>

let first\_num =500;

first\_num=501;

document.getElementById("display\_first").innerHTML = "First Number:"+ first\_num ;

const second\_num =1000;

second\_num=1001;

document.getElementById("display\_second").innerHTML = "Second Number :"+second\_num;

</script>

</body>

</html>

**Answer: Please refer Ques No: 21 before reading further**

**The output of the code snippet is:**

*First Number:501*

We will also get an Error while running the code, as we are trying to change the value of a ‘const’ variable.

**Error:** *Uncaught TypeError: Assignment to constant variable.*

**Q #24) What is the difference between ‘null’ and ‘undefined’?**

**Answer:**Both the keywords represent empty values**.**

**The differences are:**

* In ‘undefined’, we will define a variable, but we won’t assign a value to that variable. On the other hand, in ‘null’ we will define a variable and assign the ‘null’ value to the variable.
* type of (undefined) and type of (null) object.

**Q #25) What is the difference between ‘function declaration’ and ‘function expression’?**

**Answer:**

**It can be explained with an Example:**

<!DOCTYPE html>

<html>

<body>

<h2> <strong> Sample: Software Testing Help</strong> </h2>

<p style='text-decoration:underline'>Example Function Declaration</p>

<p id="display\_add"></p>

<p id="display\_sub"></p>

<script>

function add(first\_num,second\_num){

return first\_num + second\_num;

}

var substract = function sub(first\_num,second\_num){

return first\_num - second\_num;

}

var first\_num=700;

var second\_num=300;

document.getElementById("display\_add").innerHTML = "Sum of the number is:" + add(first\_num,second\_num);

document.getElementById("display\_sub").innerHTML = "Difference of the number is:" + substract(first\_num,second\_num);

</script>

</body>

</html>

As shown in the example add() is a function declaration and subtract() is a function expression. The syntax of function declaration is like a function which is saved into a variable.

Function declarations are hoisted but function expressions are not hoisted.

**Q #26) What are ‘settimeout()’?**

**Answer:**(It better to explain this answer with an example)

Consider the code snippet

Console.log (‘First Line’);

Console.log (‘Second Line’);

Console.log (‘Third Line’);

**The output of the code snippet here is:**

*First Line*  
*Second Line*  
*Third Line*

Now you introduce settimeout() method and wrap the same set of code in it.

Settimeout(function() {

Console.log (‘First Line’);

},0);

Console.log (‘Second Line’);

Console.log (‘Third Line’);

**The output of the code snippet here is:**

*Second Line*  
*Third Line*  
*First Line*

With the introduction of settimeout(), the processes become asynchronous. The first statements to be placed in the stack is Console.log (‘Second Line’), and Console.log (‘Third Line’), and they will get executed first. You need to wait until everything in the stack is completed first.

Even though ‘0’ is the timeout period, it doesn’t mean that it will be executed right away.

**Q #27) What is a Closure and How do you use it?**

**Answer:**A closure is an inner function. It can access the outer variables of a function. In Closure, within function\_1 there is another function\_2 which returns ‘A’ value and function\_1 also returns a value; say ‘B’.

Here sum() is the outer function and add () is an inner function, it can access all the variables including ‘first\_num’ ‘second\_num’ and ‘third\_num’. The outer function is calling the inner function add().

<script>

// To find the sum of two numbers using closure method

function sum( first\_num, second\_num )

{

var sumStr= 600;

function add(first\_num , second\_num)

{

return (sumStr + (first\_num + second\_num));

}

return add();

}

document.write("Result is :"+ sum(150,350));

</script>

**The output of the code snippet here is:**

*Result is: 500*

**Q #28) In the following Code snippet can you please predict the output or If you get an error; please explain the error?**

<!DOCTYPE html>

<html>

<body>

<h2> <strong> Sample: Software Testing Help</strong> </h2>

<p style='text-decoration:underline'>Example Assignmnet Statement</p>

<p id="display"></p>

<script>

var x =500;

let y,z,p,q;

q=200;

if(true){

x=y=z=p=q;

document.getElementById("display").innerHTML = "x="+ x + "<br>y :"+ y +"<br>z :"+ z+"<br>p :"+ p+"<br>q :"+ q;

}

</script>

</body>

</html>

**Answer:** The assignment statements are considered from Right to left.

**The output of the code snippet here is:**

*x=200*  
*y:200*  
*z:200*  
*p:200*  
*q:200*

**Q #29) Can you give an example where the code snippet shows the difference between test () and exec () methods?**

<!DOCTYPE html>

<html>

<body>

<h2> <strong> Sample : Software Testing Help</strong> </h2>

<p style='text-decoration:underline'>Example for exec() methods </p>

<p>Click the button to search for a pattern "How“ in the given string "Hello. Good Morning. How do you feel today?"</p>

<p>If the "How" is found, the method will return the pattern </p>

<button onclick="searchTxt()">Search</button>

<p id="result"></p>

<script>

function searchTxt() {

var str = "Hello. Good Morning. How do you feel today?";

var search\_patt = new RegExp("How");

var res = search\_patt.exec(str);

document.getElementById("result").innerHTML ="Found the pattern :"+ res;

}

</script>

</body>

</html>

**Answer:**This is an example of the test () and exec () method, Refer Ques No: 5 for more details.

**The output of the code snippet here is:**

*Found the pattern using exec (): How*  
*Using test () the result is: true*

**Q #30) Can you give an example showing JavaScript Hoisting?**

**Answer:**

<!DOCTYPE html>

<html>

<body>

<h2> <strong> Sample: Software Testing Help</strong> </h2>

<p style='text-decoration:underline'>Example for JavaScript Hoisting </p>

<p id="dispaly\_num"></p>

<script>

num = 100; // Assign value 100 to num

elem = document.getElementById("dispaly\_num");

elem.innerHTML = "Here the variables are used before declaring it." +

" <br>The value of the variable is " + num;

var num; // Declare the varaible </script>

</body>

</html>

Please refer Ques No:11 for more details

Here the variable ‘num’ is used before declaring it. But JavaScript Hoisting will allow it.

**The output of the code snippet here is:**

*Here the variables are used before declaring it.*  
*The value of the variable is 100*

**Q #31) Can you give an example showing the use of ‘debugger’ keyword in the JavaScript code?**

**Answer:**

<!DOCTYPE html>

<html>

<head>

</head>

<body>

<h2> <strong> Sample: Software Testing Help</strong> </h2>

<p style='text-decoration:underline'> Example for debug keyword </p>

<p> Here to test the code, debugger must be enabled for the browser, <br>during debugging the code below should stop executing before it goes to the next line. </p>

<p id="wait\_result"></p>

<p id="show\_result"></p>

<script>

var a = 1000;

var b = 500;

var sum = a + b;

document.getElementById("wait\_result").innerHTML = "Adding numbers......<br>Select 'Resume Script execution' to continue: ";

debugger;

document.getElementById("show\_result").innerHTML = "Sum of the numbers : "+sum;

</script>

</body>

</html>

**Note:** The debugger must be enabled for the browser to test the code. Refer Ques No: 5 for more details

This is an example of debugging keyword (Browser used: Chrome)

**The output of the code snippet here is:**

*Here to test the code, the debugger must be enabled for the browser,*  
*during debugging the code below should stop executing before it goes to the next line.*  
*Adding numbers…*  
*Select ‘Resume Script execution' to continue:*

*<Click on ‘Resume Script execution’ Button>*

*Sum of the numbers: 1500*

**Q #32) In the following Code snippet can you please predict the output or If you get an error; please explain the error?**

<!DOCTYPE html>

<html>

<body>

<h2> <strong> Sample: Software Testing Help</strong> </h2>

<p style='text-decoration:underline'>Example Type Converting </p>

<p id="display"></p>

<script>

var first\_num =500;

var first\_name='500';

if(first\_num == first\_name){

document.getElementById("display").innerHTML = "Comparison will return 'true' by Type converting Operator ";

}

</script>

</body>

</html>

**Answer:**Consider the code

If (‘100’==100) {

document. write (“It’s a Type Converting Operator”);

}

Here

  typeof(‘100’) is string

   typeof(100) is number

the ‘==’ operator will convert the number type, which is on the right side of the operator to string and compare both values

**The output of the code snippet here is:**

*Comparison will return ‘true' by Type converting Operator*

**Question 1**

**1. What is the difference between undefined and not defined in JavaScript?**

In JavaScript, if you try to use a variable that doesn't exist and has not been declared, then JavaScript will throw an error var name is not defined and script will stop executing. However, if you use typeof undeclared\_variable, then it will return undefined.

Before getting further into this, let's first understand the difference between declaration and definition.

Let's say var x is a declaration because you have not defined what value it holds yet, but you have declared its existence and the need for memory allocation.

> **var** x; *// declaring x*

> console.log(x); *//output: undefined*

Here var x = 1 is both a declaration and definition (also we can say we are doing an initialisation). In the example above, the declaration and assignment of value happen inline for variable x. In JavaScript, every variable or function declaration you bring to the top of its current scope is called hoisting.

The assignment happens in order, so when we try to access a variable that is declared but not defined yet, we will get the result undefined.

**var** x; *// Declaration*

**if**(**typeof** x === 'undefined') *// Will return true*

If a variable that is neither declared nor defined, when we try to reference such a variable we'd get the result not defined.

> console.log(y); *// Output: ReferenceError: y is not defined*

**Question 2**

**What will be the output of the code below?**

**var** y = 1;

**if** (**function** **f**(){}) {

y += **typeof** f;

}

console.log(y);

The output would be 1undefined. The if condition statement evaluates using eval, so eval(function f(){}) returns function f(){} (which is true). Therefore, inside the if statement, executing typeof f returns undefined because the if statement code executes at run time, and the statement inside the if condition is evaluated during run time.

**var** k = 1;

**if** (1) {

eval(**function** **foo**(){});

k += **typeof** foo;

}

console.log(k);

The code above will also output 1undefined.

**var** k = 1;

**if** (1) {

**function** **foo**(){};

k += **typeof** foo;

}

console.log(k); *// output 1function*

**Question 3**

**What is the drawback of creating true private methods in JavaScript?**

One of the drawbacks of creating true private methods in JavaScript is that they are very memory-inefficient, as a new copy of the method would be created for each instance.

**var** Employee = **function** (name, company, salary) {

**this**.name = name || ""; *//Public attribute default value is null*

**this**.company = company || ""; *//Public attribute default value is null*

**this**.salary = salary || 5000; *//Public attribute default value is null*

*// Private method*

**var** increaseSalary = **function** () {

**this**.salary = **this**.salary + 1000;

};

*// Public method*

**this**.dispalyIncreasedSalary = **function**() {

increaseSlary();

console.log(**this**.salary);

};

};

*// Create Employee class object*

**var** emp1 = **new** Employee("John","Pluto",3000);

*// Create Employee class object*

**var** emp2 = **new** Employee("Merry","Pluto",2000);

*// Create Employee class object*

**var** emp3 = **new** Employee("Ren","Pluto",2500);

Here each instance variable emp1, emp2, emp3 has its own copy of the increaseSalary private method.

So, as a recommendation, don’t use private methods unless it’s necessary.

**Question 4**

**What is a “closure” in JavaScript? Provide an example**

A closure is a function defined inside another function (called the parent function), and has access to variables that are declared and defined in the parent function scope.

The closure has access to variables in three scopes:

* Variables declared in their own scope
* Variables declared in a parent function scope
* Variables declared in the global namespace

**var** globalVar = "abc";

*// Parent self invoking function*

(**function** **outerFunction** (outerArg) { *// begin of scope outerFunction*

*// Variable declared in outerFunction function scope*

**var** outerFuncVar = 'x';

*// Closure self-invoking function*

(**function** **innerFunction** (innerArg) { *// begin of scope innerFunction*

*// variable declared in innerFunction function scope*

**var** innerFuncVar = "y";

console.log(

"outerArg = " + outerArg + "\n" +

"outerFuncVar = " + outerFuncVar + "\n" +

"innerArg = " + innerArg + "\n" +

"innerFuncVar = " + innerFuncVar + "\n" +

"globalVar = " + globalVar);

}*// end of scope innerFunction)(5); // Pass 5 as parameter*

}*// end of scope outerFunction )(7); // Pass 7 as parameter*

innerFunction is closure that is defined inside outerFunction and has access to all variables declared and defined in the outerFunction scope. In addition, the function defined inside another function as a closure will have access to variables declared in the global namespace.

Thus, the output of the code above would be:

outerArg = 7

outerFuncVar = x

innerArg = 5

innerFuncVar = y

globalVar = abc

**Question 5**

**Write a mul function which will produce the following outputs when invoked:**

console.log(mul(2)(3)(4)); *// output : 24*

console.log(mul(4)(3)(4)); *// output : 48*

Below is the answer followed by an explanation to how it works:

**function** **mul** (x) {

**return** **function** (y) { *// anonymous function*

**return** **function** (z) { *// anonymous function*

**return** x \* y \* z;

};

};

}

Here the mul function accepts the first argument and returns an anonymous function, which takes the second parameter and returns another anonymous function that will take the third parameter and return the multiplication of the arguments that have been passed.

In JavaScript, a function defined inside another one has access to the outer function's variables. Therefore, a function is a first-class object that can be returned by other functions as well and be passed as an argument in another function.

* A function is an instance of the Object type
* A function can have properties and has a link back to its constructor method
* A function can be stored as a variable
* A function can be pass as a parameter to another function
* A function can be returned from another function

**Question 6**

**How to empty an array in JavaScript?**

For instance,

**var** arrayList = ['a','b','c','d','e','f'];

**How can we empty the array above?**

There are a couple ways we can use to empty an array, so let's discuss them all.

**Method 1**

arrayList = []

Above code will set the variable arrayList to a new empty array. This is recommended if you don't have **references to the original array** arrayListanywhere else, because it will actually create a new, empty array. You should be careful with this method of emptying the array, because if you have referenced this array from another variable, then the original reference array will remain unchanged.

For Instance,

**var** arrayList = ['a','b','c','d','e','f']; *// Created array*

**var** anotherArrayList = arrayList; *// Referenced arrayList by another variable*

arrayList = []; *// Empty the array*

console.log(anotherArrayList); *// Output ['a','b','c','d','e','f']*

**Method 2**

arrayList.length = 0;

The code above will clear the existing array by setting its length to 0. This way of emptying the array also updates all the reference variables that point to the original array. Therefore, this method is useful when you want to update all reference variables pointing to arrayList.

For Instance,

**var** arrayList = ['a','b','c','d','e','f']; *// Created array*

**var** anotherArrayList = arrayList; *// Referenced arrayList by another variable*

arrayList.length = 0; *// Empty the array by setting length to 0*

console.log(anotherArrayList); *// Output []*

**Method 3**

arrayList.splice(0, arrayList.length);

The implementation above will also work perfectly. This way of emptying the array will also update all the references to the original array.

**var** arrayList = ['a','b','c','d','e','f']; *// Created array*

**var** anotherArrayList = arrayList; *// Referenced arrayList by another variable*

arrayList.splice(0, arrayList.length); *// Empty the array by setting length to 0*

console.log(anotherArrayList); *// Output []*

**Method 4**

**while**(arrayList.length){

arrayList.pop();

}

The implementation above can also empty arrays, but it is usually not recommended to use this method often.

**Question 7**

**How do you check if an object is an array or not?**

The best way to find out whether or not an object is an instance of a particular class is to use the toString method from Object.prototype:

**var** arrayList = [1,2,3];

One of the best use cases of type-checking an object is when we do method overloading in JavaScript. For example, let's say we have a method called greet, which takes one single string and also a list of strings. To make our greet method workable in both situations, we need to know what kind of parameter is being passed. Is it a single value or a list of values?

**function** **greet**(param){

**if**(){ *// here have to check whether param is array or not*

}**else**{

}

}

However, as the implementation above might not necessarily check the type for arrays, we can check for a single value string and put some array logic code in the else block. For example:

**function** **greet**(param){

**if**(**typeof** param === 'string'){

}**else**{

*// If param is of type array then this block of code would execute*

}

}

Now it's fine we can go with either of the aforementioned two implementations, but when we have a situation where the parameter can be single value, array, and object type, we will be in trouble.

Coming back to checking the type of an object, as mentioned previously we can use  
Object.prototype.toString

**if**( Object.prototype.toString.call( arrayList ) === '[object Array]' ) {

console.log('Array!');

}

If you are using jQuery, then you can also use the jQuery isArray method:

**if**($.isArray(arrayList)){

console.log('Array');

}**else**{

console.log('Not an array');

}

FYI, jQuery uses Object.prototype.toString.call internally to check whether an object is an array or not.

In modern browsers, you can also use

Array.isArray(arrayList);

Array.isArray is supported by Chrome 5, Firefox 4.0, IE 9, Opera 10.5 and Safari 5

**Question 8**

**What will be the output of the following code?**

**var** output = (**function**(x){

**delete** x;

**return** x;

})(0);

console.log(output);

The output would be 0. The delete operator is used to delete properties from an object. Here x is not an object but a **local variable**. delete operators don't affect local variables.

**Question 9**

**What will be the output of the following code?**

**var** x = 1;

**var** output = (**function**(){

**delete** x;

**return** x;

})();

console.log(output);

The output would be 1. The delete operator is used to delete the property of an object. Here x is not an object, but rather it's the **global variable** of type number.

**Question 10**

**What will be the output of the code below?**

**var** x = { foo : 1};

**var** output = (**function**(){

**delete** x.foo;

**return** x.foo;

})();

console.log(output);

The output would be undefined. The delete operator is used to delete the property of an object. Here, x is an object which has the property foo, and as it is a self-invoking function, we will delete the foo property from object x. After doing so, when we try to reference a deleted property foo, the result isundefined.

**Question 11**

**What will be the output of the code below?**

**var** Employee = {

company: 'xyz'

}

**var** emp1 = Object.create(Employee);

**delete** emp1.company

console.log(emp1.company);

The output would be xyz. Here, emp1 object has company as its **prototype** property. The delete operator doesn't delete prototype property.

emp1 object doesn't have **company** as its own property. You can test it console.log(emp1.hasOwnProperty('company')); //output : false. However, we can delete the company property directly from theEmployee object using delete Employee.company. Or, we can also delete the emp1 object using the \_\_proto\_\_property delete emp1.\_\_proto\_\_.company.

**Question 12**

**What is undefined x 1 in JavaScript?**

**var** trees = ["redwood","bay","cedar","oak","maple"];

**delete** trees[3];

When you run the code above and type console.log(trees); into your Chrome developer console, you will get  
["redwood", "bay", "cedar", undefined × 1, "maple"]. When you run the code in Firefox's browser console, you will get ["redwood", "bay", "cedar", undefined, "maple"]. Thus, it's clear that the Chrome browser has its own way of displaying uninitialised indexes in arrays. However, when you check trees[3] === undefined in both browsers, you will get similar output as true.

**Note:** Please remember you do not need to check for the uninitialised index of array in trees[3] === 'undefined × 1', as it will give you an error. 'undefined × 1' is just way of displaying an array's uninitialised index in Chrome.

**Question 13**

**What will be the output of the code below?**

**var** trees = ["xyz","xxxx","test","ryan","apple"];

**delete** trees[3];

console.log(trees.length);

The output would be 5. When we use the delete operator to delete an array element, the array length is not affected from this. This holds even if you deleted all elements of an array using the delete operator.

In other words, when the delete operator removes an array element, that deleted element is not longer present in array. In place of value at deleted index undefined x 1 in **chrome** and undefined is placed at the index. If you do console.log(trees)output ["xyz", "xxxx", "test", undefined × 1, "apple"] in Chrome and in Firefox ["xyz", "xxxx", "test", undefined, "apple"].

**Question 14**

**What will be the output of the code below?**

**var** bar = true;

console.log(bar + 0);

console.log(bar + "xyz");

console.log(bar + true);

console.log(bar + false);

The code will output 1, "truexyz", 2, 1. Here's a general guideline for addition operators:

* Number + Number -> Addition
* Boolean + Number -> Addition
* Boolean + Number -> Addition
* Number + String -> Concatenation
* String + Boolean -> Concatenation
* String + String -> Concatenation

**Question 15**

**What will be the output of the code below?**

**var** z = 1, y = z = **typeof** y;

console.log(y);

The output would be undefined. According to the associativity rule, operators with the same precedence are processed based on the associativity property of the operator. Here, the associativity of the assignment operator is Right to Left, so typeof y will evaluate first , which is undefined. It will be assigned to z, and then ywould be assigned the value of z and then z would be assigned the value 1.

**Question 16**

**What will be the output of the code below?**

*// NFE (Named Function Expression*

**var** foo = **function** **bar**(){ **return** 12; };

**typeof** bar();

The output would be Reference Error. To make the code above work, you can re-write it as follows:

**Sample 1**

**var** bar = **function**(){ **return** 12; };

**typeof** bar();

or

**Sample 2**

**function** **bar**(){ **return** 12; };

**typeof** bar();

A function definition can have only one reference variable as its function name. In **sample 1**, bar's reference variable points to anonymous function. In **sample 2**, the function's definition is the name function.

**var** foo = **function** **bar**(){

*// foo is visible here*

*// bar is visible here*

console.log(**typeof** bar()); *// Work here :)*

};

*// foo is visible here*

*// bar is undefined here*

**Question 17**

**What is the difference between the function declarations below?**

**var** foo = **function**(){

*// Some code*

};

**function** **bar**(){

*// Some code*

};

The main difference is the function foo is defined at run-time whereas function bar is defined at parse time. To understand this in better way, let's take a look at the code below:

Run-Time **function** **declaration**

<**script**>

**foo**(); // **Calling** **foo** **function** **here** **will** **give** **an** **Error**

**var** **foo** = **function**(){

console.log("Hi I am inside Foo");

};

</script>

<script>

Parse-Time **function** **declaration**

**bar**(); // **Calling** **foo** **function** **will** **not** **give** **an** **Error**

**function** **bar**(){

console.log("Hi I am inside Foo");

};

</script>

Another advantage of this first-one way of declaration is that you can declare functions based on certain conditions. For example:

<script>

**if**(testCondition) {*// If testCondition is true then*

**var** foo = **function**(){

console.log("inside Foo with testCondition True value");

};

}**else**{

**var** foo = **function**(){

console.log("inside Foo with testCondition false value");

};

}

</script>

However, if you try to run similar code using the format below, you'd get an error:

<script>

**if**(testCondition) {*// If testCondition is true then*

**function** **foo**(){

console.log("inside Foo with testCondition True value");

};

}**else**{

**function** **foo**(){

console.log("inside Foo with testCondition false value");

};

}

</script>

**Question 18**

**What is function hoisting in JavaScript?**

**Function Expression**

**var** foo = **function** **foo**(){

**return** 12;

};

In JavaScript, variable and functions are hoisted. Let's take function hoisting first. Basically, the JavaScript interpreter looks ahead to find all variable declarations and then hoists them to the top of the function where they're declared. For example:

foo(); *// Here foo is still undefined*

**var** foo = **function** **foo**(){

**return** 12;

};

Behind the scene of the code above looks like this:

**var** foo = undefined;

foo(); *// Here foo is undefined*

foo = **function** **foo**(){

/ Some code stuff

}

**var** foo = undefined;

foo = **function** **foo**(){

/ Some code stuff

}

foo(); *// Now foo is defined here*

**Question 19**

**What will be the output of code below?**

**var** salary = "1000$";

(**function** () {

console.log("Original salary was " + salary);

**var** salary = "5000$";

console.log("My New Salary " + salary);

})();

The output would be undefined, 5000$. Newbies often get tricked by JavaScript's hoisting concept. In the code above, you might be expecting salary to retain its value from the outer scope until the point that salary gets re-declared in the inner scope. However, due to hoisting, the salary value was undefined instead. To understand this better, have a look of the code below:

**var** salary = "1000$";

(**function** () {

**var** salary = undefined;

console.log("Original salary was " + salary);

salary = "5000$";

console.log("My New Salary " + salary);

})();

salary variable is hoisted and declared at the top in the function's scope. The console.log inside returns undefined. After the console.log, salary is redeclared and assigned 5000$.

**Question 20**

**What is the instanceof operator in JavaScript? What would be the output of the code below?**

**function** **foo**(){

**return** foo;

}

**new** foo() **instanceof** foo;

Here, instanceof operator checks the current object and returns true if the object is of the specified type.

For Example:

**var** dog = **new** Animal();

dog **instanceof** Animal *// Output : true*

Here dog instanceof Animal is true since dog inherits from Animal.prototype.

**var** name = **new** String("xyz");

name **instanceof** String *// Output : true*

Here name instanceof String is true since dog inherits from String.prototype. Now let's understand the code below:

**function** **foo**(){

**return** foo;

}

**new** foo() **instanceof** foo;

Here function foo is returning foo, which again points to function foo.

**function** **foo**(){

**return** foo;

}

**var** bar = **new** foo();

*// here bar is pointer to function foo(){return foo}.*

So the new foo() instanceof foo return false;

[Ref Link](http://stackoverflow.com/questions/2449254/what-is-the-instanceof-operator-in-javascript)

**Question 21**

**If we have a JavaScript associative array**

**var** counterArray = {

A : 3,

B : 4

};

counterArray["C"] = 1;

**How can we calculate the length of the above associative array's counterArray?**

There are no in-built functions and properties available to calculate the length of associative array object here. However, there are other ways by which we can calculate the length of an associative array object. In addition to this, we can also extend an Object by adding a method or property to the prototype in order to calculate length. However, extending an object might break enumeration in various libraries or might create cross-browser issues, so it's not recommended unless it's necessary. Again, there are various ways by which we can calculate length.

Object has the keys method which can be used to calculate the length of an object:

**We** can also calculate the length **of** an **object** by iterating through an **object** **and** by counting the **object**'s own property.

```javascript

function getSize(**object**){

**var** count = 0;

**for**(key **in** **object**){

// hasOwnProperty **method** check own property **of** **object**

**if**(**object**.hasOwnProperty(key)) count++;

}

**return** count;

}

We can also add a length method directly on Object:

Object.length = **function**(){

**var** count = 0;

**for**(key **in** object){

*// hasOwnProperty method check own property of object*

**if**(object.hasOwnProperty(key)) count++;

}

**return** count;

}

*//Get the size of any object using*

console.log(Object.length(counterArray))

**Bonus**: We can also use Underscore (recommended, As it's lightweight) to calculate object length.

[1. What is the use of let & const in JavaScript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled1)

In modern javascript let & const are different ways of creating variables. Earlier in javascript, we use the var keyword for creating variables. let & const keyword is introduced in version [ES6](https://www.onlineinterviewquestions.com/es6-interview-questions/)with the vision of creating two different types of variables in javascript one is immutable and other is mutable.  
**const:** It is used to create an immutable variable. Immutable variables are variables whose value is never changed in the complete life cycle of the program.  
**let:** let is used to create a mutable variable. Mutable variables are normal variables like var that can be changed any number of time.

[2. What are the different types of errors available in JavaScript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled2)

There are three types of errors available in JavaScript

* **Load time errors**: Errors which come up when loading a web page like improper syntax errors are known as Load-time errors and it generates the errors dynamically.
* **Run time errors**: Errors that come due to misuse of the command inside the HTML language.
* **Logical Errors**: These are the errors that occur due to the bad logic performed on a function which is having a different operation.

[3. How to get an element by class in JavaScript ?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled3)

**document.getElementsByClassName()** method is used in Javascript to get an element with a class name.

|  |  |
| --- | --- |
| getElementsByClassName() | |
| **Method Name** | getElementsByClassName |
| **Syntax** | document.getElementsByClassName('className') |
| **Parameter** | String (name of class) |
| **Output** | Array of HTMLCollection that have inputted className |

[4. What are different types of Scope Chain available in JavaScript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled4)

If we check in the program, every local scope has a connection with one or more scope in their back which forms a chain. This chain goes on until it met with the global scope which is the root of this hierarchy. As global scope doesn't have a parent, so it is on the top of the chain. This is known as scope chain.

The scope chain in JavaScript is basically used to resolve the values of the variable. Without this, it is difficult for a JavaScript to choose a certain value for a variable if there are many variables defined at different scopes.

[5. Explain MUL function in Javascript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled5)

MUL means simple multiplication of numbers. It is a techique in which you pass a one value as an argument in a function and that function returns another function to which you pass the second value and the process go on. For example: x\*y\*z can be representing as:

function mul (x) {

return function (y) { // anonymous function

return function (z) { // anonymous function

return x \* y \* z; };

};

}

[6. Write a program to reverse a string in pure JavaScript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled6)

There are many ways to reverse a string in JavaScript. These are:

**Using in-built functions:** the inbuilt function reverse() reverses the string directly. Here’ how:

str="jQuery";

str = str.split(""); //convert 'jQuery' to array

str = str.reverse(); //reverse 'jQuery' order

str = str.join(""); //then combines the reverse order values.

alert(str);

First split the string to an array, then reverse an array and after that join the characters to form a string.

**Using a loop:** First, count a number of characters in a string, then apply a decrementing loop on an original string which starts from the last character and prints each character until count becomes zero.

[7. How to redirect a page to another page in Javascript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled7)

There are several ways to redirect page to another page in JavaScript. These are:

1. **Using location.href:** It is the first approach to redirect page. In this, we can go back to access the original document.**Syntax**:window.location.href =“https://www.onlineinterviewquestions.com/”
2. **Using location.replace:** Another approach to redirect page. In this, it is not possible to navigate back to the original document by clicking on the back button as it removes the URL of the original document. **Syntax:** window.location.replace(" https://www.onlineinterviewquestions.com/;");

[8. List some Design patterns in JavaScript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled8)

The design pattern is a general reusable solution to a commonly occurring problem in software design. Some of the design patterns are:

1. **Creational design pattern:** These patterns dealt with the mechanism of object creation which optimize object creation with the basic approach.
2. **Structural design pattern:** these patterns deal with different classes and objects to provide new functionality.
3. **Behavioral Patterns:** These patterns are to improve communication between objects and to recognize patterns.
4. **Concurrency design patterns:** These patterns handle with multi-thread programming paradigms.
5. **Architectural design patterns:** These patterns used to deal with architectural designs.

[9. What is difference between Array.splice() and Array.slice() method in JavaScript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled9)

* The array.slice() removes items from the array and then return those removed items as an array whereas array.slice() method is selected items from an array and then those elements as a new array object.
* The splice() method affects the original array whereas slice() method doesn’t affect the original array.
* Splice() method takes n number of arguments whereas slice() can take only two arguments.

Syntax of splice(): array.splice(index, howmany, item1, ....., itemX)

Syntax of slice(): array.slice(start, end)

[10. How to add/remove properties to object dynamically in Javascript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled10)

You can add a property to an object using object.property\_name =value, delete object.property\_name is used to delete a property.

**Example:**

let user = new Object();

// adding a property

user.name='Anil';

user.age =25;

console.log(user);

delete user.age;

console.log(user);

[11. Explain Promise in JavaScript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled11)

A promise is an object in JavaScript which is used to produce a value that may give result in the future. The value can be resolved value or it can be a reason which tells why the value is not resolved.

A promise can be of three states:

* Fulfilled: The operation is completed and the promise has a specific value.
* Rejected: The operation is failed and promise has a reason which shows why the operation failed.
* Pending: Th operation is not fulfilled or rejected, means it has not completed yet.

[12. How to remove duplicates from JavaScript Array?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled12)

There are many ways to remove duplicates from JavaScript array. These are described below with examples:

**1. By using Set**: It is the simplest approach to remove duplicates. Set is an inbuilt object to store unique values in an array. Here's how we use set:

function uniquearray(array) {

let unique\_array= Array.from(set(array))

return unique\_array;}

As in the above code, you created a set of an array which automatically eliminates the duplicate values.

**2. By using Filter**: Another approach to remove duplicates from an array is applying filter on an array. To call filter method, it requires three arguments: array, current element, index of current element. Here’s how we use filter:

function unque\_array (arr){

let unique\_array = arr.filter(function(elem, index, self) {

return index == self.indexOf(elem); }

return unique\_array }

console.log(unique\_array(array\_with\_duplicates));

**3. By using for loop**: In this, we can use for loop to remove duplicates. In this we make an empty array in which those elements will be added from the duplicate array which are not present in this before. Thus, finally we will get an array which has unique elements. Code to implement this:

Array dups\_names = ['Ron', 'Pal', 'Fred', 'Rongo', 'Ron'];

**function** dups\_array(dups\_names) {

**let** unique = {};

names.forEach(**functio**n(i) {

**If** (!unique[i]) {

unique[i] = **true**; }

});

**return** Object.keys(unique);} // Ron, Pal, Fred, Rongo

Dups\_array(names);

These are the main three methods used in JavaScript to get a unique array.

[13. Explain few difference between null, undefined or undeclared JavaScript variable?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled13)

**Null** is a value that can be assigned to a variable or an object.

**Undefined** means a variable has been declared but no value is assigned to it. This type of variable is declared itself to be undefined.

**Undeclared** means the variable has declared without any datatype.

Null, Undefined are primitive data types whereas Undeclared is not a primitive data type.

[14. List few difference between primitive and non primitive JavaScript data types?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled14)

* The primitive data types are numbers, strings, Boolean, undefined, null and anything other than these data types are known as non-primitive such as objects and functions.
* Primitive data types are immutable while non-primitives are mutable.
* Primitives are known immutable as they can't be changed once they created but non-primitive are changeable, means once an object is created, it can be changed.
* Primitives data types are compared with their values, it means two values are strictly equal if they have the same data type and holds the same value.
* Non-primitives are not compared with values. For example, if two objects have the same properties and values, they are strictly not equal.

[15. How to add a new property in existing function JavaScript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled15)

It is easy to add a new property in existing function by just giving value to the existing function it. For example, let we have an existing object person, to give new property check the below code:

person.country= “India”;

The new property “country” has added to the object person.

[16. What is difference between deep and shallow object coping in JavaScript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled16)

Some differences are:

* Deep copy means copies all values or properties recursively in the new object whereas shallow copy copies only the reference.
* In a deep copy, changes in the new object don't show in original object whereas, in shallow copy, changes in new objects will reflect in the original object.
* In a deep copy, original objects do not share the same properties with new object whereas, in shallow copy, they do.

[17. How to call a function in every x seconds in JavaScript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled17)

In JavaScript, we use the function setInterval() to call any function in every x seconds.

**Syntax:** setInterval(function, milliseconds, param1, param2, ...)

**Function:** it is a required parameter which includes the function to be execute.

**Milliseconds:** required parameter which tells how often the function will execute.

Others are an additional parameter.

**For example:** setInterval(function (){ alert("Hello"); }, 3000);

In the above example, this function calls hello function in very 3 seconds.

[18. Explain spread operator in JavaScript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled18)

The spread operator expands an expression in places where multiple argument/variables/elements are needed to present. It represents with three dots (…).

For example:

var mid = [3, 4];

var newarray = [1, 2, ...mid, 5, 6];

console.log(newarray);

// [1, 2, 3, 4, 5, 6]

In above example, instead of appending mid array, it rather expands in the newarray with the help of spread operator. This is how spread operator works in JavaScript.

[19. How host objects are different from native objects in JavaScript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled19)

**Host objects**: These are those objects which environment gives. It means they are different for different environments. For example, browsers include objects such as windows but Node.js environments give objects such as Node List.

**Native Objects**: these are built-in objects in JavaScript. They are also known as Global Objects because they will be available to you independent of ay environment if you working in JavaScript.

[20. Explain higher-order functions in JavaScript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled20)

Higher order function is the best feature of functional programming available in JavaScript. It is the function which takes a function as an argument and returns a function as a result. Some of the inbuilt higher-order functions are mapping, filtering, reduction, zipping, etc.

[21. Please explain equality operators in JavaScript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled21)

[22. What are anonymous functions in JavaScript ?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled22)

[23. What is difference between local and global scope in JavaScript ?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled23)

[24. What is use of settimeout function in JavaScript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled24)

[25. List few advantages of using JavaScript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled25)

Few advantage og Javascript

* Javascript is executed on user's computer, the meaning is that whatever you do in Javascript will not add any processing strain on the server. and that's why it is called as the client-side programming language. And this feature makes your sites responsive for the end user and less expensive for you in terms of server traffic.
* With the help of Javascript, you can create highly responsive interfaces which will improve the user experience and provide dynamic functionality, without waiting for the server to show another page.
* If you want to make online systems available offline and sync automatically once the computer goes online, then Javascript is the best technology you can use. you can do this using the right browser add-ons (Such as Google or Yahoo Browser Plus).
* Content loading and changing it dynamically. Using Ajax in Javascript you can load content into the document if and when the user needs it, without reloading the entire page.
* Using the Principles of unobtrusive JavaScript(defensive Scripting), JavaScript can test for what is possible in your browser and react accordingly.

[26. List few Difference between JAVA and JavaScript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled26)

[27. Is JavaScript multi-threaded or single-threaded?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled27)

JavaScript is single-threaded.

[28. Is it possible to do 301 redirects in Javascript ?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled28)

JavaScript entirely runs on the client machine. 301 is response code that is sent by the server as a response. So it is not possible to do 301 Redirects In JavaScript.

[29. How to get inner Html of an element in JavaScript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled29)

InnerHTML property of HTML DOM is used to get inner Html of an element in JavaScript.

**Example Usage:**

This is inner Element

<script type="text/javascript">

var inner= document.getElementById("inner").innerHTML ;

console.log(inner); // This is inner Element

document.getElementById("inner").innerHTML = "Html changed!";

var inner= document.getElementById("inner").innerHTML ;

console.log(inner); // Html changed!

</script>

[30. Explain Event bubbling and Event Capturing in JavaScript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled30)

**Event Capture and Bubbling**: In HTML DOM API there are two ways of event propagation and determines the order in which event will be received. The two ways are Event Bubbling and Event Capturing. The first method event bubbling directs the event to its intended target, and the second is called event capture in which the event goes down to the element.

**Event Capture**

The capture procedure is rarely used but when it’s used it proves to be very helpful. This process is also called ‘trickling’. In this process, the event is captured first by the outermost element and then propagated to the innermost element. For example:

<div>

<ul>

<li></li>

</ul>

</div>

From the above example, suppose the click event did occur in the ‘li’ element, in that case capturing event it will be first handled ‘div’, then ‘ul’ and at last the target element will be hit that is ‘li’

**Event Bubbling**

Bubbling just works like the bubbles, the event gets handled by the innermost element and then propagated to the outer element.

<div>

<ul>

<li></li>

</ul>

</div>

From the above example, suppose the click event did occur in the ‘li’ element in bubbling model the event will be handled first by ‘li’ then by ‘ul’ and at last by ‘div’ element.

[31. How to import all exports of a file as an object.](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled31)

import \* as object name from ‘./file.js’ is used to import all exported members as an object. You can simply access the exported variables or methods using dot (.) operator of the object.

Example:

objectname.member1;

objectname.member2;

objectname.memberfunc();

[32. Explain Arrow functions?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled32)

An arrow function is a consise and short way to write function expressions in Es6 or above.A rrow functions cannot be used as constructors and also does not supports this, arguments, super, or new.target keywords. It is best suited for non-method functions. In general an arrow function looks like const function\_name= ()=>{}

const greet=()=>{console.log('hello');}

greet();

[33. Explain function hoisting in JavaScript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled33)

JavaScript’s default behavior that allows moving declarations to the top is called Hoisting. The 2 ways of creating functions in JavaScript are **Function Declaration** and **Function Expression**. Let’s find out more about these:

***Function Declaration***

A function with the specific parameters is known as function declarations. To create a variable in JavaScript is called declarations.

**e.g:**

hoisted(); // logs "foo"

function hoisted() {

 console.log('foo');

}

**Function Expression**

When a function is created by using an expression it is called function expression.

e.g:

notHoisted(); // TypeError: notHoisted is not a function

var notHoisted = function() {

  console.log('bar');

};

[34. How to remove duplicate values from a JavaScript array?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled34)

We can use array.indexOf method to check a value exists or not. See below example to remove duplicate values.

let duplicates = ['delhi','kanpur','kanpur','goa','delhi','new york'];

function removeDuplicatesValues(arr){

let unique\_array = [];

for(let i = 0;i < arr.length; i++){

if(unique\_array.indexOf(arr[i]) == -1){

unique\_array.push(arr[i])

}

}

return unique\_array

}

console.log(removeDuplicatesValues(duplicates));

[35. What is console.time() and console.timeEnd()? What is its syntax, and why is it used?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled35)

[36. Explain JavaScript Debounce Function?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled36)

[37. What is difference between var x =1; and x=1;?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled37)

[38. Explain JavaScript Accessors ?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled38)

JavaScript Accessors

[39. What is difference between module.exports and export?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled39)

The module is a plain JavaScript object with an exports property. Exports is a plain JavaScript variable that happens to be set to module.exports. At the end of your file, node.js will basically ‘return’ module.exports to the require function. A simplified way to view a JS file in Node could be this:

var module = { exports: {} };

var exports = module.exports;

// your code

return module.exports;

If you set a property on exports, like exports.a = 9;, that will set module.exports.a as well because objects are passed around as references in JavaScript, which means that if you set multiple variables to the same object, they are all the same object; so then exports and module.exports are the same objects.  
But if you set exports to something new, it will no longer be set to module.exports, so exports and module.exports are no longer the same objects.

Source : https://stackoverflow.com/questions/16383795/difference-between-module-exports-and-exports-in-the-commonjs-module-system

[40. What are exports and imports?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled40)

Imports and exports help us to write modular javascript code. Using Imports and exports we can split our code into multiple files. Imports allow taking only some specific variables or methods of a file. We can import methods or variables that are exported by a module. See the below example for more detail.

//index.js

import name,age from './person';

console.log(name);

console.log(age);

//person.js

let name ='Sharad', occupation='developer', age =26;

export { name, age};

[41. What is the difference between let and var?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled41)

Both var and let are used for variable/ method declaration in javascript but the main difference between let and var is that **var** is function scoped whereas **let** is block scoped.

[42. List the comparison operators supported by Javascript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled42)

Javascript supports below comparison operators

* > Greater than
* < Less than
* <= Less than or equal to
* >= Greater than or equal to
* == Equal to
* != Not Equal to
* === Equal to with datatype check
* !== Not equal to with datatype check

[43. List some Unit Testing Frameworks JavaScript](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled43)

Below is the list of few most Popular Javascript Unit Testing Frameworks:

* Unit.js
* Jasmine
* Karma
* Chai
* AVA
* Mocha
* JSUnit
* QUnit
* Jest

[44. How do you declare variables in Javascript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled44)

In Javascript variable are declared using the var keyword.A variable must begin with A **letter**, **$** or \_.

**eg.** var myVar=”Online Interview Questions”;

**PS:** All variables in Javascript are Case sensitive.

Also, read [**Advanced JavaScript Interview Questions**](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/)

[45. What are different types of Popup boxes available in Javascript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled45)

In Javascript there are 3 types of Popup Boxes are available, they are

* Alert
* Confirm
* Prompt

Read [**80+ Best Angular Js Interview Questions**](https://www.onlineinterviewquestions.com/angular-js-interview-questions/)

[46. How to convert Javascript date to ISO standard?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled46)

**toISOString()** method is used to convert javascript date to ISO standard. It converts JavaScript Date object into a string, using the ISO standard.

**Usage:**

var date = new Date();

var n = date.toISOString();

console.log(n);

// YYYY-MM-DDTHH:mm:ss.sssZ

[47. How to you change the title of the page by JavaScript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled47)

You can change the title of a webpage using setting the title property of the document object.

**Example usage**

document.title="My New Title";

[48. How to clone an object in Javascript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled48)

Object.assign() method is used for cloning an object in Javascript.Here is sample usage

var x = {myProp: "value"};

var y = Object.assign({}, x);

[49. How to encode and decode a URL in JavaScript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled49)

**encodeURI()** function is used to encode an URL in Javascript.It takes a url string as parameter and return encoded string. Note: encodeURI() did not encode characters like **/ ? : @ & = + $ #**, if you have to encode these characters too please use encodeURIComponent(). Usage:

var uri = "my profile.php?name=sammer&occupation=pāntiNG";

var encoded\_uri = encodeURI(uri);

**decodeURI()** function is used to decode an URL in Javascript.It takes a encoded url string as parameter and return decoded string. Usage:

var uri = "my profile.php?name=sammer&occupation=pāntiNG";

var encoded\_uri = encodeURI(uri);

decodeURI(encoded\_uri);

[50. Explain Typecasting in Javascript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled50)

In Programming whenever we need to convert a variable from one data type to another Typecasting is used. In Javascript, we can do this via library functions. There are basically 3 typecasts are available in Javascript Programming, they are:

* Boolean(value): Casts the inputted value to a Boolean
* Number(value): Casts the inputted value to an Integer or Floating point Number.
* String(value) : Casts the inputted value value a string

[51. List different ways of empty an array in Javascript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled51)

In Javascript, there are many ways to empty an array in Javascript, below we have listed 4 major

* By assigning an empty array.
* var arr1 =[1,4,5,6];
* arr1=[];
* By assigning array length to 0.
* var arr2 =[1,4,5,6];
* arr2.length=0;
* By poping the elements of the array.
* var arr2 =[1,4,5,6];
* while(arr.length > 0) {
* arr.pop();
* }
* By using .splice() .
* var arr =[1,4,5,6];
* arr.splice(0,arr.length)

[52. What is the ‘Strict’ mode in JavaScript and how can it be enabled?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled52)

**Strict mode** is a way to introduce better error-checking into your code. When you use strict mode, you cannot, for example, use implicitly declared variables, or assign a value to a read-only property, or add a property to an object that is not extensible.

You can enable strict mode by adding **“use strict”**; at the beginning of a file, a program, or a function. This kind of declaration is known as a directive prologue. The scope of a strict mode declaration depends on its context. If it is declared in a global context (outside the scope of a function), all the code in the program is in strict mode. If it is declared in a function, all the code in the function is in strict mode.

[53. How can you create an array in Javascript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled53)

There are 3 different ways to create an array in Javascript. They are

* By array literal  
  **usage:**

var myArray=[value1,value2...valueN];

* By creating instance of Array  
  **usage:**

var myArray=new Array();

* By using an Array constructor  
  **usage:**

var myArray=new Array('value1','value2',...,'valueN');

[54. What is Javascript BOM?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled54)

BOM stands for “Browser Object Modal” that allows Javascript to ‘talk’ to the browser, no standards, modern browsers implement similar BOMS – window, screen, location, history, navigator, timing, cookies.

[55. What does the instanceof operator do?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled55)

In Javascript **instanceof** operator checks whether the object is an instance of a class or not:

**Example Usage**

Square.prototype = new Square();

console.log(sq instanceof Square); // true

[56. How to get the primitive value of a string in Javascript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled56)

In Javascript **valueOf()** method is used to get the primitive value of a string.

**Example Usage:**

var myVar= "Hi!"

console.log(myVar.valueOf())

[57. How to get the last index of a string in Javascript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled57)

**string.length-1** is used to get the last index of a string in Javascript

**Example Usage:-**

var myString="JavascriptQuestions";

console.log(myString.length-1);

[58. List HTML DOM mouse events?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled58)

HTML DOM mouse events

* onclick
* ondblclick
* mousemove
* mousedown
* mouseover
* mouseout
* mouseup

[59. What will happen if an infinite while loop is run in Javascript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled59)

The program will crash the browser.

[60. In Javascript are calculations with fractional numbers guaranteed to be precise?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled60)

NO, calculations with fractional numbers are not guaranteed to be precise in Javascript

[61. What is the difference between the substr() and substring() functions in JavaScript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled61)

Difference between the substr() and substring() functions in JavaScript.

The substr() function has the form substr(startIndex,length). It returns the substring from startIndex and returns ‘length’ number of characters.

var s = "hello";

( s.substr(1,4) == "ello" ) // true

The substring() function has the form substring(startIndex,endIndex). It returns the substring from startIndex up to endIndex – 1.

var s = "hello";

( s.substring(1,4) == "ell" ) // true

[62. What are the primitive data types in JavaScript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled62)

A primitive is a basic data type that’s not built out of other data types. It can only represent one single value. All primitives are built-in data types by necessity, (the compiler has to know about them,) but not all built-in data types are primitives.

In JavaScript there are 5 primitive data types are available they are **undefined**, **null**, **boolean**, **string** and **number** are available.Everything else in Javascript is an object.

[63. How to calculate Fibonacci numbers in JavaScript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled63)

Calculating Fibonacci series in JavaScript

Fibonacci numbers are a sequence of numbers where each value is the sum of the previous two, starting with 0 and 1. The first few values are 0, 1, 1, 2, 3, 5, 8, 13 ,…,

function fib(n) {

var a=0, b=1;

for (var i=0; i < n; i++) {

var temp = a+b;

a = b;

b = temp;

}

return a;

}

[64. What are different types of Inheritence? Which Inheritance is followed in Javascript.](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled64)

There are two types of Inherientence in OOPS Classic and Prototypical Inheritance. Javascript follows Prototypical Inheritance.

[65. What is output of undefined \* 2 in Javascript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled65)

nan is output of undefined \* 2.

[66. Explain “use strict” ?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled66)

“use strict” is a javascript directive that is introduced in Es5. The purpose of using “use strict” directive is to enforce the code is executed in strict mode. In strict mode we can’t use a variable without declaring it. “use strict” is ignored by earlier versions of Javascript.

[67. Describe negative infinity in JavaScript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled67)

NEGATIVE\_INFINITY property represents negative infinity and is a number in javascript, which is derived by ‘dividing negative number by zero’. It can be better understood as a number that is lower than any other number. Its properties are as follows:  
– A number of objects need not to be created to access this static property.  
– The value of negative infinity is the same as the negative value of the infinity property of the global object.

The values behave differently than the mathematical infinity:

1. Any positive value, including POSITIVE\_INFINITY, multiplied by NEGATIVE\_INFINITY is NEGATIVE\_INFINITY.
2. Any negative value, including NEGATIVE\_INFINITY, multiplied by NEGATIVE\_INFINITY is POSITIVE\_INFINITY.
3. Zero multiplied by NEGATIVE\_INFINITY is NaN.
4. NaN multiplied by NEGATIVE\_INFINITY is NaN.
5. NEGATIVE\_INFINITY, divided by any negative value except NEGATIVE\_INFINITY, is POSITIVE\_INFINITY.
6. NEGATIVE\_INFINITY, divided by any positive value except POSITIVE\_INFINITY, is NEGATIVE\_INFINITY.
7. NEGATIVE\_INFINITY, divided by either NEGATIVE\_INFINITY or POSITIVE\_INFINITY, is NaN.
8. Any number divided by NEGATIVE\_INFINITY is zero.

[68. Explain JavaScript Event Delegation Model?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled68)

In JavaScript, there is some cool stuff that makes it the best of all. One of them is Delegation Model. When capturing and bubbling, allow functions to implement one single handler to many elements at one particular time then that is called event delegation. Event delegation allows you to add event listeners to one parent instead of specified nodes. That particular listener analyzes bubbled events to find a match on the child elements. Many people think it to be complicated but in reality, it is very simple if one starts understanding it.

Also, **Read Five**[**JavaScript Frameworks to learn in 2018**](https://www.onlineinterviewquestions.com/blog/5-javascript-frameworks-to-learn/)

[69. Explain Closures in JavaScript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled69)

Closures are the combination of lexical environment and function within which the function was declared. This allows JavaScript programmers to write better, more creative, concise and expressive codes. The closure will consist of all the local variables that were in-scope when the closure was created.

Sure, closures appear to be complex and beyond the scope, but after you read this article, closures will be much more easy to understand and more simple for your everyday [JavaScript](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/)programming tasks. JavaScript is  a very function-oriented language it gives the user freedom to use functions as the wish of the programmer.

[70. What close() does in Javascript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled70)

In Javascript close() method is used to close the current window. You must write window.close() to ensure that this command is associated with a window object and not some other JavaScript object.

[71. Explain what is Javascript? List some data types supported by Javascript?](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/" \l "collapseUnfiled71)

Javascript

Javascript is an object-oriented computer programming language commonly used to create interactive effects within web browsers.It is first used by the Netscape browser, that provides access to the HTML document object model (DOM), provides access to the browser object model (BOM). Javascript syntax looks a lot like java, c or c++ syntax.

Below is the list of data types supported by Javascript:-

* Undefined
* Null
* Boolean
* String
* Symbol
* Number

Commonly asked JavaScript Interview Questions | Set 1

1. **What is**[**JavaScript(JS)**](https://www.geeksforgeeks.org/how-to-be-a-javascript-developer-without-knowing-javascript/)**?**
   * JavaScript is a lightweight, interpreted programming language with object-oriented capabilities that allows you to build interactivity into otherwise static HTML pages.
2. **What are the features of JavaScript?**
   * JavaScript is a lightweight, interpreted programming language.
   * JavaScript is designed for creating network-centric applications.
   * JavaScript is complementary to and integrated with Java.
   * JavaScript is complementary to and integrated with HTML.
   * JavaScript is open and cross-platform.
3. **What are the advantages of JavaScript?**
   * **Less server interaction?** You can validate user input before sending the page off to the server.
   * **Immediate feedback to the visitors?** They don’t have to wait for a page reload to see if they have forgotten to enter something.
   * **Increased interactivity?** You can create interfaces that react when the user hovers over them with a mouse or activates them via the keyboard
4. **Why is javascript called Richer Interface?**
   * You can use JavaScript to include such items as drag-and-drop components and sliders to give a Rich Interface to your site visitors.
5. **Is javascript case-sensitive?**
   * Yes, JavaScript is a case-sensitive language. This means that language keywords, variables, function names, and any other identifiers must always be typed with a consistent capitalization of letters.
6. **How can we create an object in JS?**
7. var object =
8. {
9. name: "obj",
10. age: 10

};

1. **How can we read the properties of an object in js?**
   * Can write and read properties of an object using the dot(.) notation.
2. **How to create an array in js and how to read array elements?**  
   Can you define arrays using the array literal as follows?
   * var x = [];
   * var y = [1, 2, 3, 4, 5];

An array has a length property that is useful for iteration. Can we read elements of an array as follows?

for (var i = 0; i < x.length; i++)

1. **How many types of functions JS support?**
   * A function in JavaScript can be either named or anonymous.
2. **How to define an anonymous function?**
   * An anonymous function can be defined in a similar way as a normal function but it would not have any name.
3. **Which built-in method calls a function for each element in the array?**
   * forEach method calls a function for each element in the array.
4. **Which type of variable among global and local, takes precedence over other if names are same?**
   * A local variable takes precedence over a global variable with the same name.
5. **Difference between**[**“var” and “let”**](https://www.geeksforgeeks.org/difference-between-var-and-let-in-javascript/)**Keywords?**
   * **Var** was there from the beginning but the let was introduced in ES2015/ES6.  
     **Let** has block scope and “Var” has function scope
6. **Difference between**[**“==” and “===”**](https://www.geeksforgeeks.org/difference-between-and-operator-in-javascript/)**?**
   * ” ==” only compares values “===” comapre values and type both.
7. **Difference between “undefine” and “NULL” Keywords?**
   * When you define a var but not assign any value. typeof(undefine)=> undefine  
     Null- manually done. typeof(null)=> object
8. **What is prototypal Inheritance?**
   * Every object has a property called a prototype, where we can add methods to it and when you create another object from these the newly created object will automatically inherit its parent’s property.
9. **Which built-in method sorts the elements of an array?**
   * [Sort](https://www.geeksforgeeks.org/javascript-sort-method/) method sorts the elements of an array.
10. **Which built-in method reverses the order of the elements of an array?**
    * [Reverse](https://www.geeksforgeeks.org/javascript-array-prototype-reverse/) method reverses the order of the elements of an array ?? the first becomes the last, and the last becomes the first.
11. **What is**[**SetTimeout()**](https://www.geeksforgeeks.org/java-script-settimeout-setinterval-method/)**?**
    * When you setTimeout it becomes asynchronous and it has to wait on the stack to get everything got finished
12. **How to add one new element at end of an array in javascript?**
    * Push method adds one or more elements to the end of an array and returns the new length of the array.
13. **What is closure and how do you use it?**
    * When a function returns the other function the returning function will hold its environment and this is known as closure.
14. **Output of below statements**

filter\_none

edit

play\_arrow

brightness\_4

|  |
| --- |
| <script>  document.write({});  </script> |

1. **How can you move element in lowercase to uppercase from an array?**
   * [toUpperCase](https://www.geeksforgeeks.org/javascript-string-touppercase/) method returns the calling string value converted to upper case.

JavaScript Interview Questions And Answers

Q.1- What’s An Object In Javascript And How Do We Create Them?

A JavaScript object is an entity having state and behavior (properties and method). Since JavaScript is an object-based language, it treats everything as an object.

JavaScript is a template-based language. It doesn’t need to define a class for creating an object instead creates it directly

JavaScript supports the following three ways to create objects.

1. By Object Literal.

The syntax of creating an object using object literal is as follows.

object={property1:value1, property2:value2.....propertyN:valueN}

Here, property and value get separated by a colon “:” sign.

Let’s take an example.

<script>

std={id:1114, name:"Ram Bajaj", subject:"Physics"}

document.write(std.id+" "+std.name+" "+std.subject);

</script>

2. By Creating An Instance Of The Object (Using New Keyword).

The syntax of creating an object:

var objectname=new Object();

Here, the new keyword is used to create the object.

Let’s take an example.

<script>

var std=new Object();

std.id=1114;

std.name="Ram Bajaj";

std.subject="Physics";

document.write(std.id+" "+std.name+" "+std.subject);

</script>

3. By Using An Object Constructor.

In this method, we create a function with arguments. The value of each of these arguments can be assigned to the current object by using this keyword.

This keyword refers to the current object.

Let’s take an example of creating an object using the object constructor technique.

<script>

function std(id,name,subject){

this.id=id;

this.name=name;

this.subject=subject;

}

s=new std(1114,"Ram Bajaj","Physics");

document.write(s.id+" "+s.name+" "+s.subject);

</script>

Q-2. What Does A Scope Mean In JavaScript?

The scope determines the accessibility of variables, objects, and functions in particular part of your code.

In JavaScript, the scope is of two types.

1. Global Scope.

A variable defined outside a function comes under the Global scope. Variables defined inside the Global scope are accessible from any part of the code. Let’s see an example.

var name = 'TechBeamers';

console.log(name); // logs 'TechBeamers'

function logName() {

console.log(name); // 'name' is accessible here and everywhere else

}

logName(); // logs 'TechBeamers'

2. Local Scope.

Variables defined inside a function comes under the Local scope. Different functions can use a variable with the same name. It is because these variables are strictly bound to the function that defines it (each having different scopes) and is not accessible in other functions. Let’s see an example.

// Global Scope

function sampleFunction() {

// Local Scope #1

function sample2Function() {

// Local Scope #2

}

}

// Global Scope

function sample3Function() {

// Local Scope #3

}

// Global Scope

Q-3. What Is <This> In JavaScript?

All the OOPs languages use ‘this’ keyword to refer to an object that is currently instantiated by the class. However, in JavaScript, ‘this’ refers to an object which ‘owns’ the method. Though this varies, with how a function call happens.

Global Scope.

If no object is currently available, then ‘this’ represents the global object. In a web browser, ‘window’ is the top-level object which represents the document, location, history and a few other useful properties and methods. Let’s take a sample code.

window.Obj= "I represent the window object";

alert(window.Obj);

alert(this.Obj); // I'm the window object

alert(window === this); // true

The Scenario Of A Function Call.

In the case of a function call, ‘this’ refers to the global object.

window.Obj = "I represent the window object";

function TestFunction() {

alert(this.Obj); // I'm the window object

alert(window === this); // true

}

TestFunction();

Call Object Methods.

When an object constructor or any of its methods gets called, ‘this’ refers to an instance of an object. It is similar to any class-based language.

window.Obj = "I'm the window object";

function TestFunction() {

this.Obj = "I'm the Test object";

this.Verify1 = function() {

alert(this.Obj); // I'm the Test object

};

}

TestFunction.prototype.Verify2 = function() {

alert(this.Obj); // I'm the Test object

};

var tf= new TestFunction();

tf.Verify1();

tf.Verify2();

Q-4. What Is The Prototype Property In JavaScript?

Every JavaScript function has a prototype property (by default this property is null), that is mainly used for implementing inheritance. We add methods and properties to a function’s prototype so that it becomes available to instances of that function. Let’s take an example that calculates the perimeter of a rectangle.

function Rectangle(x, y) {

this.x = x;

this.y = y;

}

Rectangle.prototype.perimeter = function() {

return 2 \* (this.x + this.y);

}

var rect = new Rectangle(4, 2);

console.log(rect.perimeter()); // outputs '12'

Q-5. What Is Closure In JavaScript?

A closure is a JavaScript function defined inside another function. And that’s why it gets a special privilege to access three types of scope which are as follows.

* Internal Scope, i.e., the variables defined between its curly brackets
* Outer Function Scope, i.e., the variables of the enclosing function
* Global Scope, i.e., variables defined as globals

Please note that a closure can not only access the outer function variables but also see its parameters. But it can’t call the object of the outer function’s arguments. However, it can directly call the outer function’s parameters.

Here is a code example describing closure by adding a function inside another function.

function outerFunc(arg1, arg2) {

​var param = "I'm closure. ";

// Inner function accessing outer function variables and parameter​s

​function innerFunc() {

​ return arg1 + arg2 + " " + param;

}

​ ​return innerFunc();

}

​

outerFunc("arg1", "arg2");

Q-6. Why Is “Self” Needed Instead Of “This” In JavaScript?

Inner functions in JavaScript have access to all of the variables defined in the outer function. However, “this” variable is an exception. Since the nested function is just a regular function and not an object method, it’s “this” refers to the global namespace. To make it more clear, let’s look at the following example.

var aProperty = 'global';

var myObject = {

outerFun: function() {

this.aProperty = 'local';

setTimeout(function() {

console.log(this.aProperty); // outputs 'global'

}, 1);

}

};

Thus, we see that inside “setTimeout” function, “this” refers to the global object. We need a way to get a reference to the object, that is available inside the nested function. We assign the object from “this”, to another(non-special) variable, “self”. It is not a special variable and hence cannot be overwritten by other functions(like “this”). Thus on using “self” inside the inner function, we can refer to the local object. Following is the sample code.

var myObject = {

outerFun: function() {

var self = this;

this.aProperty = 'local';

setTimeout(function() {

console.log(self.aProperty); // outputs 'local'

}, 1);

}

};

Q-7. What Is An Anonymous Function And When Should You Use It?

Anonymous functions are functions that are dynamically declared at runtime. They’re called anonymous functions because they don’t have a name like normal functions.

We use the function operator to declare an anonymous function, instead of the function declaration. Also, the function operator can be used to create a new function, wherever it’s valid to put an expression. For example, we declare a new function to be supplied as an argument to a function call or to assign a property of another object.

Here’s a typical example of a named function.

function testFunction()

{

alert("Welcome!!");

}

testFunction();

Here’s the same example created as an anonymous function.

var testFunction= function()

{

alert("Zoom! Zoom! Zoom!");

}

flyToTheMoon();

Following are the key usage of anonymous functions.

* **Code brevity.**
  + Use them in
    - Callbacks, and
    - Event handlers.
* **Scope management.**
  + They are useful in the following scenario.
    - To create a temporary/private scope.
    - In Closures and Recursions.

Q-8. What Is The Difference Between “==” And “===”?

These are the operators provided by JavaScript – strict equality and Type converting equality.

Strict equality (===) returns true if the values which it is going to compare have the same data type. Taking an example, “2” will not be equal to 2  i.e. (“2″===2) will return false.

Secondly, Type converting equality (==), automatically converts the variable to value irrespective of the data type. Taking an example, here “2” will be equal to 2  i.e. (“2″===2) will return true.

Summarizing it, double equal (==) is an autotype converting equality operator while three equals (===) is a strict equality operator, i.e., it will not convert values automatically.

Q-9. What Are JavaScript Data Types?

JavaScript supports three Primary, two Composite and two Special data types. Next, we list down the data types in each of the categories.

Primary Data Types.

* String
* Number
* Boolean

Composite Data Types.

* Object
* Array

Special Data Types.

* Null
* Undefined

Q-10. What Is Prototypal Inheritance In JavaScript?

Most of the Object Oriented languages support classes and objects. Here, Classes inherit from other classes.

In JavaScript, the inheritance is prototype-based. It means that there are no classes. Instead, there is an object that inherits from another object.

JavaScript provides three different types of Prototypal Inheritance.

1. Delegation (I.E., The Prototype Chain).

A delegate prototype is an object that serves as a base for another object. When you inherit from a delegate prototype, the new object gets a reference to the prototype.

When we try to access any property, it first checks in the properties owned by the object. If that property does not exist there, it checks in the **‘[[Prototype]]’** and so on. If that property does not exist there, it checks in the **‘[[Prototype]]’** and so on. Gradually, it moves up the prototype chain, until it reaches the **<Object.prototype>** i.e., the root delegate for most of the objects.

2. Concatenative Inheritance (I.E. Mixins, Object.Assign()).

It is the process of inheriting the features of one object to another by copying the source objects properties. JavaScript calls these source prototypes by the name mixins. This process makes use of the JavaScript method Object.assign(). However, before ES6, the <.extend()> method was used.

3. Functional (Not To Be Confused With Functional Programming).

In JavaScript, a function can create an object. It’s not necessary to be a constructor(or a class). It is called a factory function. Functional inheritance produces an object from a factory and also extends it, by assigning properties.

Every type of Prototypal Inheritance supports a separate set of use-cases, applicable to it. All of them are equally useful in their ability to enable composition. It provides a has-a, uses-a, or can-do relationship as compared to the is-a relationship created with class inheritance.

Q-11. What Is Asynchronous Programming? Why Is It Important In JavaScript?

In Asynchronous programming, the engine runs in an event loop. On encountering a blocking operation, a request gets fired, and the code keeps running without blocking for the result. When its execution completes, and the response is ready, it fires an interrupt, which causes an event handler to be run, where the control flow continues. In this way, a single program thread can handle many concurrent operations in asynchronous programming.

The user interface is asynchronous by nature and spends most of the time, waiting for user input to interrupt the event loop and trigger event handlers. Node is asynchronous by default. It means that the server works by waiting for a network request in a loop. Thus, it accepts more incoming requests, while the first one is getting handled.

It is important in JavaScript because it’s suitable for user interface code and beneficial for maintaining the performance on the server.

Q-12. What Is Variable Typing In JavaScript?

JavaScript is very loosely typed language. It means that the variables are declared without a type. Its type is determined once a value is assigned to it. It can change as the variable appears in different contexts.

It’s in contrast to the strongly typed languages that require type declarations.

Consider the following examples.

/\* JavaScript Example (loose typing) \*/

var a = 13; // Number declaration

var b = "thirteen"; // String declaration

/\* Java Example (strong typing) \*/

int a = 13; // int declaration

String b = "thirteen"; // String declaration

Variable type in JavaScript is determined internally. In the above example, var ‘a’ will be of type ‘Number’ and var ‘b’ will be of type ‘String’.

Q-13. What Are The Different Ways To Create An Array In JavaScript?

There are two main ways to create an array in JavaScript.

1. Using An Array Initializer (Array Literal).

The array initializer (array literal) syntax is simple. It is a comma-separated list of values in square brackets.

Let’s see some examples.

var myArray1 = [1,2,3,4,5] // an array with 5 elements

var myArray2 = [5] // an array with 1 element

var myArray3 = [true,'Hi',[7]] // element types need not be the same.

2. Using The Array Constructor.

The Array constructor method has three different syntaxes. If we call the constructor with two or more arguments, it declares an array with array elements also initialized. If we provide only one argument to the Array constructor, it refers to the length of the new array with, elements not initialized. Lastly, the constructor without any argument creates an array with its length set to zero with elements not initialized.

Let’s see some examples.

var myArray4 = new Array(1,2,3,4,5) // an array with 5 elements

var myArray5 = new Array(20) // an empty array of length 20

var myArray6 = new Array() // an empty array of length 0

Q-14. What Are JavaScript Cookies?

A cookie is a piece of data which is sent from a website (that owns the requested web page) and gets stored locally by the browser at the user end. Cookies are needed because HTTP protocol which arranges for the transfer of web pages to your browser, is stateless. It means that HTTP has no way to keep track of the activities performed by the user at an earlier point in time. One way to resolve this issue is by using cookies. It contains the following data.

* A name-value pair containing the actual data
* An expiry date after which the cookie is no longer valid
* The domain and path of the server it should be sent to

When a request arrives at the server for a web page that maintains a cookie, the server appends the cookie to the HTTP header to send it across. The server-side programs can then read out the information included in it and decide that you have the right to view the page or not and other user preferences.

Thus, every time you visit the site that maintains the cookies, your information is available there.

Q-15. Which Built-In Method Adds One Or More Elements To The End Of An Array And Returns The New Length Of The Array?

The push() method adds one or more elements to the end of an array and returns the new length of the array.

Q-16. What Is The Naming Conventions For The Variables In JavaScript?

We have to follow the below rules while naming the variables in JavaScript.

* Do not use any of the JavaScript reserved keywords as a name for your variable. For example, the “break” or “boolean” is the JavaScript keywords, and if used as variable names, it’s invalid.
* JavaScript variable names should not start with a numeral (0-9). It must begin with a letter or the underscore character. For example, **123var** is an invalid variable name, but **\_123var** is a valid one.
* Also, JavaScript variable names are case sensitive. For example, test and Test are two different variables.

Q-17. How Will You Create A Cookie Using JavaScript?

The simplest way to create a cookie is to assign a string value to the <**document.cookie**> object.

Its syntax is as follows.

document.cookie = "key1 = value1; key2 = value2; expires = date";

Here, “expires” attribute is optional. We have to provide a date or time value for this attribute.

If we provide a valid value for the date or time, then the cookie will expire at the given date or time, and it will not be accessible after that.

Q-18. How To Read A Cookie Using JavaScript?

To read a Cookie, we have to access the value of the <document.cookie> object. This <document.cookie> string maintains a list of <name = value> pairs that is separated with semicolons.

Where,

"name" is the name of a cookie and

"value" is its string value.

We use String <split()> function to break the <document.cookie> Object to sub-strings. Each of these sub-strings contains a key-value pair which represents the information related to a Cookie.

Q-19. How To Delete A Cookie Using JavaScript?

To delete a Cookie, we have to set its expiry date to a time that occurred in the past. If attempts are made to read a deleted Cookie then, nothing is returned.

Q-20. How Do You Submit A Form Using JavaScript?

We use the following in the web page code to submit a form.

document.forms[0].submit();

Here <0> refers to the index of the form. If we have more than one forms on a web page, then the index for the first one will be “0”, for the second form, its value will be “1” and so on.

Q-21. Why Should You Not Prefer To Use Global Variables In JavaScript And How Can You Prevent It?

The principal issue in using a global variable is that someone else can create another variable with the same name. And you may not know it until the duplicate could overwrite the value of your variable.

To avoid using globals, follow any of the following approaches.

**1.** Create a single global variable that holds all your other variables.

var myGlobalList = {};

myGlobalList.first = "test";

**2.** Enclose all of your code in a self-executing method/function so that any variable declared inside remain in the function scope.

(function(){

var test = "myvar";

})();

Q-22. What Are The Different Objects Used In JavaScript?

JavaScript uses a hierarchical structure, applicable to all the objects created in a document. Following are the objects, used in JavaScript that shows the relationship of one object to another.

Window Object.

It is the topmost object in the hierarchy. It refers to the content area of the browser window that consists of HTML documents. Each frame is also a window that has some actions inside it.

Document Object.

A Document object represents the HTML document that the window will display. It has various properties that refer to other objects, which allow access to and modification of content in the document.

Form Object.

A form object is used to take user data as input for processing. It corresponds to an HTML input form constructed with the **<FORM>…</FORM>** tag.

Q-23. What Do We Achieve By Deferring The Loading Of JavaScript?

During the page load operation, by default, the parsing of the HTML code remains paused until the execution of the script has not stopped.

Now, if the script is heavy or the server is slow, then this will result in a delay in displaying the web page.

By deferring the loading of JavaScript, we instruct the browser to load the script, only after the loading of the web page(DOM) has finished. It reduces the loading time of the webpage, and it gets displayed faster.

In this way, it makes the page load quicker, without the average user even realizing that some JavaScript load after the DOM has loaded.

Let’s see an example.

<script>

// this piece of code runs first

</script>

<script defer="defer">

//do stuff, but defer it (runs last)

</script>

<script>

//do more stuff (runs second)

</script>

Here the middle block will execute once the page has loaded even though it appears before the last block.

Q-24. What Is The Strict Mode In JavaScript?

Strict Mode imposes a layer of constraint on JavaScript. It provides the following enhancements.

* JavaScript will throw an error if we try to use the elements of a deprecated language.
* To use a variable, it has become mandatory to declare it.
* It disallows duplicate property and parameter names.
* The **eval()** method is safer to use, but still considered evil in some cases.
* It deprecates the “with” statement.
* JavaScript will throw an error if we try to assign a value to a read-only property.
* It decreases the global namespace pollution.

To enable strict mode, we have to add, “use strict” directive to the code. The physical location of the “strict” directive determines its scope. If used at the beginning of the js file, its scope is global. However, if we declare strict mode at the first line in the function block, its scope restricts to that function only.

Q.25- What Are Event Handlers In JavaScript And How To Use Them?

JavaScript event handlers are functions that bind to a specific HTML DOM event. And events are the part of the HTML document object model (DOM). An event can take place in one of the following cases.

Due to user actions on a web page.

<onclick()>, <onmouseover()>, <onkeydown()>

Some events are callbacks triggered by the browser when a page changes its state.

<onload()>, <onunload()>, <onresize()>

Whenever the DOM receives an event, it calls the JavaScript event handler function bind to the event. To use a handler function, we can assign it to the desired event as an attribute of the target HTML element. Please follow the below example.

<HTML>

<BODY>

<FORM>

<INPUT TYPE=”button” VALUE=”Test” onClick=”window.alert ('Event Handler')”>

</FORM>

</BODY>

</HTML>

Q-26. What Does A JavaScript Function Result When It Has No Return Statement?

If a function doesn’t use the return statement, then it returns undefined as the return value.

However, the browser won’t print anything in its console as it ignores if the expression returns an undefined value. But if we explicitly call such a function inside the **console.log()** method, then it’ll print exact value.

function test(){}

console.log(test()); // will print 'undefined'

Q-27. What Is EncodeURI() Function?

The **encodeURI()** function is used to encode a URI. This function encodes all special characters, except these **< , / ? : @ & = + $ #>.**

Let’s See An Example.

var uri="//www.techbeamers.com/how to make a website using javaScript";

var encodedURI = encodeURI(uri);

console.log(encodedURI);

Output.

http://www.somedomain.com/how%20to%20make%20a%20website%20using%20javaScript

We see that JavaScript encodes the space between the words in the **<uri>** variable as **<%20>**. Thus, the encodeURI function is used to encode special reserved characters and other non-ASCII characters in the URI.

Q-28. How Does The <Array()> Differ From <[]> While Creating A JavaScript Array?

Both the **<Array()>** and **<[]>** works almost the same in JavaScript.

If we use them as is (i.e., without any argument) to create an array object, then they will result in an array object of zero length. Also, if we pass a string or a list of strings as arguments, even then the result will be similar.

However, they differ when the input argument is of integer type. In that case, the **<Array(n)>** statement will create an uninitialized array of size of n. Whereas, the **<[n]>** statement will create an array of size **<1>**and assign **<n>** as value to the first element.

Q-29. How Does Variable Hoisting Take Place In JavaScript?

In JavaScript, variable declarations are the first piece of code that executes irrespective of their location. Hence, it doesn’t matter whether we declare a variable at the top or anywhere else. This functionality which moves the declaration to the top either inside a function or in the global code is known as hoisting.

out = 2;

var out;

function myfunc() {

var in = "inside"

// some code

}

// ...

JavaScript will interpret the above in the following manner.

var out;

out = 2;

function myfunc() {

var in;

// some code

in = "inside";

}

Q-30. How Will You Replace All Occurrences Of A String In JavaScript?

We can use String’s **<replace()>** method to substitute any string. There are the following two ways to use this method.

Pass The Input String As A Regular Expression.

str = "ghktestlllltest-sdds"

str = str.replace(/test/g, '');

alert(str)

Use RegExp Class To Create A Regular Expression And Pass It.

String.prototype.replaceAll = function(find, replace) {

var target = this;

return target.replace(new RegExp(find, 'g'), replace);

};

str = "ghktestlllltest-sdds"

str = str.replaceAll('test', '');

alert(str)

**ECMASCRIPT IQ**

**Q1. What is ES or ECMAScript or ES6?**

When JavaScript was developed, the name itself was chosen for marketing reasons since Java was getting popular around the time. But to standardize the language and its specification, it was submitted to ECMA international (European Computer Manufacturers Association), a body for standardization of information and communication technology. Eventually the language standardized in ECMA is called ECMAScript of ES. The first edition was released on June 1997.

The ES6 is the sixth edition of the language and was released on June 2015. It was initally known as ECMAScript 6 (ES6) and later renamed to ECMAScript 2015. This edition includes many new features like class, modules, iterators, for/of loop, arrow functions, typed arrays, promises, reflection.

In the next year on June 2016, ECMAScript 6 (ES6) was released and later renamed to ECMAScript 2015. This edition intended to continue the themes of language reform, includes two new features: exponentiation operator (\*\*) and Array.prototype.includes.

And then on June 2017, ECMAScript 2017 (ES2017), the eighth edition released which includes features for concurrency and atomics, syntactic integration with promises (async/await).

**Q2. What are some of the features of ES6?**

Some of the new features of ES6 are:

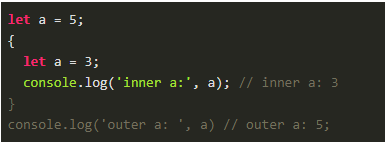
* Support for constants (also known as “immutable variables”)
* Block-Scope support for both variables, constants, functions
* Arrow functions
* Extended Parameter Handling
* Template Literals and Extended Literals
* Enhanced Regular Expression
* Destructuring Assignment
* Modules, Classes, Iterators, Generators
* Enhanced Object Properties
* Support for Map/Set & WeakMap/WeakSet
* Promises, Meta-Programming ,Internationalization and Localization

**Q3. What is let and const? And how it differs from var?**

Previously, when we declare any variable using var, it was function scoped. Meaning the variable can be accessed within the function. This leads to wrap the code in a function whenever we need to create a new scope.

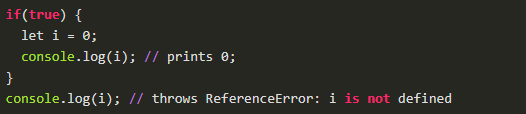
But let and const uses block scoping. This means the variable declared using these keywords only exist within the innermost block that surronds them.

https://miro.medium.com/max/60/1*BZFIX1601t27Ttj76ZVD0Q.png?q=20



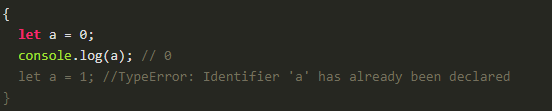
If we declare a let variable inside a block like if condition, for loop, it can be accessed within the block.

https://miro.medium.com/max/60/1*HmQ-fgtGyMYnCsmrFZiOMg.png?q=20



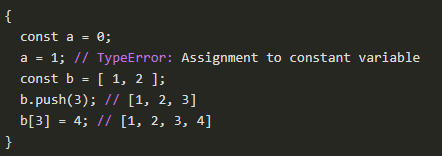
Also we can not re-declare the same variable with the same scope.

https://miro.medium.com/max/60/1*JP6jvGFvTjqoLIc32Xxt5A.png?q=20



Unlike const, const is immutable. It means the value must be given at the time of the declaration and it can not be re-assigned or changes. Although we can not change the value of the const but we can mutate them.

https://miro.medium.com/max/60/1*E2T3qhEpruD3fFcEmYGi4A.png?q=20

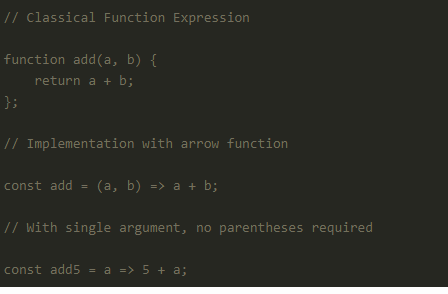


It is always a good practice to use let and const over var and if we are not changing the value of the variable we should use const.

**Q4. What is Arrow function? What are all its uses? How it differs from normal function?**

Arrow functions are a short-hand notation for writing functions in ES6. The arrow function definition consists of a parameter list ( ... ), followed by the =>marker and a function body. For single-argument functions, the parentheses may be omitted.

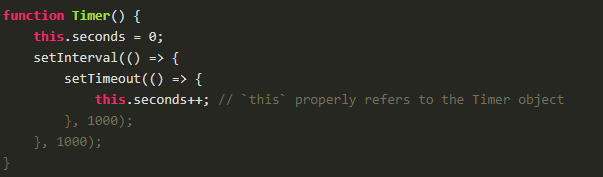
https://miro.medium.com/max/60/1*AwxBvPeMuKSLPvIIeqI1xw.png?q=20



If the arrow function is implemented with “concise body” (without {}), it does not need an explicit return statement. Note the omitted { } after the =>.

Arrow functions behavior with this keyword varies from that of normal functions. Each function in JavaScript defines its own this context but arrow functions capture the this value of the nearest enclosing context.

https://miro.medium.com/max/60/1*WiJjD4EJ6Qe2Jo-LLR3YYg.png?q=20



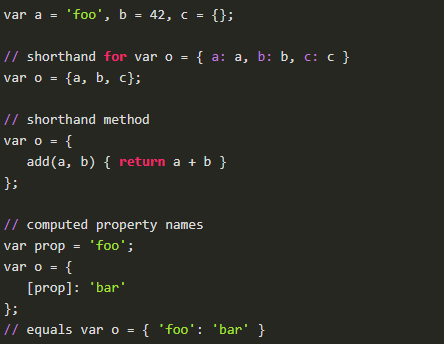
There are four fundamental differences between arrow functions and function functions

* They close over this, and do not have their own versions.
* They can have a concise body (without { }) rather than a verbose one (but they can have a verbose body as well).
* They cannot be used as constructors. E.g., you can’t use new with an arrow function. Hence arrow functions do not have a protoype property on them.
* There is no generator syntax for arrow functions. E.g., there is no arrow equivalent to function \*foo() { ... }.

**Q5. What are all the new changes in Object literal notations in ES6?**

ES6 allows declaring object literals by providing shorthand syntax for initializing properties from variables and defining function methods. It also enables the ability to have computed property keys in an object literal definition.

https://miro.medium.com/max/60/1*CKPyW4XHFB6zczUGEcNKpg.png?q=20

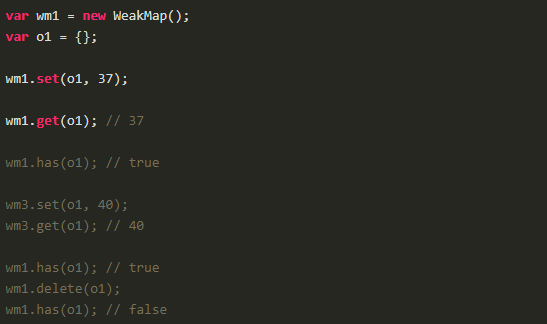


**Q6. What is WeakMap in ES6?**

The WeakMap is same as Map where it is a collection of key/value pairs. But in WeakMap, the **keys must be objects** and the values can be arbitrary values. The object references in the keys are held weakly, meaning that they are a target of garbage collection (GC) if there is no other reference to the object anymore. The WeakMap API is the same as the Map API.

However, One difference to Map objects is that WeakMap keys are not enumerable. And there are no methods giving us a list of keys. If they were, the list would depend on the state of garbage collection, introducing non-determinism. If we want to have a list of keys, we should use a Map.

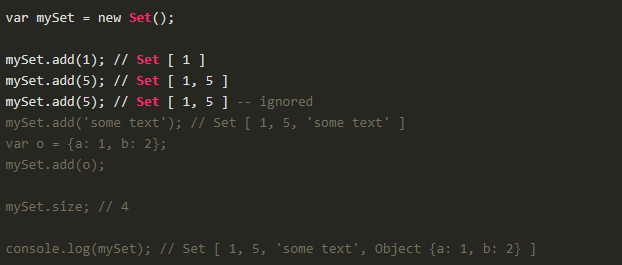
https://miro.medium.com/max/60/1*pauXMAJgZKJd-qJ5cE2a0A.png?q=20



**Q7. What is Set?**

Set objects are collections of unique values. Duplicate values are ignored, as the collection must have all unique values. The values can be primitive types or object references.

https://miro.medium.com/max/60/1*1iUpvH_7Q5F6QmQn9kVHAw.png?q=20

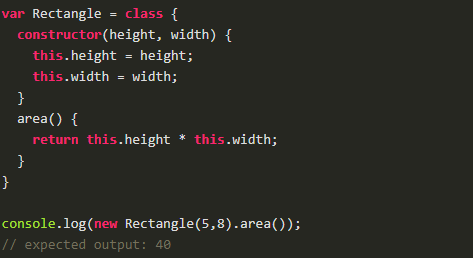


Also, NaN and undefined can also be stored in a Set. NaN is considered the same as NaN (even though NaN !== NaN).

**Q8. What is Class expression?**

The Class expression is one way to define a class in ES6. Similar to function expressions, class expressions can be named or unnamed. If named, the name of the class is local to the class body only. JavaScript classes use prototype-based inheritance.

https://miro.medium.com/max/60/1*nZDGrAliSc2cvTTdTeJOKA.png?q=20



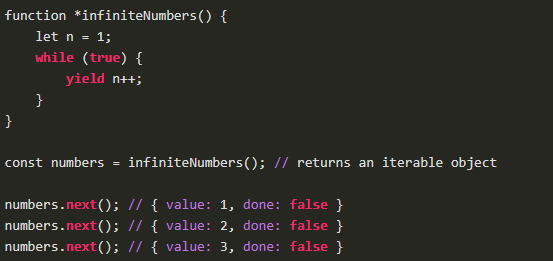
A class expression has a similar syntax to a class statement (declaration). However, with class expressions, you are able to omit the class name (“binding identifier”), which you can’t with class statements. Additionally, class expressions allow you to redefine/re-declare classes and don’t throw any type errors like class declaration. The constructor property is optional. And, typeof the classes generated using this keyword will always be “function”.

**Q9. What is Generator function?**

Generator functions are a new feature in ES6 that allow a function to generate many values over time by returning an object which can be iterated over to pull values from the function one value at a time.

A generator function returns an **iterable object** when it’s called. It is written using the new \* syntax as well as the new yield keyword introduced in ES6.

https://miro.medium.com/max/60/1*ljit-vBI8XK3d3VkVcOL6g.png?q=20



Each time yield is called, the yielded value becomes the next value in the sequence. Also, note that generators compute their yielded values on demand, which allows them to efficiently represent sequences that are expensive to compute, or even infinite sequences.

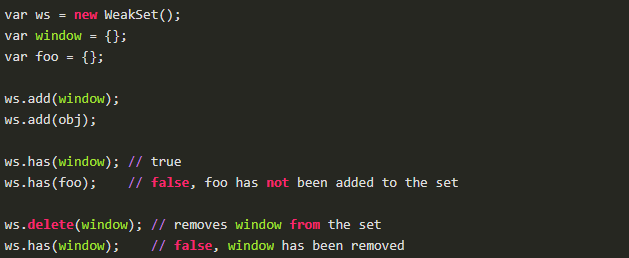
**Q10. What is WeakSet?**

The WeakSet object lets you store weakly held objects in a collection.

WeakSet objects are collections of objects. An object in the WeakSet may occur only once; it is unique in the WeakSet's collection. The main differences to the Set object are:

* Unlike Set, WeakSets are collections of **objects only** and not of arbitrary values of any type.
* The WeakSet is weak: References to objects in the collection are held weakly. If there is no other reference to an object stored in the WeakSet, they can be garbage collected. That also means that there is no list of current objects stored in the collection. WeakSets are not enumerable.

https://miro.medium.com/max/60/1*vYy-hn3o9wvt1-eVhnKvXA.png?q=20



**1. What is ES6?**

**Es6** or ECMASCRIPT 2015 is sixth major release of ECMAScript language which comes with a lot of new features and syntax for writing web applications in [Javascript](https://www.onlineinterviewquestions.com/advanced-javascript-interview-questions/). As currently, not all browsers support ES6, they support pre-versions of ES6. So to write web applications in ES6 that will support all Browsers we needed tools like Babel and Webpack.

**2. Explain Destructuring Assignment in ES6?**

**Destructing assignment** in another improvement in Es6. It allows us to extract data from array and objects into separate variables.

Example

let full\_name =['John','Deo'];

let [first\_name,last\_name]=full\_name;

console.log(first\_name,last\_name);

// outputs John Deo

Another example

let c=[100,200,330,400];

let [a,...b]=c;

console.log(a,b);

// outputs 100 [200, 330, 400]

**3. What are template literals in Es6?**

**Template literals** are the string with embedded code and variables inside. Template literal allows concatenation and interpolation in much more comprehensive and clear in comparison with prior versions of ECMAScript.

Let see an example of concatenating a string in JavaScript.

var a="Hello";

var b="John";

var c = a+ " " + b;

Console.log(c); //outputs Hello John;

In ES6 concatenation and interpolation is done by backtick “ in a single line. To interpolate a variable simply put in to {} braces forwarded by $ sign.>/p>

// In ES6

let a="Hello";

let b="John";

let c=`${a} ${b}`;

console.log(c); //outputs Hello John;

**4. Explain Constants in Es6?**

**Constants** also are known as immutable variables are a special type of variables whose content is not changed. In Es6 a constant is defined using const keyword. Constants in Es6 enable protection to overwrite a variable value, improve performance and helps programmers to write readable and cleaner code.

**Example**

In Es6

const WEBSITE\_URL = "http://www.abc.com";

WEBSITE\_URL="new url"; // generate an error;

console.log(WEBSITE\_URL);

In prior version of Es6

// and only in global context and not in a block scope

Object.defineProperty(typeof global === "object" ? global : window, "WEBSITE\_URL", {

value: "http://www.abc.com", enumerable: true,

writable: false,

configurable: false

});

console.log(WEBSITE\_URL);

Also, Read [Best Node JS Interview Questions](https://www.onlineinterviewquestions.com/node-js-interview-questions/)

**5. What is Set in ES6?**

Set is a collection of unique values. The values could be also primitives or object references.

**Creating a Set in Javascript**

let set = new Set();

set.add(1);

set.add('1');

set.add({ key: 'value' });

console.log(set); // Set {1, '1', Object {key: 'value'}}

**6. Explain Generator function in ES6?**

Generators are functions that can be exited and later re-entered. Their context (variable bindings) will be saved across re-entrances. A function keyword followed by an asterisk defines a generator function, which returns a Generator object.

Generator Function Example.

function\* generator(i) {

yield i;

yield i + 10;

}

var gen = generator(10);

console.log(gen.next().value);

// expected output: 10

console.log(gen.next().value);

**Further reading:** https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/function\*

**7. Explain WeakMap in ES6?**

WeakMaps provide a way to extend objects from the outside without interfering with garbage collection. Whenever you want to extend an object but can't because it is sealed - or from an external source - a WeakMap can be applied. WeakMap was also introduced by ES6 in 2015

Further reading : https://stackoverflow.com/questions/29413222/what-are-the-actual-uses-of-es6-weakmap

**8. What is the use of let & const in JavaScript?**

In modern javascript let & const are different ways of creating variables. Earlier in javascript, we use the var keyword for creating variables. let & const keyword is introduced in version [ES6](https://www.onlineinterviewquestions.com/es6-interview-questions/)with the vision of creating two different types of variables in javascript one is immutable and other is mutable.  
**const:** It is used to create an immutable variable. Immutable variables are variables whose value is never changed in the complete life cycle of the program.  
**let:** let is used to create a mutable variable. Mutable variables are normal variables like var that can be changed any number of time.

**9. List some new features of ES6**

New Features in ES6.

1. Support for constants (also known as “immutable variables”)
2. Block-Scope support for both variables, constants, functions
3. Arrow Functions
4. Extended Parameter Handling
5. Template Literals
6. Extended Literals
7. Enhanced Regular Expression
8. Enhanced Object Properties
9. Destructuring Assignment
10. Modules, Classes, Iterators, Generators
11. Support for Map/Set & WeakMap/WeakSet
12. Promises, Meta-Programming ,Internationalization & Localization

Read More from http://es6-features.org/

**10. What is Babel?**

Babel is one of the most popular JavaScript transpilers and becomes the industry standard. It allows us to write ES6 code and convert it back in pre-Es6 JavaScript that browser supports.

For example look the below code snippet.  
**In ES6 (ECMASCRIPT 2015)**

const PI = 3.141593;

PI > 3.0 ;

export{PI};

**In ES5 after conversion**

"use strict";

Object.defineProperty(exports, "\_\_esModule", {

value: true

});

var PI = 3.141593;

PI > 3.0;

exports.PI = PI;

**11. List steps to install Babel?**

**Installation:** In order to install Babel, you require node.js and NPM. Make sure [Node.js](https://www.onlineinterviewquestions.com/node-js-interview-questions/) is installed on your server.

To check node installed or not run below commands on your terminal.

node -v

npm -v

Installing Babel : We can install Babel CLI locally by running below command on terminal.

npm install --save-dev babel-cli

**12. What is Webpack?**

Webpack allows you to run an environment that hosts babel. Webpack is opensource javascript module bundler which takes modules with dependencies and generates static assets representing those modules.

**13. List benefits of using Webpack?**

Benefits of using Webpack.

1. It bundles your multiple modules and packs it into a single .js file.
2. It comes with integrated dev server. A small [express](https://www.onlineinterviewquestions.com/node-js-interview-questions/)app for local development. You simply include one Javascript tag pointed to the server, like localhost:8080/assets/bundle.js, and get live code updating and asset management for free.

**14. What is Spread Operator in ES6?**

Spread Operator provides a new way to manipulate array and objects in Es6.A Spread operator is represented by … followed by the variable name.

Example :

let a =[7,8,9];

let b=[1,2,3,...a,10];

console.log(b); // [1,2,3,7,8,9,10]

So spread operator spreads the contents of variable a and concatenates it in b.

Another Example

function print(...z){

console.log(z);

}

print(1,2,3,4);//[1,2,3,4]

**15. How to create a Javascript class in ES6?**

In Es6 you can create a class using the Class keyword.Below is sample javascript class in ES6.

class User{

constructor(name,age) {

this.name = name;

this.age = age;

}

getData() {

console.log(this.name + " is " + this.age + " years old !");

}

}

var user = new User("foo", 7);

s1.getData();

**Part 1 – ES6 Interview Questions (Basic)**

This first part covers basic ES6 Interview Questions and Answers

**Q1.Define ES6 and mention the new features of ES6?**

Answer:  
Refer the introduction part for definition of ES6. Below are the new features listed:

* Constants (Immutable variables)
* Scoping
* Arrow functions
* Extended parameter handling
* Template literals
* Extended literals
* Modules
* Classes
* Enhanced Regular expressions
* Enhanced object properties.
* Destructuring Assignment
* Symbol Type
* Iterators
* Generator
* Map/Set & WeakMap/WeakSet
* Typed Arrays
* Built-in Methods
* Promises
* Metaprogramming
* Internationalization and Localization.

**Q2.How do you use ES6 or the best way to use ES6 in a project?**

 Popular Course in this category

[**Software Development Course - All in One Bundle**600+ Online Courses | 3000+ Hours | Verifiable Certificates | Lifetime Access   
4.6 (3,144 ratings)](https://www.educba.com/software-development/courses/software-development-course/?btnz=edu-blg-inline-banner1)

[Course Price   
₹19999 ~~₹150000~~   
**View Course**](https://www.educba.com/software-development/courses/software-development-course/?btnz=edu-blg-inline-banner1)

Related Courses

[Programming Languages Course](https://www.educba.com/software-development/courses/programming-languages-course/?btnz=edu-blg-inline-banner1)[C Programming Course](https://www.educba.com/software-development/courses/c-programming-course/?btnz=edu-blg-inline-banner1)[Selenium Training Certification](https://www.educba.com/software-development/courses/selenium-training-certification/?btnz=edu-blg-inline-banner1)

Answer:  
As it is mentioned that ES6 is not supported by all the browsers, so to use ES6 script we need to convert into the ES5 script which is supported by all the browsers. To convert into a pre-ES6 script, we required transpilers like Babel. Babel is a popular javascript transpiler used for this purpose, which allows ES-6 code to convert into an ES-5 script to support by all the browsers.

Let us move to the next ES6 Interview Questions

**Q3.What are Constants in ES6?**

Answer:  
Constants are also referred to as Immutable variable. It means that the value of a constant variable cannot be changed. The value that has been assigned at the time of declaration remains unchanged. For e.g. const X= 5.0, here the value of X remains 5 every time and it cannot be changed.

**Q4.What are Block Scoped variables and functions?**

Answer:  
This is the common ES6 Interview Questions which is asked in an interview. The variables and function that are defined as indefinite blocks. It means these can be used where the variables and functions are defined or declared. If we have declared variable and function in any function block then their scope will be limit to that function only, they cannot be accessible outside the block/function. ‘Const’ keyword cannot change the value of a variable. ‘let’ keyword allows variable value to be re-assigned, it can be in for loop or arrays.

**Q5.Explain briefly about Arrow functions?**

Answer:  
Arrow functions support expressions bodies and statement bodies which returns the value of an expression and makes the syntax more expressive. Arrow functions have lexical ‘this’ feature as well. Arrow (=>) is used as part of the syntax. Lexical this is declared or defined where the function is written. It comes under the umbrella of lexical scope; lexical scope has access to variables which are in its parent scope.

**Part 2 – ES6 Interview Questions (Advanced)**

Let us now have a look at the advanced ES6 Interview Questions and Answers.

**Q6.Explain about Webpack and benefits of using Webpack?**

Answer:  
Webpack is used to bundle javascript files that can be used in a browser. Webpack processes the application and builds a dependency graph to map each module of the project requirement and generated the bundles. It allows you to run that environment which has been hosted babel. The advantages of using web pack are that it bundles multiple modules and packs into single JavaScript file. It integrated the dev server which helps in updating code and asset management.

**Q7.Explain about Default parameter values, Rest parameter, Spread operator?**

Answer:  
Default parameter values are used to initialize the functions with default values. The value of a parameter can be anything like null value, number or function.  
Rest parameter is used to retrieve all the arguments to invoke the function. It means we can push the items of different categories separately. Rest parameter uses the rest parameter to combine parameters into single array parameter.  
A spread operator is donated by … and then the variable name has been provided. E.g. ‘…X’ syntax of spread operator. It has been used to manipulate objects and array in ES6 and to copy the enumerable properties from one object to another.

Let us move to the next ES6 Interview Questions.

**Q8.Explain about Internationalization and localization?**

Answer:  
These are the APIs which are standard API of JavaScript that helps in different tasks like collation, Number formatting, Currency formatting, Date and time formatting.  
**Collation: –** It is used for searching within a set of strings and sorting a set of strings. It is parameterized by locale and aware of Unicode.  
**Number Formatting:** Numbers can be formatted with localized separators and digit grouping. The other things that include are style formatting, numbering system, percent, and precision.  
**Currency formatting:** Numbers can be formatted mainly with currency symbol, with localized separators and digit grouping.  
**Date and time formatting:** it has been formatted with localized separators and ordering. The format can be short, long and other parameters like locale and time zone.

**Q9.What is a Destructuring assignment and explain in brief?**

Answer:  
This is the frequently asked ES6 Interview Questions which is asked in an interview. Destructuring assignment is used to bind the set of variable to the corresponding values. It mainly refers to the use of patterns to extract the parts of an object. A destructuring assignment has different forms like array matching, object matching, shorthand notation, object matching, deep matching, object and array matching, default values, parameter context matching and fail-soft *destructuring.* Some are explained as:  
Array matching/object matching, shorthand notation/ object matching, deep matching: – It is intuitive and flexible of arrays into individual variables during an assignment.  
Object and Array matching: – it is simple and defined default values for destructuring of objects and arrays.

**Q10.Explain briefly about classes, modules and proxies?**

Answer:  
Classes are based on the [OOP style that is object-oriented programming](https://www.educba.com/functional-programming-vs-oop/). The class declaration makes the patterns easier to use. It supports inheritance, base class access, static methods, and constructors.  
**Modules:** it defines the patterns from popular javascript module loaders. It supports for exporting or importing the values from or to modules without the global namespace. It supports marking the value as the default exported value and max-min values.  
**Proxies:** It enables object creation with a wide variety of behaviors available to host objects. It can be used for logging, profiling, etc.

Q1: Could you explain the difference between ES5 and ES6

Topic: **JavaScript**  
Difficulty: ⭐⭐⭐

* **ECMAScript 5 (ES5)**: The 5th edition of ECMAScript, standardized in 2009. This standard has been implemented fairly completely in all modern browsers
* **ECMAScript 6 (ES6)/ ECMAScript 2015 (ES2015)**: The 6th edition of ECMAScript, standardized in 2015. This standard has been partially implemented in most modern browsers.

Here are some key differences between ES5 and ES6:

* **Arrow functions** & **string interpolation**:  
  Consider:
* const greetings = (name) => {
* return `hello ${name}`;
* }

and even:

const greetings = name => `hello ${name}`;

* **Const**.   
  Const works like a constant in other languages in many ways but there are some caveats. Const stands for ‘constant reference’ to a value. So with const, you can actually mutate the properties of an object being referenced by the variable. You just can’t change the reference itself.
* const NAMES = [];
* NAMES.push("Jim");
* console.log(NAMES.length === 1); // true
* NAMES = ["Steve", "John"]; // error
* **Block-scoped variables**.  
  The new ES6 keyword let allows developers to scope variables at the block level. Let doesn’t hoist in the same way var does.
* **Default parameter values** Default parameters allow us to initialize functions with default values. A default is used when an argument is either omitted or undefined — meaning null is a valid value.
* // Basic syntax
* function multiply (a, b = 2) {
* return a \* b;
* }
* multiply(5); // 10
* **Class Definition and Inheritance**  
  ES6 introduces language support for classes (class keyword), constructors (constructor keyword), and the extend keyword for inheritance.
* **for-of operator**  
  The for...of statement creates a loop iterating over iterable objects.
* **Spread Operator** For objects merging
* const obj1 = { a: 1, b: 2 }
* const obj2 = { a: 2, c: 3, d: 4}
* const obj3 = {...obj1, ...obj2}
* **Promises**  
  Promises provide a mechanism to handle the results and errors from asynchronous operations. You can accomplish the same thing with callbacks, but promises provide improved readability via method chaining and succinct error handling.
* const isGreater = (a, b) => {
* return new Promise ((resolve, reject) => {
* if(a > b) {
* resolve(true)
* } else {
* reject(false)
* }
* })
* }
* isGreater(1, 2)
* .then(result => {
* console.log('greater')
* })
* .catch(result => {
* console.log('smaller')
* })
* **Modules exporting & importing** Consider module exporting:
* const myModule = { x: 1, y: () => { console.log('This is ES5') }}
* export default myModule;

and importing:

import myModule from './myModule';

🔗 **Source:** FullStack.Cafe  
💡 **Don't miss:** [116 More JavaScript Interview Questions & Answers](https://www.fullstack.cafe/JavaScript)

Q2: What is IIFEs (Immediately Invoked Function Expressions)?

Topic: **JavaScript**  
Difficulty: ⭐⭐⭐

It’s an Immediately-Invoked Function Expression, or IIFE for short. It executes immediately after it’s created:

(function IIFE(){

console.log( "Hello!" );

})();

// "Hello!"

This pattern is often used when trying to avoid polluting the global namespace, because all the variables used inside the IIFE (like in any other normal function) are not visible outside its scope.

🔗 **Source:** stackoverflow.com  
💡 **Don't miss:** [116 More JavaScript Interview Questions & Answers](https://www.fullstack.cafe/JavaScript)

Q3: When should I use Arrow functions in ES6?

Topic: **JavaScript**  
Difficulty: ⭐⭐⭐

I'm now using the following rule of thumb for functions in ES6 and beyond:

* Use function in the global scope and for Object.prototype properties.
* Use class for object constructors.
* Use => everywhere else.

Why use arrow functions almost everywhere?

* **Scope safety**: When arrow functions are used consistently, everything is guaranteed to use the same thisObject as the root. If even a single standard function callback is mixed in with a bunch of arrow functions there's a chance the scope will become messed up.
* **Compactness**: Arrow functions are easier to read and write. (This may seem opinionated so I will give a few examples further on).
* **Clarity**: When almost everything is an arrow function, any regular function immediately sticks out for defining the scope. A developer can always look up the next-higher function statement to see what the thisObject is.

🔗 **Source:** stackoverflow.com  
💡 **Don't miss:** [116 More JavaScript Interview Questions & Answers](https://www.fullstack.cafe/JavaScript)

Q4: What is the motivation for bringing Symbols to ES6?

Topic: **JavaScript**  
Difficulty: ⭐⭐⭐

Symbols are a new, special kind of object that can be used as a unique property name in objects. Using Symbol instead of string's allows different modules to create properties that don't conflict with one another. Symbols can also be made private, so that their properties can't be accessed by anyone who doesn't already have direct access to the Symbol.

Symbols are a new **primitive**. Just like the number, string, and boolean primitives, Symbol have a function which can be used to create them. Unlike the other primitives, Symbols do not have a literal syntax (e.g how string have '') - the only way to create them is with the Symbol constructor in the following way:

let symbol = Symbol();

In reality, Symbol's are just a slightly different way to attach properties to an object - you could easily provide the well-known Symbols as standard methods, just like Object.prototype.hasOwnProperty which appears in everything that inherits from Object.

🔗 **Source:** stackoverflow.com  
💡 **Don't miss:** [116 More JavaScript Interview Questions & Answers](https://www.fullstack.cafe/JavaScript)

|  |  |
| --- | --- |
| All JavaScript Q&A |  |

Q5: What are the benefits of using spread syntax in ES6 and how is it different from rest syntax?

Topic: **JavaScript**  
Difficulty: ⭐⭐⭐

ES6's spread syntax is very useful when coding in a functional paradigm as we can easily create copies of arrays or objects without resorting to Object.create, slice, or a library function. This language feature is used often in Redux and rx.js projects.

function putDookieInAnyArray(arr) {

return [...arr, 'dookie'];

}

const result = putDookieInAnyArray(['I', 'really', "don't", 'like']); // ["I", "really", "don't", "like", "dookie"]

const person = {

name: 'Todd',

age: 29,

};

const copyOfTodd = { ...person };

ES6's rest syntax offers a shorthand for including an arbitrary number of arguments to be passed to a function. It is like an inverse of the spread syntax, taking data and stuffing it into an array rather than unpacking an array of data, and it works in function arguments, as well as in array and object destructuring assignments.

function addFiveToABunchOfNumbers(...numbers) {

return numbers.map(x => x + 5);

}

const result = addFiveToABunchOfNumbers(4, 5, 6, 7, 8, 9, 10); // [9, 10, 11, 12, 13, 14, 15]

const [a, b, ...rest] = [1, 2, 3, 4]; // a: 1, b: 2, rest: [3, 4]

const { e, f, ...others } = {

e: 1,

f: 2,

g: 3,

h: 4,

}; // e: 1, f: 2, others: { g: 3, h: 4 }

🔗 **Source:** github.com/yangshun  
💡 **Don't miss:** [116 More JavaScript Interview Questions & Answers](https://www.fullstack.cafe/JavaScript)

|  |  |
| --- | --- |
| Create Full Stack Resume |  |

Q6: What are the differences between ES6 class and ES5 function constructors?

Topic: **JavaScript**  
Difficulty: ⭐⭐⭐

Let's first look at example of each:

// ES5 Function Constructor

function Person(name) {

this.name = name;

}

// ES6 Class

class Person {

constructor(name) {

this.name = name;

}

}

For simple constructors, they look pretty similar.

The main difference in the constructor comes when using inheritance. If we want to create a Student class that subclasses Person and add a studentIdfield, this is what we have to do in addition to the above.

// ES5 Function Constructor

function Student(name, studentId) {

// Call constructor of superclass to initialize superclass-derived members.

Person.call(this, name);

// Initialize subclass's own members.

this.studentId = studentId;

}

Student.prototype = Object.create(Person.prototype);

Student.prototype.constructor = Student;

// ES6 Class

class Student extends Person {

constructor(name, studentId) {

super(name);

this.studentId = studentId;

}

}

It's much more verbose to use inheritance in ES5 and the ES6 version is easier to understand and remember.

🔗 **Source:** github.com/yangshun  
💡 **Don't miss:** [116 More JavaScript Interview Questions & Answers](https://www.fullstack.cafe/JavaScript)

|  |  |
| --- | --- |
| offline\_bolt | Trending Dev Jobs |

Hey Team Leads! **Hiring a new Dev for your Team?** Let's repost your JD here![Post a Job](javascript:;)

Q7: What's the difference between .call and .apply?

Topic: **JavaScript**  
Difficulty: ⭐⭐⭐

Both .call and .apply are used to invoke functions and the first parameter will be used as the value of this within the function. However, .call takes in comma-separated arguments as the next arguments while .apply takes in an array of arguments as the next argument. An easy way to remember this is C for call and comma-separated and A for apply and an array of arguments.

function add(a, b) {

return a + b;

}

console.log(add.call(null, 1, 2)); // 3

console.log(add.apply(null, [1, 2])); // 3

🔗 **Source:** github.com/yangshun  
💡 **Don't miss:** [116 More JavaScript Interview Questions & Answers](https://www.fullstack.cafe/JavaScript)

|  |  |
| --- | --- |
| All JavaScript Q&A |  |

Q8: Why should we use ES6 classes?

Topic: **JavaScript**  
Difficulty: ⭐⭐⭐

Some reasons you might choose to use **Classes**:

* The syntax is simpler and less error-prone.
* It's **much** easier (and again, less error-prone) to set up inheritance hierarchies using the new syntax than with the old.
* class defends you from the common error of failing to use new with the constructor function (by having the constructor throw an exception if thisisn't a valid object for the constructor).
* Calling the parent prototype's version of a method is much simpler with the new syntax than the old (super.method() instead of ParentConstructor.prototype.method.call(this) or Object.getPrototypeOf(Object.getPrototypeOf(this)).method.call(this)).

Consider:

// \*\*ES5\*\*

var Person = function(first, last) {

if (!(this instanceof Person)) {

throw new Error("Person is a constructor function, use new with it");

}

this.first = first;

this.last = last;

};

Person.prototype.personMethod = function() {

return "Result from personMethod: this.first = " + this.first + ", this.last = " + this.last;

};

var Employee = function(first, last, position) {

if (!(this instanceof Employee)) {

throw new Error("Employee is a constructor function, use new with it");

}

Person.call(this, first, last);

this.position = position;

};

Employee.prototype = Object.create(Person.prototype);

Employee.prototype.constructor = Employee;

Employee.prototype.personMethod = function() {

var result = Person.prototype.personMethod.call(this);

return result + ", this.position = " + this.position;

};

Employee.prototype.employeeMethod = function() {

// ...

};

And the same with ES6 classes:

// \*\*\*ES2015+\*\*

class Person {

constructor(first, last) {

this.first = first;

this.last = last;

}

personMethod() {

// ...

}

}

class Employee extends Person {

constructor(first, last, position) {

super(first, last);

this.position = position;

}

employeeMethod() {

// ...

}

}

🔗 **Source:** stackoverflow.com  
💡 **Don't miss:** [116 More JavaScript Interview Questions & Answers](https://www.fullstack.cafe/JavaScript)

Q9: What is the preferred syntax for defining enums in JavaScript?

Topic: **JavaScript**  
Difficulty: ⭐⭐⭐

Since 1.8.5 it's possible to seal and freeze the object, so define the above as:

var DaysEnum = Object.freeze({

"monday": 1,

"tuesday": 2,

"wednesday": 3,

...

})

or

var DaysEnum = {

"monday": 1,

"tuesday": 2,

"wednesday": 3,

...

}

Object.freeze(DaysEnum)

and voila! JS enums.

However, this doesn't prevent you from assigning an undesired value to a variable, which is often the main goal of enums:

let day = DaysEnum.tuesday

day = 298832342 // goes through without any errors

🔗 **Source:** stackoverflow.com  
💡 **Don't miss:** [116 More JavaScript Interview Questions & Answers](https://www.fullstack.cafe/JavaScript)

|  |  |
| --- | --- |
| Create Full Stack Resume |  |

Q10: Explain the difference between Object.freeze() vs const

Topic: **JavaScript**  
Difficulty: ⭐⭐⭐

const and Object.freeze are two completely different things.

* const applies to **bindings** ("variables"). It creates an immutable binding, i.e. you cannot assign a new value to the binding.
* const person = {
* name: "Leonardo"
* };
* let animal = {
* species: "snake"
* };
* person = animal; // ERROR "person" is read-only
* Object.freeze works on **values**, and more specifically, *object values*. It makes an object immutable, i.e. you cannot change its properties.
* let person = {
* name: "Leonardo"
* };
* let animal = {
* species: "snake"
* };
* Object.freeze(person);
* person.name = "Lima"; //TypeError: Cannot assign to read only property 'name' of object
* console.log(person);

🔗 **Source:** stackoverflow.com  
💡 **Don't miss:** [116 More JavaScript Interview Questions & Answers](https://www.fullstack.cafe/JavaScript)

|  |  |
| --- | --- |
| All JavaScript Q&A |  |

Q11: What is generator in JS?

Topic: **JavaScript**  
Difficulty: ⭐⭐⭐

Generators are functions which can be exited and later re-entered. Their context (variable bindings) will be saved across re-entrances. Generator functions are written using the function\* syntax. When called initially, generator functions do not execute any of their code, instead returning a type of iterator called a Generator. When a value is consumed by calling the generator's next method, the Generator function executes until it encounters the yield keyword.

The function can be called as many times as desired and returns a new Generator each time, however each Generator may only be iterated once.

function\* makeRangeIterator(start = 0, end = Infinity, step = 1) {

let iterationCount = 0;

for (let i = start; i < end; i += step) {

iterationCount++;

yield i;

}

return iterationCount;

}

1. **Question 1. What Is Es6?**

**Answer :**

Es6 or ECMASCRIPT 2015 is sixth major release of ECMAScript language which comes with a lot of new features and syntax for writing web applications in javascript. As currently, not all browsers support ES6, they support pre-versions of ES6.SO to write web applications in ES6 that will support all Browsers we needed tools like Babel and Webpack.

1. **Question 2. List Some New Features Of Es6?**

**Answer :**

**New Features in ES6:**

* 1. Support for constants (also known as “immutable variables”)
  2. Block-Scope support for both variables, constants, functions
  3. Arrow Functions
  4. Extended Parameter Handling
  5. Template Literals
  6. Extended Literals
  7. Enhanced Regular Expression
  8. Enhanced Object Properties
  9. Destructuring Assignment
  10. Modules, Classes, Iterators, Generators
  11. Support for Map/Set & Weak Map/Weak Set
  12. Promises, Meta-Programming, Internationalization & Localization.

[Shell Scripting Interview Questions](https://www.wisdomjobs.com/e-university/shell-scripting-interview-questions.html)

* **Question 3. What Is Babel?**

**Answer :**

Babel is the one of the most popular javascript transpiler and becomes the industry standard. It allows us to write ES6 code and convert it back in pre-Es6 javascript that browser supports.

For example look the below code snippet.

In ES6 (ECMASCRIPT 2015)

const PI = 3.141593;

PI > 3.0;

export {PI};

In ES5 after conversion

"use strict";

Object.defineProperty (exports, "\_\_esModule", {

  Value: true

});

var PI = 3.141593;

PI > 3.0;

exports. I = PI;

* **Question 4. List Steps To Install Babel?**

**Answer :**

Installation

In order to install Babel, you require node.js and NPM. Make sure Node.js is installed on your server.

To check node installed or not run below commands on your terminal.

node -v

npm -v

Installing Babel

We can install Babel CLI locally by running below command on terminal.

npm install --save-dev babel-cli

[Shell Scripting Tutorial](https://www.wisdomjobs.com/e-university/shell-scripting-tutorial-174.html)

* **Question 5. What Is Webpack?**

**Answer :**

Webpack allow you to run an environment that hosts babel.Webpack is open source javascript module bundler which takes modules with dependencies and generates static assets representing those modules.

[Java Script Interview Questions](https://www.wisdomjobs.com/e-university/java-script-interview-questions.html)

* **Question 6. List Benefits Of Using Webpack?**

**Answer :**

**Benefits of using Webpack:**

* 1. It bundles your multiple modules and packs it into a single .js file.
  2. It comes with integrated dev server. A small express app for local development.You simply include one Javascript tag pointed to the server, like localhost:8080/assets/bundle.js, and get live code updating and asset management for free.
* **Question 7. Explain Constants In Es6?**

**Answer :**

Constants also are known as immutable variables are a special type of variables whose content is not changed. In Es6 a constant is defined using const keyword. Constants in Es6 enable protection to overwrite a variable value, improve performance and helps programmers to write readable and cleaner code.

**Example:**

In Es6

const WEBSITE\_URL = "http://www.abc.com";

WEBSITE\_URL="new url"; // generate an error;

console.log (WEBSITE\_URL);

In prior version of Es6

// and only in global context and not in a block scope

Object.defineProperty(type of global === "object" ? global : window, "WEBSITE\_URL", {

value: "http://www.abc.com", enumerable: true,

writable:     false,

configurable: false

});

console.log (WEBSITE\_URL);

[Java Script Tutorial](https://www.wisdomjobs.com/e-university/java-script-tutorial-209.html) [Node.js Interview Questions](https://www.wisdomjobs.com/e-university/nodedotjs-interview-questions.html)

* **Question 8. What Are Template Literals In Es6?**

**Answer :**

Template literals are the string with embedded code and variables inside. Template literal allows concatenation and interpolation in much more comprehensive and clear in comparison with prior versions of Ecma script.

Let see an example of concatenating a string in javascript.

var a="Hello";

var b="John";

var c = a+ " " + b;

Console.log(c); //outputs Hello John;

In ES6 concatenation and interpolation is done by back tick “ in a single line. To interpolate a variable simply put in to {} braces forwarded by $ sign.>/p>

// In ES6

let a="Hello";

let b="John";

let c=`${a} ${b}`;

console.log(c); //outputs Hello John;

* **Question 9. What Is Spread Operator In Es6?**

**Answer :**

Spread Operator provides a new way to manipulate array and objects in Es6.A Spread operator is represented by … followed by the variable name.

**Example:**

let a =[7,8,9];

let b=[1,2,3,...a,10];

console.log(b); // [1,2,3,7,8,9,10]

So spread operator spreads the contents of variable a and concatenates it in b.

Another Example

function print(...z){

 console.log (z);

}

print(1,2,3,4);//[1,2,3,4]

[Javascript Advanced Interview Questions](https://www.wisdomjobs.com/e-university/javascript-advanced-interview-questions.html)

* **Question 10. Explain Destructuring Assignment In Es6?**

**Answer :**

Destructing assignment in another improvement in Es6. It allows us to extract data from array and objects into separate variables.

**Example:**

let full name =['John','Deo'];

let [first\_name,last\_name]=full name;

console.log (first\_name,last\_name);

// outputs John Deo

Another example

let c=[100,200,330,400];

let [a,...b]=c;

console.log (a,b);

// outputs 100 [200, 330, 400]

[Node.js Tutorial](https://www.wisdomjobs.com/e-university/nodedotjs-tutorial-1161.html)

* **Question 11. How To Create A Javascript Class In Es6?**

**Answer :**

In Es6 you can create a class using the Class keyword. Below is sample javascript class in ES6.

class User{

    constructor(name, age) {

        this.name  = name;

        this. age = age;

    }

   get Data() {

        console.log(this.name + " is " + this. age + " years old !");

    }

}

var user = new User("foo", 7);

s1.getData();

Javascript ES6 Interview Questions: ECMAScript 6 is also known as ES6 and ECMAScript 2015.

**Q:- What Is ES6?**

* ES6 is also known as ECMAScript 6 and ECMAScript 2015.
* ES6 or ECMAScript 2015 is a 6th major release of ECMAScript.
* If your browser doesn't support ES6, then you can write ES6 commpatible code using Babel and Webpack.
* ECMAScript 2018 (ES2018) - ES9 is the latest verison of ECMAScript.

**Q:- What Is Babel?**

Babel is the one of the most popular javascript **transpiler**

It allows us to write ES6 code and convert it back in pre-ES6 javascript that browser supports.

**Q:- What Is Webpack?**

Webpack allow you to run an environment that hosts babel.  
Webpack is open source javascript module bundler which takes modules with dependencies and generates static assets representing those modules.

**Benefits of using Webpack:**

1. It bundles your multiple modules and packs it into a single .js file.
2. It comes with integrated dev server. A small express app for local development.You simply include one Javascript tag pointed to the server, like localhost:8080/assets/bundle.js, and get live code updating and asset management for free.

 You may also like - [**Node Js Interview Questions Answers**](https://www.fullstacktutorials.com/interviews/node-js-interview-questions-29.html)

**Q:- List Some New Features in ES6?**

**New Features in ES6:**

Top 10 ES6 Features that every JavaScript Developer must know.

1. Let and Const Keywords
2. Arrow functions
3. Template Literals
4. Object Literals
5. Default Parameters
6. Destructuring Assignment
7. Rest and Spread Operators
8. Modules, Classes, Iterators, Generators in ES6
9. Promises in ES6
10. Support for Map/Set & WeakMap/WeakSet

**1. Let and Const Keywords:**

**Const Keyword:**

* **const** keyword used to define constants in ES6.
* it is also known as immutable variables.
* The value of a constant cannot change through reassignment, and it can't be redeclared.
* code
* source

1. const MY\_CONST = "Javascript ES6 Interview Questions";
2. console.log(MY\_CONST); *//Javascript ES6 Interview Questions*
3. MY\_CONST = "ES6 Interview Questions";
4. console.log(MY\_CONST); *// Uncaught TypeError: Assignment to constant variable, because we cannot re-assign a new value to a constant*

**Let Keyword:**

ES6 introduces the new **let** keyword that allows us to declare local variables in the scope of a block.

let allows you to declare variables that are limited in scope to the block, statement, or expression on which it is used.

* code
* source

1. let a = 10;
2. funtion testMe(){
3. let a = 20;
4. console.log(a); *// 20*
5. }
6. console.log(a); *// 10*

**2. Arrow Functions:**

* An arrow function has a shorter syntax than a regular function.
* In Arrow function there is no existence of **this** keyword.
* code
* source

1. (param1, param2, …, paramN) => {
2. *//statements or code*
3. }

* code
* source

1. *//in ES5*
2. var sum = function(a, b) {
3. return a + b;
4. };
5. console.log(sum(10,40)); *// 50*
6. ----------------------------
7. *//in ES6*
8. var total = (x, y) => {
9. return (x + y);
10. }
11. console.log(total(20,30)); *// 50*

 You may also like - [**Advanced JavaScript Interview Questions**](https://www.fullstacktutorials.com/interviews/advanced-javascript-interview-questions-and-answers-for-experienced-12.html)

**3. Template literals:**

ES6 introduce a new and easy to use string templates with placeholders for variables.

* Template literals are enclosed by the back-tick (` `).
* You can use multi-line strings.
* You can use a new syntax ${variable\_name} inside of the back-ticked string.
* Before ES6 it was called Template Strings.
* code
* source

1. *//Example-1*
2. var num1 = 10;
3. var num2 = 20;
4. console.log(`The sum of ${num1} and ${num2} is ${num1 + num2}.`);
5. *//Output: The sum of 10 and 20 is 30*
6. *//Example-2*
7. var TempBody = `Hello Users! Your are reading about ES6 Features,
8. In ES6, you are implementing Template Literals.
9. Let me show you the sum of ${num1} and ${num2} is ${num1 + num2}.
10. Hope you have understood the concept of Template Literals.`;
12. console.log(TempBody);

**4. Object literals:**

A JavaScript object literal is a comma-separated list of name-value pairs wrapped in curly braces.

* code
* source

1. var myObject = {
2. title: 'ES6 Interview Questions and Answers',
3. topic: 'Object Literals',
4. status: true,
5. myFunction: () => {
6. *// Block of Code*
7. }
8. };

**5. Default Parameters:**

Default function parameters allow named parameters to be initialized with some default values if no value or undefined is passed.

* code
* source

1. *//3rd parameter has default value zero.*
2. let addNumbers = (a, b, c = 0) => {
3. return a + b + c;
4. }
5. console.log(addNumbers(1, 2, 2)); *//5*
6. console.log(addNumbers(5, 20)); *//25*

**6. Destructuring Assignment**

Destructuring assignment is a special syntax that allows you to "unpack" arrays or objects into a bunch of variables, as sometimes they are more convenient.

* code
* source

1. *// we have an array with the first\_name middle\_name and last\_name*
2. let arr = ["Full", "Stack", "Tutorials"]
3. *// destructuring assignment*
4. let [first\_name, middle\_name, last\_name] = arr;
5. console.log(first\_name); *// Full*
6. console.log(middle\_name); *// Stack*
7. console.log(last\_name); *// Tutorials*

 You may also like - [**React Js Interview Questions**](https://www.fullstacktutorials.com/interviews/react-js-interview-questions-answers-31.html)

**7. REST and Spread operator:**

The spread operator, or the three dots, (…) is a new operator introduced in ES6. It allows the expansion of an iterable (e.g., Array) into its constituent elements.

* code
* source

1. const t = [1, 2, 3];
2. const s = [...t, 4, 5, 6];
3. console.log(s); *// [1, 2, 3, 4, 5, 6];*

**Example:- Remove duplicates items from an array using ES6 Set?**

The Set object lets you store unique values of any type, whether primitive values or object

Here’s a one-liner to remove duplicates from an array. (ES6, of course!)

* code
* source

1. const numbers = [1, 2, 3, 4, 5, 5, 5, 5, 5, 5];
2. function removeDuplicates(array) {
3. return [...new Set(array)];
4. }
5. console.log(removeDuplicates(numbers)); *// [1, 2, 3, 4, 5]*

**Set** was introduced in **ES6** and is similar to the Sets we encounter in Mathematics: They can’t have duplicates. After we have done that, we simply convert the Set back to an array by using the spread operator.

**8. Class In ES6:**

* code
* source

1. class User{
2. constructor(name, age) {
3. this.name = name;
4. this. age = age;
5. }
6. getUserDetails() {
7. console.log(this.name + " is " + this. age + " years old!");
8. }
9. }
10. var userObj = new User("Full Stack Tutorials", 5);
11. console.log(userObj.getUserDetails());
12. *//Output: Full Stack Tutorials is 5 years old!*

**Generators**

A generator is a function that can stop midway and then continue from where it stopped. In short, a generator appears to be a function but it behaves like an iterator.

* code
* source

1. function\* process() {
2. console.log('Start process 1');
3. console.log('Pause process2 until call next()');
4. yield;
5. console.log('Resumed process2');
6. console.log('Pause process3 until call next()');
7. yield;
8. console.log('Resumed process3');
9. console.log('End of the process function');
10. }
11. let \_process = process();

**yield:**

* **yield** keyword simply helps to **pause** and **resume** a function in any time **asynchronously**.
* Additionally it helps to **return value** from a **generator function**.

**Usages:**

* Lazy evaluation
* Infinite sequences
* Asynchronous control flows

**9. Promise:**

A promise is an object which can be returned synchronously from an asynchronous function.

Promise overcome the problem of callback hell.

**Promise States:**

* Fulfilled: onFulfilled() will be called (e.g. resolve() was called).
* Rejected: onRejected() will be called (e.g. reject() was called).
* Pending: initial state, neither fulfilled nor rejected.

**Promise Consumers: then, catch, finally**

* code
* source

1. var promise = new Promise(function(resolve, reject) {
2. const A = "fullstacktutorials";
3. const B = "fullstacktutorials"
4. if(A === B) {
5. resolve();
6. } else {
7. reject();
8. }
9. });
10. promise.
11. then(function () {
12. console.log('Success, Your promise has been resolved successfully');
13. }).
14. catch(function () {
15. console.log('Something went wrong!');
16. });

**10. Map:**

The Map object is a simple key/value map.

Any value (both objects and primitive values) may be used as either a key or a value. Syntax [key, value].

* code
* source

1. hash = new Map()
2. hash.set("FullStackTutorials", "Learn Full Stack Tutorials free Online, Javascript ES6 Interview Questions")
3. hash.set(1, 200);
4. console.log(hash);
5. *//Output: Map(2) {"FullStackTutorials" => "Learn Full Stack Tutorials free Online, Javascript ES6 Interview Questions", 1 => 200}*

**How would you filter an Array of Objects in ES6?**

* code
* source

1. var testArray = [{"name":"Jai", "age": 26},
2. {"name":"Shiv", "age": 22},
3. {"name":"Anuj", "age": 5 },
4. {"name":"Jenny", "age": 20 }];
5. var data = testArray.filter(person => person.age > 18); *//Filters the Array of Objects to condition set*
6. console.log(data);
7. */\**
8. *0:{name: "Jai", age: 26}*
9. *1:{name: "Shiv", age: 22}*
10. *2:{name: "Jenny", age: 20}*
11. *length:3*
12. *\*/*

**Q #1) What is HTML5?**

**Answer:** HTML5 is the latest version of the HyperText Markup Language that can be referred to the WWW (World Wide Web) primary language, this markup language enhances a text file with bits of code, and this code which we can say as “markup” describes the structure of the document.

HTML5 provides some standard features like that of CSS, HTML, JavaScript, and DOM, which in turn will reduce the requirement of external plugins. It’s more markup to replace scripting, better error handling, etc. HTML5 is device independent.

**HTML5 introduces some new features that can be used to change the way of user interaction with documents including:**

* Adding new parsing rules to enhance flexibility.
* Adding New attributes.
* Allow offline editing.
* Support (Web SQL), – A common standard for storing data in SQL databases.
* Support Protocol and MIME handler registration.

**Q #2) What is the difference between HTML and HTML5?**

**Answer: Difference between HTML and HTML5 are given below:**

| **HTML5** | **HTML** |
| --- | --- |
| HTML5 has high-level video and audio support. | High-level video and audio support is not a part of the version and specifications in the previous HTML. |
| Canvas, SVG and other virtual vector graphics are supported in HTML5. | In HTML, if we want to implement vector graphics, that was only possible by using third party library like VML, Silver-light, etc. |
| SVG and MathML can be used in text. | This is not possible in HTML. |
| Web SQL database, application cache and web storage is used as permanent storage. | Browser cache can be used as temporary storage. |
| HTML5 is more mobile friendly. | HTML is less mobile friendly. |
| Doctype declaration is simple and easy. | Doctype declaration is long and complicated.. |
| Allows drag and drop effect. | Does not allow drag and drop effect. |
| Attributes of Async, charset, and ping are available. | These attributes are not available in HTML. |
| HTML5 support javascript to run in the background. | Does not support javascript to run within the web browser. |
| We can draw shapes like rectangle, circle, and triangle in HTML5. | It is not possible to draw shapes like rectangle, circle, triangle etc. |

**Q #3) What is <!DOCTYPE>? What are the different types of <!DOCTYPE> that are available?**

**Answer:** The <!DOCTYPE> declaration provides instruction to the web browser to understand what information it should be display, and the need to start with <!DOCTYPE> declaration. In HTML5, DOCTYPE declaration is very short, and case-insensitive, and <!DOCTYPE html> is written at the top of every HTML5 page.

**The following DOCTYPE are also supported in HTML5:**

* <!DocTYpe html>
* <!dOCtype html>
* <!doctype html>

**There are 3 types of DOCTYPES as mentioned below:**

* Strict Doctype
* Frameset Doctype
* Transitional Doctype

**Q #4) What are the New tags in Media Elements in HTML5?**

**Answer: The new tags in Media Elements in HTML5 are enlisted below**:

* **<audio>:** Apply for multimedia contents like sounds, audio streams or music, embed audio content without the requirement of any additional plug-in like flash player.
* **<video>:** Apply for video content like video streams or movie clip, embed video content etc.
* **<source>:** Apply for multiple media resources in media elements, such as audio, video, picture etc.
* **<embed>**: Apply for an external application or embedded content (a plug-in).
* **<track>:** Apply for text tracks in the media elements such as video or audio. This tag is used for subtitles or caption files while the video media is playing.

**Q #5) What is a tag in HTML5?**

**Answer:** A tag is a special content in HTML5, which is surrounded by an angle bracket (<,>). A slash (/) symbol is used to close the tag after completing the block.

**For Example**

<title> This is my Browser </title>

An Html5 tag is a set of characters that develop a formatted command for a web page. These formatted commands communicate and send the instruction to the Browser.

**Q #6) What is the minimum number of HTML5 tags that are required to create a Web page?**

**Answer:** Minimum 3  HTML5 tags are required to create a Web page, such as (<HEAD>, <BODY>, <HTML>).

**Q #7) What is the importance of Drag and Drop in HTML5?**

**Answer:** Drag and Drop is the most important User Interface concept which makes it easy to grab an object and Drag it at the place you want with the help of a mouse click.

Some common features that are mostly used by Drag and Drop operation include move, link or copy.

We can drag an image using elements, type = **<img draggable = “true”>**, to make an image draggable and set the draggable image attribute to true.

**Q #8) Explain new Form input types in HTML5.**

**Answer: HTML5 has 14 new forms input types:**

* **Date:**This is a Date picker, we can pick a date by using type = **“date”.**
* **Week:**This is a Week picker, we can pick a week by using type = **“week”.**
* **Month:**This is a Month picker, we can pick a month by using type = **“month”.**
* **Time:**This is a Time picker, we can pick the time by using type = **“time”.**
* **Datetime:**This is a combined date and time, we can pick the combination of date and time by using type = **“datetime”.**
* **Datetime-local:**A combined local date and time, we can pick the combination of local date and time using type = **“DateTime-local”.**
* **Email:**Allows one or more Email Addresses, we can enter multiple email addresses using type = **“email”.**
* **Tel:**Allows different phone numbers around the world. A phone number is validated by the client-side. We can enter a phone number using type = **“tel”.**
* **Search:**Allows to search queries by input text. We can enter multiple queries using type = **“search”.**
* **Number:**Allows inserting a numerical value with additional attributes such as min, max. etc., and we can enter multiple numerical values using type = **“number”.**
* **Url:**A url input type, that is used for the web address. In a single url, we can use multiple attributes using type = **“url”.**
* **Color:**Allows to select multiple colors, we can pic multiple color using type = **“color”.**
* **Range:**Allows to insert a numerical value within a specific range, Range is similar to the number but it is much specific. We can enter a numerical value within a range using type = **“range”.**
* **Placeholder:**Allows to display a short hint (usually in a light color) in the input fields, before we enter the value. We can write a short hint in the input field by using type = **“placeholder”.**

**Q #9) What is image map in html5?**

**Answer:** Image maps are a combination of URL and images, where clicking on these images (clickable area of the image) will open different new web pages.

**Two types of image maps are available in HTML5, i.e. client side and server side:**

**The client-side image map** is created by using two elements **<area>** and **<map>**, where the map holds the map information and the area element takes the attributes to define each section of the map. **Server-side image map** created by using **<usemap>** attribute, the usemap attribute is the name of our map.

**Q #10) How do you write a copyright symbol on a web browser page?**

**Answer:** In order to write a copyright symbol, we need to type © or © in an HTML5 file.

**Q #11) How to optimize website assets?**

**Answer:** We need to understand some basic optimization rules, in order to optimize website assets. Initially, we should decrease the download size and make fewer http requests.

**To optimize website assets we can follow the below techniques:**

* File compression
* File concatenation
* CDN Hosting
* Offloading assets
* Re-organizing
* Refining code

**Q #12) What is the use of MathML Element in HTML5?**

**Answer:** The word MathML (Mathematical Markup Language) is a markup language, that is used to show scientific and mathematical expression on the web. MathML is a form of XML (extensible markup language) to describe the Math notation.

We can use **<math>…</math>** tags inside the HTML5 documents for implementing MathML element.

**Example:** print a²+2b+5=0 using HTML5 code.

|  |  |  |
| --- | --- | --- |
| 1 | <!doctype html> | |
| 2 | <html> |  |

|  |  |  |
| --- | --- | --- |
| 3 | <head> |  |
| 4 | <meta charset = “UTF-8”> | |

|  |  |  |
| --- | --- | --- |
| 5 | <title> MathMl Example </title> | |
| 6 | </head> |  |

|  |  |  |
| --- | --- | --- |
| 7 | <body> | |
| 8 |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| 9 | <math xmlns=<a href="http://www.w3.org/1998/Math/MathML">http://www.w3.org/1998/Math/MathML</a>”> | | |
| 10 | | <mrow> |  |

|  |  |
| --- | --- |
| 11 | <mrow> |
| 12 | <msup> |

|  |  |
| --- | --- |
| 13 | <mi> a </mi> |
| 14 | <mn> 2 </mn> |

|  |  |  |
| --- | --- | --- |
| 15 | </msup> |  |
| 16 | <mo> + </mo> | |

|  |  |  |
| --- | --- | --- |
| 17 | <mrow> |  |
| 18 | <mn> 2 </mn> | |

|  |  |  |
| --- | --- | --- |
| 19 | <mo> </mo> |  |
| 20 | <mi> b </mi> | |

|  |  |  |
| --- | --- | --- |
| 21 | </mrow> |  |
| 22 | <mo> + </mo> | |

|  |  |  |
| --- | --- | --- |
| 23 | <mn> 5 </mn> | |
| 24 | </mrow> |  |

|  |  |
| --- | --- |
| 25 | <mo> = </mo> |
| 26 | <mn> 0 </mn> |

|  |  |
| --- | --- |
| 27 | </mrow> |
| 28 | </math> |

|  |  |  |
| --- | --- | --- |
| 29 |  |  |
| 30 | </body> | |

|  |  |
| --- | --- |
| 31 | </html> |

**Note:**If MathML is used by an application that conforms to the Namespace in an XML Recommendation, then the following namespace should be used:

http://www.w3.org/1998/Math/MathML

**Q #13) What are the various formatting tags in HTML5?**

**Answer: HTML5 has some old and new formatting tags as given below:**

* **Marked text:**Represents highlighted text for Reference purposes. We can use **<marks**> tags for text highlight.
* **Deleted text:**Specifies the deleted block of text. We can use **<del>** tags to implement a deleted text.
* **Emphasized text:** Defines the emphasized text. We can use **<em>** tags to implement an emphasized text.
* **Inserted text:**Inserts a block of text into a document. We can use **<ins>** tags to implement an inserted text.
* **Small text:** Display inserted text in a small size. We can use **<small>** tags to implement a small text.
* **Superscript text:**This is a superscripted text. We can use **<sup>** tags to implement a superscript text.
* **Subscript text:** This is a subscripted text. We can use **<sub>** tags to implement a superscript text.

**Q #14) Why do we use HTML5?**

**Answer:** HTML5 supports animation, drawing, audio, video, etc and it easily embeds a video on the web page. It does not require any additional software like Flash for watching videos.

**Some of the important reasons to use HTML5 are given below:**

* Legacy and cross-browser support
* Better interactions
* Smarter storage
* Cleaner code

**Q #15) What is a hyperlink? Does it only apply to text?**

**Answer:** The hyperlink is a link that allows a user to move from one web page to another web page when clicked. Hyperlink concept is used on text and as well as image, and we can convert an image into a link with the help of <a href = “ ”….</a>> tags.

**Example to Create a hyperlink in an image (clickable image) is shown below.**

|  |  |  |
| --- | --- | --- |
| 1 | <!DOCTYPE html> | |
| 2 | <html> |  |

|  |  |  |
| --- | --- | --- |
| 3 | <head> |  |
| 4 | <title> Image Hyperlink Example </title> | |

|  |  |  |
| --- | --- | --- |
| 5 | </head> | |
| 6 | <body> |  |

|  |  |
| --- | --- |
| 7 |  |
| 8 |  |

|  |  |  |  |
| --- | --- | --- | --- |
| 9 | Click the following link | | |
| 10 | |  |  |

|  |  |  |
| --- | --- | --- |
| 11 | <a href = “https://en.wikipedia.org/wiki/HTML5” target = “\_self”> | |
| 12 | <img src = “image.png” alt = “Wiki” border = “0/”> |  |

|  |  |  |
| --- | --- | --- |
| 13 | </a> |  |
| 14 | </body> | |

|  |  |
| --- | --- |
| 15 | </html> |

**Q #16) Explain the concept of web storage in HTML5.**

**Answer:** Web storage provides the facility to store the data of our web applications locally into the user’s browser. It can store up to 10 MB data. Web storage helps to increase the performance of our applications.

**There are two types of web storage that are used to store data locally in HTML5:**

* **Local storage:**This stores the data that will not expire or clear automatically when a user closes or reopens a browser.
* **Session Storage:** This stores data for one session only(i.e, a user who is surfing the internet or website). Once the browser is closed, session data will automatically delete from the web browser.

**Q #17) Explain Geolocation API in HTML5.**

**Answer:** Geolocation API is used to locate a user’s geographical position.

For privacy reasons, the user is asked for permission to report the location information. Use the **navigator.geolocation.get current position()** method to get the user’s position and geographical coordinates (longitude and latitude number).

**Example to Return the latitude and longitude position of a user is shown below.**

|  |  |  |
| --- | --- | --- |
| 1 | <!DOCTYPE html> | |
| 2 | <html> |  |

|  |  |  |
| --- | --- | --- |
| 3 | <body> |  |
| 4 | <p>Click The My Location Button to get your Location.</p> | |

|  |  |  |
| --- | --- | --- |
| 5 | <button onclick="getLocation()"> My Location </button> | |
| 6 | <p id="location"></p> |  |

|  |  |  |
| --- | --- | --- |
| 7 | <script> |  |
| 8 | var x = document.getElementById("location"); | |

|  |  |  |  |
| --- | --- | --- | --- |
| 9 | function getLocation() { | |  |
| 10 | | if (navigator.geolocation) { | |

|  |  |  |
| --- | --- | --- |
| 11 | navigator.geolocation.getCurrentPosition(showPosition); | |
| 12 | } else { |  |

|  |  |  |
| --- | --- | --- |
| 13 | x.innerHTML = "Geolocation is not supported by this browser."; | |
| 14 | } |  |

|  |  |  |
| --- | --- | --- |
| 15 | } |  |
| 16 | function showPosition(position) { | |

|  |  |  |
| --- | --- | --- |
| 17 | x.innerHTML = "Latitude: " + position.coords.latitude + | |
| 18 | "<br>Longitude: " + position.coords.longitude; |  |

|  |  |  |
| --- | --- | --- |
| 19 | } |  |
| 20 | </script> | |

|  |  |
| --- | --- |
| 21 | </body> |
| 22 | </html> |

**Example Explained:**

* Check, if Geolocation is supported or not.
* If Geolocation is supported, run the getCurrentPosition() method. If Geolocation is not supported, display the error message to the user.
* If getCurrentPosition() method is successful, it returns coordinates and functions specified in the parameter.
* The showPosition() function, gets the output – Longitude and Latitude.

**Q #18) Explain HTML5 Graphics.**

**Answer:** HTML5 supports two types of graphics i.e. Canvas and SVG.

**#1) Canvas:**<canvas> element is used to design graphics on the web page, and it has several methods available for drawing circles, boxes, adding images and text. 300 px X 150 px (width X height) is the default pixel size of canvas.

**Example to Draw square Box using canvas element is shown below.**

|  |  |  |
| --- | --- | --- |
| 1 | <!DOCTYPE HTML> | |
| 2 | <html> |  |

|  |  |  |
| --- | --- | --- |
| 3 | <head> |  |
| 4 | <style> | |

|  |  |  |
| --- | --- | --- |
| 5 | #mycanvas{border:2px solid pink;} | |
| 6 | </style> |  |

|  |  |  |
| --- | --- | --- |
| 7 | </head> | |
| 8 | <body> |  |

|  |  |  |  |
| --- | --- | --- | --- |
| 9 | <canvas id = “mycanvas” width = “150” height = “150”></canvas> | | |
| 10 | | </body> |  |

|  |  |
| --- | --- |
| 11 | </html> |

**#2) SVG: <svg>**Scalable Vector Graphics, are mostly used for graphical application and scalable vector type diagrams, such as X, Y coordinate system, two-dimensional and pie charts. This makes it faster and lightweight. SVG follows XML format.

**Example to draw a Rectangle using SVG element is shown below.**

|  |  |  |
| --- | --- | --- |
| 1 | <!DOCTYPE html> | |
| 2 | <html> |  |

|  |  |  |
| --- | --- | --- |
| 3 | <body> |  |
| 4 | <svg width = “400” height = “150”> | |

|  |  |  |
| --- | --- | --- |
| 5 | <rect width = “400” height = “150” style = “fill:rgb(200,220,255);stroke-width:10;stroke:rgb(0,0,0)”/> | |
| 6 | </svg> |  |

|  |  |
| --- | --- |
| 7 | </body> |
| 8 | </html> |

**Q #19) What are the advantages of using HTML5?**

**Answer:** HTML5 is the advanced version of HTML. HTML5 enables to create easier or interactive websites by embedding video, audio, and graphics on the web page.

HTML5 support multimedia technology and graphical content to the web without using any third party plugins.

**Some of the most important features added by HTML5 include:**

* Geolocation
* Offline Application Cache
* Client-side database
* Error Handling
* New Structure and new multimedia elements.
* Browser Support and compatibility.

**Supports Some New Application Programming Interface (API) like:**

* Browser History Management
* Drag and Drop
* 2D drawing on a web page
* Time media playback

**Supported Applications include:**

* Web Workers – JavaScript
* Local File Access
* Application Cache
* Local data storage
* Local SQL databases

**Q #20) How to create a link that will connect to another web browser page when clicked in HTML5?**

**Answer:** Use the **<href>** tag to create hyperlinks, and these are used to connect to another web page. We can create hyperlink by using type = **<a href = “ url of website”> text</a>** tag. When we click the text then, the linked url of the web page will open.

**For Example**, the code below is a link that leads to the Yahoo home page, located at the address https://in.yahoo.com

|  |  |  |
| --- | --- | --- |
| 1 | <!DOCTYPE html> | |
| 2 | <html> |  |

|  |  |  |
| --- | --- | --- |
| 3 | <body> | |
| 4 |  |  |

|  |  |  |
| --- | --- | --- |
| 5 | <h2> Yahoo link </h2> | |
| 6 |  |  |

|  |  |  |
| --- | --- | --- |
| 7 | <a href = “https://in.yahoo.com”> Visit Yahoo Home Page </a> | |
| 8 |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| 9 | </body> | |  |
| 10 | | </html> | |

**Q #21) How many web browsers does HTML5 support?**

**Answer:** Most of the latest version of Apple Safari, Google Chrome, Opera, Internet Explorer, and Mozilla Firefox are supported by HTML5.

**Q #22) What are the frequently useful Api in HTML5?**

**Answer:**

**A list of the most frequently useful API's in HTML5 include:**

* Media API
* Data Transfer API
* Application Cache API
* User Interaction
* History API
* Constraint Validation API
* Command API
* Text Track API

**Q #23) How many tags were removed in HTML5?**

**Answer:**

**The list of the tags that are removed completely in HTML5 include:**

* <frame>
* <frameset>
* <noframes>
* <big>
* <basefront>
* <applet>
* <center>
* <bigcenter>
* <acronym>
* <basefont>
* <dir>
* <tt>
* <font>
* <strike>

**Q #24) Which video and audio formats are used for embedding on web page?**

**Answer: Formats of video and audio that are used for embedding on the web page are given below:**

* **Video:**MPEG4, Ogg, WebM.
* **Audio:** WAV, Ogg Vorbis, MP3.

**Q #25) List out the page structure elements of HTML5.**

**Answer: Page structure elements of HTML5 are given below:**

* **<header>:**Represents the header section and stores the starting information about the web page.
* **<footer>:**Represents the footer section (last portion) of the page.
* **<nav>:**Represents the navigation elements of the HTML page.
* **<article>:**It is a set of information.
* **<section>:**It is a set of instruction that is used inside the article block to define the basic structure of a page.
* **<aside>:**Sidebar content of the page.

**Q #26) Explain some of the common lists to design a web page.**

**Answer: The common lists to design a web page include:**

* Directory list
* Definition list
* Ordered list
* Menu list
* Unordered list

**Different**– Different tags are used to compose each list.

**Q #27) What is the use of output tag in HTML5?**

**Answer:** **<output>** tag is used to represent the different types of output and result.

**Q #28) Which element provides the autocomplete feature in a textbox?**

**Answer:** In HTML5 **<Datalist>** element, provides the autocomplete feature in a textbox.

**Q #29) How to Embed Video and Audio in Html5?**

**Answer:**

**Video:**

**Example** to Embed a video in HTML5 is shown below.

|  |  |  |
| --- | --- | --- |
| 1 | <!DOCTYPE html> | |
| 2 | <html> |  |

|  |  |  |
| --- | --- | --- |
| 3 | <body> |  |
| 4 | <video width = “300” height = “250” controls> | |

|  |  |  |
| --- | --- | --- |
| 5 | <source src = “MyMovie.mp4” type = “video/mp4”> | |
| 6 | </video> |  |

|  |  |
| --- | --- |
| 7 | </body> |
| 8 | </html> |

**Audio:**

**Example** to Embed an Audio in HTML5 is shown below.

|  |  |  |
| --- | --- | --- |
| 1 | <!DOCTYPE html> | |
| 2 | <html> |  |

|  |  |  |
| --- | --- | --- |
| 3 | <body> |  |
| 4 | <audio controls> | |

|  |  |  |
| --- | --- | --- |
| 5 | <source src = “song.mp3” type = “audio/mpeg”> | |
| 6 | </audio> |  |

|  |  |
| --- | --- |
| 7 | </body> |
| 8 | </html> |

**Q #30) What are the migrated tags from HTML4 to HTML5?**

**Answer:**

**List of the migrated tags from HTML4 to HTML5 are given below:**

**Typical HTML4**         **Typical HTML5**

<div id = “footer”>                <footer>  
<div id = “header”>              <header>  
<div id = “menu”>                <nav>  
<div id = “post”>                   <article>  
<div id = “content”>            <section>

**Q #31) What are the part of HTML5 Technologies?**

**Answer: List of the part of HTML5 Technologies are given below:**

* Web Workers
* Web Storage
* SVG
* CSS3
* Server-Sent Events(SSE)
* Microdata
* Web Intents
* Web Sockets
* Offline Application
* Geolocation
* File API
* Web Messaging
* Drag and Drop
* Canvas 2D

**Q #32) What is the difference between SVG and Canvas elements?**

**Answer:**

| **SVG** | **Canvas elements** |
| --- | --- |
| It is Object Model-based. | It is pixel based. |
| Is suitable for using large rendering areas. | Is suitable for using small rendering areas. |
| SVG provides any support for event handlers . | Canvas does not provide any recourse for event handlers. |
| Modification is allowed through script and CSS. | Modification is allowed through script only. |
| SVG has Better scalability | Canvas has poor scalability. |
| SVG is Vector based (composed of shapes). | Canvas is Raster based (composed of a pixel). |
| SVG is not suitable for Games graphics. | Canvas is suitable for games graphics. |
| SVG does not depend on resolution. | Canvas is completely dependent on resolution. |
| SVG is capable for API animation. | Canvas has no any API for animation. |
| SVG is suitable for printing with high quality and any resolution. | Canvas is not suitable for printing high quality and high resolution. |

**Q #33) What is the use of <figure> tag in HTML5?**

**Answer:** The figure tag is used to add an image in the document on a web page.

**Q #34) What is Microdata in HTML5?**

**Answer:** Microdata is a new simple semantic syntax, that is used to add the nested groups of name and value pair of data to documents, that are commonly based on the page content. Microdata is used for new global attributes.

**Q #35) Explain Meta tags.**

**Answer:** Meta tags are used to provide useful information to our web pages.

**Some of the tags include:**

* **Title:** Provides a title to the web page.
* **Style:**Inserts some styles and CSS details to the web page.
* **Link:** Defines the relationship between one page to another page and an external source.

Some Useful Abbreviations

* **XML:**Extensible Markup Language
* **W3C:**World Wide Web Consortium
* **SQL:**Structured Query Language
* **JPEG:**Joint Photographic Expert Group
* **IP:**Internet Protocol
* **HTTP:**Hypertext Transfer Protocol
* **href:**Hypertext Reference
* **FTP:**File Transfer Protocol
* **API:**Application Programming Interface
* **IDE:**Integrated Development Environment
* **WEFT:**Web Embedding Fonts Tool
* **DOM:**Document Object Model
* **Url:**Uniform Resource Locator

**Conclusion**

HTML5 can be considered as the foundation of the web technologies, this is the primary technology that is used to create web pages.

A lot of professionals who are intending to build a career in web technology must learn HTML5. HTML5 is not only the substructure of web technologies, but it is also used to develop mobile applications. Terminologically, HTML5 is not a programing language, rather it’s a markup language.

What is HTML?

Introduced by Tim Berners-Lee, HTML stands for HyperText Markup Language. It is an open source technology, which means it is not owned by anyone in particular and can be used without any cost. HTML provides data to the user in a way that the user can query or lookup for more information just by clicking a highlighted word (a link or image etc.) or by navigating from one HTML page to another. It is made up of tags (<>) which annotate text.

HTML Versions

HTML 1.0: It was the first release but had very limited usage.

HTML 2.0: Released by adding some new features to HTML 1.0, this was the standard application for website development.

HTML 3.0: Due to improper standards of the existing version of HTML,  popular browser developer Netscape Navigator had some of its own tags. This is when HTML 3.0 was introduced.

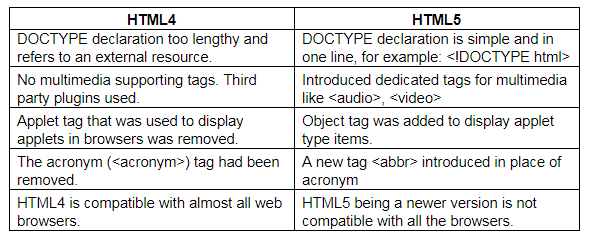
HTML 3.2: This version stopped the non-standardized HTML by introducing World Wide Web Consortium (in short W3C) to keep the language standardized.

HTML 4.0: Created an evolution as the browser support was undertaken by Microsoft in their Internet Explorer (IE).

HTML 5.0: The current version of HTML which was introduced especially to work with any type of devices like mobile, tablets, large monitors etc.

**Explore**[Complete Information About HyperText Markup Language](https://www.greycampus.com/blog/programming/complete-information-about-hypertext-markup-language)

Difference between HTML4 and HTML5



Levels of Interview

When it comes to interviews for web developers, HTML plays a minor role as it is very easy to learn and use. The interview can also last for only one round asking questions about some of the regularly used tags or listing out the differences between HTML versions (4 & 5). Below are some of the commonly asked interview questions for HTML5 web application:

HTML5 Interview Questions and Answers

1. What is the need of mentioning <!DOCTYPE HTML> at the beginning of an HTML page?

The <!DOCTYPE HTML> indicates the version of HTML which in this case is 5. The earlier versions like HTML4 were based on Standard Generalized Markup Language (SGML) and required a Document Type Definition (DTD). HTML5 however, is not based on SGML and doesn’t need a DTD.

2. List out the new Structural Elements in HTML5.

HTML5 introduced few structural elements to define different parts of a web page like article, header, footer, main, mark and many more.

These are sections in which a web page is usually divided into. In HTML4, the same was implemented using div with an id/class name to it.

3. What is the use of the article tag?

It defines space for an article that is independently distributable. For example, A magazine or newspaper article.

4. How can we produce a horizontal line on a webpage?

A horizontal line can be produced on a web page using the <hr> tag. The other point to note about the hr tag is that it doesn’t have an end tag as most of the HTML tags do.

5. Why is the div tag so useful in developing HTML web pages?

The HTML div tag (ie <div>...</div>) is used to create a section of a web page. It can hold another sections or elements in it. It is also possible to hold direct text in it, but not the one that is used.

6. Define the types of HTML Headings.

HTML holds 6 types of headings starting from h1 to h6. The heading one (h1) is the maximum font and h6 is the minimum font.

7. How can we group heading elements?

All the heading tags or elements can be grouped together in hgroup tag:

<hgroup>

* <h1>Main heading</h1>
* <h2>Subheading</h2>

</hgroup>

8. Define span tag.

A span tag is similar to div section but is used to divide the screen into even smaller sections. <span>...</span>

9. Define Meta tags.

Meta tags are the tags that provide more information to our web pages. Some of the tags that are examples for meta are:

* **Link**: Defines a relation between the web page and an external source.
* **Style**: Inserts styles (CSS) details to the web page.
* **Title**: Provides a title to the web page.

**You may also like:**[Top 5 Skills That Make You A Sure Shot Programmer](https://www.greycampus.com/blog/programming/top-five-skills-that-make-you-a-sure-shot-programmer)

10. What are the most commonly used List tags?

The list tags are used to list out any information. There are types in listing out the information using the tags, these are:

* **Ordered List**: Used as <ol>...</ol> and provides the list in a numbered format.
* **Unordered List**: Used as <ul>...</ul> and provides the list in a bullet format.

NOTE: both the above use list item tags ie <li>...</li> in them

* **Definition List**: Used as <dl>...</dl> which provides a list in it and takes Definition Term (ie <dt>...</dt>) and a detailed definition (ie <dd>...</dd>).

11. What is the best way to group form elements?

The <fieldset> tag can be used to group some of the form elements. When the purpose is to provide some HTML elements as a group to the user then, fieldset tag is the best one to use. It takes another tag in it <legend> which provides a title to the grouped elements.

12. List few input type attributes that are new in HTML5.

An input tag can take some attributes based on the requirement. ie <input type=”text”> creates a textbox that takes text/number/special characters into it. The attribute type can take more values, these are:

* **Datetime**: Gives a datetime accepted textbox.
* **Placeholder**: Places an editable text in the textbox.
* **Number**: Takes only numbers and no text and special characters.
* **Tel**: Takes inputs like a telephone number.
* **Email**: Takes text with email format.

13. How is it possible to insert a copyright symbol in a web page?

As there is no direct key to type and insert the copyright symbol in a web page, it has to be given by combining different characters. This can be done in multiple ways like:

* &copy.
* & #169.

14. What is a hyperlink?

A hyperlink is a text/image on a webpage, which when clicked redirects to a new webpage.

15. Explain the type and value attributes for a list item (ie <li>).

The type attribute helps to change the listing/numbering of the list items. It is also used to change the numbering index the value attribute is used.

16. Describe the way in which color can be applied to an element in an HTML page.

HTML elements can be colored in three ways:

* Color name ie red/blue etc.
* Hex Color ie #efefef, and
* RGB(10,234,128) which is a combination of red, green, blue.

17. Explain the two web storage in HTML5.

In HTML 5 data storage can be done in two ways:

* **Session Storage**: The current session (i.e., a user who is browsing the website) details are stored. Once the user closes the browser, the storage is cleared.
* **Local Storage**: Data stored in local storage will not be cleared automatically or when the user closes the browser.

18. Explain Graphics in HTML5.

HTML5 has two types of providing graphics, SVG and Canvas. SVG is helpful when using devices like tablets, iPads etc. Whereas, Canvas is an area in the document or HTML page where graphics can be drawn.

**You may also like:**[Top 40 Spring Interview Questions and Answers (Updated for 2018)](https://www.greycampus.com/blog/programming/top-spring-interview-questions-and-answers)

19. What are the Server-Sent Events (SSE)?

The events that come from the server to the browser are termed as Server Side Events.

20. List out the new attributes provided in HTML5 for the form tag.

HTML5 has introduced two new attributes for the form tag which are:

* autocomplete.
* non-validate.

21. What is the usage of a novalidate attribute for the form tag that is introduced in HTML5?

The form attribute novalidate is a Boolean type which takes true/false values. If disabled or made false, the form data is not validated when submitted.

22. Why do we need to autocomplete attribute for the form tag?

The autocomplete attribute is usually applied to text boxes with which the user can fill the textboxes with the previous values given to it.

23. How can you hide  HTML source code?

It is not possible to hide the HTML source code as the browser needs to understand the code.

24. How can a static HTML page redirect the visited a new page?

The meta tag has an attribute called URL which takes you to a redirected link.

25. What is Cell Padding?

Cell Padding refers to the space between the cell wall and cell border.

26. What are the different kinds of Doctypes available?

The three kinds of Doctypes which are available:

* Strict Doctype
* Transitional Doctype
* Frameset Doctype

27. How do we comment HTML code in an HTML page?

HTML code can be commented using <!-- all the code written here will be commented -->

28. List the API’s available in HTML5.

The available API’s in HTML are:

* DOM Level 1
* DOM Level 2 Core
* DOM Level 2 HTML
* DOM Level 3

29. Explain Application Cache.

Application Cache is a mechanism in which the web applications can also run offline. One of the advantages of using the application cache is that the resources load faster.

30. When do we use a meter tag?

This tag is also known as a gauge and is used to define a scalar measurement which is in a known range.

31. What is the usage of the cite tag?

The cite tag is used to represent the title of a work (like a book, paper, poem etc). This tag is mostly used as an inline tag.

**Explore**[8 Up-and-Coming Programming Languages Developers Should Know](https://www.greycampus.com/blog/programming/eight-up-and-coming-programming-languages-developers-should-know)

32. What is a contenteditable attribute?

This attribute is a new feature in HTML5 using which any element can be edited directly from the browser.

33. How are Web Workers APIs used in HTML5?

Using Web Workers APIs provide a way to run something in the background with which certain tasks can be performed without distracting the user interface.

34. What is MathML in HTML5?

MathML stands for Mathematical Markup Language and is used to show mathematical and scientific content on the web page. This expression should be inserted in a math tag.

35. Explain WHATWG.

WHATWG stands for **Web HyperText Application Technology Working Group** which is an organization that develops and also maintains the HTML and API for web applications.

36. Explain Microdata in HTML5.

Using Microdata, the metadata is nested within existing content on web pages. Search engines extract the microdata from a web page to provide a good browsing experience.

**Related reading**[Top 55 Java Interview Questions and Answers for 2018](https://www.greycampus.com/blog/programming/top-fifty-five-Java-interview-questions-and-answers)

37. List some of the formatting elements in HTML5.

HTML5 holds some new and old formatting elements, all together here is the list:

* Marked text
* Emphasized text
* Subscripts
* Superscripts
* Small text
* Deleted text
* Inserted text

38. What is the difference between HTML and XHTML?

HTML stands for HyperText Markup Language and XHTML stands for eXtensible HyperText Markup Language which is defined in XML format.

39. What is an API in HTML5?

API stands for Application Programming Interfaces and is a way to create applications. They use pre-built components. Using the available APIs, developers can integrate the features into their websites.

40. List the API’s available in HTML5.

The HTML5 standard introduced a number of new APIs. Here is a small list:

* Constraint Validation API
* Text Track API
* Media API
* Command API
* User Interaction
* Data Transfer API
* History API
* Application Cache API

41. What is a manifest file in HTML?

With the introduction of application cache, a web application is accessible without an internet connection. Every page of the web application should hold the manifest attribute to get cached.

42. What is the difference between the font size units em and rem?

By default, the font size in a browser is 16px and can be increased or decreased in the browser settings. 1em equals the font-size given in the browser. Whereas, the rem value depends on root element that is HTML tag.

43. What are the two methods that are used in form elements action attribute?

A form has an action attribute that takes two values, Post and Get (default). The Post method is used to submit (insert\update\delete) data whereas, Get method is used to get some data.

44. What is Web SQL?

HTML5 has introduced a database connection to store values on the client side. Web SQL is such a database in the browser which holds client information. It is important to note that Web SQL does not store critical data (like passwords) in it.

45. What are the minimum number of HTML tags are required to create an HTML page?

A minimum of 3 tags (HTML, head, and body) are required to create a working HTML page.

**Read more**[Top 51 Javascript Interview Questions and Answers You Must Prepare in 2018](https://www.greycampus.com/blog/programming/top-javascript-interview-questions-and-answers)

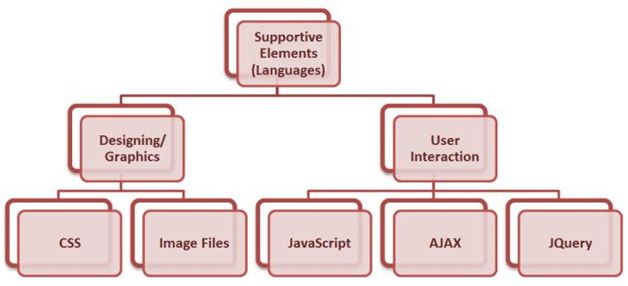
Interview Tips

HTML is considered as a very minor technology, but many elements have been introduced with the release of the latest HTML version 5 and there is now a very high chance that the professionals will be posed with more questions based on HTML5 in the job interviews. It is, therefore, advised that both freshers and experienced must know and understand the differences between the version 4 & 5.

Conclusion

HTML can be considered as the very base of web technologies as this is the very first technology that is used to create a web page. Professionals who are willing to make their career in web technology must learn HTML. Not only is HTML one of the cornerstones of web technology, but is also used to develop applications for phones and tablets too.

Question 1: What is HTML5?

**Answer**  
  
HTML5 is the latest version of HTML and XHTML with new features like Drawing, Animation, Video and Audio etc. It is used to solve some common structural problems encountered with HTML 4.1. It gives more flexibility to both the web developers, the web designers and enables more exciting and interactive websites in addition to more powerful and efficient applications. The HTML 5 <! doctype html> is recognized by all modern browsers.  
  
HTML5 brings a whole new dimension to the web world. It can embed video on the web-pages without using any special software like Flash. HTML5 is developed in such a way that the developers are not required to waste their time and efforts in creating an error free web page. Firefox, Chrome, Opera, Safari and Internet Explorer all support <! doctype html>.  
  
**Why do we use HTML5?**  
  
The main benefit of HTML5 is that it supports Drawing, Animation, Video and Audio.  
  
The web developers can decrease the complexity and the time to create applications with animations, play music (audio and video), high quality drawings and other rich content using HTML 5 because it can embed video on the web-pages without using any special software like Flash.  
  
HTML5 is far easier for the web designers and the web developers as it tells them how a web page is structured.  
  
  
  
For more details, visit following link:

* [Introduction to HTML5](http://www.c-sharpcorner.com/UploadFile/7b0949/introduction-to-html5/)

Question 2: What are HTML5 Graphics?

**Answer**  
  
In HTML5, there are the following two types of graphics:

* **Scalable Vector Graphics (SVG)**

SVG provides a big benefit; basically people are now using high-resolution devices (iPads and Monitors) so it becomes impressive as designs, logos and charts can scale accordingly. The HTML tag <svg> is a container for SVG graphics. SVG is used for drawing paths, circle, boxes, text and graphic images.

* **Canvas**

A canvas is a rectangular area on HTML page for drawing graphics on the fly via JavaScript. The default size of the canvas is 300 px × 150 px (width × height). The HTML tag <canvas> is a container for Canvas graphics.  
  
**Where to use Canvas and SVG**  
  
Canvas is procedural whereas SVG is declarative. Some reasons to consider SVG instead of canvas are:

* SVG is scalable, provides the facility of auto scaling icon, logo and chart.
* SVG is not supported by the languages whereas canvas elements are manipulated using client-side JavaScript.
* DOM handling. It's easy to attach event handlers and manipulate elements like you would for another HTML block. To move an item, you simply change its coordinates but this is not true for a Canvas.

For more details, visit following link:

* [Understanding HTML5 Graphics](http://www.c-sharpcorner.com/UploadFile/22d000/understanding-of-html5-graphics/)

Question 3: What is DataList Tag in HTML?

**Answer**  
  
A <datalist> tag can be used to create a simple auto-complete feature for a webpage.<datalist> is a newly defined HTML tag that comes with the HTML 5 specification. By using this <datalist> tag, we can define a list of data and then we can bind it with an <input> tag.   
  
A <datalist> tag specifies a list of predefined options for an <input> element. After binding it, the user will see a drop down list in which all the predefined options will be there for the input. When the user types a character or a string, the user will automatically get the data which depends on the input string or a character.  
  
The main feature of this <datalist> tag is to auto-complete the <input> element.  
  
**Example**  
  
Suppose we have a TextBox for the country.

1. <input type="text" list="countries" name="country" />

**Complete Example**

1. <!DOCTYPE html>
2. <html lang="en">
4. <body>
5. Please Select Country: <input type="text" list="countries" name="country" />
6. <datalist id="countries">
7. <option value="India">India</option>
8. <option value="United States"></option>
9. <option value="United Kingdom"></option>
10. <option value="China"></option>
11. <option value="Nepal"></option>
12. <option value="Afghanistan"></option>
13. <option value="Iceland"></option>
14. <option value="Indonesia"></option>
15. <option value="Iraq"></option>
16. <option value="Ireland"></option>
17. <option value="Israel"></option>
18. <option value="Italy"></option>
19. <option value="Swaziland"></option>
20. </datalist>
21. </body>
23. </html>

For more details, visit following link:

* [DataList Tag in HTML 5](http://www.c-sharpcorner.com/UploadFile/75a48f/datalist-tag-in-html-5/)

Question 4: Describe Form Input Types in HTML5?

**Answer**  
  
HTML5 is the new standard for HTML that has 13 new input types for forms. Using these new input types, we can create more interactive and easy-to-use web forms. It also provides better data validation, input control, color picker controls and many others.  
  
The new input types are:

* Time
* Date
* Datetime
* Datetime-local
* Week
* Month
* Email
* Color
* Number
* Range
* Search
* Telephone
* URL

**Input Type Range**  
  
An input type range is used for input fields that should contain a value from a range of numbers which is done by declaring a minimum and a maximum value. You can also set restrictions on which numbers are required to be accepted.

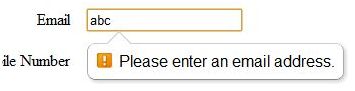
1. <input type="range" name="age" min="18" max="35" />

Type Range  
  
**Input Type URL**  
  
A type URL is used for the input fields that should contain a URL address. The value of the URL field is automatically validated when the form is submitted.

1. <input type="url" name="mainpage" />

   
**Input Type email**  
The email type is used for the input fields that should contain an Email address and automatically validates it when submitted.

1. <input type="email" name="emailid" />

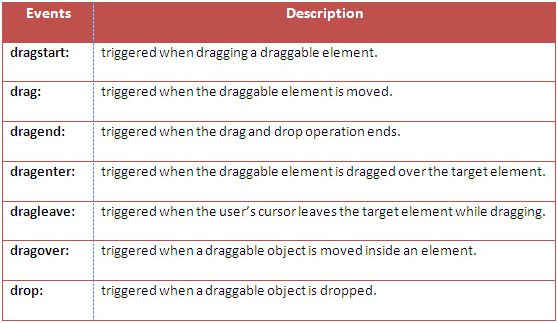
  
  
**Input Type Tel**  
  
Input Type Tel defines a field to enter a telephone number and automatically validates when the form is submitted.

1. <input type="tel" name="usrtel" />

For more details, visit following link:

* [HTML5 Input Types](http://www.c-sharpcorner.com/UploadFile/867331/html5-input-types/)

Question 5: What is the use of Drag and Drop in HTML5?

**Answer**  
  
Drag and drop is a very common feature and convenient to users. Simply, you need to grab an object and put it at the place you want. This feature is commonly used by many of the online examination websites wherein you have the options to pick up the correct answer, drag it to the answers place holder and drop it.  
  
The Drag and Drop API comes with seven new events to track a drag and drop. The events are dragstart, drag, dragend, dragenter, dragleave, dragover and drop that are triggered during the various stages of the drag and drop operation. These events are listed below:   
  
   
  
For more details, visit following link:

* [HTML5 Drag and Drop](http://www.c-sharpcorner.com/UploadFile/1e050f/html5-drag-and-drop/)

Question 6: What is HTML 5 Web Storage?

**Answer**  
  
HTML5 Web Storage, also known as DOM Storage is a way to preserve state on either the client or server which makes it much easier to work against the stateless nature of HTTP.  
  
**Advantages of HTML5 Web Storage:**

1. It can store 5 to 10 MB data. That is far more than what cookies have.
2. Web storage data is never transferred with HTTP request, so it increases the performance of the application.

**Web Storage Strengths and Weaknesses**  
Strengths

* Apps can work both online and offline.
* API is easy to learn and use.
* Has the ability to hook in to the browser events such as offline, online and storage change.
* Has less overhead than cookies; no extra header data is sent with the browser requests.
* Provides more space than what cookies offer so increasingly complex information can be kept.

Weaknesses

* Data is stored as a simple string; manipulation is needed to store objects of different types such as Booleans, Objects, Ints and Floats.
* It has a default 5MB limit; more storage can be allowed by the user, if required.
* It can be disabled by the user or systems administrator.
* Storage can be slow with the complex sets of data.

**HTML5 Web Storage Methods**

* **setItem(key,value):**Adds a key/value pair to the sessionStorage object.
* **getItem(key):** Retrieves the value for a given key.
* **clear():** Removes all key/value pairs for the sessionStorage object.
* **removeItem(key):** Removes a key/value pair from the sessionStorage object.
* **key(n):**Retrieves the value for a key.

**Getting a Value**  
There are two methods to retrieve a key/value pair as well:

1. sessionStorage.getItem('someKey');
2. //returns 'someValue'
3. sessionStorage.someKey;
4. //returns 'someValue'

For more details, visit following link:

* [HTML5 Web Storage](http://www.c-sharpcorner.com/UploadFile/18ddf7/html5-web-storage/)

Question 7: What are the types of Web Storage in HTML5?

**Answer**  
  
There are two types of Web Storage,

1. Session Storage
2. Local Storage

As its name implies, it stores data of current session only which means the data stored in session storage clears when the browser is closed. To access session storage in JavaScript, the following methods are available.

1. To store data in session storage, setItem () function is used.
   1. sessionStorage.setItem (‘key’,’value’);

**Example:**

* 1. sessionStorage.setItem (‘username’,’ABC’);

We can only store strings in Session Storage. To save the objects in session, first convert the object into JSON string and then store this string in Session Sorage as in the following,

* 1. sessionStorage.setItem (‘**object**’, JSON.stringify(**object**));

1. To retrieve the data from Session Storage getItem() function is used.
   1. sessionStorage.getItem(‘key’);

**Example:**

* 1. var username= sessionStorage.getItem(‘username’);

If JSON string is stored in Session Storage then you can convert it into object as below.

* 1. var **object**=JSON.parse(sessionStorage.getItem(‘**object**’));

1. To delete a particular key from Session Storage, removeItem function is used.
   1. sessionStorage.removeItem(‘key’);

**Example:**

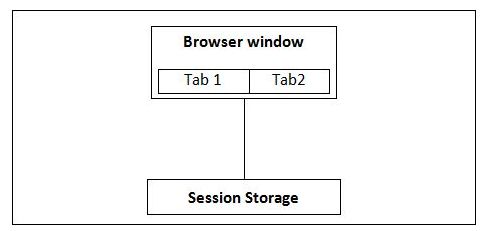
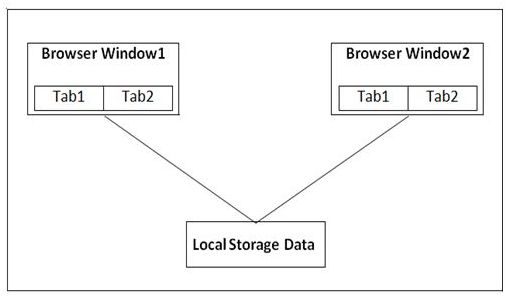
* 1. sessionStorage.removeItem(‘username’);

1. To delete all the keys from Session Storage, clear function is used as shown below
   1. sessionStorage.clear();

To get all KEY/VALUE pairs from Session Storage, you can loop through Session Storage like the following.

* 1. **for** (var i = 0; i < sessionStorage.length; i++)
  2. {
  3. var key = sessionStorage.key(i);
  4. var value = sessionStorage.getItem(key);
  5. }

**Local Storage**  
Local Storage is a second type of HTML Web Storage. Like Session Storage, it also stores data in KEY / VALUE pair of strings. The following points helps to compare Session Storage and Local Storage.

1. Session Storage stores the data for only current session of the browser, when the browser closes data in, Session Storage clears. On the other hand, the data stored in Local Storage is not deleted automatically when the current browser window is closed. Data in Local Storage clears only when it is manually deleted.
2. The data in Session Storage is accessible only in current window of the browser but the data in the Local Storage can be shared between multiple windows of the browser.  
     
    **Figure:**Session Storage  
     
    **Figure:**Local Storage

The following functions are used to access Local Storage in JavaScript:

1. To store data in Local Storage, setItem() function is used.
   1. localStorage.setItem (‘key’,’value’);

**Example:**

* 1. localStorage.setItem (‘username’,’ABC’);

We can only store strings in Local Storage. To save objects in Local Storage, first convert the object into JSON string and then store this string in Local Storage as shown below:

1. localStorage.setItem (‘**object**’, JSON.stringify(**object**));

To retrieve the data from Local Storage, getItem() function is used.

1. localStorage.getItem(‘key’);

**Example:**

1. var username= localStorage.getItem(‘username’);

For more details, visit following link:

* [HTML5 Web Storage - Part 2 (Local Storage)](http://www.c-sharpcorner.com/UploadFile/3194c4/html5-web-storage-part-2-local-storage/)
* [HTML5 Web Storage Part 1 - Session Storage](http://www.c-sharpcorner.com/UploadFile/3194c4/html5-web-storage-part-1-session-storage/)
* [Difference Between Local Storage, Session Storage And Cookies](http://www.c-sharpcorner.com/UploadFile/cd7c2e/difference-between-local-storage-session-storage-ans-cookie/)

Question 8: What is Audio Tag in HTML 5?

**Answer**  
  
This new element allows you to deliver audio files directly through the browser, without the need for any plug-ins. The Audio tag is a new tag introduced in HTML5. You can use it to play audio sound like .mp3, wav, and .ogg. I have used five types of sound formats to show which formats are compatible for the browsers. A WAV file is a common sound format that is supported by most browser versions.  
  
**Syntax**  
  
*<audio src="URL" controls> </audio>*  
  
**Syntax for more than one audio format**  
  
*<audio controls="controls" >  
<source src="URL1" type="audio/mp3" />  
<source src="URL2" type="audio/wma" />  
<source src="URL3" type="audio/x-wav" />  
</audio>*

|  |  |
| --- | --- |
| **New Element** | **Specific Attributes** |
| autobuffer | This Boolean attribute indicates whether or not the browser should begin buffering audio right away. |
| autoplay | This is Boolean attribute indicate whether or not the file should start playing audio as soon as it can. |
| loop | This Boolean attribute indicates whether or not apply repetition on playing audio. |
| src | This attribute is used to specify the URL (location of the audio file) of the audio to show. |
| controls | This Boolean attribute specify whether or not the browser should display audio controls (such as play/pause, volume and seek). |

|  |  |  |  |
| --- | --- | --- | --- |
| **HTML5 Event Attributess** | | | |
| onabort | onblur | oncanplay | oncanplaythrough |
| onchange | onclick | oncontextmenu | ondblclick |
| ondrag | ondragend | ondragenter | ondragleave |
| ondragover | ondragstart | ondrop | ondurationchange |
| onemptied | onended | onerror | onfocus |
| onformchange | onforminput | oninput | oninvalid |
| onkeydown | onkeypress | onkeyup | onload |
| onloadeddata | onloadedmetadata | onloadstart | onmousedown |
| onmousemove | onmouseout | onmouseover | onmouseup |
| onmousewheel | onpause | onplay | onplaying |
| onprogress | onratechange | onreadystatechange | onscroll |
| onseeked | onseeking | onselect | onshow |
| onstalled | onsubmit | onsuspend | ontimeupdate |
| onvolumechange | onwaiting |  |  |

For more details, visit following link:

* [Audio tag in HTML5](http://www.c-sharpcorner.com/UploadFile/rohatash/audio-tag-in-html5/)

Question 9: What is a Video Tag in HTML 5?

**Answer**One of the exciting features of HTML5 is <video>tag.Video Tag was introduced in HTML5. Video tag is used to add video or media files in the HTML document. It is not present in HTML 4.01. Before this, it was not possible to show a video/movie on the Web Page.For example, you can embed music video or a movie on your web page for the visitor to listen and watch it.  
  
HTML5 video tag accepts the attribute that specifies how the video should be played. You can write content in <video> tag. as <video> tag always occur in pair. Any content between opening and closing tag is "fallback content". This content is displayed only by those browsers that does not support video tag. Now, most of the video files are shown by plug-in.  
  
**Syntax**  
Syntax of <video>tag in HTML5:  
 *<video src="URL"></video>*  
OR  
  
*<video>  
<source src="URL">  
<source src="URL">  
</video>*  
  
**Attributes of the <video> tag**

|  |  |  |
| --- | --- | --- |
| **Attribute** | **Value** | **Description** |
| autoplay | autoplay | Video will start playing automatically. |
| autobuffer | autobuffer | Video will start buffering automatically. |
| loop | loop | Video automatically start over again when done. |
| controls | controls | In order to show the controls. |
| poster | URL of the image | URL(address) of the image. |
| src | URL | Address of the video. |
| width | pixel | Defining the width of the video. |
| height | pixel | Defining the height of the video. |

For more details, visit following link:

* [Video Tag in HTML5](http://www.c-sharpcorner.com/UploadFile/18ddf7/video-tag-in-html5/)

Question 10: What are the media elements in HTML 5?

**Answer**   
  
The following are the 5 most popular media elements.  
  
**Audio**

* Audio element is used to define or create a music element in your simple HTML page.
* It supports all the browsers like Internet Explorer 9.0 and above, Chrome, Opera, Firefox and Safari.
* This tag defines music or any other audio stream formats.

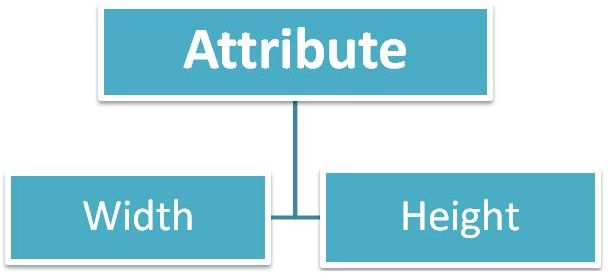
**Video**  
  
The Video element creates a video element in your HTML page. It supports all the browsers like Internet Explorer 9.0 and above, Chrome, Opera, Firefox and Safari. This tag defines music or any other video stream formats.  
  
**Track**  
This element is useful in both the previous elements  i.e AUDIO and VIDEO. This element helps to define tracks or we can say simple sectors for the <audio> and <video> elements.  
  
**Source**  
Like the track element, the Source element must be used in <audio> and <video> elements to do the control property and structure of the tracks.  
  
**Embed**  
  
It is also called a container because as the name suggests, it is used for defining the containers for the external applications or we can say plug-ins for the Applications.  
  
For more details, visit following link:

* [HTML5 Media Elements](http://www.c-sharpcorner.com/UploadFile/2072a9/html5-media-elements/)

Question 11: What is the use of Canvas element in HTML5?

**Answer**  
  
The canvas element is used to draw graphics by making use of JavaScript. It is defined with the <canvas> tag. We write the following code.  
  
This tag is nothing but a container for graphics. We need to use a script to provide the structure or shapes of the graphics.

* We can have multiple Canvas elements in one HTML page.
* The real power of the Canvas element is the scripting for which use JavaScript.
* For the complete functionality of the HTML5 Canvas tag, we require an API which is made by writing JavaScript code that provides access to Canvas element functionality.

**HTML5 Canvas | Methods**  
There are several methods for various functionalities like drawing shapes, adding images, graphics and so on in a Canvas element.  
  
  
  
**What are the features of HTML5 Canvas?**

* Flexibility
* Interactivity
* Multimedia Options
* Animation
* Platform Support
* Free and Accessible Dev Tools
* Code Once, Run Everywhere

For more details, visit following link:

* [HTML5 Canvas](http://www.c-sharpcorner.com/UploadFile/2072a9/html5-canvas/)

Question 12: What is the use of Fieldset tag in HTML5?

**Answer**  
The <fieldset> tag groups related form elements. It is like a box. In other words, it draws a box around related elements.  
  
It must start with a <legend>tag because the <legend> tag defines the title of the field set.  
  
By using the <fieldset>tag and the <legend> tag, you can make your form much easier to understand for the users.  
  
**Syntax**The syntax of the <fieldset> tag in HTML5 is:  
  
*<fieldset>Controls</fieldset>***Browser Support**  
The <Fieldset> tag is supported by all major browsers.  
  
**Attributes of <fieldset> tag**  
  
HTML5 has added some new attributes; they are:

|  |  |  |
| --- | --- | --- |
| **Attribute** | **Value** | **Description** |
| disabled | disabled | Specify fieldset will be displayed or not |
| name | text | Specify name of field set |
| form | name of form | Define it is related to the form |

**Example:** In this example, we create a fieldset in a form. We used the <legend> tag to define the caption for the <fieldset> element.  
  
**Code:**

1. <html>
3. <body>
4. <form>
5. <fieldset>
6. <legend>Personal Information</legend>
7. First Name: <input type="text" />
8. <br/><br/> Last Name: <input type="text" />
9. <br/><br/> p\_Address: <input type="text" />
10. </fieldset>
11. </form>
12. </body>
14. </html>

For more details, visit following link:

* [HTML tags which are deprecated in HTML5](http://www.c-sharpcorner.com/UploadFile/18ddf7/fieldset-tag-in-html5/)

Question 13: What are the HTML tags which deprecate in HTML5?

**Answer**One of the main points on which HTML5 wins over XHTML2.0 is “backward compatibility”. XHTML2.0 sought to enforce well-written code by using very harsh error handling. If a page returns error based on syntax, the user agent will stop parsing the code.  
  
An HTML5 specification states that certain HTML tags should not be used but it is only a guideline to the HTML authors. The implementations, however, must support these tags to be backward compatible.  
  
The tags that are deprecated are the following:

* <basefont>
* <big>
* <center>
* <font>
* <s>
* <strike>
* <tt>
* <u>
* <frame>
* <frameset>
* <noframe>
* <acronym>
* <applet>
* <isindex>
* <dir>

Several tag attributes are also removed. Few of the most notable ones are:

|  |  |
| --- | --- |
| **Element** | **Attribute removed** |
| a,link | rev, charset |
| img | longdesc, name |
| html | version |
| th | abbr |
| td | scope |
| all block level elements | align |
| body | background |
| img | hspace, vspace |
| table, tr, th, td | bgcolor |
| table | border, cell padding, cell spacing |
| td, th | height, width |
| table | valign |

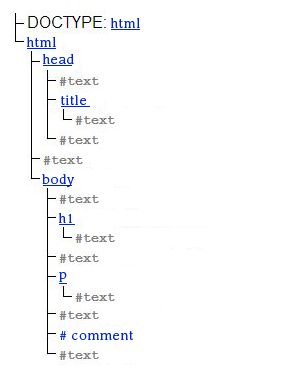
For more details, visit following link:

* [Learn HTML5 - Part 1: Introducing HTML5 APIs](http://www.c-sharpcorner.com/blogs/html-tags-which-are-deprecated-in-html5)

Question 14: What are the new APIs provided by the HTML 5 standard?

**Answer**  
**HTML5 APIs**  
After HTML's evolution, later on in 1998 parts of the API for HTML developed by the browser vendors and release named DOM Level 1 followed by DOM Level 2 Core, DOM Level 2 HTML and finally DOM Level 3.  
  
**Note:**One thing you need to ensure is that HTML is not only used to write code as tags and angle brackets. The HTML5 specification also defines how JavaScript is interacting with those angle brackets using the Document Object Model (DOM). There is a DOM API for each corresponding tag.  
  
Web browsers, or you can say HTML user agents, parse your written markup into a DOM tree that represents the specific document in memory and each DOM tree contains several kinds of nodes, in particular a DOCTYPE node, elements etc.  
  
Let's use an example of HTML markup snippet and understand the DOM concept.  
  
**HTML markup snippet**

1. <!DOCTYPE HTML>
2. <html>
4. <head>
5. <title>Your Page Title</title>
6. </head>
8. <body>
9. <h1>Text</h1>
10. <p>Text</p>
11. <!-- Comment -->
12. </body>
14. </html>

**DOM tree of HTML markup snippet**  
  
  
  
In HTML5 you can use many APIs. Some of them are:

* Web Workers API
* Server-sent Events API
* WebSocket API
* Cross-document Messaging API
* Drawing
* Audio/Video
* Drag and drop
* Autofocus
* Editable
* Client-side storage
* Geolocation

For more details, visit following link:

* [Application Cache in HTML5](http://www.c-sharpcorner.com/UploadFile/manish1231/learn-html5-part-1-introducing-html5-apis/)

Question 15: What is Application Cache in HTML5?

**Answer**  
  
Yet HTML5 has many new special elements and attributes but one of the best feature of HTML5 is "Application Cache", that enables us to make a offline session of a web application. It allows to fetch few or all the content of a website such as HTML files, Images, JavaScript, CSS ...etc. This features speeds up the site performance. This is achieved with the help of a file, defined as follows:

1. <! doctype html>
2. <html manifest="example.appcache">
3. ....
4. ....
5. .....
6. </html>

For more details, visit following link:

* [Meter Tag in HTML 5](http://www.c-sharpcorner.com/blogs/application-cache-in-html5)

Question 16: What is a meter tag? What is the difference between progress tag and a meter tag?

**Answer**The meter tag is used to represent a scalar measurement within a known range. The value can be fractional.   
 **Examples**  
  
Disk uses, the relevance of a query result, the fraction of a voting population to have selected a specific candidate.  
  
**Difference between progress tag and meter tag**  
A progress tag represents the completion progress of a task whereas the meter tag is used to represent gauges. We can think that a progress tag represents a dynamic data whereas a meter tag represents a static data.   
  
**Note:**

1. According to the W3C, the meter tag should not be used to indicate progress as to indicate the progress, we have the progress tag.
2. The meter tag also does not represent a scalar value of an arbitrary range; for example, it would be wrong to use this to report a weight, or height, unless there is a known maximum value.

**Syntax**  
The Meter tag is an inline element, the same as a header, progress and so on.  
  
*<meter></meter>*  
**Attributes**The meter tag has 6 more attributes as shown in the following table:

|  |  |  |
| --- | --- | --- |
| **Attribute** | **Value** | **Description** |
| Min | Floating Point Number | Specifies the lower bound, Default value is 0.0 |
| Max | Floating Point Number | Specifies the upper bound, Default value is 1.0 |
| Low | Floating Point Number | This represents the upper bound of the low end |
| High | Floating Point Number | This represents the lower bound of the high end |
| Value | Floating Point Number | Specifies the current value |
| Optimum | Floating Point Number | Specifies that what measurements value is the best value |

For more details, visit following link

* [Scalable Vector Graphics (SVG) in HTML5](http://www.c-sharpcorner.com/UploadFile/75a48f/meter-tag-in-html-5/)

Question 17: What is the use of Scalable Vector Graphics (SVG) in HTML5?

**Answer**  
Scalable Vector Graphics (SVG) are the part of the vector-based family of graphics. There are various forms of Raster-based graphics available that stores the color definition for each pixel in an array of data. Some of the most common Raster-based formats used on the web today are JPEG (Joint Photographic Experts Group), GIF (Graphics Interchanged Format) and PNG (Portable Network Graphics). SVG is a language for to describe 2D vector graphics in XML.  
  
**Basics of SVG**Creation of an SVG image is a very different process. To create any other Raster images like JPEG, PNG or GIF, we use image editing software like Photoshop and so on but SVG images are XML based file so they can be created in any other text editor. There is a tool also available (inkspace). By using this tool, you can draw and create SVG images very conveniently.  
  
**Basic Shapes Created by SVG**You can use SVG XML tag to create shapes.

|  |  |
| --- | --- |
| **Element** | **Description** |
| line | Creates Simple line |
| circle | Creates Circle |
| rect | Creates Rectangle |
| ellipse | Creates Ellipse |
| polygon | Creates Polygon |
| polyline | Creates Multiline Shape |
| path | Creates Arbitrary Path |
| text | Allows to Creates Text |

For more details, visit following link

* [Learn HTML5 - Part 3: Server-sent Events APIs](http://www.c-sharpcorner.com/UploadFile/75a48f/scalable-vector-graphics-svg-in-html5/)

Question 18: Why do we need HTML 5 Server-Sent Events?

**Answer**Server-Sent Events (SSE) is a recent HTML5 specification in combination with the EventSource API designed for streaming updates. Prior to that, you might be familiar with the bidirectional communication channel, known as WebSockets, used very much and tons of server implementations are available on the internet. However, the second server-push technology of HTML5 yet stays in the shadows.  
  
To enable efficient server-to-client streaming of event data as it is basically text-based. For example: real-time notifications or updates, like Facebook, Twitter, stock exchange updates and and so on are generated on the server. Basically, we have the following two types of components that SSE introduces:

* EventSource Interface
* Event Stream

The EventSource Interface allows the client to receive push notifications from the server as DOM events and the "Event Stream" data format is used to deliver the individual updates.  
  
**SSE flow chart**  
In the following figure, you will see how SSE works after the client/server is connected.   
  
  
  
For more details, visit following link:

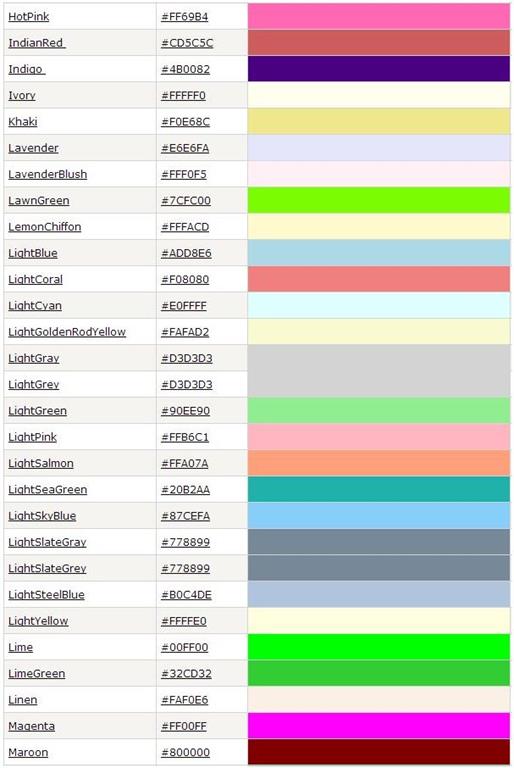
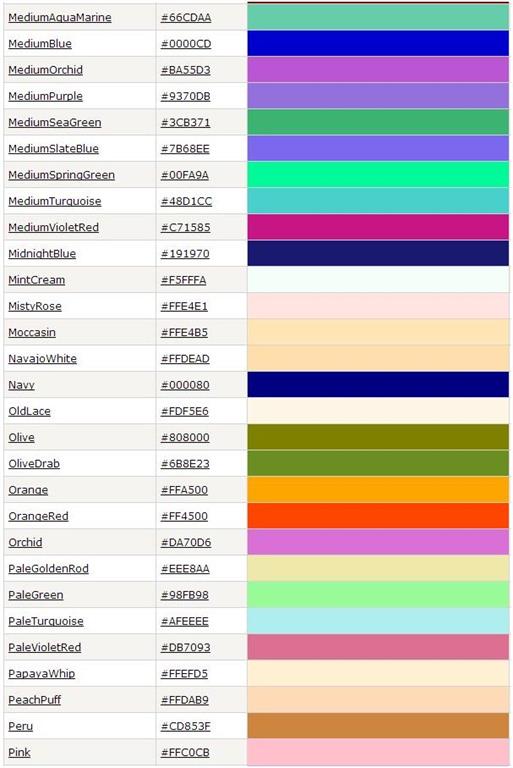
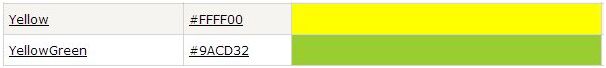
* [Cite Tag in HTML5](http://www.c-sharpcorner.com/UploadFile/manish1231/learn-html5-part-3-server-sent-events-apis/)

Question 19: What is the use of cite tag in HTML5?

**Answer**The <cite> tag indicates a citation. It represents the title of a work (e.g. a book, paper, essay, poem, score, song, script, film, TV show, game, painting, sculpture , play , exhibition , etc.). The <cite> tag is an inline tag that indicates "defining a citation". The text within the <cite> tag is shown in Italics. The cite tag must have a start and end tag.  
  
In this tag the "title" attribute defines the title of the Text within the <cite></cite> tags.  
  
In HTML5 , the <cite> tag defines the cited title of a work whereas HTML 4.01 implied that the <cite> tag can be used to indicate the name of a person.  
  
**Declaring Syntax***<cite title="value">Some Text Here</cite>*  
 **Browser support**The <cite> tag is supported in all major browsers (e.g. Internet Explorer, Google Chrome, Mozilla Firefox. Safari, etc ).  
  
For more details, visit following link

* [Color Name in HTML5](http://www.c-sharpcorner.com/UploadFile/18ddf7/cite-tag-in-html5/)

Question 20: What are the colors names in HTML 5?

**Answer:** You can see all 147 colors in the following list:  
  
  
  
  
  
  
  
  
For more details, visit following link:

* [Smooth Wave in HTML5](http://www.c-sharpcorner.com/UploadFile/da6585/color-name-in-html5/)

Question 21: What are Waves in HTML?

**Answer**A sine wave is a mathematical function that is repeats at a regular interval of time. The function is used in many fields including mathematics, physics, and engineering. We can also say that a sine wave is a smooth wave.  
  
It has the following properties:

1. The sine wave is blue whenever the value is positive.
2. The sine wave is red whenever the value is red.
3. The thickness of the line is directly proportional to the absolute value of the magnitude of the wave. For example, where the sine value reaches 0, the wave is absent.

On the X-axis, we will map the angle Theta. Theta will vary from 0 degree to 1040 degrees.  
  
On the Y-axis, we will map the sin (Theta). For this, we will use the Math function Math.sin. The Math.sin function takes angles in radians. So the angle is first multiplied by PI / 180.  
  
For more details, visit following link:

* [HTML 5 Web SQL Database](http://www.c-sharpcorner.com/UploadFile/18ddf7/smooth-wave-in-html5/)

Question 22: What is Web SQL Database in HTML 5?

**Answer**  
Web SQL is a very interesting feature, even though it isn't part of the HTML 5 specification but it is a separate specification and it can still help to develop Web Applications. Web SQL is used to manipulate a client-side database. Since I am saying that it is good to use, there is a disclaimer for its use; it is risky because it stores data at the client side, not on the server side. So always remember, don't store information sensitive to the server inside it.   
  
**Note:** A Web SQL database only works in the latest versions of Safari, Google Chrome and Opera browsers.  
  
**Core Methods of Web SQL**  
  
The following are the 3 core methods of Web SQL that I will cover:

* openDatabase
* transaction
* executeSql

**Creating and Opening Databases**  
Using the openDatabase method, you can create an object for the database. If the database doesn't exist then it will be created and an object for that database will be created. You also don't need to worry about closing the connection with the database.  
  
To create and open the database, you need to use the following syntax.  
  
*var dbObj = OpenDatabase('[Database\_Name]', '[Version\_Number]', '[Text\_Description]', '[size]', '[Creation\_Callback]’)*For more details, visit following link

* [Vibration API in HTML5](http://www.c-sharpcorner.com/UploadFile/75a48f/html-5-web-sql-database/)

Question 23: What is HTML5 Contenteditable Attribute?

**Answer**One of the new features in HTML 5 is the contenteditable attribute. In HTML 5, any element can be editable. By using some JavaScript event handler, you can transform your web page into a fast rich text-box. This feature is mainly applied in Content Management Systems. By using this, you can edit content directly from the browser.   
  
The contenteditable attribute is an enumerated attribute whose keywords are the empty string, true and false. The empty string and the true keyword equates to the true state. The false keyword implies the false state. In addition, there is a third state, the inherit state, which is the missing value default (and the invalid value default).  
 **States of content editable attribute**  
  
According to WHATWG.org, there are the following 3 states of the contenteditable attribute:

|  |  |  |
| --- | --- | --- |
| **State** | **Description** | **How to write?** |
| true | Indicates that element is editable | contenteditable=" " / contenteditable="true" |
| false | Indicates that element is not editable | contenteditable="false" |
| inherit | ndicates that the element will be editable if and only if, its immediate parent element is editable | contenteditable="inherit" |

For more details, visit following link

* [Battery Status API in HTML5](http://www.c-sharpcorner.com/UploadFile/75a48f/html5-contenteditable-attribute/)

Question 24: What is Vibration API in HTML5?

**Answer**Vibration is a simple, a nice way of alert when you get a new message or a phone call. It is especially useful when you are in a noisy environment or the place where you feel the ringing would be a distraction to others.  
  
It is interesting to know that HTML5 is now providing us to play with the vibration on the devices but the HTML5 Vibrate API supports only the recent version of Firefox & Chrome.  
  
To check the vibration, API support in browsers as shown below,

1. navigator.vibrate = navigator.vibrate || navigator.mozVibrate ||
2. navigator.webkitVibrate || navigator.msVibrate;
3. **if** (navigator.vibrate) {
4. // supports vibration API.
5. }

**Vibration Syntax**  
  
Vibration basic syntax is,  
  
*navigator.vibrate(long | [long]);*  
The vibrate function accepts milliseconds or array of milliseconds.  
  
**Example**

1. // vibrate for 1000 ms
2. navigator.vibrate(1000);
3. // same like above but in array of ms
4. navigator.vibrate([1000]);

In the above examples, we are setting the device to vibrate 1000 milliseconds.  
  
For more details, visit following link:

* [Geolocation In HTML5](http://www.c-sharpcorner.com/blogs/vibration-api-in-html5)

Question 25: What is the Battery Status API in HTML5?

**Answer**When a users downloads an application for their devices, they are more conscious of the battery usage of the application. So as a mobile application developer, you should consider the battery usage of your Application.  
  
If you are developing a Web Application for a mobile device then your choice is to use HTML5’s Battery Status API, if you are concerned about the user’s device battery status/charging levels. Yes, HTML5 provides an API for a device's battery.  
W3.org says: “The Battery Status API specification defines a means for the web developers to programatically to determine the battery status of the hosting device”.  
Check for Battery Status API  
  
You can check whether the battery status API is supported by the browser or not as shown below.

1. var battery = navigator.battery || navigator.webkitBattery || navigator.mozBattery || navigator.msBattery;
3. **if** (battery) {
4. // Battery Status API is supported
5. }

The battery status API is currently supported by the latest version of Chrome and Firefox.  
  
**Properties**  
  
There are four basic properties available in the battery status API.

1. **Charging:**Charging is a type of Boolean and a read only that indicates whether the device is charging the battery. The default value is true.
2. **ChargingTime:** ChargingTime is type is double and a read only that gives you the remaining time in seconds to charge the device battery fully. The default value is 0.
3. **DischargingTime:** DischargingTime is the type of double and read only that represents the remaining time for a complete discharge of the device battery. The default value is calculated based on the other property values.
4. **Level:** Level is a type of double and read only that represents the battery level in the scale of 0 - 1.0. The default value is 1.0.

For more details, visit following link:

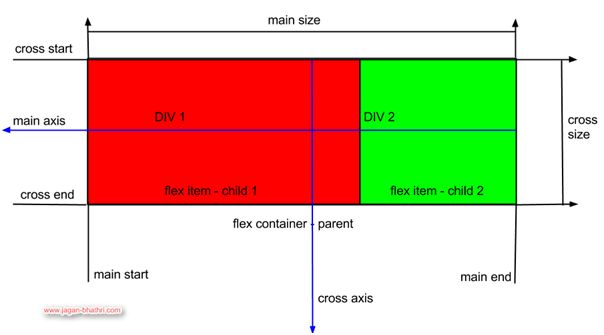
* [Flexbox in HTML5: An Overview](http://www.c-sharpcorner.com/UploadFile/dbd951/battery-status-api-in-html5/)

Question 26: What is the use of Geolocation API of HTML5?

**Answer**The Geolocation API of HTML5 helps to identify the user’s location. It can be used to provide location-specific information. For privacy reasons, the user is asked for permission to report location information. The HTML 5 Geolocation API provides the geographical location of the user. There are many techniques used to identify the location of the user. A desktop browser generally uses WiFi or IP based positioning techniques whereas a mobile browser uses cell triangulation, GPS and A-GPS (Assistive GPS) to triangulate between mobile phone towers and public masts to determine the location and WiFi based positioning techniques and so on.   
  
The Geolocation API will use any of these techniques to identify the user’s location. The Geolocation API protects the user’s privacy by mandating that the user permission should be sought and obtained before sending the location information of the user to any website. Hence, the user will be prompted with a popover or dialog requesting for the user’s permission to share the location information. The user can accept or deny the request.  
  
   
  
**Geolocation Object**  
The Geolocation API is published through the navigator.geolocation object. If this object is present then the geolocation service works.   
  
var geolocation=navigator.geolocation;  
  
The geolocation object is a service object that allows widgets to retrieve information about the geographic location of the device.  
  
For more details, visit following link:

* [Special Symbols in HTML5](http://www.c-sharpcorner.com/UploadFile/a17322/geolocation-html5/)

Question 27: What is Flexbox in HTML 5?

**Answer**Flexbox is not a single property but a set of properties on the parent element and their children. Basically the parent is a container. It is probably a div called a flex container and the children are the elements called flex items.  
  
  
  
The preceding picture shows the Flexible Box Module. Let us discuss the attributes of the flexible box.

* **Main axis:**The main axis is the default flow direction for the flex items.
* **Main-start and Main-end: T**he main-start and main-end are the starting point and ending point for the flex items to flow in the flex container.
* **Cross axis:** The cross axis is perpendicular to the main axis.
* **Cross-start and Cross-end:**The flex items are placed from the start at the cross-start point and ends at cross-end point.
* **Main size:** The flex items width or height in the main dimension is the main size of the flexbox.
* **Cross size:**The flex items width or height in the cross dimension is the cross size of the flexbox.

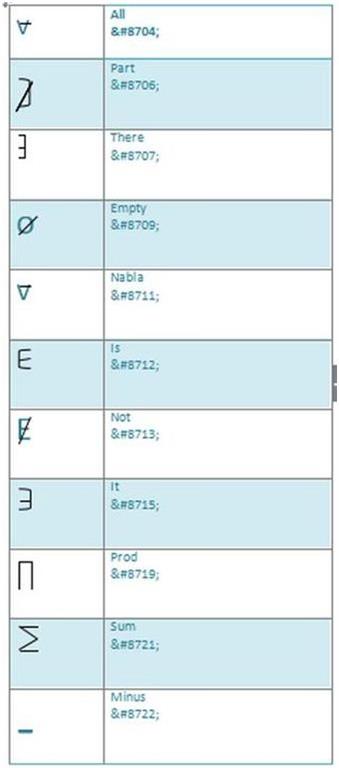
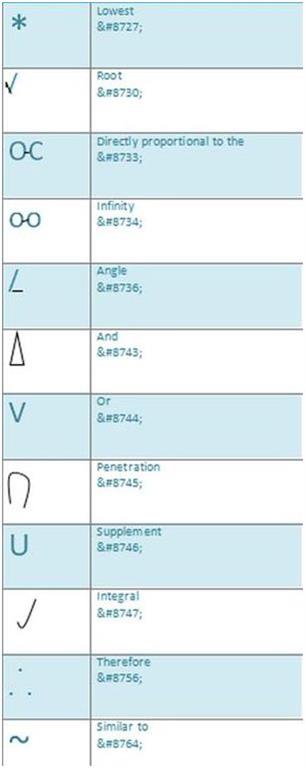
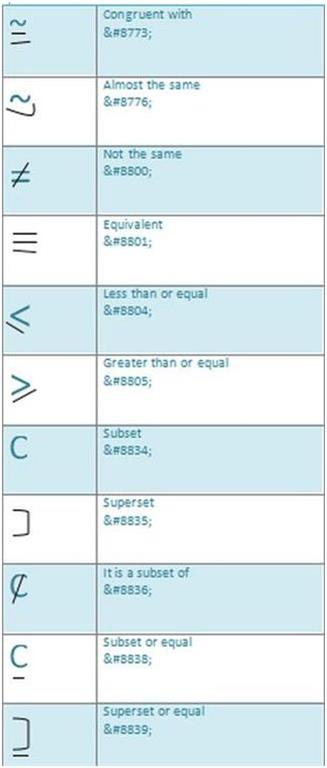
For more details, visit following link:

* [Learn HTML5 - Part 2: Web Workers](http://www.c-sharpcorner.com/UploadFile/dbd951/flexbox-in-html5-an-overview/)

Question 28: What are the special symbols in HTML 5?

**Answer**Special symbols, characters and any other related stuff play very important role in web development or any other kind of design and development work, since we require several special symbols or characters in all these activities directly or indirectly. Thus HTML5 provides several special symbols in its new features. These special symbols are presented in the form of several predefined set of codes.  
  
These special symbols provide these functionalities during web or any other type of design and development work or procedure.

* Extra features
* Description of the content
* Explains mathematical terms
* Reduce the content
* Provides proper explanation

**Special Symbols**  
Some of these symbols are as follows:  
  
  
  
  
  
  
  
For more details, visit following link:

* [Progress Bar in HTML 5](http://www.c-sharpcorner.com/UploadFile/2072a9/special-symbols-in-html5/)

Question 29: What are Web Workers APIs in HTML 5?

**Answer**  
Web Workers APIs provide a way in JavaScript to run something in the background that can do tasks without interfering with the user interface. As per the W3C standard "It is a JavaScript script executed from an HTML page that runs in the background, independently of other user-interface scripts that may also have been executed from the same HTML page. Web workers are able to utilize multi-core CPUs more effectively."  
  
**Types of Web Workers**

* Dedicated workers
* Shared workers

**Dedicated workers**

* A dedicated worker is accessible from the parent script that created it
* Wide browser support: All
* It is simply tied to its main connection

**Shared workers**

* A shared worker can be accessed from any script of the same origin.
* Limited browser support: Chrome 4.0+, Safari 5.0+ and Opera 10.6+.
* It can work with multiple connections.

Basically Web Workers work in the following three steps:

* First it should be executed on separate threads.
* It should be hosted in separate files from the main page.
* Finally a Worker object needs to be instantiated to call them.

For more details, visit following link:

* [Getting Started With Font-Awesome](http://www.c-sharpcorner.com/UploadFile/manish1231/learn-html5-part-2-web-workers/)

Question 30: Describe Progress Bar in HTML 5?

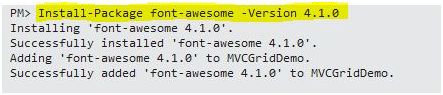
**Answer**Sometimes a task is running within a program that might take a while to complete. A user-friendly program provides some information to the user that the task is happening. It also tells how long the task might take and how much the task has been done or completed. One of the best way to show all of these activities is with the Progress Bar control.   
  
**Example**  
  
When you install a software or upload a file onto a website or when you download a file, you have seen a progress bar showing how the installation is progressing i.e download or upload has been done or how much is remaining.  
In a simple way, the progress bar indicates the progress of a specific task.  
  
In HTML 5, there is the element "progress" that displays the progress of a task.  
  
**Syntax**  
*<progress></progress>*  
**Attribute**It has 2 more attributes as shown in the following table:

|  |  |  |
| --- | --- | --- |
| **Attribute** | **Value** | **Description** |
| max | Floating Point Number | Specifies how much work the task requires in total, (Default value is 1.0) |
| value | Floating Point Number | Specifies how much of the task has been completed (This value will be in between 0.0<=value<=max) |

For more details, visit following link:

* [Various Kinds of Mouse Pointers For HTML Controls](http://www.c-sharpcorner.com/UploadFile/75a48f/progress-bar-in-html-5/)

Question 31: What is Font-Awesome in HTML 5?

**Answer**  
These is a simple text that can easily be manipulated like normal texts using the fonts. When you stretch or enlarge an image (PNG) icon, it becomes blurred, it looks bad in a simple language. If you enlarge any font from the package of Fontawesome then it would never become blurred or pixalated. Moreover, no width and no height is required as used in images as attributes as only by increasing the font-size, it manipulates the icon. The best thing is this package can be downloaded from the Nuget Package or the Package Manager console too. The following image will show how to install the package from the PM Console.  
  


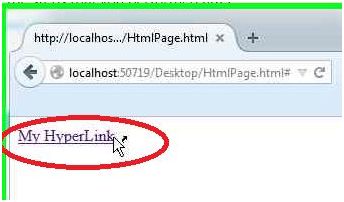
For more details, visit following link:

* [Getting Started With Font-Awesome](http://www.c-sharpcorner.com/UploadFile/0ef46a/getting-started-with-font-awesome/)

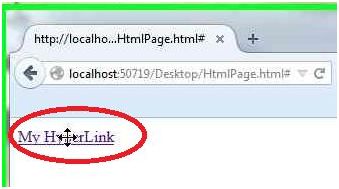
Question 32: What are the Kinds of Mouse Pointers For HTML Controls?

**Answer**Various Kinds of Mouse Pointers in HTML

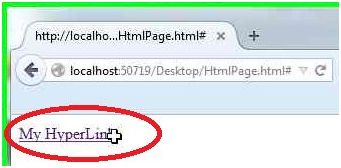
* **Alias:** It is used for a shortcut or to an alias somewhere.
  1. <a href="#" style="cursor:alias;">My HyperLink</a>

Visual Studio Tool Tip for this Mouse pointer.  
  
Alias  
  
After running the page, check the mouse pointer.  
  
 

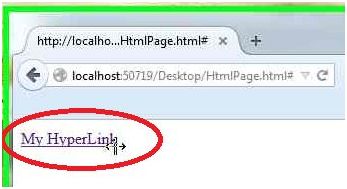
* **All:**Scroll: It is used to indicate to scroll in any direction.
  1. <a href="#" style="cursor: all-scroll;">My HyperLink</a>

Visual Studio Tool Tip for this Mouse pointer.  
  
All  
  
After running the page, check the mouse pointer.  
  


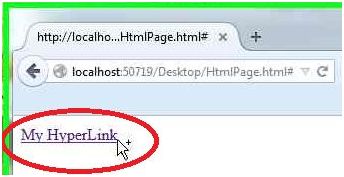
* **Cell:** It is used to select a cell or a set of cells.
  1. <a href="#" style="cursor: cell;">My HyperLink</a>

Visual Studio Tool Tip for this Mouse pointer.  
  
Cell  
  
After running the page, check the mouse pointer.  
  


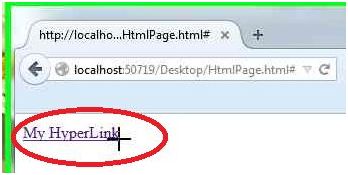
* **Col:** Resize: It is used to resize the columns horizontally.
  1. <a href="#" style="cursor: col-resize;">My HyperLink</a>

Visual Studio Tool Tip for this Mouse pointer.  
  
Col  
  
After running the page, check the mouse pointer.  
  


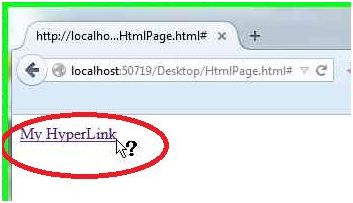
* **Copy:** It indicates that some text has been copied.
  1. <a href="#" style="cursor: copy;">My HyperLink</a>

Visual Studio Tool Tip for this Mouse pointer.  
  
Copy  
  
After running the page, check the mouse pointer.  
  


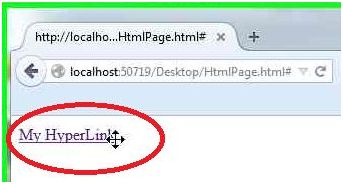
* Crosshair: With a "+" sign.
  1. <a href="#" style="cursor: crosshair;">My HyperLink</a>

Visual Studio Tool Tip for this Mouse pointer  
  
Crosshair  
  
After running the page, check the mouse pointer.  
  


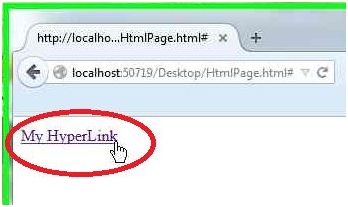
* **Help:** It is used to indicate the help.
  1. <a href="#" style="cursor help;">My HyperLink</a>

Visual Studio Tool Tip for this Mouse pointer.  
  
HyperLink  
  
After running the page, check the mouse pointer.  
  
 

* Move: It is used to indicate that something is moving.
  1. <a href="#" style="cursor: move;">My HyperLink</a>

Visual Studio Tool Tip for this Mouse pointer.  
  
Mouse pointer  
  
After running the page, check the mouse pointer.  
  


* Pointer: It is used to indicate a pointer.
  1. <a href="#" style="cursor: pointer;">My HyperLink</a>

Visual Studio Tool Tip for this Mouse pointer.  
  
Tool Tip  
  
After running the page, check the mouse pointer.  
  


For more details, visit following link:

* [Various Kinds of Mouse Pointers For HTML Controls](http://www.c-sharpcorner.com/UploadFile/a20beb/different-kinds-of-mouse-pointers-for-html-controls/)

Question 33: What are Frames in HTML?

**Answer**Frames allow multiple HTML documents to be present as independent windows within a main browser. They allow you to present two or more documents at once.  
 **Example**

1. <HTML>
3. <HEAD>
4. <TITLE>FRAME EXAMPLE</TITLE>
5. </HEAD>
6. <frameset cols="25%,\*,25%">
7. <frame src="FRAME1.html">
8. <frame src="FRAME2.html">
9. <frame src="FRAME3.html">
10. </frameset>
12. </HTML>

A <Frames document> can be declared using the <FRAMESET> element. A regular HTML Frameset document has a start element and an end element. This element can contain one or more elements. The SRC attribute of the frameset element points to the document that you want to display in a frame.The ROWS AND COLS attributes of frameset elements defines the layout of the frame.

1. <HTML>
3. <HEAD>
4. <TITLE>FRAM EXAMPLE</TITLE>
5. </HEAD>
6. <FRAMESET cols="20%, 80%">
7. <FRAMESET rows="100, 200">
8. <FRAME src="contents\_of\_frame1.html">
9. <FRAME src="contents\_of\_frame2.gif">
10. </FRAMESET>
11. </FRAMESET>

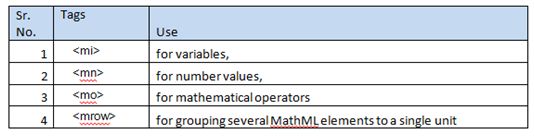
**Type of Frame-**  
There are two types of frames based on their layout.

1. *Vertical Frames*
2. *Horizontal Frames*

For more details, visit following link:

* [Working With Frames in HTML](http://www.c-sharpcorner.com/UploadFile/87b416/working-with-frames-in-html/)

Question 34: Describe MathML in HTML5?

**Answer**The Mathematical Markup Language (MathML) is a markup language to show mathematical and scientific content on the Web. HTML5 allows us to use MathML elements inside a document using <math>...</math> tags. A mathematical expression must be inserted into the element <math> with a specified namespace as in the following:  
 *<math xmlns="http://www.w3.org/1998/Math/MathML"> </math>*  
  
In HTML5 we can simply write <math></math>. The basic elements of the <math> tag are as follows:  
  
  
  
For more details, visit following link:

* [MathML With HTML5](http://www.c-sharpcorner.com/UploadFile/mscratnesh/mathml-with-html5926/)

Question 35: What are the HTML lists?

**Answer**In HTML, there are the following types of lists:

* **Unordered Lists (<ul>) -** The list items are marked with the bullets.
* **Ordered Lists (<ol>) -** The list items are marked with the numbers or letters.
* **Definition Lists(<dl>)-**This arranges your items in the same way as they are arranged in a dictionary.

All lists must contain one or more list elements.  
 **Unordered HTML Lists**  
An unordered list is a collection of related items that have no special order or sequence. An unordered list starts with the <ul> element. Each list item starts with the <li> element.  
  
The list items will be marked with bullets (black circles):

1. <!DOCTYPE html>
2. <html>
4. <head>
5. <title>HTML Unordered List</title>
6. </head>
8. <body>
9. <ul>
10. <li>JavaScript</li>
11. <li>ASP.NET</li>
12. <li>CSS</li>
13. <li>HTML</li>
14. </ul>
15. </body>
17. </html>

**Ordered HTML Lists**  
  
If you are required to put your items in a numbered list instead of bullet, then HTML ordered list will be used. This list is created using<ol> tag. The numbering starts at one and is incremented by one for each successive ordered list element starting with the <li> tag.  
  
**Example:**

1. <!DOCTYPE html>
2. <html>
4. <head>
5. <title>HTML Ordered List</title>
6. </head>
8. <body>
9. <ol>
10. <li>Beetroot</li>
11. <li>Ginger</li>
12. <li>Potato</li>
13. <li>Radish</li>
14. </ol>
15. </body>
17. </html>

**HTML Definition Lists**  
  
HTML supports a list style which is called definition lists where entries are listed like in a dictionary. The definition list is the ideal way to present a glossary, list of terms or other name/value list.  
  
Definition List makes use of the following three tags:

* **<dl> -** Defines the start of the list
* **<dt> -** A term
* **<dd> -** Term definition
* **</dl> -**Defines the end of the list

**Example:**

1. <!DOCTYPE html>
2. <html>
4. <head>
5. <title>HTML Definition Lists</title>
6. </head>
8. <body>
9. <dl>
10. <dt><b>ASP.NET</b></dt>
11. <dd>ASP.NET **is** easy to Learn</dd>
12. <dt><b>FTP</b></dt>
13. <dd>This stands **for** File Transfer Protocol</dd>
14. </dl>
15. </body>
17. </html>

For more details, visit following link:

* [Overview Of HTML Lists](http://www.c-sharpcorner.com/UploadFile/fa97a9/overview-of-html-lists/)

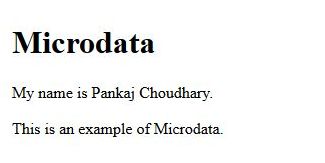
Question 36: What is Microdata in HTML 5?

**Answer**Microdata is used to nest metadata within an existing content on the web pages. This mechanism allows machine-readable data to be embedded in HTML documents in an easy-to-write manner, with an unambiguous parsing model. Microdata allows us to define our own customized elements and start embedding custom properties in our web pages. It’s purpose is not to make a new widget appear on our web page but to help automated programs like Google understand and better handle the content of our web pages.  
  
A Microdata consists of a group of name-value pairs. The group of name-value pairs is called items, each name-value property is called as a property and properties are represented by regular elements.  
  
Microdata defines five Global HTML attributes that can be applied to any HTML5 tag. Microdata introduces the following five global attributes that can be available for any element to use and give context for machines about our data.

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| Itemscope | Creates the Item and indicates that descendants of this element contain information about it. The itemscope attribute is a Boolean attribute that tells that there is Microdata on this page. |
| Itemtype | A valid URL of a vocabulary that describes the item and its properties context. |
| Itemprop | This attribute defines a property for the item.Indicates that its containing tag holds the value of the specified item property. |
| Itemid | Define a unique identifier of the item. |
| Itemref | Provides a list of element ids with additional properties elsewhere in the document. This attribute gives a list of additional elements to find the name-value pairs of the item. |

**Example:**

1. <html>
3. <head>
4. <title>Introduction to Microdata</title>
5. </head>
7. <body>
8. <div id="myTab">
9. </div>
10. <h1>Microdata</h1>
11. <div itemscopeitemtype="http://schema.org/Person">
12. <p>My name **is**
13. <spanitemprop="name">Pankaj Choudhary</span>.</p>
14. </div>
15. <div itemscopeitemtype="http://schema.org/Person">
16. <p>This **is** an example of
17. <span itemprop="name">Microdata</span>.</p>
18. </div>
19. </body>
21. </html>

**Output:**  
  
  
  
For more details, visit following link:

* [HTML5 Microdata](http://www.c-sharpcorner.com/UploadFile/f0b2ed/html5-microdata/)

Question 37: What is the difference between HTML5 vs HTML4.01?

**Answer**

1. **HTML4.01 vs. HTML5**In HTML4.01, we specify a Doctype element like this:  
     
   *<! Doctypehtml ”-//W3C//DTD HTML4.01 Transitional//EN”>*  
     
   In HTML5, we declare this element as:  
    *<!Doctype html>*
2. **HTML4.01 vs. HTML5**In HTML4.01, we generally need to mention a complete referenced tag line for enabling functionality of JavaScript or External CSS.  
     
   For example: For enabling JavaScript functionality we do:  
     
   *<script type=”text/javascript” src=”abc.js”>  
   </script>*  
   For HTML5, we only specify a source, not the type, as in:  
     
   *<script src=”abc.js”>  
   </script>*

**HTML5 | Tags**HTML5 makes our work easier and more precise using as it uses several new tags. These tags directly work without any link or reference.  
  
For example: For including a structuring element and a navigation or any other section in our HTML page, we need to create divisions or the margins for that so that these can work properly in HTML4. Like:

1. <html>
2. <head>
3. <title></title>
4. </head>
5. <body>
6. <div **class**="header">
7. <div **class**="nav">
8. <ol **class**="nav-list">
9. <li>
10. <a href="cshub.somee.com" title="Cshub" alt="Cshub">IT Encyclopedia </a>
11. </li>
12. <li>
13. <a href="picmaniac.brinkster.net" title=" Picmaniac" alt=" Picmaniac">Pic Maniac
14. </a>
15. </li>
16. <li>
17. <a href="jaiswalabhishek.blogspot.in" title="Google Says" alt=" Google Says">Google
18. Says </a>
19. </li>
20. </ol>
21. </div>
22. </div>
23. </body>
24. </html>

Using HTML5, we can do that as:

1. <html>
3. <head>
4. <title></title>
5. </head>
7. <body>
8. <header>
9. <nav>
10. <ol id="nav-list">
11. <li> <a href="cshub.somee.com" title="Cshub" alt="Cshub"> IT Encyclopedia </a></li>
12. <li><a href="picmaniac.brinkster.net" title=" Picmaniac" alt=" Picmaniac"> Pic Maniac </a></li>
13. <li> <a href="jaiswalabhishek.blogspot.in" title="Google Says" alt=" Google Says"> Google Says</a></li>
14. </ol>
15. </nav>
16. </header>
17. </body>
19. </html>

For more details, visit following link:

* [HTML5 vs HTML4.01](http://www.c-sharpcorner.com/UploadFile/2072a9/html5-vs-html4-01/)

Question 38: Discuss keyboard shortcut in HTML 5?

**Answer**For Displaying the Keyboard text, we can also create the keyboard shortcut to perform various operations such as clicking a link or a button. We can use the accesskey attribute when defining the element to provide the keyboard shortcut to that element or control. Let's create a Web Page, named "accesskey.html" and understand ‘How to create a keyboard shortcut.  
  
**First We will Write a Code for ‘accesskey.html:**

1. <!DOCTYPE HTML>
2. <html>
4. <head>
5. <title>Key Board Shortcut</title>
6. </head>
8. <body>
9. <h1>
10. Use the Shortcut keys to access the Content
11. </h1>
12. <p>
13. Press the <kbd>Alt + W</kbd> keys to navigate the following link :
14. </p>
15. <a href="XYZ.html" accesskey="w" target="">Open XYZ.html file. </a>
16. <p>
17. Press the <kbd>ALT + Z </kbd>keys to focus on the following text field
18. </p>
19. Enter Your Name :
20. <input type="text" name="name" accesskey="z">
21. <p>
22. Press the <kbd>ALT+S</kbd> keys to click the button to submit the form:</p>
23. <input type="submit" accesskey="s" onclick="alert('Form submitted successfully.')">
24. </body>
26. </html>

Here we have used a hyperlink, a text field and a button on the web page and assigned a keyboard shortcut using the accesskey attribute. The output of the above code is given below :  
  
  
  
Now When we press ALT + \* Key:

* **ALT + W :**The focus goes on the hyperlink and we areredirected to the page specified by the hyperlink (XYZ.html)
* **ALT+Z :** When we press ALT+Z key then get a focus on the Enter your Name text field.
* **ALT+S:**This focuses on Submit button and give an alert box :

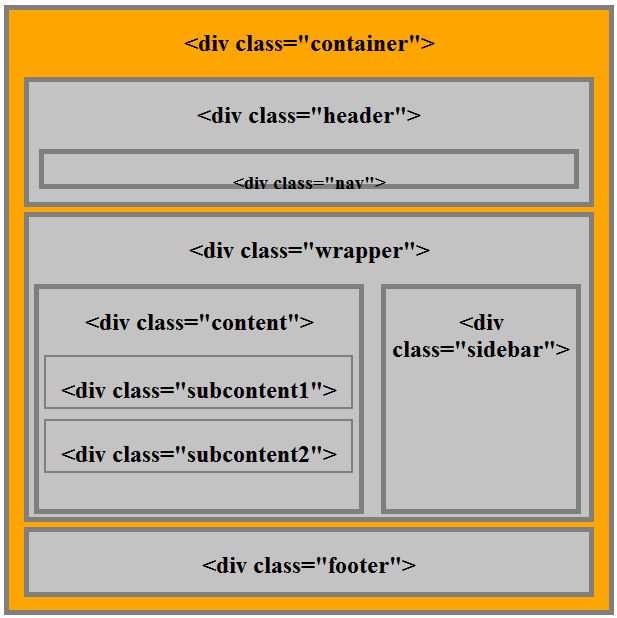
For more details, visit following link:

* [Keyboard Shortcut in HTML](http://www.c-sharpcorner.com/blogs/keyboard-shortcut-in-html1)

Question 39:What are HTML5 Semantic Elements?

**Answer**HTML5 introduced a new set of Semantic Elements. It help developers to define the structure of a page in an easy way.  
  
Semantic means 'meaning' and Semantic Element means elements with a meaning. A Semantic Element helps the developer and the browser to understand its meaning. There are two types of Semantic Elements:

* **Semantic:**These elements clearly define their content like <form>, <img>, <table> and so on.
* **Non-semantic:** These elements have no definition, they don't define anything about their content like <span> and <div>.

**Traditional HTML Layouts**Developers have been using <div> elements to manage entire structure of a HTML page that includes elements like a header, an article, a footer or a sidebar and you need to use an ID or class attributes to assign the role of the div elements. Let's design a traditional HTML layout before heading to HTML5 Semantic Elements.  
  
  
  
**New HTML5 Semantic Elements**  
HTML5 provides some new Semantic Elements that are semantically meaningful to describe a webpage layout. It's easy to understand and organize our code. It helps search engines to organize content more efficiently as well. The following is a list of the new HTML5 elements:

* <header>
* <nav>
* <section>
* <article>
* <aside>
* <figcaption>
* <figure>
* <footer>

For more details, visit following link:

* [Working With Semantic Elements in HTML5 With Layout Examples : Part 1](http://www.c-sharpcorner.com/UploadFile/b5be7f/working-with-new-semantic-elements-in-html5-along-with-html/)

Question 40: What is the difference between Article Vs Section tags in HTML5?

**Answer**He <article> element is a special type of <section>. <article>has more specific meaning as compared to <section> that is, it is independent, self-contained block of related content.  
  
We can use<section>, but we mostly use <article> to give more specific meaning to the content.  
  
<section> is only a block of a related content and <div> is only a block of content. We need to mention here that the pubdate attribute does not apply in <section>.

The pubdate attribute is used as one time attribute in <article>.If it is present, it indicates that the <time> element is the date, the <article> was published. It can be written in several ways, the most popular being:  
  
*pubdate  
pubdate="pubdate"*  
To decide, which of three elements is appropriate, choose the first suitable option:

1. Would the content make sense on its own in a feed reader? If so use <article>
2. Is the content related? if so use <section>
3. Finally, if there's no semantic relationship use <div>

Here is an example of <section> containing <article>:

1. <h1>Articles on: Fruit</h1>
2. <article>
3. <h2>Apple</h2>
4. <p>The apple **is** the pomaceous fruit of the apple tree...</p>
5. </article>
6. <article>
7. <h2>Orange</h2>
8. <p>The orange **is** a hybrid of ancient cultivated origin, possibly between pomelo and tangerine...</p>
9. </article>
10. <article>
11. <h2>Banana</h2>
12. <p>Bananas come **in** a variety of sizes and colors when ripe, including yellow, purple, and red...</p>
13. </article>
14. </section>

For more details, visit following link:

* [Difference Between Article and Section Tag](http://www.c-sharpcorner.com/blogs/difference-betweenarticleand-section-tag2)

Question 41: What are the advantages and limitations of HTML5 Web Worker

**Answer**  
  
**Advantages**  
We know the web browsers increased a lot over the past few years and it is primarily because of lot of work done is being done on its engines, ex- V8 (Google), Chakra (Microsoft). The JavaScript so far runs in a single thread. The problem with single threaded architecture is that it blocks the code and UI becomes unresponsive in case of running a complex script. There are various ways to solve this problem:

* Offload work to server but to make apps faster, fat client is preferred.
* Use asynchronous calls but many complex ecysystem of async calls & promises can lead to callback hell.
* Leverage multi threading. Interesting!

Web Workers solve this issue by providing capability of multi threading in JavaScript.   
  
**Features available to WW**  
  
Due to the multi-threaded behavior, a WW only has an access to the following subset of JavaScript features:

* The navigator object
* The location object
* XMLHttpRequest
* setTimeOut/clearTimeOut/setInterval/clearInterval
* importScripts
* spawning other WW

**Limitations of WW**  
It has some limitations apart from it's multithreading advantage, i.e., it can't access:

* DOM elements
* window object
* document object
* parent object

For more details, visit following link:

* [HTML 5 Web Workers](http://www.c-sharpcorner.com/UploadFile/9c2805/html5-web-workers-ww/)

Question 42: How can we use Details Tag in HTML5?

**Answer**The **<details> tag** is a new tag in HTML 5. The <details> tag specifies additional information or controls about the documents that the user can view or hide on demand. The contents of the details tag is hidden by default. The header is visible and can be shown when the user clicks on the header portion of <details> tag. The contents of the details tag are made visible by adding an open attribute to the <details> tag. The tag can also be used along with the summary tag to make your own header so that the user can expand or collapse the details of the document when required. This tag is only supported by Chrome.  
  
**Syntax**  
  
*<details open="open">  
<summary></summary>  
<p></p>  
</details>***Element-Specific Attribute**

|  |  |  |
| --- | --- | --- |
| **Attributes** | **Values** | **Description** |
| open | open | This Boolean attribute specifies whether the details of the data should be shown to the user or not. |

For more details, visit following link-:

* [Details Tag in HTML5](http://www.c-sharpcorner.com/uploadfile/667ddf/details-tag-in-html5/)

Question 43: What is figure tag in HTML 5?

**Answer**Figure tag is useful to markup a photo in the document. It is used to specify the self-contained block, images etc.  
Its content is belong to main document. But if we want to remove it, we can remove it independently as it does not effect the rest of the document.

1. <!DOCTYPE html>
2. <html>
3. <body>

<p>Are you going to the Mindcracker MVP Summit 2012 at Noida or interested in learning the daily progress of the summit? Check out daily updates here and let the saga begin.</p>  
  
*<figure>  
<imgsrc="1.jpg" alt="C# Corner" width="100" height="100" />  
</figure>*  
  
</body>  
</html>  
  
For more details, visit following link:

* [figure tag in HTML5](http://www.c-sharpcorner.com/blogs/figure-tag-in-html5)

Question 44: What are the formatting elements in HTML 5?

**Answer**Following is a list of formatting elements:

* Emphasized text
* Marked text
* Small text
* Deleted text
* Inserted text
* Subscripts
* Superscripts

Emphasized text– The syntax of Emphasized text element is <em> which defines emphasized text.  
  
**Example:**

1. <!DOCTYPE html>
2. <html>
4. <head>
5. <title>HTML formatting element sample</title>
6. </head>
8. <body>
9. <p>Hi. This **is** Raj Kumar Beniwal Here (This **is** normal text) </p>
10. <p><em> Hi. This **is** Raj Kumar Beniwal Here (This **is** normal text) </em> </p>
11. </body>
13. </html>

**Marked text–** The syntax of marked text element is <mark> which defines marked or highlighted text.  
  
**Example:**

1. <!DOCTYPE html>
2. <html>
4. <head>
5. <title>HTML formatting element sample</title>
6. </head>
8. <body>
9. <h2>Hi. This **is** <mark>Raj Kumar</mark> Beniwal Here. </h2>
10. </body>
12. </html>

**Small text–**The syntax of small text element is <small> which defines small text.  
  
**Example:**

1. <!DOCTYPE html>
2. <html>
4. <head>
5. <title>HTML formatting element sample</title>
6. </head>
8. <body>
9. <h1>Hi. This **is** <small>Raj Kumar</small> Beniwal Here. </h1>
10. </body>
12. </html>

**Deleted text–** The syntax of deleted text is <del> which defines deleted or removed text.  
  
**Example:**

1. <!DOCTYPE html>
2. <html>
4. <head>
5. <title>HTML formatting element sample</title>
6. </head>
8. <body>
9. <h1>Hi. This **is** <del>Raj Kumar</del> Beniwal Here. </h1>
10. </body>
12. </html>

**Inserted text–**The syntax of inserted text element is <ins> which defines inserted or added text.  
  
**Example:**

1. <!DOCTYPE html>
2. <html>
4. <head>
5. <title>HTML formatting element sample</title>
6. </head>
8. <body>
9. <h1>Hi. This **is** Raj <ins>Kumar</ins> Beniwal here. </h1>
10. </body>

**Subscripts text–**The syntax of subscripts text element is <sub> which defines subscripted text.  
  
**Example:**

1. <!DOCTYPE html>
2. <html>
4. <head>
5. <title>HTML formatting element sample</title>
6. </head>
8. <body>
9. <h1>Hi. This **is** <sub>Raj Kumar</sub> Beniwal here. </h1>
10. </body>
12. </html>

**Superscripts text–**The syntax of superscript text element is (sup> which defines superscripted text.  
  
**Example:**

1. <!DOCTYPE html>
2. <html>
4. <head>
5. <title>HTML formatting element sample</title>
6. </head>
8. <body>
9. <h1>Hi. This **is** <sup>Raj Kumar</sup> Beniwal here. </h1>
10. </body>
12. </html>

For more details, visit following link:

* [HTML 5 – Some important formatting elements](http://www.c-sharpcorner.com/blogs/html-5-some-important-formatting-elements)

Question 45: Why to use IndexedDB instead of Web SQL in HTML 5?

**Answer**This IndexedDB has been introduced with [HTML5](http://www.c-sharpcorner.com/technologies/html5). This allows a user to store large amount of data in the browser. It has been proved that IndexedDB is more powerful and efficient than other storage mechanisms like Local Storage and Session Storage. IndexedDB is an API which helps the developers to do some database operations in the client side like creating a database, opening the transaction, creating tables, inserting values to the tables, deleting the values, and reading the data. If you need any other way to save some data in the client side, you can use storage mechanisms introduced in HTML5.   
  
W3C has announced that use of Web SQL is obsolete and deprecated, hence it is not recommended to use Web SQL in your Applications. Most of the modern web browsers like Mozilla do not support the use of Web SQL and this is also a great limitation of Web SQL.  
  
Now we have an alternative to Web SQL. IndexedDB is more efficient and faster than Web SQL. The following are some of the main advantages of IndexedDB.

* It stores the data as Key-Pair values
* It is asynchronous
* It is non-relational
* Can access the data from the same domain

For more details, visit following link:

* [Introduction to IndexedDB](http://www.c-sharpcorner.com/UploadFile/65794e/introduction-to-indexeddb/)

Question 46: How you can Use Modernizr in HTML 5?

**Answer**  
  
Modernizr is an open source JavaScript library that helps to detect features of HTML5 and CSS3.  
  
You can download the Modernizr library from this [link](http://modernizr.com/).  
  
The good thing about the Modernizr JS is that we can download this depending on the requirements. In other words, if an application needs to check for only few features of HTML5 and CSS3 then select those features and download the file. That JavaScript file will contain only the source code for the selected features.   
 **How to use Modernizr JS.**  
On every page, you need to add the following tag:  
  
JavaScript    
Add the Modernize.js like we add any JavaScript file as shown below:  
Modernize.js  
In the following source code, if the HTML5 Canvas tag is supported then the following picture will draw on the Canvas object else the same kind of picture will be shown using the image tag.  
  
   
  
For more details, visit following link-

* [How to Use Modernizr](http://www.c-sharpcorner.com/UploadFile/0e28e5/how-to-use-modernizr/)

Question 47: What is the difference between Canvas and SVG elements?

**Answer**  
  
SVG is a part of the Vector based family of graphics. It is completely different from Raster based graphics. The most common Raster-based formats used on the web today are JPEG, GIF and PNG. SVG is a W3C recommendation. SVG graphics do NOT lose any quality if they are zoomed or resized.  
  
The <canvas> element:

* Helps the browser to draw shapes and images without any plugin.
* Used to draw graphics.
* Has several methods to draw paths, boxes, circles, characters and add images.

**Comparison Of SVG and Canvas**

|  |  |
| --- | --- |
| **SVG** | **Canvas** |
| Object Model-based (SVG elements are similar to HTML elements) | Pixel-based (the canvas is essentially an image element with a drawing API) |
| Multiple graphical elements that become part of the Document Object Model (DOM) | Single HTML element similar to <img> in behavior |
| Visual presentation created with markup and modified by CSS or programmatically through script | Visual presentation created and modified programmatically through script |
| Event model/user interaction is object-based at the level of primitive graphic elements, suchg as lines, rectangles and paths | Event model/user interaction is coarse, at the canvas element only; interactions must be manually programmed from mouse coordinates |
| SVG markup and object model directly support accessibility | The API does not support accessibility; markup-based techniques must be used in addition to canvas |

**Some other Differences between Canvas and SVG elements**

|  |  |
| --- | --- |
| **Canvas** | **SVG** |
| Canvas draws 2D graphics, on the fly (with a JavaScript). | SVG defines the graphics in XML format. |
| Resolution dependent. | Resolution independent. |
| Canvas is rendered pixel by pixel. | In SVG, each drawn shape is remembered as an object. |

For more details, visit following link:

* [HTML5 SVG](http://www.c-sharpcorner.com/UploadFile/18ddf7/html5-svg/)

Question 48: What is the use of Language attribute in HTML 5?

**Answer**The Language attribute-

* The document language can be declared in the <html> tag.
* The language is declared in the language attribute.
* Declaring a language is important for accessibility applications (screen readers) and search engines.

**Example**

1. <!DOCTYPE html>
2. <html lang="en-US">
4. <body>
5. <h1>My First Heading</h1>
6. <p>My first paragraph.</p>
7. </body>
9. </html>

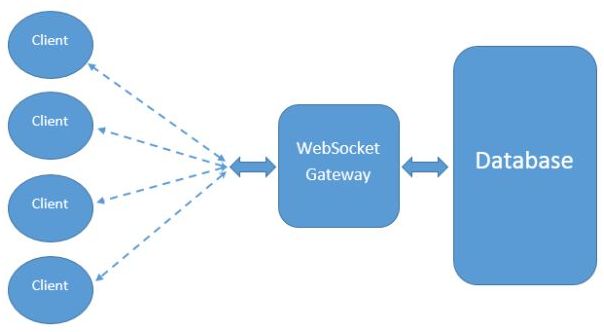
The first two letters specify the language (en). If there is a dialect, use two more letters (US).  
  
For more details, visit following link:

* [HTML For Beginners: Part 3](http://www.c-sharpcorner.com/UploadFile/fc2f0e/html-for-beginnerspart-3/)

Question 49: What is the use of WebSocket API?

**Answer**WebSockets provide a rich protocol to perform bi-directional communication and we can create a full-duplex communication channel that can be operated through a single socket over the Web and for that reason its more attractive for the things like games, messaging apps and for real-time updates in both directions.  
  
WebSocket is basically used to reduce the overhead of HTTP, since it has its own protocol defined by IETF and an API for the server communication. By using them, the client notifies the WebSocket server with the recipients ID of an event and the server immediately notifies all the active clients and the last clients processes the event when the given recipient ID matches the client ID.  
  
**Key Features of WebSocket**

* WebSocket is a rich protocol to perform bi-directional communication.
* It’s a full-duplex communication channel that can operate through a single socket over the Web. Your request reuses the same connection from the client to the server and the server to the client.
* It makes a single request and the single request greatly reduces the latency over polling.
* Communication becomes more efficient since bandwidth, CPU power and latency is saved.
* You can build other standard protocols on top of its protocol.
* WebSocket is a feature that makes HTML5 more advanced.
* WebSocket is about simplicity.

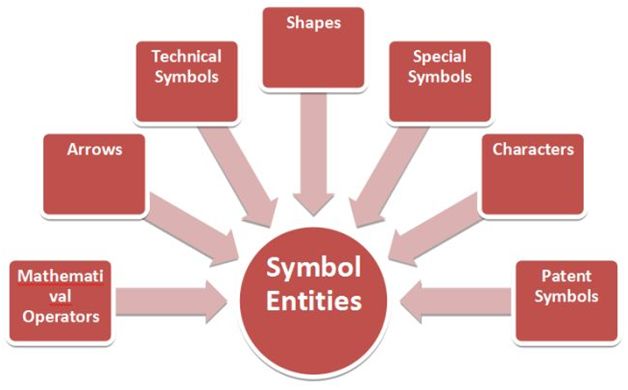
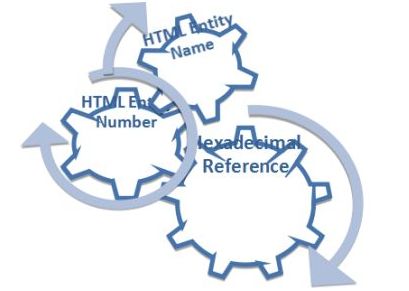
   
  
For more details, visit following link:

* [Learn HTML5 - Part 4: WebSocket API](http://www.c-sharpcorner.com/UploadFile/manish1231/learn-html5-part-4-websocket-api/)

Question 50: What are HTML 5 Symbol Entities?

**Answer**There is a wide category of HTML symbols for various purposes and applications like:

* Mathematical operators
* Arrows
* Technical symbols
* Shapes
* Special Symbols and so on

 **Functionalities | HTML5 Symbols**  
  
Several special symbols are not present on a normal keyboard. We can use special symbols along with several combinations of attributes and the entities available in HTML5. There are generally 3 ways or procedures through which we can access or use all the functionalities provided by HTML5.   
 **Methods | HTML5 Symbols**  
  
To add these symbols to an HTML page, these methods are:  
  
  
  
For more details, visit following link:

* [HTML5 Symbol Entities](http://www.c-sharpcorner.com/UploadFile/2072a9/html5-symbol-entities/)

Question 51: What is Template tag in HTML 5?

**Answer**W3C has introduced a new "template" tag that provides a mechanism to define HTML markup fragments as prototypes. In practice, a template can be used to insert fragments of HTML code into your page, for example:

1. <template id="rowTemplate">
2. <tr>
3. <td **class**="record"></td>
4. <td></td>
5. <td></td>
6. </tr>
7. </template>

**Features**  
  
The following are the features of the template tag:

* The template code can be defined nearly anywhere; the head, body or even a frameset.
* Templates will not be displayed
* Templates are not considered to be a part of the document. In other words, using document.getElementById(“mytablerow”) will not return child nodes.
* Templates are inactive until used. In other words, enclosed images will not download, media will not play, scripts will not run and so on.

**Using templates**  
To use a template, it must be cloned and inserted into the DOM. For example, assuming the following HTML:

1. <table id="testTable">
2. <thead>
3. <tr>
4. <td>
5. ID
6. </td>
7. <td>
8. name
9. </td>
10. <td>
11. twitter
12. </td>
13. </tr>
14. </thead>
15. <tbody>
16. <!-- rows to be appended here -->
17. </tbody>
18. </table>
19. <!-- row template -->
20. <template id="rowTemplate">
21. <tr>
22. <td **class**="record"></td>
23. <td></td>
24. <td></td>
25. </tr>
26. </template>

Use the following to clone the new row and append it to the table in JavaScript:

1. // get tbody and row template
2. var t = document.querySelector("#testTabletbody"),
3. row = document.getElementById("rowTemplate");
4. // modify row data
5. var td = row.getElementsByTagName("td");
6. td[0].textContent = "1";
7. td[1].textContent = "Sunny";
8. td[2].textContent = "@sunny\_delhi";
9. // clone row and insert into table
10. t.appendChild(row.content.cloneNode(**true**));

For more details, visit the following link:

* [Template Tag in HTML5](http://www.c-sharpcorner.com/UploadFile/370e35/template-tag-in-html5/)

Question 52: What is the difference between XHTML and HTML?

**Answer**  
  
**XHTML**XHTML stands for ”eXtensible Hyper Markup Language”. It is an extended version of HTML 4.01. XHTML is HTML which is defined in XML format .Therefore, it is an XML application. XHTML is basically supported by almost all the available major browsers like Internet Explorer, Opera, Safari, and Chrome.  
  
There are many web pages available on the internet that contain “bad HTML” means those don't follow the actual principle of HTML but still execute very well.  
  
For example,

1. <html>
3. <head>
4. <title>
5. This **is** a bad HTML page
6. </title>
7. </head>
9. <body>
10. <h1>Bad HTML
11. <p> This **is** a web page that contain bad HTML.
12. </body>
13. </html>

**HTML**

* HTML is proposed by Tin-Berners-Lee in 1987.
* HTML is developed by W3C and WHATWG.
* HTML files are saved with .html and .htm extension.
* HTML is an SGML (Standard Generalized Markup Language) application.

**Example**

1. <html>
3. <head>
4. <title>This **is** an XHTML page </title>
5. </head>
7. <body>
8. <p> This **is** my first XHTML page
9. </body>
11. </html>

What is HTML5?

HTML5 is the next major revision of the HTML standard superseding HTML 4.01, XHTML 1.0, and XHTML 1.1. HTML5 is a standard for structuring and presenting content on the World Wide Web.

Name some of the new features of HTML5.

HTML5 introduces a number of new elements and attributes that helps in building a modern websites. Following are great features introduced in HTML5 −

* **New Semantic Elements −** These are like <header>, <footer>, and <section>.
* **Forms 2.0 −** Improvements to HTML web forms where new attributes have been introduced for <input> tag.
* **Persistent Local Storage −** To achieve without resorting to third-party plugins.
* **WebSocket −** A a next-generation bidirectional communication technology for web applications.
* **Server-Sent Events −** HTML5 introduces events which flow from web server to the web browsers and they are called Server-Sent Events (SSE).
* **Canvas −** This supports a two-dimensional drawing surface that you can program with JavaScript.
* **Audio & Video −** You can embed audio or video on your web pages without resorting to third-party plugins.
* **Geolocation −** Now visitors can choose to share their physical location with your web application.
* **Microdata −** This lets you create your own vocabularies beyond HTML5 and extend your web pages with custom semantics.
* **Drag and drop −** Drag and drop the items from one location to another location on a the same webpage.

Which browsers support HTML5?

The latest versions of Apple Safari, Google Chrome, Mozilla Firefox, and Opera all support many HTML5 features and Internet Explorer 9.0 will also have support for some HTML5 functionality.

The mobile web browsers that come pre-installed on iPhones, iPads, and Android phones all have excellent support for HTML5.

Is HTML5 backward compatible with old browsers?

Yes! HTML5 is designed, as much as possible, to be backward compatible with existing web browsers. New features build on existing features and allow you to provide fallback content for older browsers.

It is suggested to detect support for individual HTML5 features using a few lines of JavaScript.

Are HTML tags case sensitive?

No!

What is the purpose of 'section' tag in HTML5?

This tag represents a generic document or application section. It can be used together with h1-h6 to indicate the document structure.

What is the purpose of 'article' tag in HTML5?

This tag represents an independent piece of content of a document, such as a blog entry or newspaper article.

What is the purpose of 'aside' tag in HTML5?

This tag represents a piece of content that is only slightly related to the rest of the page.

What is the purpose of 'header' tag in HTML5?

This tag represents the header of a section.

What is the purpose of 'footer' tag in HTML5?

This tag represents a footer for a section and can contain information about the author, copyright information, et cetera.

What is the purpose of 'nav' tag in HTML5?

This tag represents a section of the document intended for navigation.

What is the purpose of 'dialog' tag in HTML5?

This tag can be used to mark up a conversation.

What is the purpose of 'figure' tag in HTML5?

This tag can be used to associate a caption together with some embedded content, such as a graphic or video.

What are custom attributes in HTML5?

A custom data attribute starts with data- and would be named based on your requirement. Following is the simple example−

<div class="example" data-subject="physics" data-level="complex">

...

</div>

The above will be perfectly valid HTML5 with two custom attributes called data-subject and data-level. You would be able to get the values of these attributes using JavaScript APIs or CSS in similar way as you get for standard attributes.

What is Web Forms 2.0?

Web Forms 2.0 is an extension to the forms features found in HTML4. Form elements and attributes in HTML5 provide a greater degree of semantic mark-up than HTML4 and remove a great deal of the need for tedious scripting and styling that was required in HTML4.

What is the purpose of datetime input control in Web form 2.0?

It represents a date and time (year, month, day, hour, minute, second, fractions of a second) encoded according to ISO 8601 with the time zone set to UTC.

What is the purpose of datetime-local input control in Web form 2.0?

It represents a date and time (year, month, day, hour, minute, second, fractions of a second) encoded according to ISO 8601 with no time zone information.

What is the purpose of date input control in Web form 2.0?

It represents a date (year, month, day) encoded according to ISO 8601.

What is the purpose of month input control in Web form 2.0?

It represents a date consisting of a year and a month encoded according to ISO 8601.

What is the purpose of week input control in Web form 2.0?

It represents a date consisting of a year and a week number encoded according to ISO 8601.

What is the purpose of time input control in Web form 2.0?

It represents a time (hour, minute, seconds, fractional seconds) encoded according to ISO 8601.

What is the purpose of number input control in Web form 2.0?

This control accepts only numerical value. The step attribute specifies the precision, defaulting to 1.

What is the purpose of range input control in Web form 2.0?

The range type is used for input fields that should contain a value from a range of numbers.

What is the purpose of email input control in Web form 2.0?

This accepts only email value. This type is used for input fields that should contain an e-mail address. If you try to submit a simple text, it forces to enter only email address in email@example.com format.

What is the purpose of url input control in Web form 2.0?

This accepts only URL value. This type is used for input fields that should contain a URL address. If you try to submit a simple text, it forces to enter only URL address either in http://www.example.com format or in http://example.com format.

What is the purpose of 'output' tag in HTML5?

HTML5 introduced a new element <output> which is used to represent the result of different types of output, such as output written by a script.

What is the purpose of 'placeholder' attribute in HTML5?

HTML5 introduced a new attribute called placeholder. This attribute on <input> and <textarea> elements provides a hint to the user of what can be entered in the field. The placeholder text must not contain carriage returns or line-feeds.

What is the purpose of 'autofocus' attribute in HTML5?

This is a simple one-step pattern, easily programmed in JavaScript at the time of document load, automatically focus one particular form field.

What is the purpose of 'required' attribute in HTML5?

HTML5 introduced a new attribute called required which would insist to have a value in an input control.

Can you use SVG tags directly in HTML5 without any plugin?

Yes! HTML5 allows embeding SVG directly using <svg>...</svg> tag.

Can you use MathML tags directly in HTML5 without any plugin?

Yes! The HTML syntax of HTML5 allows for MathML elements to be used inside a document using <math>...</math> tags.

What are the drawbacks of cookies?

Cookies have following drawbacks−

* Cookies are included with every HTTP request, thereby slowing down your web application by transmitting the same data.
* Cookies are included with every HTTP request, thereby sending data unencrypted over the internet.
* Cookies are limited to about 4 KB of data . Not enough to store required data.

What do you mean by session storage in HTML5?

HTML5 introduces the sessionStorage attribute which would be used by the sites to add data to the session storage, and it will be accessible to any page from the same site opened in that window i.e. session and as soon as you close the window, session would be lost.

What do you mean by local storage in HTML5?

HTML5 introduces the localStorage attribute which would be used to access a page's local storage area without no time limit and this local storage will be available whenever you would use that page.

When a session storage data gets deleted?

The Session Storage Data would be deleted by the browsers immediately after the session gets terminated.

When a local storage data gets deleted?

local storage data has no time limit. To clear a local storage setting you would need to call localStorage.remove('key'); where 'key' is the key of the value you want to remove. If you want to clear all settings, you need to call localStorage.clear() method.

What is Server Side Events in HTML5?

Along with HTML5, WHATWG Web Applications 1.0 introduces events which flow from web server to the web browsers and they are called Server-Sent Events (SSE). Using SSE you can push DOM events continously from your web server to the visitor's browser.

The event streaming approach opens a persistent connection to the server, sending data to the client when new information is available, eliminating the need for continuous polling.

Server-sent events standardizes how we stream data from the server to the client.

How to utilize a server-sent event in HTML5?

To use Server-Sent Events in a web application, you would need to add an <eventsource> element to the document.

The src attribute of <eventsource> element should point to an URL which should provide a persistent HTTP connection that sends a data stream containing the events.

The URL would point to a PHP, PERL or any Python script which would take care of sending event data consistently.

What are the steps of server side scripts for SSE?

server side script should send Content-type header specifying the type text/event-stream as follows−

print "Content-Type: text/event-stream\n\n";

After setting Content-Type, server side script would send an Event − tag followed by event name. Following example would send Server-Time as event name terminated by a new line character.

print "Event: server-time\n";

Final step is to send event data using Data − tag which would be followed by integer of string value terminated by a new line character as follows−

$time = localtime();

print "Data: $time\n";

What are web sockets?

Web Sockets is a next-generation bidirectional communication technology for web applications which operates over a single socket and is exposed via a JavaScript interface in HTML 5 compliant browsers.

Once you get a Web Socket connection with the web server, you can send data from browser to server by calling a send() method, and receive data from server to browser by an onmessage event handler.

Following is the API which creates a new WebSocket object.

var Socket = new WebSocket(url, [protocal] );

Here first argument, url, specifies the URL to which to connect. The second attribute, protocol is optional, and if present, specifies a sub-protocol that the server must support for the connection to be successful.

What is the purpose of Socket.readyState atribute of WebSocket?

The readonly attribute readyState represents the state of the connection. It can have the following values:

* A value of 0 indicates that the connection has not yet been established.
* A value of 1 indicates that the connection is established and communication is possible.
* A value of 2 indicates that the connection is going through the closing handshake.
* A value of 3 indicates that the connection has been closed or could not be opened.

What is the purpose of Socket.bufferedAmount atribute of WebSocket?

The readonly attribute bufferedAmount represents the number of bytes of UTF-8 text that have been queued using send() method.

What is the purpose of 'canvas' tag in HTML5?

HTML5 element <canvas> gives you an easy and powerful way to draw graphics using JavaScript. It can be used to draw graphs, make photo compositions or do simple (and not so simple) animations.

What is the purpose of 'audio' tag in HTML5?

HTML5 supports <audio> tag which is used to embed sound content in an HTML or XHTML document. The current HTML5 draft specification does not specify which audio formats browsers should support in the audio tag. But most commonly used audio formats are ogg, mp3 and wav.

You can use <source> tag to specify media along with media type and many other attributes. An audio element allows multiple source elements and browser will use the first recognized format.

What is the purpose of 'video' tag in HTML5?

HTML5 supports <video> tag which is used to embed a video file in an HTML or XHTML document.The current HTML5 draft specification does not specify which video formats browsers should support in the video tag. But most commonly used video formats are−

* **Ogg −** Ogg files with Thedora video codec and Vorbis audio codec.
* **mpeg4 −** MPEG4 files with H.264 video codec and AAC audio codec.

You can use <source> tag to specify media along with media type and many other attributes. An audio element allows multiple source elements and browser will use the first recognized format.

What is Geolocation API in HTML?

HTML5 Geolocation API lets you share your location with your favorite web sites. A Javascript can capture your latitude and longitude and can be sent to backend web server and do fancy location-aware things like finding local businesses or showing your location on a map.

Today most of the browsers and mobile devices support Geolocation API. The geolocation APIs work with a new property of the global navigator object ie. Geolocation object which can be created as follows:

var geolocation = navigator.geolocation;

The geolocation object is a service object that allows widgets to retrieve information about the geographic location of the device.

What is purpose of getCurrentPosition() method of geolocation object of HTML5?

This method retrieves the current geographic location of the user.

What is purpose of watchPosition() method of geolocation object of HTML5?

This method retrieves periodic updates about the current geographic location of the device.

What is purpose of clearPosition() method of geolocation object of HTML5?

This method cancels an ongoing watchPosition call.

What are Web Workers?

Web Workers do all the computationally expensive tasks without interrupting the user interface and typically run on separate threads.

Web Workers allow for long-running scripts that are not interrupted by scripts that respond to clicks or other user interactions, and allows long tasks to be executed without yielding to keep the page responsive.

**1) What is HTML?**

HTML is short for HyperText Markup Language and is the language of the World Wide Web. It is the standard text formatting language used for creating and displaying pages on the Web. HTML documents are made up of two things: the content and the tags that format it for proper display on pages.

**2) What are tags?**

Content is placed in between HTML tags in order to properly format it. It makes use of the less than symbol (<) and the greater than symbol (>). A slash symbol is also used as a closing tag. For example:



|  |  |
| --- | --- |
| 1 | <strong>sample</strong> |

**3) Do all HTML tags come in a pair?**

No, there are single HTML tags that do not need a closing tag. Examples are the <img> tag and <br> tags.

**4) What are some of the common lists that can be used when designing a page?**

You can insert any or a combination of the following list types:  
– ordered list  
– unordered list  
– definition list  
– menu list  
– directory list  
Each of this list types makes use of a different tag set to compose

**5) How do you insert a comment in HTML?**



Comments in HTML begins with “<!–“nd ends with “–>”. For example:



|  |  |
| --- | --- |
| 1 | <!-- A SAMPLE COMMENT --> |

**6) Do all character entities display properly on all systems?**

No, there are some character entities that cannot be displayed when the operating system that the browser is running on does not support the characters. When that happens, these characters are displayed as boxes.

**7) What is an image map?**

Image map lets you link to many different web pages using a single image. You can define shapes in images that you want to make part of an image mapping.

**8 ) What is the advantage of collapsing white space?**

White spaces are a blank sequence of space characters, which is treated as a single space character in HTML. Because the browser collapses multiple spaces into a single space, you can indent lines of text without worrying about multiple spaces. This enables you to organize the HTML code into a much more readable format.

**9) Can attribute values be set to anything or are there specific values that they accept?**

Some attribute values can be set to only predefined values. Other attributes can accept any numerical value that represents the number of pixels for a size.

**10) How do you insert a copyright symbol on a browser page?**

To insert the copyright symbol, you need to type &copy; or & #169; in an HTML file.

**11) How do you create links to sections within the same page?**

Links can be created using the <a> tag, with referencing through the use of the number (#) symbol. For example, you can have one line as <a href=”#topmost”>BACK TO TOP</a>, which would result in the words “BACK TO TOP” appearing on the webpage and links to a bookmark named topmost. You then create a separate tag command like <a name=”topmost”> somewhere on the top of the same webpage so that the user will be linked to that spot when he clicked on “BACK TO TOP”.

**12) Is there any way to keep list elements straight in an HTML file?**

By using indents, you can keep the list elements straight. If you indent each subnested list in further than the parent list that contains it, you can at a glance determine the various lists and the elements that it contains.

**13) If you see a web address on a magazine, to which web page does it point?**

Every web page on the web can have a separate web address. Most of these addresses are relative to the top-most web page. The published web address that appears within magazines typically points this top-most page. From this top level page, you can access all other pages within the website.

**14) What is the use of alternative text in image mapping?**

When you use image maps, it can easily become confusing and difficult to determine which hotspots correspond to which links. Using alternative text lets, you put a descriptive text on each hotspot link.

**15) Do older HTML files work on newer browsers?**

Yes, older HTML files are compliant to the HTML standard. Most older files work on the newer browsers, though some features may not work.

**16) Does a hyperlink apply to text only?**

No, hyperlinks can be used in the text as well as images. That means you can convert an image into a link that will allow users to link to another page when clicked. Surround the image within the <a href=” “>…</a> tag combinations.

**17) If the user’s operating system does not support the needed character, how can the symbol be represented?**

In cases wherein their operating system does not support a particular character, it is still possible to display that character by showing it as an image instead.

**18) How do you change the number type in the middle of a list?**

The <li> tag includes two attributes – type and value. The type attribute can be used to change the numbering type for any list item. The value attribute can change the number index.

**19) What are style sheets?**

Style sheets enable you to build consistent, transportable, and well-defined style templates. These templates can be linked to several different web pages, making it easy to maintain and change the look and feel of all the web pages within site.

**20) State  bullet types available in HTML**

With ordered lists, you can select to use some different list types including alphabetical and Roman numerals. The type attribute for unordered lists can be set to disc, square, or circle.

**21) How do you create multicolored text in a webpage?**

To create text with different colors, use the <font color=”color”>…</font> tags for every character that you want to apply color. You can use this tag combination as many times as needed, surrounding a single character or an entire word.

**22) Why are there both numerical and named character entity values?**

The numerical values are taken from the ASCII values for the various characters, but these can be difficult to remember. Because of this, named character entity values were created to make it easier for web page designers to use.

**23) Write an HTML table tag sequence that outputs the following:  
50 pcs 100 500  
10 pcs 5 50**

Answer:



|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12 | <table>  <tr>  <td>50 pcs</td>  <td>100</td>  <td>500</td>  </tr>  <tr>  <td>10 pcs</td>  <td>5</td>  <td>50</td>  </tr>  </table> |

**24) What is the advantage of grouping several checkboxes together?**

Although checkboxes don’t affect one another, grouping checkboxes together help to organize them. Checkbox buttons can have their name and do not need to belong to a group. A single web page can have many different groups of checkboxes.

**25) What will happen if you overlap sets of tags?**

If two sets of HTML tags are overlapped, only the first tag will be recognized. You will find this problem when the text does not display properly on the browser screen.

**26) What are applets?**

Applets are small programs that can be embedded within web pages to perform some specific functionality, such as computations, animations, and information processing. Applets are written using the Java language.

**27) What if there is no text between the tags or if a text was omitted by mistake? Will it affect the display of the HTML file?**

If there is no text between the tags, then there is nothing to format. Therefore no formatting will appear. Some tags, especially tags without a closing tag like the <img> tag, do not require any text between them.

**28) Is it possible to set specific colors for table borders?**

You can specify a border color using style sheets, but the colors for a table that does not use style sheets will be the same as the text color.

**29) How do you create a link that will connect to another web page when clicked?**

To create hyperlinks, or links that connect to another web page, use the href tag. The general format for this is: <a href=”site”>text</a>  
Replace “site” with the actual page URL that is supposed to be linked to when the text is clicked.

**30) What other ways can be used to align images and wrap text?**

Tables can be used to position text and images. Another useful way to wrap text around an image is to use style sheets.

**31) Can a single text link point to two different web pages?**

No. The <a> tag can accept only a single href attribute, and it can point to only a single web page.

**32) What is the difference between the directory and menu lists and the unordered list?**

The key difference is that the directory and menu lists do not include attributes for changing the bullet style.

**33) Can you change the color of bullets?**

The bullet color is always the same as that of the first character in the list item. If you surround the <li> and the first character with a set of <font> tags with the color attribute set, the bullet color, and the first character will be a different color from the text.

**34) What are the limits of the text field size?**

The default size for a text field is around 13 characters. However, if you include the size attribute, you can set the size value to be as low as 1. The maximum size value will be determined by the browser width. If the size attribute is set to 0, the size will be set to the default size of 13 characters.

**35) Do <th> tags always need to come at the start of a row or column?**

Any <tr> tag can be changed to a <th> tag. This causes the text contained within the <th> tag to be displayed as bold in the browser. Although <th> tags are mainly used for headings, they do not need to be used exclusively for headings.

**36) What is the relationship between the border and rule attributes?**

Default cell borders, with a thickness of 1 pixel, are automatically added between cells if the border attribute is set to a nonzero value. Likewise, If the border attribute is not included, a default 1-pixel border appears if the rules attribute is added to the <table> tag.

**37) What is a marquee?**

A marquee allows you to put a scrolling text in a web page. To do this, place whatever text you want to appear scrolling within the <marquee> and </marquee> tags.

**38) How do you create text on a webpage that will allow you to send an email when clicked?**

To change text into a clickable link to send email, use the mailto command within the href tag. The format is as follows:



|  |  |
| --- | --- |
| 1 | <A HREF=”mailto:youremailaddress”>text to be clicked</A> |

**39) Are <br> tags the only way to separate sections of text?**

No. The <br> tag is only one way to separate lines of text. Other tags, like the <p> tag and <blockquote> tag, also separate sections of text.

**40) Are there instances where the text will appear outside of the browser?**

By default, the text is wrapped to appear within the browser window. However, if the text is part of a table cell with a defined width, the text could extend beyond the browser window.

**41) How are active links different from normal links?**

The default color for normal and active links is blue. Some browsers recognize an active link when the mouse cursor is placed over that link; others recognize active links when the link has the focus. Those that don’t have a mouse cursor over that link is considered a normal link.

**42) Do style sheets limit the number of new style definitions that can be included within the brackets?**

Style sheets do not limit the number of style definitions that can be included within the brackets for a given selector. Every new style definition, however, must be separated from the others by a semicolon symbol.

**43) Can I specify fractional weight values such as 670 or 973 for font weight?**

Implementation largely depends on the browser, but the standard does not support fractional weight values. Acceptable values must end with two zeroes.

**44) What is the hierarchy that is being followed when it comes to style sheets?**

If a single selector includes three different style definitions, the definition that is closest to the actual tag takes precedence. Inline style takes priority over embedded style sheets, which takes priority over external style sheets.

**45) Can several selectors with class names be grouped together?**

You can define several selectors with the same style definition by separating them with commas. This same technique also works for selectors with class names.

**46) What happens if you open the external CSS file in a browser?**

When you try to open the external CSS file in a browser, the browser cannot open the file, because the file has a different extension. The only way to use an external CSS file is to reference it using <link/> tag within another HTML document.

**47) How do you make a picture into a background image of a web page?**

To do this, place a tag code after the </head> tag as follows:



|  |  |
| --- | --- |
| 1 | <body background = “image.gif”> |

Replace image.gif with the name of your image file. This will take the picture and make it the background image of your web page.

**48) What happens if the list-style-type property is used on a non-list element like a paragraph?**

If the list-style-type property is used on a non-list element like a paragraph, the property will be ignored and do not affect the paragraph.

**49) When is it appropriate to use frames?**

Frames can make navigating a site much easier. If the main links to the site are located in a frame that appears at the top or along the edge of the browser, the content for those links can be displayed in the remainder of the browser window.

**50) What happens if the number of values in the rows or cols attribute doesn’t add up to 100 percent?**

The browser sizes the frames relative to the total sum of the values. If the cols attribute is set to 100%, 200% the browser displays two vertical frames with the second being twice as big as the first.

**51) Which browsers support HTML5?**

The latest versions of Google Chrome, Apple Safari, Mozilla Firefox, and Opera all support most of the HTML5 features.

**52) Name two new tags included in the HTML 5**

<Video> and <Audio> are new tags which are included in HTML5 version. They are mainly used as a replacement for Flash, Silverlight, and similar technologies to play multimedia items.

**53) Do you know which are two semantic tags are included in HTML5 version?**

The <article> and <section> tags are two new tags that are included in HTML5. Articles can be composed of multiple sections that can have multiple articles. An article tag represents a full block of content which is a section of a bigger whole.

**54) What is <figure> in HTML5?**

This tag represents a piece of self-contained flow content. It is mostly used as a single unit as a reference the main flow of the document.

**55) What is the use of Canvas element?**

The canvas element helps to build charts, graphs, bypass Photoshop to create 2D images and place them directly into HTML5 code.

**56) What are the new FORM elements which are available in HTML5?**

The new Form elements in HTML5 offers much better functionality than the earlier versions.

The tags given provided to carry out these functions are:

1) <datalist> – This tag is use to specify a list of options for input controls.

2) <keygen> – This tag represents a key-pair generator field.

3) <output> – It represents the result of any scripting calculation.

**57) Tell me two benefits of HTML5 Web Storage**  
Two main benefits of HTML5 Web Storage:

* It can store up to 10 MB data which is certainly more than what cookies have.
* Web storage data cannot be transferred with the HTTP request. It helps to increase the performance of the application.

**58) What are two types of Web Storage in HTML5?**

Two storage types of HTML5 are:

**Session Storage:**

It stores data of current session only. It means that the data stored in session storage clear automatically when the browser is closed.

**Local Storage:**

Local storage is another type of HTML5 Web Storage. In local storage, data is not deleted automatically when the current browser window is closed.

**59) What is the Application Cache in HTML5 and why it is used?**

The Application Cache concept means that a web application is cached. It can be accessible without the need for internet connection.

**Some advantages of Application Cache:**

1. Offline browsing – Web users can also use the application when they are offline.
2. Speed – Cached resources load quicker
3. Reduce the server load – The web browser will only download updated resources from the server.

**60) Explain five new input types provided by HTML5 for forms?**

Following are the important, new data types offered by HTML5:

1. Date: It allows the user to select a date.
2. datetime-local: This input type allows the user to select a date and time without time zone.
3. datetime: This input type allows the user to select a date and time with time zone.
4. month: It enables the user to select a month and year
5. email: These input fields used to contain an e-mail address.

What is the relationship between SGML,HTML , XML and XHTML?

SGML (Standard generalized markup language) is a standard which tells how to specify document markup. It’s only a Meta language which describes how a document markup should be. HTML is a markup language which is described using SGML.

So by SGML they created DTD which the HTML refers and needs to adhere to the same. So you will always find “DOCTYPE” attribute at the top of HTML page which defines which DTD is used for parsing purpose.

Hide   Copy Code

<!--!doctype-->

Now parsing SGML was a pain so they created XML to make things better. XML uses SGML. For example in SGML you have to start and end tags but in XML you can have closing tags which close automatically (“”).

XHTML was created from XML which was used in HTML 4.0. So for example in SGML derived HTML “  
” is not valid but in XHTML it’s valid. You can refer XML DTD as shown in the below code snippet.

Hide   Copy Code

<!--!doctype--><!--!doctype-->

What is HTML 5?

HTML 5 is a new standard for HTML whose main target is to deliver everything without need to any additional plugins like flash, Silverlight etc. It has everything from animations, videos, rich GUI etc.  
HTML5 is cooperation output between World Wide Web Consortium (W3C) and the Web Hypertext Application Technology Working Group (WHATWG).

If I do not put <! DOCTYPE html> will HTML 5 work?

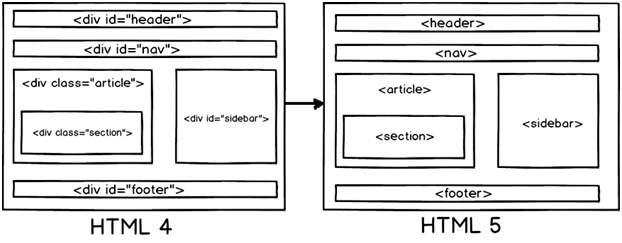
No, browser will not be able to identify that it’s a HTML document and HTML 5 tags will not function properly.

Which browsers support HTML 5?

Almost all browsers i.e. Safari, Chrome, Firefox, Opera, Internet Explorer support HTML 5.

How is the page structure of HTML 5 different from HTML 4 or previous HTML?

A typical web page has headers, footers, navigation, central area and side bars. Now if we want to represent the same in HTML 4 with proper names to the HTML section we would probably use a DIV tag.  
But in HTML 5 they have made it more clear by creating element names for those sections which makes your HTML more readable.



Below are more details of the HTML 5 elements which form the page structure.

* <header>: Represents header data of HTML.
* <footer>: Footer section of the page.
* <nav>: Navigation elements in the page.
* <article>: Self-contained content.
* <section>: Used inside article to define sections or group content in to sections.
* <aside>: Represent side bar contents of a page.

What is datalist in HTML 5?

Datalist element in HTML 5 helps to provide autocomplete feature in a textbox as shown below.



Below is the HTML code for DataList feature:-

Hide   Copy Code

<input list="Country">

<datalist id="Country">

<option value="India">

<option value="Italy">

<option value="Iran">

<option value="Israel">

<option value="Indonesia">

</datalist>

What are the different new form element types in HTML 5?

There are 10 important new form elements introduced in HTML 5:-

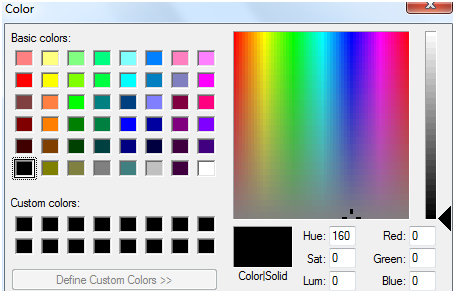
1. Color.
2. Date
3. Datetime-local
4. Email
5. Time
6. Url
7. Range
8. Telephone
9. Number
10. Search

Let’s understand these elements step by step.

If you want to show color picker dialog box.

Hide   Copy Code

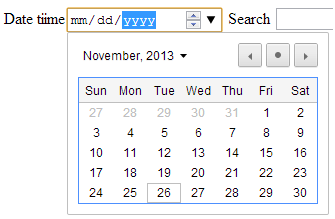
<input type="color" name="favcolor">



If you want to show calendar dialog box.

Hide   Copy Code

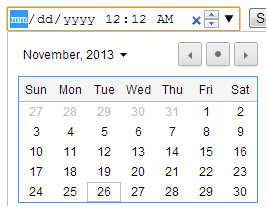
<input type="date" name="bday">



If you want to show calendar with local time.

Hide   Copy Code

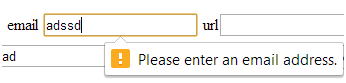
<input type="datetime-local" name="bdaytime">



If you want to create a HTML text with email validation we can set the type as “email”.

Hide   Copy Code

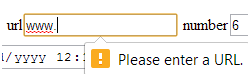
<input type="email" name="email">



For URL validation set the type as “url” as shown in the below HTML code.

Hide   Copy Code

<input type="url" name="sitename">



For URL validation set the type as “url” as shown in the below HTML code.

If you want to display textbox with number range you can set type to number.

Hide   Copy Code

<input type="number" name="quantity" min="1" max="5">

Image 9 for 40 important HTML 5 Interview questions with answers

If you want to display a range control you can use type as range.

Hide   Copy Code

<input type="range" min="0" max="10" step="2" value="6">

Image 10 for 40 important HTML 5 Interview questions with answers

Want to make text box as search engine box.

Hide   Copy Code

<input type="search" name="googleengine">

What to only take time input.

Hide   Copy Code

<input type="time" name="usr\_time">

If you want to make text box to accept telephone numbers.

Hide   Copy Code

<input type="tel" name="mytel">

What is output element in HTML 5?

Output element is needed when you need calculation from two inputs to be summarized in to a label. For instance you have two textboxes( see the below figure) and you want to add numbers from these textboxes and send them to a label.

Image 11 for 40 important HTML 5 Interview questions with answers

Below goes the code of how to use output element with HTML 5.

Hide   Copy Code

<form onsubmit="return false" öninput="o.value = parseInt(a.value) + parseInt(b.value)">

<input name="a" type="number"> +

<input name="b" type="number"> =

<output name="o" />

</form>

You can also replace “parseInt” with “valueAsNumber” for simplicity. You can also use “for” in the output element for more readability.

Hide   Copy Code

<output name="o" for="a b"></output>

What is SVG?

SVG stands for scalable vector graphics. It’s a text based graphic language which draws images using text, lines, dots etc. This makes it lightweight and renders faster.

Can we see a simple example of SVG using HTML 5?

Let’s say we want to display the below simple line using HTML 5 SVG.



Below is how the code of HTML 5. You can see the SVG tag which encloses the polygon tag for displaying the star image.

Hide   Copy Code

<svg id="svgelem" height="[object SVGAnimatedLength]" xmlns="http://www.w3.org/2000/svg">

<line style="stroke: rgb(255, 0, 0); stroke-width: 2px;" y2="[object SVGAnimatedLength]" x2="[object SVGAnimatedLength]" y1="[object SVGAnimatedLength]" x1="[object SVGAnimatedLength]">

</line>

What is canvas in HTML 5?

Canvas is an HTML area on which you can draw graphics.

</canvas>

**Get access to canvas area**

To draw on the canvas area we need to first get reference of the context section. Below is the code for canvas section.

Hide   Copy Code

var c=document.getElementById("mycanvas");

var ctx=c.getContext("2d");

**Draw the graphic**

Now once you have access to the context object we can start drawing on the context. So first call the “move” method and start from a point , use line method and draw the line and then apply stroke over it.

Hide   Copy Code

<a name="WhatisthedifferencebetweenCanvasandSVGgraphics">What is the difference between Canvas and SVG graphics? </a>

***Note: -****If you see the previous two questions both canvas and SVG can draw graphics on the browser. So in this question interviewer wants to know when will you use what.*

|  |  |
| --- | --- |
| **SVG** | **Canvas** |
| Here’s it’s like draw and remember. In other words any shape drawn by using SVG can be remembered and manipulated and browser can render it again. | Canvas is like draw and forget. Once something is drawn you cannot access that pixel and manipulate it. |
| SVG is good for creating graphics like CAD software’s where once something is drawn the user wants to manipulate it. | Canvas is good for draw and forget scenarios like animation and games. |
| This is slow as it needs to remember the co-ordinates for later manipulations. | This is faster as there is no intention of remembering things later. |
| We can have event handler associated with the drawing object. | Here we cannot associate event handlers with drawing objects as we do not have reference of them. |
| Resolution independent. | Resolution dependent. |

How to draw rectangle using Canvas and SVG using HTML 5?

HTML 5 code Rectangle code using SVG.

Hide   Copy Code

<svg xmlns="http://www.w3.org/2000/svg" version="1.1">

<rect style="fill: rgb(0, 0, 255); stroke-width: 1px; stroke: rgb(0, 0, 0);" height="[object SVGAnimatedLength]" width="[object SVGAnimatedLength]">

</rect>

HTML 5 Rectangle code using canvas.

Hide   Shrink Image 13 for 40 important HTML 5 Interview questions with answers   Copy Code

var c=document.getElementById("myCanvas");

var ctx=c.getContext("2d");

ctx.rect(20,20,150,100);

ctx.stroke();

<svg xmlns="http://www.w3.org/2000/svg" version="1.1">

<circle fill="red" stroke-width="2" stroke="black" r="[object SVGAnimatedLength]" cy="[object SVGAnimatedLength]" cx="[object SVGAnimatedLength]">

var canvas = document.getElementById('myCanvas');

var context = canvas.getContext('2d');

var centerX = canvas.width / 2;

var centerY = canvas.height / 2;

var radius = 70;

context.beginPath();

context.arc(centerX, centerY, radius, 0, 2 \* Math.PI, false);

context.fillStyle = 'green';

context.fill();

context.lineWidth = 5;

context.strokeStyle = '#003300';

context.stroke();

<!DOCTYPE html>

<html>

<body önload="DrawMe();">

<svg height="[object SVGAnimatedLength]" width="[object SVGAnimatedLength]">

<circle id="circle1" cx="[object SVGAnimatedLength]" cy="[object SVGAnimatedLength]" r="[object SVGAnimatedLength]" style="stroke: none; fill: rgb(255, 0, 0);">

</body>

<script>

var timerFunction = setInterval(DrawMe, 20);

alert("ddd");

function DrawMe()

{

var circle = document.getElementById("circle1");

var x = circle.getAttribute("cx");

var newX = 2 + parseInt(x);

if(newX > 500)

{

newX = 20;

}

circle.setAttribute("cx", newX);

}

</script>

</html></circle>

What are selectors in CSS?

Selectors help to select an element to which you want to apply a style. For example below is a simple style called as ‘intro” which applies red color to background of a HTML element.

Hide   Copy Code

<style>

.intro

{

background-color:red;

}

</style>

To apply the above “intro” style to div we can use the “class” selector as shown in the below figure.

Hide   Copy Code

<div class="intro">

<p>My name is Shivprasad koirala.</p>

<p>I write interview questions.</p>

</div>

How can you apply CSS style using ID value?

So let’s say you have a HTML paragraph tag with id “mytext” as shown in the below snippet.

Hide   Copy Code

<p id="mytext">This is HTML interview questions.</p>

You can create a style using “#” selector with the “id” name and apply the CSS value to the paragraph tag. So to apply style to “mytext” element we can use “#mytext” as shown in the below CSS code.

Hide   Copy Code

<style>

#mytext

{

background-color:yellow;

}

</style>

**Quick revision of some important selectors.**

Set all paragraph tags back ground color to yellow.

Hide   Copy Code

P,h1

{

background-color:yellow;

}

Sets all paragraph tags inside div tag to yellow background.

Hide   Copy Code

div p

{

background-color:yellow;

}

Sets all paragraph tags following div tags to yellow background.

Hide   Copy Code

div+p

{

background-color:yellow;

}

Sets all attribute with “target” to yellow background.

Hide   Copy Code

a[target]

{

background-color:yellow;

}

<a href="http://www.questpond.com">ASP.NET interview questions</a>

<a href="http://www.questpond.com" target="\_blank">c# interview questions</a>

<a href="http://www.questpond.org" target="\_top">.NET interview questions with answers</a>

Set all elements to yellow background when control gets focus.

Hide   Copy Code

input:focus

{

background-color:yellow;

}

Set hyperlinks according to action on links.

Hide   Copy Code

a:link {color:green;}

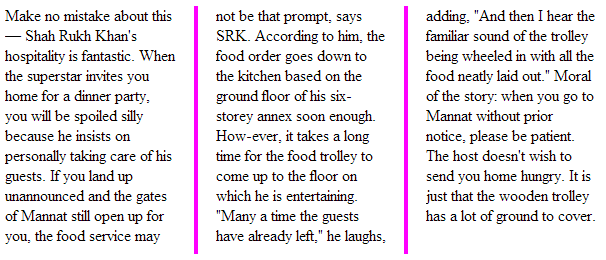
a:visited {color:green;}

a:hover {color:red;}

a:active {color:yellow;}

What is the use of column layout in CSS?

CSS column layout helps you to divide your text in to columns. For example consider the below magazine news which is one big text but we need to divide the same in to 3 columns with a border in between. That’s where HTML 5 column layout comes to help.



To implement column layout we need to specify the following:-

* How many columns we want to divide the text in to ?

To specify number of columns we need to us column-count. “webkit” and “moz-column” are needed for chrome and firefox respectively.

Hide   Copy Code

-moz-column-count:3; /\* Firefox \*/

-webkit-column-count:3; /\* Safari and Chrome \*/

column-count:3;

* How much gap we want to give between those columns ?

Hide   Copy Code

-moz-column-gap:40px; /\* Firefox \*/

-webkit-column-gap:40px; /\* Safari and Chrome \*/

column-gap:20px;

* Do you want to draw a line between those columns , if yes how much thick ?

Hide   Copy Code

-moz-column-rule:4px outset #ff00ff; /\* Firefox \*/

-webkit-column-rule:4px outset #ff00ff; /\* Safari and Chrome \*/

column-rule:6px outset #ff00ff;

Below is the complete code for the same.

Hide   Copy Code

<style>

.magazine

{

-moz-column-count:3; /\* Firefox \*/

-webkit-column-count:3; /\* Safari and Chrome \*/

column-count:3;

-moz-column-gap:40px; /\* Firefox \*/

-webkit-column-gap:40px; /\* Safari and Chrome \*/

column-gap:20px;

-moz-column-rule:4px outset #ff00ff; /\* Firefox \*/

-webkit-column-rule:4px outset #ff00ff; /\* Safari and Chrome \*/

column-rule:6px outset #ff00ff;

}

</style>

You can then apply the style to the text by using the class attribute.

Hide   Copy Code

<div class="magazine">

Your text goes here which you want to divide in to 3 columns.

</div>

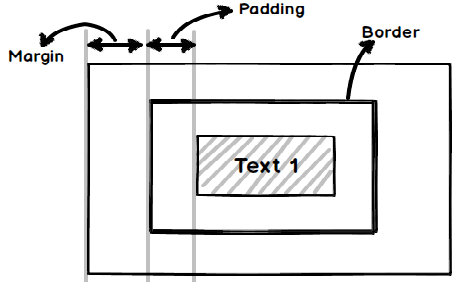
Can you explain CSS box model?

CSS box model is a rectangular space around a HTML element which defines border, padding and margin.

Border: - This defines the maximum area in which the element will be contained. We can make the border visible, invisible, define height and width etc.

Padding: - This defines the spacing between border and element.

Margin: - This defines the spacing between border and any neighboring elements.



For instance below is a simple CSS code which defines a box with border , padding and margin values.

Hide   Copy Code

.box {

width: 200px;

border: 10px solid #99c;

padding: 20px;

margin: 50px;

}

Now if we apply the above CSS to a DIV tag as shown in the below code , your output would be as shown in the figure below. I have created two test “Some text” and “Some other text” so that we can see how margin property functions.

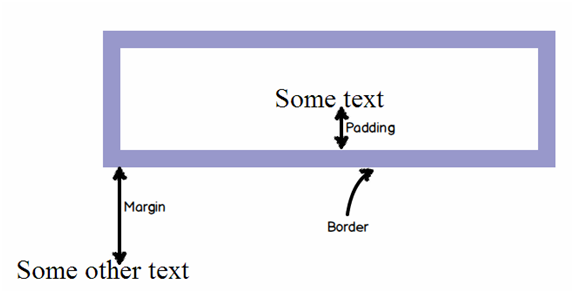
Hide   Copy Code

<div align="middle" class="box">

Some text

</div>

Some other text



Can you explain some text effects in CSS 3?

Here the interviewer is expecting you to answer one of two text effects by CSS.Below are two effects which are worth noting.

Shadow text effect

Hide   Copy Code

.specialtext

{

text-shadow: 5px 5px 5px #FF0000;

}

Image 17 for 40 important HTML 5 Interview questions with answers

Word wrap effect

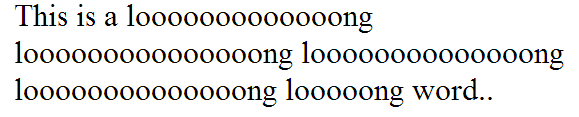
Hide   Copy Code

<style>

.breakword

{word-wrap:break-word;}

</style>



What are web workers and why do we need them ?

Consider the below heavy for loop code which runs above million times.

Hide   Copy Code

function SomeHeavyFunction()

{

for (i = 0; i < 10000000000000; i++)

{

x = i + x;

}

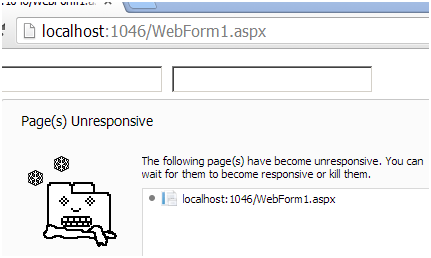
}

Let’s say the above for loop code is executed on a HTML button click. Now this method execution is synchronous. In other words the complete browser will wait until the for loop completes.

Hide   Copy Code

<input type="button" onclick="SomeHeavyFunction();" />

This can further lead to browser getting freezed and unresponsive with an error message as shown in the screen below.



So if we can move this heavy for loop in a JavaScript file and run it asynchronously that means the browser does need to wait for the loop then we can have a more responsive browser. That’s what web worker are for.

Web worker helps to execute JavaScript file asynchronously.

What is local storage concept in HTML 5?

Many times we would like to store information about the user locally in the computer. For example let’s say user has half-filled a long form and suddenly the internet connection breaks off. So the user would like you to store this information locally and when the internet comes back.He would like to get that information and send it to the server for storage.

Modern browsers have storage called as “Local storage” in which you can store this information.

How can we add and remove data from local storage?

Data is added to local storage using “key” and “value”. Below sample code shows country data “India” added with key value “Key001”.

Hide   Copy Code

localStorage.setItem(“Key001”,”India”);

To retrieve data from local storage we need to use “getItem” providing the key name.

Hide   Copy Code

var country = localStorage.getItem(“Key001”);

You can also store JavaScript object’s in the local storage using the below code.

Hide   Copy Code

var country = {};

country.name = “India”;

country.code = “I001”;

localStorage.setItem(“I001”, country);

var country1 = localStorage.getItem(“I001”);

If you want to store in JSON format you can use “JSON.stringify” function as shown in the below code.

Hide   Copy Code

localStorage.setItem(“I001”,JSON.stringify(country));

What is the lifetime of local storage?

Local storage does not have a life time it will stay until either the user clear it from the browser or you remove it using JavaScript code.

What is the difference between local storage and cookies?

|  |  |  |
| --- | --- | --- |
|  | **Cookies** | **Local storage** |
| **Client side / Server side.** | Data accessible both at client side and server side. Cookie data is sent to the server side with every request. | Data is accessible only at the local browser side. Server cannot access local storage until deliberately sent to the server via POST or GET. |
| **Size** | 4095 bytes per cookie. | 5 MB per domain. |
| **Expiration** | Cookies have expiration attached to it. So after that expiration the cookie and the cookie data get’s deleted. | There is no expiration data. Either the end user needs to delete it from the browser or programmatically using JavaScript we need to remove the same. |

What is WebSQL?

WebSQL is a structured relational database at the client browser side. It’s a local RDBMS inside the browser on which you can fire SQL queries.

Is WebSQL a part of HTML 5 specification?

No, many people label it as HTML 5 but it’s not part of HTML 5 specification. The specification is based around SQLite.

So how can we use WebSQL?

The first step we need to do is open the database by using “OpenDatabase” function as shown below. The first argument is the name of the database, the next is the version, then a simple textual title and finally the size of the database.

Hide   Copy Code

var db=openDatabase('dbCustomer','1.0','Customer app’, 2 \* 1024 \* 1024);

To execute SQL we then need to use “transaction” function and call “executeSql” function to fire SQL.

Hide   Copy Code

db.transaction(function (tx)

{

tx.executeSql('CREATE TABLE IF NOT EXISTS tblCust(id unique, customername)');

tx.executeSql('INSERT INTO tblcust (id, customername) VALUES(1, "shiv")');

tx.executeSql('INSERT INTO tblcust (id, customername) VALUES (2, "raju")');

}

In case you are firing “select” query you will get data is “results” collection which we can loop and display in the HTML UI.

Hide   Copy Code

db.transaction(function (tx)

{

tx.executeSql('SELECT \* FROM tblcust', [], function (tx, results) {

for (i = 0; i < len; i++)

{

msg = "<p><b>" + results.rows.item(i).log + "</b></p>";

document.querySelector('#customer).innerHTML += msg;

}

}, null);

});

What is fallback in Application cache?

Hide   Copy Code

<a name="WhatisfallbackinApplicati>FALLBACK:

/home/ /homeoffline.html </a></pre>

<h2><a name=" whatisfallbackinapplicati=""></a>

**1. What is HTML5 ?**

**HTML5**is the most recent version of the HTML(Hypertext Markup Language). It is a language for structuring and displaying content for the World Wide Web, a core technology of the Internet.

**WHATWG**(Web Hypertext Application Technology Working Group) another gathering of **W3C**individuals felt that W3C is not giving careful consideration to this present reality improvement needs of dialect, it has begun dealing with the new determination of HTML-HTML5. Consequently, HTML5 is another adaptation of HTML 4.01 and XHTML 1.0 concentrating on the necessities of Web application designers and in addition tending to issues found in the present details.

Specifically, HTML5 includes numerous new syntactical features. New elements , like <section>, <article>, <header>, and <nav>, are the essential parts of semantic substance of documents. These additionally incorporate the <video>, <audio>, and <canvas> tags, as well as the integration of SVG content. These components are intended to make it simple to incorporate and handle interactive media and graphical substance on the web without resorting to restrictive modules and APIs. While a few components and traits have been expelled. A few components, for example, <a>, <cite> and <menu> have been changed, redefined or standardized. The APIs and DOM are no longer reconsideration but are fundamental parts of the HTML5 specification.

**2. What is < !DOCTYPE>? Is it necessary to use in HTML5 ?**

The < !DOCTYPE> is an instruction to the web browser about what version of HTML the page is written in. AND The < !DOCTYPE> tag does not have an end tag and It is not case sensitive.

The < !DOCTYPE> declaration must be the very first thing in HTML5 document, before the tag. As In HTML 4.01, all < ! DOCTYPE > declarations require a reference to a Document Type Definition (DTD), because HTML 4.01 was based on Standard Generalized Markup Language (SGML). Where as **HTML5**is not based on SGML, and therefore does not require a reference to a Document Type Definition (DTD).

**3. List out the advantages of HTML5 ?**

The top 5 from some web research would appear to offer the following **HTML5**features;

* Mutuality
* Cleaner mark-up / Improved Code
* Improved Semantics
* Elegant forms and web apps
* Offline Application cache

**4. What is Canvas in HTML5 ? How to write a Canvas ?**

**Canvas**is a element of **HTML5**which uses JavaScript to draw graphics on a web page. A canvas is a rectangular area. Each and every pixel of it can be controlled by us. There are several methods for drawing paths, boxes, circles, characters, and adding images by using canvas.

To add canvas tag to our HTML document we need id, width and height. Below is the example how to write a basic canvas tag to your HTML document.

<canvas id="myFirstCanvas" width="100" height="100"> </canvas>

**5. What is HTML5 Geolocation ? How to use it ?**

**HTML5**nGeolocation is used to locate a user’s position  
The HTML5 Geolocation API is used to get the geographical position of a user.  
Since this can compromise user privacy, the position is not available unless the user approves it.

Use the **getCurrentPosition()** method to get the user’s position.  
The example below is a simple **Geolocation**example returning the latitude and longitude of the user’s position:

<script>

var x=document.getElementById("demo");

function getLocation()

{

if (navigator.geolocation)

{

navigator.geolocation.getCurrentPosition(showPosition);

}

else{x.innerHTML="Geolocation is not supported by this browser.";}

}

function showPosition(position)

{

x.innerHTML="Latitude: " + position.coords.latitude +"

Longitude: " + position.coords.longitude;

}

</script>

**6. What are the rules established for HTML5 ?**

Some rules for HTML5 were established:

* New features should be based on HTML, CSS, DOM, and JavaScript
* Reduce the need for external plugins (like Flash)
* Better error handling
* More markup to replace scripting
* HTML5 should be device independent
* The development process should be visible to the public.

**7. What is the sessionStorage Object in HTML5 ? How to create and access?**

The **sessionStorage**object stores the data for one session. The data is deleted when the user closes the browser window. like below we can create and access a sessionStorage here we created “blogName” as session

<script type="text/javascript">

sessionStorage.blogName="OnlineInterviewQuestions";

document.write(sessionStorage.name);

</script>

**8. What is the difference between Html5 application cache and regulate Html browser cache?**

The new **HTML5** specification allows browsers to prefetch some or all of website assets such as HTML files, images, CSS, JavaScript, and so on, while the client is connected. It is not necessary for the user to have accessed this content previously, for fetching this content. In other words, application cache can prefetch pages that have not been visited at all and are thereby unavailable in the regular browser cache. Prefetching files can speed up the site’s performance, though you are of course using bandwidth to download those files initially.

**9. What are the new Apis provided by the Html5 standard? Give a brief description of each**

Below are the list of new Api’s provided by the Html 5 standard.

* **Canvas :** Canvas consists of a drawable region defined in HTML code with height and width attributes. JavaScript code may access the area through a full set of drawing functions similar to other common 2D APIs, thus allowing for dynamically generated graphics. Some anticipated uses of the canvas include building graphs, animations, games, and image composition.
* Timed media playback
* Offline storage database
* Document editing
* Drag-and-drop
* Cross-document messaging
* Browser history management
* MIME type and protocol handler registration

**10. What is the use of LocalStorage in HTML5 ?**

Before HTML5 LocalStores was done with cookies. Cookies are not very good for large amounts of data, because they are passed on by every request to the server, so it was very slow and in-effective.  
In HTML5, the data is NOT passed on by every server request, but used ONLY when asked for. It is possible to store large amounts of data without affecting the website’s performance.and The data is stored in different areas for different websites, and a website can only access data stored by itself.  
And for creating localstores just need to call localStorage object like below we are storing name and address

<script type="text/javascript">

localStorage.name="ABC";

localStorage.address="New Delhi India.";

document.write(localStorage.address);

</script>

**11. Can you list the new input type attributes in HTML5 ?**

Yes, we can use below new input type Attribute in HTML5

* **tel**The input is of type telephone number
* **search**The input field is a search field
* **url**a URL
* **email**One or more email addresses
* **datetime**A date and/or time
* **date**A date
* **month**A month
* **week**A week
* **time** The input value is of type time
* **datetime-local** A local date/time
* **number** A number.
* **range**A number in a given range.
* **color**A hexadecimal color, like #82345c
* **placeholder**Specifies a short hint that describes the expected value of an input field.

**12. List out the new features of HTML5?**

Some of the most interesting new features in HTML5:

* The <canvas> element for 2D drawing
* The <video> and <audio> elements for media playback
* Support for local storage
* New content-specific elements, like <article>, <footer>, <header>, <nav>, <

**Q #1) What is CSS?**

**Answer:**CSS outline the style of an HTML webpage, it is a language by which we can set the behavior of an HTML webpage. It describes how the HTML content will be shown on screen.

CSS controls the layout of several HTML web pages. CSS is referred to as the Cascading Style Sheet.

**Q #2) Name all the modules which are used in the current version of CSS.**

**Answer: There are several modules in CSS as stated below:**

* Selectors
* Box Model
* Backgrounds and Borders
* Text Effects
* 2D/3D Transformations
* Animations
* Multiple Column Layout
* User Interface.

**Q #3) Distinguish between CSS2 and CSS3.**

**Answer:**There are several differences between CSS2 and CSS3.

1. CSS3 is divided into two various sections which are called as a module. Whereas in CSS2 everything accedes into a single document with all the information in it.
2. CSS3 modules are supported almost on every browser and on the other hand modules of CSS and CSS2 are not supported in every browser.
3. In CSS3 we will find that many graphics related characteristics have been introduced like “Border-radius or box-shadow, flexbox.
4. In CSS3, a user can precise multiple background images on a webpage by using properties like background-image, background-position, and background-repeat styles.

**Q #4) Cite different types of CSS.**

**Answer: There are three types of CSS as mentioned below.**

* **External –**These are written in separate files.
* **Internal –**These are cited at the top of the web page code document.
* **Inline –** These are written right next to the text.

**Q #5) Why is the external style sheet useful?**

**Answer:**External style sheet is very useful as we write all the styling codes in a single file and it can be used anywhere by just referencing the link of that external style sheet file.

So if we do any changes in that external file, then the changes can also be observed on the webpage. So we can say that it is very useful and it makes your work easy while working on larger files.

**Q #6) What are the uses of embedded style sheet**?

**Answer:**Embedded style sheet gives us the privilege to define styles at one place in an HTML document.

We can generate multiple classes using an embedded style sheet to use on multiple tag types of a web page and also there is no extra downloading required for importing the information.

**Example:**

<!DOCTYPE html>

<html>

<head>

<style type="text/css">

p {

  font-family: georgia, serif;

  font-size: x-large;

  color:#ff9900;

  }

a:hover {

  color: LimeGreen;

  text-decoration: none;

  }

</style>

</head>

<body>

<p>Embedded style sheet gives us the privilege to define styles at one place in a HTML document.

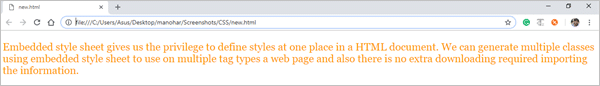
We can generate multiple classes using embedded style sheet to use on multiple tag types a web page

and also there is no extra downloading required importing the information.

</p>

</body>

</html>



**Q #7) How to use CSS selector?**

**Answer:**By using CSS selector, we can choose the content which we want to style so that we can say that it is a bridge between the style sheet and the HTML files.

The syntax for CSS selector is “select” HTML elements created on their id, class, type etc.

**Q #8) Explain the concept of Tweening.**

**Answer:**Tweening is the process in which we create intermediate frames between two images to get the appearance of the first image which develops into the second image.

It is mainly used for creating animation.

**Q #9) Define CSS image scripts.**

**Answer:**CSS image scripts is a group of images which is placed into one image.

It reduces the load time and request number to the server while projecting multiple images into a single web page.

**Q #10) Explain the term Responsive web design.**

**Answer:**It is a method in which we design and develop a web page according to the user activities and conditions which are based on various components like the size of the screen, portability of the web page on different device etc.

Hence it is done by using different flexible layouts and grids.

**Q #11) What are CSS counters?**

**Answer:**CSS counters are variables which can be incremented by rules of CSS that inspector track how many times the variable has been used.

**Q #12) What is CSS specificity?**

**Answer:**CSS specificity is a score or rank that decides which style declaration has to be used to an element.

(\*) this universal selector has low specificity while ID selectors have high specificity.

**There are four categories in CSS which authorized the specificity level of the selector.**

* Inline style
* IDs
* Classes, Attributes, and pseudo-classes.
* Elements and pseudo elements.

**Q #13) How can we calculate specificity?**

**Answer:**To calculate specificity we will start with 0, then we have to add 1000 for each ID and we have to add 10 to the attributes, classes or pseudo-classes with each element name or pseudo-element and later we have to add 1 to them.

**Example:**

h2

             #content h2

           <div id=”content”>

              <h2 style=”color:#FF0000”>heading</h2>

            </div>

**Q #14) How do we make a rounded corner by using CSS?**

**Answer:**We can make a rounded corner by using the property “border-radius”. We can apply this property to any element.

**Example:**

<html>

<head>

<style>

#rcorners1 {

    border-radius: 25px;

    background: #715751;

    padding: 20px;

    width: 200px;

    height: 150px;

}

</style>

</head>

<body>

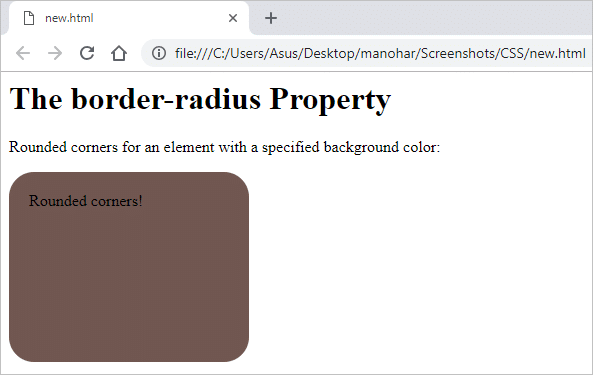
<h1>The border-radius Property</h1>

<p>Rounded corners for an element with a specified background color:</p>

<p id="rcorners1">Rounded corners!</p>

</body>

</html>



**Q #15) How will you add border images to an HTML element?**

**Answer:**We can set the image to be used as the border image alongside an element by using the property of CSS “border-image”.

**Example:**

#borderimg {

    border: 15px solid transparent;

    padding: 20px;

    border-image: url(border.png) 30 round;

}

**Q #16) What are gradients in CSS?**

**Answer:**It is a property of CSS which allows you to display a smooth transformation between two or more than two specified colors.

There are two types of gradients that are present in CSS. They are:

* Linear gradient
* Radial Gradient

**Q #17) What is CSS flexbox?**

**Answer:**It allows you to design a flexible responsive layout structure without using any float or positioning property of CSS. To use CSS flexbox you need to define a flex container initially.

**Example:**

<!DOCTYPE html>

<html>

<head>

<style>

.flex-container {

  display: flex;

  background-color: #f4b042;

}

.flex-container > div {

  background-color: #d60a33;

  margin: 10px;

  padding: 20px;

  font-size: 30px;

}

</style>

</head>

<body>

<div class="flex-container">

  <div>1</div>

  <div>2</div>

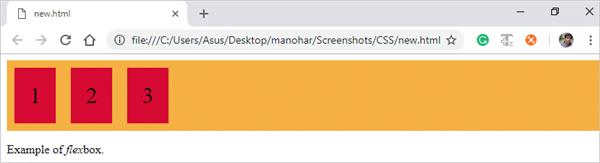
  <div>3</div>

</div>

<p> Example of  <em>flex</em>box.</p>

</body>

</html>



**Q #18) Write all the properties of flexbox.**

**Answer:** There are several properties of flexbox that are used in the HTML webpage.

**They are:**

* flex-direction
* flex-wrap
* flex-flow
* justify-content
* align-items
* align-content

**Q #19) How to align image vertically in a division which spans vertically in the whole webpage?**



**Answer:**It can be done by using the syntax verticle-align: middle in the <div1> element and even we can bind the two text spans around with another span and after this, we have to use verticle-align: middle in the content #icon.

**Q #20) What is the difference between padding and margin?**

**Answer:**In CSS, the margin is the property by which we can create space around elements. We can even create space to the exterior defined borders.

**In CSS we have margin property as follows:**

* margin-top
* margin-right
* margin-bottom
* Margin-left

**Margin property has some defined values as shown below.**

* **Auto –** using this property browser calculates the margin.
* **Length –** It sets the margin values in px,pt,cm etc.
* **% –** It sets the width % of the element.
* **Inherit –** By this property we can inherit the margin property from the parent element.

In CSS, padding is the property by which we can generate space around an element’s content as well as inside any known border.

**CSS padding also has properties like,**

1. Padding-top
2. Padding-right
3. Padding-bottom
4. Padding-left

Negative values are not allowed in padding.

div {

padding-top: 60px;

padding-right: 40px;

padding-bottom: 50px;

padding-left: 70px;

}

**Q #21) What is the use of the Box Model in CSS?**

**Answer:**In CSS, the box model is a box that binds all the HTML elements and it includes features like margins, border, padding, and the actual content.

By using box model we will get the authority to add the borders all around the elements and we can also define the space between the elements.

**Q #22) How can we add icons to the web page?**

**Answer:**We can add icons to the HTML webpage by using an icon library like font-awesome.

We have to add the name of the given icon class to any inline HTML element. (<i> or <span>) . Icons in the icon libraries are scalable vectors that can be customized with CSS.

**Q #23) What is a CSS pseudo class?**

**Answer:**It is a class which is used to define a special state of an HTML element.

This class can be used by styling an element when a user snooped over it and also it can style an HTML element when it gets the focus.

selector:pseudo-class {

property:value;

}

**Q #24) Explain the concept of pseudo-elements in CSS.**

**Answer:**It is a feature of CSS which is used to style the given parts of an element.

**For Example**, we can style the first letter or line of an HTML element.

selector::pseudo-element {

property:value;

}

**Q #25) What is CSS opacity?**

**Answer:**It is the property which elaborates the transparency of an element.

By this property, we can transparent the image that can take the values from 0.0-1.0, if the value is lower then the image is more transparent. IE8 and earlier version of the browser can take the values from 0-100.

img {

opacity: 0.6;

filter: alpha(opacity=60); /\* For IE8 and earlier \*/}

**Q #26) Write all the position states used in CSS.**

**Answer: In CSS, there are four position states as stated below:**

* Static(default)
* Relative
* Fixed
* absolute

**Q #27) What are navigation bars in CSS?**

**Answer:**By using navigation bars we can make an ordinary HTML page into a user-specific and more dynamic web page.

Basically, it is a list of links, hence use of <ul> and <li> elements makes the perfect sense.

ul {

list-style-type: none;

margin: 0;

padding: 0;

}

**Q #28) What are the differences between relative and absolute in CSS?**

**Answer:**The main difference between relative and absolute is that “relative” is used for the same tag in CSS and it means that if we write the left:10px then the padding will shift to 10px in the left while absolute is totally relative to the non-static parent.

It means if we write left:10px then the result will be 10px far from the left edge of the parent element.

**Q #29) Define ‘important’ declarations used in CSS.**

**Answer:**Important declarations are defined as that declaration which is having more importance than the normal declaration.

While executing, these declarations override the declaration which is having less importance.

**Example,** if there are two users Having important declaration then one of the declarations will override the declaration of another user.

**For Example:**

Body {background: #FF00FF !important; color: blue}

In this body background has more weight than the color.

**Q #30) Define different cascading methods that can be used inside the cascading order.**

**Answer: Cascading order is itself a sorting method that allows many other different sorting methods:**

**#1) Sort by origin:** There are some rules which can provide the alternate way which can be defined as:

* Normal weight of style sheet of a particular provider will be overridden by the increased weight of the user's style sheet.
* Stylesheet rules of a particular user will be overridden by the normal width of the provider’s style sheet.

**#2) Sort by selector's specificity:** Less specific selector is been overridden by the more specific selector.

**Example**, A Contextual selector is less specific in comparison to the ID selector which is a more specific one and with that contextual selector is been overridden by the ID selector.

**#3) Sort by order specified:** This comes in the scenario when the two selectors are same in weight and the other properties than the specification which will be seen for overriding.

**Example:**

All other styles will be seen overridden if the style attribute is used for inline style.

And also if the link element is used for external style, then it will override the imported style.

**Q #31) Differentiate between inline and block element.**

**Answer:**Inline element does not have an element to set width and height and also it does not have the line break.

**Example:** em, strong, etc.

**Block element specification:**

* They do have the line break.
* They define the width by setting a container and also allow setting height.
* It can also contain an element that occurs in the inline element.

**Example:**

width and height  
max-width and max-height  
min-width and min-height  
hi (i=1-6)- heading element  
p- Paragraph element.

**Q #32) How is the concept of inheritance applied in CSS?**

**Answer:**Inheritance is a concept in which the child class will inherit the properties of its parent class.

It is a concept which is been used in many languages and is the easy way of defining the same property again.

It is used in CSS to define the hierarchy from the top level to the bottom level. Inherited properties can be overridden by the children class if the child uses the same name.

**Example:**

Body {font-size: 15pt;}

**And another definition is being defined in the child class**

Body {font-size: 15pt;}  
H1 {font-size: 18pt;}

All the paragraph text will be displayed using the property and will be defined in the body except for the H1 style which will show the text in font 18 only.

**Q #33) Differentiate between the ID and class.**

**Answer:**Both id and class is been used in HTML and assigns the value from CSS.

**Please find below the differences:**

* Id is a kind of element which uniquely assigns a name to a particular element whereas class has an element with a certain set of properties which can be used for the complete block.
* The id can be used as an element because it can uniquely identify it whereas class is also defined to block the element and applies too many tags wherever it is used.
* Id provides the restriction to use its properties to one specific element whereas in class the inheritance is applied to a specific block or group of the element.

1) What is CSS?

CSS stands for Cascading Style Sheet. It is a popular styling language which is used with HTML to design websites. It can also be used with any XML documents including plain XML, SVG, and XUL.[More details...](https://www.javatpoint.com/what-is-css)

2) What is the origin of CSS?

SGML (Standard Generalized Markup Language) is the origin of CSS. It is a language that defines markup languages.

3) What are the different variations of CSS?

Following are the different variations of CSS:

* CSS1
* CSS2
* CSS2.1
* CSS3
* CSS4

4) How can you integrate CSS on a web page?

There are three methods to integrate CSS on web pages.

1. Inline method - It is used to insert style sheets in HTML document
2. Embedded/Internal method - It is used to add a unique style to a single document
3. Linked/Imported/External method - It is used when you want to make changes on multiple pages.

[More details...](https://www.javatpoint.com/how-to-add-css)

5) What are the advantages of CSS?

* Bandwidth
* Site-wide consistency
* Page reformatting
* Accessibility
* Content separated from presentation

6) What are the limitations of CSS?

* Ascending by selectors is not possible
* Limitations of vertical control
* No expressions
* No column declaration
* Pseudo-class not controlled by dynamic behavior
* Rules, styles, targeting specific text not possible

7) What are the CSS frameworks?

CSS frameworks are the preplanned libraries which make easy and more standard compliant web page styling. The frequently used CSS frameworks are: -

* Bootstrap
* Foundation
* Semantic UI
* Gumby
* Ulkit

8) Why background and color are the separate properties if they should always be set together?

There are two reasons behind this:

* It enhances the legibility of style sheets. The background property is a complex property in CSS, and if it is combined with color, the complexity will further increase.
* Color is an inherited property while the background is not. So this can make confusion further.

9) What is Embedded Style Sheet?

An Embedded style sheet is a CSS style specification method used with HTML. You can embed the entire stylesheet in an HTML document by using the STYLE element. [More details...](https://www.javatpoint.com/internal-css)

1. **<style>**
2. body {
3. background-color: linen;
4. }
5. h1 {
6. color: red;
7. margin-left: 80px;
8. }
9. **</style>**

10) What are the advantages of Embedded Style Sheets?

* You can create classes for use on multiple tag types in the document.
* You can use selector and grouping methods to apply styles in complex situations.
* No extra download is required to import the information.

11) What is a CSS selector?

It is a string that identifies the elements to which a particular declaration apply. It is also referred as a link between the HTML document and the style sheet. It is equivalent of HTML elements. There are several different types of selectors in CSS: -

* CSS Element Selector
* CSS Id Selector
* CSS Class Selector
* CSS Universal Selector
* CSS Group Selector

[More details...](https://www.javatpoint.com/css-selector)

12) Name some CSS style components.

Some CSS Style components are:

* Selector
* Property
* Value

13) What is the use of CSS Opacity?

The CSS opacity property is used to specify the transparency of an element. In simple word, you can say that it specifies the clarity of the image. In technical terms, Opacity is defined as the degree to which light is allowed to travel through an object. For example:

1. **<style>**
2. img.trans {
3. opacity: 0.4;
4. filter: alpha(opacity=40); /\* For IE8 and earlier \*/
5. }
6. **</style>**

14) Explain universal selector.

The universal selector matches the name of any of the element type instead of selecting elements of a specific type.

1. **<style>**
2. \* {
3. color: green;
4. font-size: 20px;
5. }
6. **</style>**

15) Which command is used for the selection of all the elements of a paragraph?

The p[lang] command is used for selecting all the elements of a paragraph.

16) What is the use of % unit?

It is used for defining percentage values.

17) Name the property used to specify the background color of an element.

The background-color property is used to specify the background color of the element. For example:

1. **<style>**
2. h2,p{
3. background-color: #b0d4de;
4. }
5. **</style>**

18) Name the property for controlling the image repetition of the background.

The background-repeat property repeats the background image horizontally and vertically. Some images are repeated only horizontally or vertically.

1. **<style>**
2. body {
3. background-image: url("paper1.gif");
4. margin-left:100px;
5. }
6. **</style>**

19) Name the property for controlling the image position in the background.

The background-position property is used to define the initial position of the background image. By default, the background image is placed on the top-left of the webpage.

You can set the following positions:

1. center
2. top
3. bottom
4. left
5. right
6. background: white url('good-morning.jpg');
7. background-repeat: no-repeat;
8. background-attachment: fixed;
9. background-position: center;

20) Name the property for controlling the image scroll in the background.

The background-attachment property is used to specify if the background image is fixed or scroll with the rest of the page in the browser window. If you set fixed the background image, then the image not move during scrolling in the browser. Let's take an example with the fixed background image.

1. background: white url('bbb.gif');
2. background-repeat: no-repeat;
3. background-attachment: fixed;

21) What is the use of ruleset?

The ruleset is used to identify that selectors can be attached with other selectors. It has two parts:

* Selector - Selector indicates the HTML element you want to style.
* Declaration Block - The declaration block can contain one or more declarations separated by a semicolon.

22) What is the difference between class selectors and id selectors?

An overall block is given to class selector while id selectors take only a single element differing from other elements.

CSS Class Selector

1. **<style>**
2. .center {
3. text-align: center;
4. color: blue;
5. }
6. **</style>**

CSS Id Selector

1. **<style>**
2. #para1 {
3. text-align: center;
4. color: blue;
5. }
6. **</style>**

[More details...](https://www.javatpoint.com/css-selector)

23) What are the advantages of External Style Sheets?

* You can create classes for reusing it in many documents.
* By using it, you can control the styles of multiple documents from one file.
* In complex situations, you can use selectors and grouping methods to apply styles.

[More details...](https://www.javatpoint.com/external-css)

24) What is the difference between inline, embedded and external style sheets?

**Inline**: Inline Style Sheet is used to style only a small piece of code.

Syntax

1. **<htmltag** style="cssproperty1:value; cssproperty2:value;"**>** **</htmltag>**

**Embedded**: Embedded style sheets are put between the <head>...</head> tags.

Syntax

1. **<style>**
2. body {
3. background-color: linen;
4. }
5. h1 {
6. color: red;
7. margin-left: 80px;
8. }
9. **</style>**

**External**: This is used to apply the style to all the pages within your website by changing just one style sheet.

Syntax

1. **<head>**
2. **<link** rel="stylesheet" type="text/css" href="mystyle.css"**>**
3. **</head>**

25) What is RWD?

RWD stands for Responsive Web Design. This technique is used to display the designed page perfectly on every screen size and device, for example, mobile, tablet, desktop and laptop. You don't need to create a different page for each device.

26) What are the benefits of CSS sprites?

If a web page has a large number of images that take a longer time to load because each image separately sends out an HTTP request. The concept of CSS sprites is used to reduce the loading time for a web page because it combines the various small images into one image. It reduces the number of HTTP requests and hence the loading time.

27) What is the difference between logical tags and physical tags?

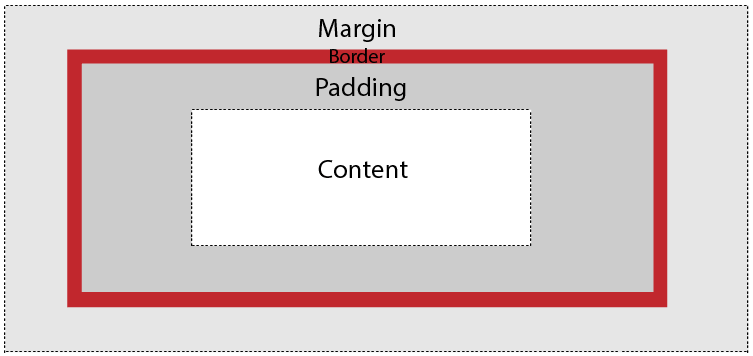
* Physical tags are referred to as presentational markup while logical tags are useless for appearances.
* Physical tags are newer versions, on the other hand, logical tags are old and concentrate on content.

28) What is the CSS Box model and what are its elements?

The CSS box model is used to define the design and layout of elements of CSS.

The elements are:

* Margin - It removes the area around the border. It is transparent.
* Border - It represents the area around the padding
* Padding - It removes the area around the content. It is transparent.
* Content - It represents the content like text, images, etc.



29) What is the float property of CSS?

The CSS float property is used to move the image to the right or left along with the texts to be wrapped around it. It doesn't change the property of the elements used before it.

To understand its purpose and origin, let's take a look at its print display. In the print display, an image is set into the page such that text wraps around it as needed.

Its web layout is also just similar to print layout.

[More details...](https://www.javatpoint.com/css-float)

30) How to restore the default property value using CSS?

In short, there is no easy way to restore to default values to whatever a browser uses.

The closest option is to use the 'initial' property value, which restores the default CSS values, rather than the browser's default styles.

31) What is the purpose of the z-index and how is it used?

The z-index helps to specify the stack order of positioned elements that may overlap one another. The z-index default value is zero and can take on either a positive or negative number.

An element with a higher z-index is always stacked above than a lower index.

Z-Index can take the following values:

* **Auto:** Sets the stack order equal to its parents.
* **Number:** Orders the stack order.
* **Initial:** Sets this property to its default value (0).
* **Inherit:** Inherits this property from its parent element.

32) Explain the difference between visibility: hidden and display: none?

**visibility: hidden** hides the element, but it occupies space and affects the layout of the document.

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. h1.vis {
6. visibility: visible;
7. }
9. h1.hid {
10. visibility: hidden;
11. }
12. **</style>**
13. **</head>**
14. **<body>**
15. **<h1** class="vis"**>**It is visible**</h1>**
16. **<h1** class="hid"**>**It is hidden**</h1>**
18. **<p>**Note - Second heading is hidden, but it still occupy space.**</p>**
19. **</body>**
20. **</html>**

**display: none** also hides the element but not occupy space. It will not affect the layout of the document.

1. <!DOCTYPE html**>**
2. **<html>**
3. **<head>**
4. **<style>**
5. h1.vis {
6. display: block;
7. }
9. h1.hid {
10. display: none;
11. }
12. **</style>**
13. **</head>**
14. **<body>**
15. **<h1** class="vis"**>**It is visible**</h1>**
16. **<h1** class="hid"**>**It is hidden**</h1>**
18. **<p>**Note - Second heading is hidden and not occupy space.**</p>**
19. **</body>**
20. **</html>**

33) What do you understand by W3C?

W3C stands for World Wide Web Consortium. Its purpose is to deliver the information of the World Wide Web. It also develops rules and guidelines for the Web.

34) What is tweening?

It is the process of generating intermediate frames between two images.

It gives the impression that the first image has smoothly evolved into the second one.

It is an important method used in all types of animations.

In CSS3, Transforms (matrix, translate, rotate, scale) module can be used to achieve tweening.

35) What is the difference between CSS2 and CSS3?

The main difference between CSS2 and CSS3 is that CSS3 is divided into different sections which are also known as modules. Unlike CSS2, CSS3 modules are supported by many browsers.

Apart from that, CSS3 contains new General Sibling Combinators which is responsible for matching the sibling elements with the given elements.

**1. What is CSS ?**

The full form of CSS is Cascading Style Sheets. It is a styling language which is simple enough for HTML elements. It is popular in web designing, and its application is common in XHTML also.

**2. What is the origin of CSS ?**

Standard Generalized Markup Language marked the beginning of style sheets in 1980s.

**3. What are the different variations of CSS ?**  
The variations for CSS are:

* CSS 1
* CSS 2
* CSS 2.1
* CSS 3
* CSS 4

**4. What are the limitations of CSS ?**

Limitations are:

* Ascending by selectors is not possible
* Limitations of vertical control
* No expressions
* No column declaration
* Pseudo-class not controlled by dynamic behavior
* Rules, styles, targeting specific text not possible

**5. What are the advantages of CSS ?**

Advantages are:

* Bandwidth
* Site-wide consistency
* Page reformatting
* Accessibility
* Content separated from presentation



*CSS*

**6. What are CSS frameworks?**

It is a pre-planned libraries, which allows easier and more standards-compliant webpage styling, using CSS language.

**7. How block elements can be centered with CSS1?**

Block level elements can be centered by:

The margin-left and margin-right properties can be set to some explicit value:



|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17 | BODY {    width: 40em;    background: fluorescent;    }    P {    width: 30em;    margin-right: auto;    margin-left: auto    } |

In this case, the left and right margins will be each, five ems wide since they split up the ten ems left over from (40em-30em). It was unnecessary for setting up an explicit width for the BODY element; it was done here for simplicity.

**8. Who maintains the CSS specifications?**

World Wide Web Consortium maintains the CSS specifications.

**9. In how many ways can a CSS be integrated as a web page?**

CSS can be integrated in three ways:

* Inline: Style attribute can be used to have CSS applied HTML elements.
* Embedded: The Head element can have a Style element within which the code can be placed.
* Linked/ Imported: CSS can be placed in an external file and linked via link element.

**10. What benefits and demerits do External Style Sheets have?**  
Benefits:

* One file can be used to control multiple documents having different styles.
* Multiple HTML elements can have many documents, which can have classes.
* To group styles in composite situations, methods as selector and grouping are used.

Demerits:

* Extra download is needed to import documents having style information.
* To render the document, the external style sheet should be loaded.
* Not practical for small style definitions.

**11. Discuss the merits and demerits of Embedded Style Sheets?**  
Merits of Embedded Style Sheets:

* Multiple tag types can be created in a single document.
* Styles, in complex situations, can be applied by using Selector and Grouping methods.
* Extra download is unnecessary.

Demerits of Embedded Style Sheets:

* Multiple documents cannot be controlled.

**12. What does CSS selector mean?**

A string equivalent of HTML elements by which declarations or a set of it, is declared and is a link that can be referred for linking HTML and Style sheet is CSS selector.

**13. Enlist the media types CSS allows?**

The design and customization of documents are rendered by media. By applying media control over the external style sheets, they can be retrieved and used by loading it from the network.

**14. Differentiate logical tags from physical tags?**

* While physical tags are also referred to as presentational mark-up, logical tags are useless for appearances.
* Physical tags are newer versions while logical tags are old and concentrate on content.

**15. Differentiate Style Sheet concept from HTML?**

While HTML provides easy structure method, it lacks styling, unlike Style sheets. Moreover, style sheets have better browser capabilities and formatting options.

**16. Describe ‘ruleset’?**

Ruleset : Selectors can be attached to other selectors to be identified by ruleset.

It has two parts:

* Selector, e.g. R and
* declaration {text-indent: 11pt}

**17. Comment on the Case-sensitivity of CSS ?**

Although, there are no case-sensitivity of CSS, nevertheless font families, URL’s of images, etc is. Only when XML declarations along with XHTML DOCTYPE are being used on the page, CSS is case -sensitive.

**18. Define Declaration block?**

A catalog of directions within braces consisting of property, colon and value is called declaration block.  
e.g.: [property 1: value 3]

**19. Enlist the various fonts’ attributes?**

They are:

* Font-style
* Font-variant
* Font-weight
* Font-size/line-height
* Font-family
* Caption
* Icon

**20. Why is it easy to insert a file by importing it?**

Importing enables combining external sheets to be inserted in many sheets. Different files and sheets can be used to have different functions. Syntax:

@import notation, used with <Style> tag.

**21. What is the usage of Class selector?**

Selectors that are unique to a specific style, are called CLASS selectors. Declaration of style and association with HTML can be made through this. Syntax:

Classname  
it can be A-Z, a-z or digits.  
.top {font: 14em ;}, class selector  
<Body class= “top”> this class is associated with element </body>

**22. Differentiate Class selector from ID selector?**

While an overall block is given to class selector, ID selector prefers only a single element differing from other elements. In other words, ID are uniques while classes are not. Its possible that an element has both class and ID.

**23. Can more than one declaration be added in CSS?**

Yes, it can be achieved by using a semicolon.

**24. What is Pseudo-elements ?**

Pseudo-elements are used to add special effects to some selectors.  CSS in used to apply styles in HTML mark-up. In some cases when extra mark-up or styling is not possible for the document, then there is a feature available in CSS known as pseudo-elements. It will allow extra mark-up to the document without disturbing the actual document.

**25. How to overrule underlining Hyperlinks?**

Control statements and external style sheets are used to overrule underlining Hyperlinks.

E.g.:



|  |  |
| --- | --- |
| 1  2  3  4  5  6  7 | B {    text-decoration: none;    }    <B href="career.html" style="text-decoration: none">link text</B> |

**26. What happens if 100% width is used along with floats all across the page?**

While making the float declaration, 1 pixel is added every time it is used in the form of the border, and   even more float is allowed thereafter.

**27. Can default property value be restored through CSS? If yes, how?**

In CSS, you cannot revert back to old values due to lack of default values. The property can be re- declared to get the default property.

**28. Enlist the various Media types used?**

Different media has different properties as they are case insensitive.

They are:

* Aural – for sound synthesizers and speech
* Print – gives a preview of the content when printed
* Projection- projects the CSS on projectors.
* Handheld- uses handheld devices.
* Screen- computers and laptop screens.

**29. What is CSS Box Model and what are its elements?**

This box defines design and layout of elements of CSS. The elements are:

**Margin**: the top most layer, the overall structure is shown  
**Border**: the padding and content option with a border around it is shown.  Background color affects the border.  
**Padding**: Space is shown. Background colour affects the border.  
**Content**: Actual content is shown.

**30. What is contextual selector?**

Selector used to select special occurrences of an element is called contextual selector. A space separates the individual selectors. Only the last element of the pattern is addressed in this kind of selector. For e.g.: TD P TEXT {color: blue}

**31. Compare RGB values with Hexadecimal color codes ?**

A color can be specified in two ways:

* A color is represented by 6 characters i.e. hexadecimal color coding. It is a combination of numbers and letters and is preceded by #.       e.g.: g {color: #00cjfi}
* A color is represented by a mixture of red, green and blue. The value of a color can also be specified. e.g.: rgb(r,g,b): In this type the values can be in between the integers 0 and 255. rgb(r%,g%,b%):  red, green and blue percentage is shown.

**32. Define Image sprites with context to CSS ?**

When a set of images is collaborated into one image, it is known as ‘Image Sprites’. As the loading every image on a webpage consumes time, using image sprites lessens the time taken and gives information quickly.

CSS coding:



|  |  |
| --- | --- |
| 1 | img.add { width: 60px; height: 55px; background: url (image.god) 0 0; } |

In this case, only the part needed is used. The user can save substantial margin and time through this.

**33. Compare Grouping and Nesting in CSS ?**

Grouping:  Selectors can be grouped having the same values of property and the code be reduced.  
E.g. :



|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17 | h1 {    color: blue;    }    h2 {    color: blue;    }    p {    color: blue;    } |

It can be seen from the code that every element shares the same property. Rewriting can be avoided by writing each selector separated by a comma.

Nesting: Specifying a selector within a selector is called nesting.



|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25 | P    {    color: red;    text-align: left;    }    .marked    {    background-color: blue;    }    .marked p    {    color: green;    } |

**34. How can the dimension be defined of an element ?**

Dimension properties can be defined by:

* Height
* Max-height
* Max-width
* Min-height
* Min-width
* Width

**35. Define float property of CSS?**

By float property, the image can be moved to the right or the left along with the text to be wrapped around it. Elements before this property is applied do not change their properties.

**36. How does Z index function?**

Overlapping may occur while using CSS for positioning HTML elements. Z index helps in specifying the overlapping element. It is a number which can be positive or negative, the default value being zero.

**37. What is graceful degradation?**

In case the component fails, it will continue to work properly in the presence of a graceful degradation. The latest browser application is used when a webpage is designed. As it is not available to everyone, there is a basic functionality, which enables its use to a wider audience. In case the image is unavailable for viewing, text is shown with the alt tag.

**38. What is progressive enhancement?**

It’s an alternative to graceful degradation, which concentrates on the matter of the web. The functionality is same, but it provides an extra edge to users having the latest bandwidth. It has been into prominent use recently with mobile internet connections expanding their base.

**39. How can backward compatibility be designed in CSS?**

HTML sheet methods is collaborated with CSS and used accordingly.

**40. How can the gap under the image be removed?**

As images being inline elements are treated same as texts, so there is a gap left, which can be   removed by:

CSS



|  |  |
| --- | --- |
| 1 | img { display: block ; } |

**41. Why is @import only at the top?**

@import is preferred only at the top, to avoid any overriding rules. Generally, ranking order is followed in most programming languages such as Java, Modula, etc. In C, the # is a prominent example of a @import being at the top.

**42. Which among the following is more precedent: CSS properties or HTML procedures?**

CSS is more precedent over HTML procedures. Browsers, which do not have CSS support, display HTML attributes.

**43. What is Inline style?**

The Inline style in a CSS is used to add up styling to individual HTML elements.

**44.  How comments can be added in CSS?**

The comments in CSS can be added with /\* and \*/.

**45. Define Attribute Selector ?**

It is defined by a set of elements, value and its parts.

**46. Define property?**

A style, that helps in influencing CSS. E.g. FONT. They have corresponding values or properties within them, like FONT has different style like bold, italic etc.

**47.  What is Alternate Style Sheet?**

Alternate Style Sheets allows the user to select the style in which the page is displayed using the view>page style menu. Through Alternate Style Sheet, user can see a multiple version of the page on their needs and preferences.

**48. Are quotes mandatory in URL’s?**

Quotes are optional in URL’s, and it can be single or double.

**49. What is at-rule?**

Rule, which is applicable in the entire sheet and not partly, is known as at-rule. It is preceded by @ followed by A-Z, a-z or 0-9.

**50. How can CSS be cascaded to mix with user’s personal sheet?**

Properties can be a set in recommended places and the document modified for CSS to mix with user’s   personal sheet.

**Part 1 – CSS3 Interview Questions (Basic)**

This first part covers basic CSS3 interview questions and answers

**1.In how many ways can a CSS be integrated as a web page?**

Answer:  
CSS can be integrated into three ways

* **Inline:** Style attribute can be used to have CSS applied to HTML elements.
* **Embedded:** the head element can have a style element within which the code can be placed.
* **Linked/Imported:** CSS can be placed in an external file and linked via link element.

**2. What benefits and demerits do external style sheets have?**

Answer:

Benefits:

* One file can be used for many different documents having multiple styles.
* Multiple HTML elements can have many different documents which can have classes.
* To group styles in composite situations, methods as selectors and grouping can be used.

Demerits

* An extra download is needed to import documents having style information.
* To render the document, external style sheet should be loaded.
* Not practical for small style definitions.

Let us move to the next CSS3 interview questions

**3. What are the merits and demerits of embedded style sheets?**

Answer:  
This is the basic CSS3 interview questions that have been frequently asked in an interview.  Following is the merit and demerit are as follows:

Merits:

* Multiple tag types can be created in a single document.
* Styles, in a complex situation, can be applied by using the Selector and grouping methods.
* Extra download in unnecessary.

Demerits

* Multiple documents cannot be controlled.

**4. What are the advantages and limitations of CSS?**

Answer:

Advantages

* Bandwidth
* Site-wide consistency.
* Page reformatting.
* Accessibility
* Content separated from presentation.

Disadvantages

* Ascending be selectors is not possible.
* Limitations of vertical control.
* No expressions.
* No column declaration.
* Pseudo-class not controlled by dynamic declarations.
* Rules, styles targeting specific text not possible.

**5. Can more than one declaration be added to CSS?**

Answer:  
Yes, it can be achieved using a semicolon;

**Part 2 – CSS3 interview questions (Advanced)**

Let us now have a look at the [advanced CSS3](https://www.educba.com/what-is-css3/) interview questions.

**6. Differentiate logical tags from physical tags?**

Answer:

* Physical tags are also referred to as presentational markup; logical tags are not used for presentations.
* Physical tags are newer version while logical tags are old and concentrate on content.

**7. What is a contextual selector?**

Answer:

Selector used to select special occurrences of an element is called a contextual selector. Space separates the individual selectors. Only the last element of the pattern is addressed in this kind of selector.

Let us move to the next CSS3 interview questions

**8. How does the Z index function?**

Answer:  
Overlapping may occur while using CSS for positioning HTML elements. Z index helps in specifying the overlapping element. It is a number which can be positive or negative, the default value being zero.

**9. Is it possible to make a class selector for a particular element? How?**

Answer:  
Yes, we can make a class selector for a particular element. For example

h2.className{  
Color: #FFFF;  
}  
In above example, whenever classname ‘className’ is found under element ‘h2’ apply white color.

**10. How can we create text shadow or box shadow in CSS3?**

Answer:  
Box shadow can be created  
box-shadow: 5px 5px 2px #ffff;  
text-shadow: 5px 5px 2px #ffff;

**11.What are new texts added in CSS3?**

Answer:

* Word wrap
* Text-overflow
* Word-break

1. **Question 1. What Is Wrapping In Css3?**

**Answer :**

Wrapping is a vital property for proper display of contents in web pages. If wrapping is disabled then the user could not display and view long lines that goes outside the window boundary and thus becomes useless.

1. **Question 2. What Is The Syntax Of Opacity In Css3?**

**Answer :**

style="opacity:0.4;filter:alpha(opacity=40)"  
Firefox uses the property opacity:x for transparency, while IE uses filter:alpha (opacity=x).

[XML Interview Questions](https://www.wisdomjobs.com/e-university/xml-interview-questions.html)

1. **Question 3. How Do You Manage The Complete Site In Center Of The Browser, Which Is Completely Using Divs Using The Help Of Css2 Or Css3?**

**Answer :**

In CSS file   
1. body{width:100%;}  
2. Create a div tag and assign a class, say 'container' and place all your content in this div tag.  
3. Assign properties to this class in CSS file  
container{width:777px; margin:auto;}

1. **Question 4. How Is Multi-column Feature Used In Css3?**

**Answer :**

By using the Multi-Column feature of CSS3 the web designers can get their output in multiple columns with definitions defined for each column like column-width, column-count.

[XML Tutorial](https://www.wisdomjobs.com/e-university/xml-tutorial-181.html)

1. **Question 5. What Is The Syntax Of Word Wrap In Css3?**

**Answer :**

The general syntax word-wrap property of CSS3 is as follows:  
word-wrap: normal| break-word  
The default initial value is normal in the above syntax.

[HTML Interview Questions](https://www.wisdomjobs.com/e-university/html-interview-questions.html)

1. **Question 6. How Flexibility Is Achieved More In Css3?**

**Answer :**

Flexibility achieved is in greater ratio in CSS3 because of the feature of handling multiple style sheets in CSS3 and because of the modularized approach of CSS3.

1. **Question 7. What Are The Values That Can Be Taken By Property White-space Of Css3?**

**Answer :**

The five values that can be taken by property white-space of CSS3 are normal, pre, no wrap, pre-wrap and pre-line.

[CSS3 Tutorial](https://www.wisdomjobs.com/e-university/css3-tutorial-205.html) [Dreamweaver Interview Questions](https://www.wisdomjobs.com/e-university/dreamweaver-interview-questions.html)

1. **Question 8. What Are Cascading Style Sheets?**

**Answer :**

Cascading style sheets help the designer in various ways. It primarily helps to design style sheets for content and HTML pages. It can be applied to any XML document or markup language.

1. **Question 9. Explain About Some Of The Features Of Css?**

**Answer :**

Some of the features used by CSS are to define a layout, colors, fonts, etc. It separates the content from the design part which makes the content much more visible thereby improving productivity of the entire process. It can make the same page appear in different formats.

[UI Developer Interview Questions](https://www.wisdomjobs.com/e-university/ui-developer-interview-questions.html)

1. **Question 10. Explain About The Rules Present In Style Sheets?**

**Answer :**

A huge list of rules is defined for style sheets. Each rule consists of selectors and declaration block. A declaration block consists of declarations which can be defined and separated by using a semicolon within curly braces. Also each declaration consists of a colon, a property and a semicolon.

[HTML Tutorial](https://www.wisdomjobs.com/e-university/html-tutorial-206.html)

1. **Question 11. Explain About The Hover Element?**

**Answer :**

A hover element is activated when the user moves a pointer over the area where HTML coding is defined. This style is being extensively used for online advertising. It is appended to #elementid:hover.

[CSS Interview Questions](https://www.wisdomjobs.com/e-university/css-interview-questions.html)

1. **Question 12. State Some Of The Uses Of Css?**

**Answer :**

CSS offers many possibilities to a designer and developer. Some of them are  
1) Moving of the entire style sheet coding to a separate external folder.  
2) Repetition and code reuse features make CSS rich.  
3) Presentation and structure are made different by CSS etc.

[XML Interview Questions](https://www.wisdomjobs.com/e-university/xml-practice-tests-181-326158)

1. **Question 13. State The Different Type Of Author Styles?**

**Answer :**

A web page author, developer, designer, etc can describe styles in three different forms they are  
1) External style sheets  
2) Embedded style  
3) Inline styles  
4) Local and default style sheet.  
A default style sheet consists of the style imparted by the browser which the user is accessing for the information.

[CSS Tutorial](https://www.wisdomjobs.com/e-university/css-tutorial-1198.html)

1. **Question 14. What Are The Main Goals Of Applying Style Sheets?**

**Answer :**

The main use of style sheet is to give the user a greater degree of control over his presentation. If the user does not like a certain font he can change it with flexibility without any difficulty.

Also a developer can minimize the effects which a browser can have on the application or the text part of the webpage.

Control over the entire web page without any external influences, separation of the page from the HTML code part.

1. **Question 15. Explain About Css1?**

**Answer :**

W3C recommended adoption of CSS1in 1996. Some of the capabilities and features present in CSS1 are as follows.  
1) Font properties  
2) Background colors, text colors, etc  
3) Spacing between text, words, etc  
4) Tables, images alignment  
5) Border, margin, padding and positioning of elements  
6) Attributes are uniquely identified and classified generically.

[CSS Advanced Interview Questions](https://www.wisdomjobs.com/e-university/css-advanced-interview-questions.html)

1. **Question 16. What Are The Additional Features Which Were Present In Css2 Compared To Css1?**

**Answer :**

Many of the new features were present in CSS2 some of them are as follows.  
1) Positioning of elements will be according to the relative, fixed and the absolute positioning.  
2) Inclusion of large number of media functions.  
3) Shadows, bidirectional support for text and aural style sheets.

[CSS Advanced Tutorial](https://www.wisdomjobs.com/e-university/css-advanced-tutorial-1199.html)

1. **Question 17. Explain About Css Filter?**

**Answer :**

CSS filter solves designers and coders difficulties. With the help of these filters one can write entirely different code or modify the existing design patterns so that browsers receive CSS specifications which they support, thereby not allowing the browsers to make or create any unnecessary changes.

[WordPress Interview Questions](https://www.wisdomjobs.com/e-university/wordpress-interview-questions.html)

1. **Question 18. Explain About Internet Explorer Box Model Bug?**

**Answer :**

Designers had a very tough time dealing with this bug, because it will not display box widths appropriately on the webpage. It displayed the block to be narrow in size. This bug can be rectified by using CSS filter and hacks. Also correct usage and documentation of XHTML can solve the problems.

[HTML Interview Questions](https://www.wisdomjobs.com/e-university/html-interview-questions.html)

1. **Question 19. Explain About Vertical Control Limitation?**

**Answer :**

Vertical control limitations are always a nightmare for a designer. Horizontal positioning of text or element is always easy but vertically positioning an element always leads to convoluted and impossible tasks. CSS has unsupported rules for vertical placement of elements.

[WordPress Tutorial](https://www.wisdomjobs.com/e-university/wordpress-tutorial-1204.html)

1. **Question 20. Explain About Absence Of Expressions?**

**Answer :**

CSS currently does not have the ability to specify property values as simple expressions. For example if you want to calculate the size of all the columns but you have a restriction on the sum of all columns. This problem has been solved by many browsers by proprietary expressions which did the job well.

[Pure.CSS Interview Questions](https://www.wisdomjobs.com/e-university/puredotcss-interview-questions.html)

1. **Question 21. Explain About Lack Of Orthogonality?**

**Answer :**

Lack of orthogonality explains about the various multiple jobs which are either undefined correctly for a specific application or used multiple times for the same job. CSS defines that internal elements of a table will not have margins so usage of border spacing is appropriate which leads to confusion.

1. **Question 22. Explain About The Ease Of Maintenance With Css.**

**Answer :**

An effective style sheet has effective use of inheritance and cascading style sheets. When changes are applied to a single element of this global style sheet every element of the web page changes. This made the maintenance easy because here we can change the style of the whole page by effecting a change to a single element.

[Pure.CSS Tutorial](https://www.wisdomjobs.com/e-university/puredotcss-tutorial-1224.html)

1. **Question 23. Explain About Float Containment?**

**Answer :**

CSS does not support float elements as it can cause overflow of the elements. Floats also tend to vary according to the web browser resolution and size but positions do not. There are some multiple properties which can define about float but they have their own limitations.

[XHTML Interview Questions](https://www.wisdomjobs.com/e-university/xhtml-interview-questions.html)

1. **Question 24. State Some Limitations Of Style Sheets?**

**Answer :**

Style sheets do have its own share of limitations some of them are as follows: -  
1) Inconsistent browser support  
2) Vertical control limitations  
3) Margin collapsing, float containment, control of element shapes, etc  
4) Lack of column declaration and variables are some of the limitations present in CSS.

[Dreamweaver Interview Questions](https://www.wisdomjobs.com/e-university/dreamweaver-interview-questions.html)

1. **Question 25. What Is Id Selector?**

**Answer :**

ID selector is an individually identified (named) selector to which a specific style is declared. Using the ID atthbute the declared style can then be associated with one and only one HTML element per document as to differentiate it from all other elements ID selectors are created by a character# followed by the selectors name. The name can contain characters a-z. A-Z. digits 0-9. period. hyphen. escaped characters, Unicode characters 161 -255. as well as any Unicode character as a numeric code, however they cannot start with a dash or a digit.

#abc 123 {color red: background: black)

This and only this element can be identified as abcl23

[XHTML Tutorial](https://www.wisdomjobs.com/e-university/xhtml-tutorial-1261.html)

1. **Question 26. What Is Css Rule ‘ruleset’?**

**Answer :**

There are two types of CSS rules: ruleset and at-rule. Ruleset identifies selector or selectors and declares style which is to be attached to that selector or selectors. For example P {text-indent: 10pt) is a CSS rule. CSS rulesets consist of two parts: selector e.g. P and declaration. e.g. {text-indent: 10pt)  
P {text-indent: 10pt) - CSS rule (ruleset)  
{text-indent: 10pt) - CSS declaration  
text-indent - CSS property  
10pt-CSS value

1. **Question 27. Display And Visibility Properties Are Used To Hide And Show Elements In Any Page. Then How They Are Different From Each Other?**

**Answer :**

As said both the properties are used to hide and Show elements but they are different in the way they both work visibility property, set to hidden will still occupy the space in the layout but display:none does not take up the space in the page

[UI Developer Interview Questions](https://www.wisdomjobs.com/e-university/ui-developer-interview-questions.html)

1. **Question 28. How To Page Break After An Html Element In Css?**

**Answer :**

Use following code snippet  
<p style=”page-break-after: always’>Place your text</p>

After above code. the rest content will appear in the next page. (It will not be visible as next page in browser but on the printer and in Print Preview, you will see them as next page)

1. **Question 29. What Is The Default Value Of “position” Attribute In Css?**

**Answer :**

Default value Is “static “.

1. **Question 30. What Are The Values Of “position” Attribute In Css?**

**Answer :**

Possible values are static. relative, absolute, fixed. Inherit.

**1. What are the different versions of CSS ?**

Different versions of CSS are:

* CSS 1
* CSS 2
* CSS 2.1
* CSS 3
* CSS 4

**2. In how many ways can a CSS be integrated as a web page?**

CSS can be integrated in three ways:

* **Inline**: term is used when the CSS code have attribute of HTML elements

<p style="colour:skyblue;"> hello world!</p>

* **External:** separate CSS file is created in the workspace and later linking them in every web page that is created
* <head>
* <link rel="text/css"
* href="your\_CSS\_file\_location"/>
* </head>
* **Internal**: the head element of the web page has internal CSS implemented in it
* <head>
* <style>
* p{
* color:lime;
* background-color:black;
* }
* </style>
* </head>

**3. What is CSS?**

Cascading style sheets or CSS is a web designing language simple for HTML elements. The application is commonly known as XHTML. It is basically used to simplify the process and make the web page look presentable.

**4. What benefits and demerits do External Style Sheets have?**

The advantages of External Style sheets are:

* The style of several documents can be controlled from site by using them.
* Multiple HTML elements can have many documents, where classes can be created.
* To group styles in complex situations, selector and grouping methods are used.

Demerits of external css are as follows:

* Extra download is needed to import documents having style information.
* To render the document, the external style sheet should be loaded.
* Not practical for small style definitions.

**5. Discuss the merits and demerits of Embedded Style Sheets?**

There are several merits and demerits of embedded style sheets:  
**Merits of Embedded Style Sheets:**

* Multiple tag types can be created in a single document.
* Styles, in complex situations, can be applied by using Selector and Grouping methods.
* Extra download is unnecessary.

**Demerits of Embedded Style Sheets:**

* Multiple documents cannot be controlled.

**6. What is CSS Box Model and what are its elements?**

The CSS box defines the design and the layout of elements of CSS. The several elements are:

**Margin**: transparent area outside border  
**Border**: the padding and content option with a border around it is shown.  
**Padding**: Space is around content. Padding is transparent.  
**Content**: box where text and images appear.  
e.g.:

div{

width: 300px;

border: 25px solid yellow;

padding: 25px;

margin: 25px;

}

**7. What is the purpose of the z-index and how is it used?**

The z-index helps specify the stack order of positioned elements that may overlap one another. The z-index default value is zero, and can take on either a positive or negative number.

An element with a higher z-index is always stacked above than a lower index.

Z-Index can take the following values:

* **Auto:** Sets the stack order equal to its parents.
* **Number:** Orders the stack order.
* **Initial:** Sets this property to its default value (0).
* **Inherit:** Inherits this property from its parent element.

**8. What are the benefits of CSS sprites?**

Benefits of using CSS sprites are

* It is a technique where one has a large image containing a set of small images. Those images can be broken down with the help of CSS to disintegrate into multiple images.
* It helps large images or pages to load faster hence, saving a lot of time. It cuts back HTTP requests. It is also flexible over website’s layout and design.
* The concept of CSS sprites is used to reduce the loading time for a web page because it combines the various small images into one image. It reduces the number of http requests and hence the loading time.

**9. How can you integrate CSS on a web page?**

CSS can be integrated in three ways:

* Inline: term is used when the CSS code have attribute of HTML elements

<p style="colour:skyblue;"> My Sky!</p>

* External: separate CSS file is created in the workspace and later linking them in every web page that is created

<head>

<link rel="text/css" href="your\_CSS\_file\_location"/>

</head>

* Internal: the head element of the web page has internal CSS implemented in it

<head>

<style>

p{

color:lime;

background-color:black;

}

</style>

</head>

**10. Compare RGB values with Hexadecimal color codes?**

A color can be specified in two ways:

* A color is represented by 6 characters i.e. hexadecimal color coding. It is a combination of numbers and letters and is preceded by #. e.g.: g {color: #00cjfi}
* A mixture of red, green and blue represents a color. The value of a color can also be specified.

e.g.: rgb(r,g,b):  
In this type the values can be in between the integers 0 and 255. rgb(r%,g%,b%): red, green and blue percentage is shown.

**11. Enlist the various Media types used?**

Different media has different properties as they are case insensitive.  
They are:

* Aural – for sound synthesizers and speech
* Print – gives a preview of the content when printed
* Projection- projects the CSS on projectors.
* Handheld- uses handheld devices.
* Screen- computers and laptop screens.

**12. What is Pseudo-elements?**

Pseudo-elements are keyword added to the selector that allows one o style a specific part of the selected element. CSS in used to apply styles in HTML mark-up. In some cases when extra mark-up or styling is not possible for the document, then there is a feature available in CSS known as pseudo-elements. It will allow extra mark-up to the document without disturbing the actual document. It can be used for;

1. To style the first letter, line or element
2. To insert a content

Syntax

Selector: :pseudo-element {

property1 :value;

property2 :value;

}

**13. What is the usage of Class selector?**

A name preceded by a full stop is considered as a class selector in CSS. Selectors that are unique to a specific style, are called CLASS selectors. Declaration of style and association with HTML can be made through this. An ID identifies a single element whereas a class identifies more than one element.

Syntax for the class selector:

**Classname:**it can be A-Z, a-z or digits.

.top {font: 14em ;} //class selector

<Body class= "top"> //this class is associated with element </body>

**14. Enlist the various fonts’ attributes?**

They are:

* Font-style
* Font-variant
* Font-weight
* Font-size/line-height
* Font-family
* Caption
* Icon

**15. Comment on the Case-sensitivity of CSS ?**

Basically it is not case sensitive but the class names are considered as case sensitive in HTML 4.01 nevertheless font families, URL’s of images, etc is. Only when XML declarations along with XHTML DOCTYPE are being used on the page, CSS is case -sensitive.

**16. Enlist the media types CSS allows?**

Media is one of the most important features of CSS. Media renders the design and customization of documents. By applying media control over the external style sheets, they can be retrieved and used by loading it from the network. Some of the media types are:

* All – for all devices
* Aural – speech synthesizers
* Braille – feedback devices
* Embossed – paged braille printers
* Handheld – typically small screen, limitation of bandwidth
* Projection – basically for projectors
* Tv – television type devices
* Screen – color computer screens

**17. Differentiate logical tags from physical tags?**

There are several pointers that make local tags different from physical tags:

|  |  |
| --- | --- |
| **Physical Tags** | **Local Tags** |
| Physical tags are used to indicate how a particular character is to be formatted | logical tags are used to indicate by the visually impaired and put emphasis on the text. |
| Physical tags are also referred to as presentational mark-up | Logical tags are useless for appearances |
| Physical tags are newer versions | Logical tags are old and concentrate on content |

**18. Differentiate Style Sheet concept from HTML?**

While HTML provides easy structure method, it lacks styling, unlike Style sheets. Moreover, style sheets have better browser capabilities and formatting options. CSS works better on bigger pages and as the pages grow the benefits become more and more visible. HTML is basically for smaller pages. Due to modularity, CSS has become popular it makes the process simple and webpages more presentable and is not meant for HTML alone.

**19. Describe ‘rule set’?**

It is an instruction that tells browser on how to render a specific element on the HTML page. It consists of a selector with a declaration block that follows. Rule set: Selectors can be attached to other selectors to be identified by rule set.

It has two parts:

* Selector, e.g. R and
* Declaration block {text-indent: 11pt}

**20. What are the limitations of CSS?**

There are several limitations of CSS such as:

* CSS can’t fulfill turning completeness hence, it can never perform logical like ‘if/else’, for/while, etc, or arithmetical tasks
* One cannot read files using CSS
* It cannot provide total control over document display and allows the contents of the page to come through whatever the browser is used.
* Ascending by selectors is not possible
* Limitations of vertical control
* No expressions as it is a text-based coding language
* No column declaration
* Pseudo-class not controlled by dynamic behavior
* Rules, styles, targeting specific text not possible

**21. What are the advantages of CSS?**

There are a number of advantages of CSS,

* It gives lots of flexibility for setting the properties of the element
* Easy maintenance
* It allows separation of content of the HTML document from the style and layout of the content basically
* Loading of pages at a faster pace
* Compatibility with multiple device
* Increases the website’s adaptability and makes it compatible to future browsers

CSS is designed to allow presentation and content to be separated, including layout, colors, and fonts.

This divergence can enhance the availability of content, provide more flexibility and control in presentation specifications, allow different web pages to share proofreading by stipulating the relevant CSS in a separate.css file and lessen the complexity and repetition of major structural content.

**1. What is the difference between CSS and CSS3 ?**

**CSS3**is upgraded version of **CSS**with new future like Selectors, Box Model, Backgrounds and Borders, Text Effects,2D/3D Transformations, Animations, Multiple Column Layout, User Interface etc.

**Below we have listed some difference between CSS and CSS3**

|  |  |
| --- | --- |
| **CSS** | **CSS 3** |
| CSS is the basic version so it doesn’t support responsive design, and cannot handle media queries. | CSS3 supports responsive design, and can also handle media queries. This feature adds entirely new responsive design capabilities to the CSS repertoire. |
| `CSS cannot split into varied modules | Css3 can split into modules. |
| CSS is comparatively slower than CSS3 | It is faster than all its previous versions |
| The user cannot create 3D transformations and animations using CSS | The user can easily create 3D transformations, transitions, and animations using CSS3.  For the very first time, elements can move on screen without the help of Flash code or JavaScriopt. Elements can also change their size and color. |
| It has old and standard colors. | New colors have been added in CSS3. It supports RGBA, HSLA, HSL and gradient colors.  Another achievement for CSS3 is that it can now support the effect of rounded image corners. |
| It doesn’t have any box-sizing tool. The user has to follow the standard procedures to align the text. | All the alignment problems are now fixed by the Box- Sizing tool. This tool allows its user to get the right size for their element without having to subtract dimensions for padding and borders. |
| [CSS](https://www.onlineinterviewquestions.com/css-interview-questions/)or cascading style sheets is the key element for web designing. Both CSS and CSS3 is more or less the same thing i.e. both are web designing tools. | [CSS3](https://www.onlineinterviewquestions.com/css3-interview-questions/)is nothing but an upgraded version of CSS. Just like [HTML5](https://www.onlineinterviewquestions.com/html5-interview-questions/)is an upgraded version of the basic HTML. This version comes with the above-mentioned features and hence is a better tool for web designing. |

**2. What is the syntax of opacity in CSS3?**

style="opacity:0.4;filter:alpha(opacity=40)"

Firefox uses the property opacity:x for transparency, while IE uses

**3. What are CSS3 Transitions?**

CSS3 transitions allow you to change property values smoothly (from one value to another), over a given duration.

div {

-webkit-transition: width 2s, height 4s; /\* Safari \*/

transition: width 2s, height 4s;

}

**4. What is the word wrap/word wrapping in CSS3?**

**word wrap/word wrapping** is used to allow long words to be able to break and wrap onto the next line in css3 we used word-wrap property like below class

.wrapWord{word-wrap:break-word;}

**5. Explain What are the values that can be taken by property white-space of CSS3?**

The five values that can be taken by property white-space of CSS3 are

* Normal
* pre
* nowrap
* pre-wrap
* pre-line

**6. Explain How flexibility is achieved more in CSS3?**

Flexibility achieved is in the greater ratio in CSS3 because of the feature of handling multiple style sheets in CSS3 and because of the modularized approach of CSS3.

**7. What is the CSS3 animation?**

When the animation is created in the @keyframe, bind it to a selector, otherwise, the animation will have no effect.

Bind the animation to a selector by specifying at least these two CSS3 animation properties:

* Specify the name of the animation
* Specify the duration of the animation

**8. List out CSS3 modules?**

Below is list of most important CSS3 modules are

* Selectors
* Box Model
* Backgrounds and Borders
* Text Effects
* 2D/3D Transformations
* Animations
* Multiple Column Layout
* User Interface

**9. What are the possible values of the “Position” attributes?**

The possible value of the “Position” attributes are

* absolute
* fixed
* inherit
* relative
* static

**10. List some advantages to CSS3 animations over script-based animation?**

Advantages of using CSS3 animations over script-based animation techniques are as follows:

1. Easy to use and anybody can create them without the knowledge of JavaScript.
2. Executes well even under reasonable system load. As simple animations perform poorly in JavaScript, the rendering engine uses the frame-skipping techniques to allow smooth flow of animation.
3. Allows the browser to control the animation sequence, optimize performance and efficiency by reducing the update frequency of animations executing in tabs that aren’t currently visible.

**11. What is CSS3 Flexbox?**

Flexible boxes, or flexbox, is a new layout mode in CSS3. Flexbox consists of flex containers and flex items.  
A flex container is declared by setting the display property of an element to either flex (rendered as a block) or inline-flex (rendered as inline).Inside a flex container, there are one or more flex items.

**Q #1) Define Bootstrap.**

**Answer:** Bootstrap is a front-end framework that is used for creating HTML, CSS, and JS web applications. Its layout is very responsive, easy and fast to use. It mostly focuses on building a mobile application with having design templates for creating UI like Dropdown, Forms, Buttons, Alerts Tab, etc.

**Q #2) Why Bootstrap is used for Mobile Web Development?**

**Answer:** It is used for Mobile Web development because it has responsive designs and templates which is easy to use.

**Q #3) Explain the features of Bootstrap.**

**Answer: Its Features include:**

* Open Source for use
* Compatibility with all browsers.
* Responsive designs
* Easy to use and fast.

**Q #4) Define the key components of Bootstrap.**

**Answer: Its components include:**

* Scaffolding – It has the grid system, background, styles.
* JS Plugins – Contains JS and jQuery plugins.
* Customize – Can customize frameworks.
* CSS – Contains CSS files.

**Q #5) What do you understand by Bootstrap container?**

**Answer:** Bootstrap container behaves like a container where you can put HTML code and it is a part within the page where the content of the site can be placed to make it responsive and fast.

**Q #6) What do you mean by Class loaders?**

**Answer:** ClassLoaders load java classes into JVM (Java Virtual Machine) and translate the classes into its correspondent binary form.

**Q #7) How many types of layouts are there in Bootstrap?**

**Answer:** There are two types of layouts in Bootstrap.

**They are:**

* Fluid Layout
* Fixed Layout

**Q #8) Define Fluid Layout.**

**Answer:** Fluid Layout is useful when you need to make an app which involves the full width of the screen i.e. Fluid Layout adjusts itself according to the browser size.

**Q #9) Define Fixed Layout.**

**Answer:** Fixed layout is responsive and easy to use but just like the fluid layout, it cannot adjust itself according to the browser size. Fixed Layout should be 940 px in most cases.

**Q #10) How can you display a code in Bootstrap?**

**Answer:** You can display the code in two ways i.e. by using the <code>tag and by using the <pre>tag.

**Q #11) When will you use <code>tag and <pre>tag?**

**Answer:** <code>tag is used to show the code inline and <pre>tag is used to show code with multiple lines.

**Q**#**12) What is a progress bar in bootstrap?**

**Answer:** Progress bar is used with HTML tag style in HTML element using <**progress**> keyword. In bootstrap we used html5 <**progress**> with CSS classes that have special features in bootstrap, that is only made for the progress bar.

**Q**#**13) Name the contextual classes that are used with the progressive bar in bootstrap.**

**Answer:** The contextual classes used with progressive bar are as follows.

* Progress-success
* Progress-info
* Progress-warning
* Progress-danger

**Q**#**14) What are responsive utility classes in Bootstrap?**

**Answer:** Responsive utility classes in bootstrap are a set of classes which are used to conceal or exhibit the HTML elements based on screen resolution that discerns by media query in bootstrap.

**Example:** “hidden-md-down”, It hides

**Q** #**15) What are the different button styles in Bootstrap?**

**Answer:** In bootstrap there are seven styles which we can use with the bootstrap button.

* .btn-default.
* .btn-primary
* .btn-success
* .btn-info
* .btn-warning.
* .btn-danger.
* .btn-link.

**Q**#**16) What are Bootstrap alerts?**

**Answer:** This is used to create presume alert messages, which adds style to the messages to look more noticeable to the user.

There are four classes in alerts i.e .alert-success, .alert-info, .alert-warning, .alert-danger.

**Q**#**17) Define Bootstrap thumbnails.**

**Answer:** It is a way to use the layout images, videos, text etc. in a grid system. We can create thumbnails by adding a tag with the class **.thumbnails**around the image.

This will add four pixels of padding and a grey border.

**Q**#**18) Explain Modal plugin in Bootstrap.**

**Answer:** A model is an inherited window that is layered over its parent window. This is used to augment the user experience and adds different functionalities to the users.

Model windows are created with the help of the modal plugin.

**Q**#**19) Which class is used for pagination in Bootstrap?**

**Answer:** To add pagination on the webpage we have to use the class **.pagination**.

**Q**#**20) Explain what is Bootstrap collapsing elements.**

**Answer:** It allows you to collapse any particular element without using any JavaScript code.

To use this feature in bootstrap you have to add data-toggle=” collapse” to the controller element along with a data target to automatically assign the control of a collapsible element. We can use this by writing **.collapse(options)**etc.

**Q**#**21) What is Bootstrap Well?**

**Answer:** Bootstrap well is a form of container which thrives or makes the content to look recessed on the web page. It also wraps the content by using **.well**class.

**Q**#**22) Explain the uses of carousel plugin in Bootstrap.**

**Answer:** Carousel plugin in bootstrap is used to make sliders in the web pages or your site. There are several carousel plugins that are used in bootstrap to display large contents within a small space by adding sliders.

**Example:** **.carousel(options), .carousel(‘pause’), .carousel(cycle’), .carousel(‘prev’), .carousel(‘next’)**.

**Q** #**23) What will be the output of the below code and why?**

<div class="progress">

<div class="progress-bar progress-bar-success" style="width: 65%">

<span class="sr-only">75% successfully completed</span>

</div>

<div class="progress-bar progress-bar-warning" style="width: 20%">

<span class="sr-only">30% completed with warnings</span>

</div>

<div class="progress-bar progress-bar-danger" style="width: 15%">

<span class="sr-only">15% did not complete</span>

</div>

</div>

**Answer:** If we place multiple bars with the same **.progress**parent element, Bootstrap will pile them into one single progress bar. As we know, in bootstrap the sum of the progress bar is 100 %. So, the progress bar will give the result as full width and fully populated.

**Q**#**24) How can we customize links of pagination in Bootstrap?**

**Answer:**We can customize the links by using **.disabled** for unclickable links and **.active** for indicating the current page.

**Q** **#25) Explain input group in Bootstrap.**

**Answer:** Input group in bootstrap are put out from controls. By using an input group, we can easily add prepended and appended text or button to the text-based inputs.

We can prepend and append elements to a **.form-control**by taking all the elements in a <div> under a class**.input-group.** After that, place your extra content inside a <span> in same <div> by using class **.input-group-addon**after this you can place the <span> element either before or after the input element.

**Q**#**26) Write the ways to create a tabbed navigation menu in Bootstrap.**

**Answer:** We can create a tabbed navigation by making a basic unordered list with the base class of **.nav** and after this, we can add class **.nav-tabs**.

**Q** #**27) In Bootstrap, how can you create a pills navigation menu?**

**Answer:** Pills navigation menu in bootstrap is created by making an unordered list at first with the base class of **.nav**and after this add the class **.nav-pills**.

**Q**#**28) How navbar works in Bootstrap?**

**Answer:** In bootstrap, navbar is an eminent feature to make responsive meta component that works as navigation headers for your application and site. In mobile view, navbar collapses and become horizontal as the available viewport width increases.

**Q**#**29) How we can create a navbar in Bootstrap?**

**Answer:** To create a navbar in a bootstrap at first, we have to add the classes **.navbar**, **.navbar-default**to the <nav> tag. After this, we have to add the role=”navigation” to the above element, and this will help in accessibility.

We have added a header class **.nav-header** to the <div> element, which will include an <a> element with a class navbar brand. From this, we will get a text with a larger size.

**Q**#**30) What is Bootstrap breadcrumb?**

**Answer:** Bootstrap breadcrumb is an efficient way to show hierarchy-based information for a site. This can show the dates of publishing, categories or tags in a blog. They also tell the user about the current page location within a navigational hierarchy.

So we can say that Bootstrap breadcrumb is simply an unordered list with a class of **.breadcrumb**.

**Q #31) What are labels in Bootstrap?**

**Answer:** Bootstrap labels are used for offering counts, tips or other things to provide markup on web pages. To use the label in Bootstrap we use the class **.labels** to indicate the labels.

**Q** **#32) What are badges in Bootstrap?**

**Answer:** Badges are homogeneous to labels, the main difference between them is corners are more rounded. The main work of badges in the bootstrap is to highlight new or unread items. To use badges just add <span class=”badge”> to links and bootstrap navs.

**Q**#**33) What is a jumbotron in Bootstrap?**

**Answer:** It is used to increase the size of headings and to add a lot of margins for landing page content. To create a jumbotron we have to create a container <div> with the class of **.jumbotron.**

**Q**#**34) How can we make image responsive in Bootstrap?**

**Answer:** After the updates in Bootstrap, currently the feature to make an image responsive has been added, we can do this by adding a class **.img-responsive**to the <img> tag. This class makes the width max-width =100%; and height=auto; to the image so that it matches nicely to the parent element.

**Q**#**35) What do you mean by normalize in Bootstrap?**

**Answer:** Bootstrap normalize is a small CSS file which is used to make cross-browser consistency.

**Q** #**36) What is lead body copy in Bootstrap?**

**Answer:** It is used to add some ascent to the paragraph if we add class=”lead”. This will enlarge the font size and a taller line height.

**Q**#**37) What are panels in Bootstrap?**

**Answer:** Panels are components that are used when you want to put your DOM component in a box. So, to retrieve a basic panel we just need to add class.panel to the <div> element. We can also add class.panel-default to this element.

**Q**#**38) How will you create a Bootstrap panel with heading?**

**Answer:** There are two ways by which we can add panel heading.

First is, we can directly use **.panel-heading**class to add heading container in a panel and the second way is by using any heading tag like <h1> to <h6> with a **.panel-title** class to give more styles on the heading.

**Q**#**39) What is a scrollspy plugin in Bootstrap?**

**Answer:** It is an auto-updating nav plugin which allows in fetching section of the page based on the scroll position. This can be done by **the .active class** to the navbar based scroll position.

**Q**#**40) What is the work of affix plugin in Bootstrap?**

**Answer:** This plugin allows a <div> to be attached to a location on the page.

Use of the social icon in a page is an example for this in which we see that the icons will start in a location, but when the page hits on a certain mark it will block the <div> in place and will stop the scrolling for rest of the page.

**Q**#**41) What is grid system in Bootstrap?**

**Answer:** By using the grid system, we can make up to 12 columns across a page. There are different classes which have been defined for this for the UI purpose.

**Q**#**42) What are Grid classes in the Bootstrap?**

**Answer:** There are four grid classes in Bootstrap.

**They are:**

* xs (It is used for phone screens less than 786px wide).
* sm (It is used for the tablet screens which are greater than 786px wide).
* md (It is for small laptop screen of size equal to or greater than 992px wide).
* LG ( It is for laptop and desktop screens which are equal to greater than 1200px wide).

**Q**#**43) What are global styles that are used in Bootstrap Default Typography?**

**Answer:** In Bootstrap the global default font-size is 14px and the line height is 1.428. The default font changes to Helvetica and Arial are with sans-serif fallback and all these styles are applicable for both body and all paragraphs.

**Q**#**44) What will be the output of the below code?**

<div

class="row">

<div class="col-xs-12 col-md-3">.col-xs-12 .col-md-3</div>

</div>

**Answer:** The output of this will give the grids for extra small devices as we can see in the snippet col-xs-12 that has been used and it will also give the grids for desktop devices and above as the class col-md-3 has been used.

**Q**#**45) What dependencies does Bootstrap require to work properly?**

**Answer:** jQuery is the only dependency that bootstrap requires for working properly and this is only for JavaScript plugins in bootstrap.

**Q**#**46) Explain what the below code will do?**

<a href="#">Home <span class="badge">36</span></a>

**Answer:** This code will produce a link with an inline badge which will give an important notification to the user like number received, message received or the number of requests etc.

**Q**#**46) What are the two codes that are used for code display in Bootstrap?**

**Answer:** The codes are <code> tag and <pre> tag.

**Q**#**47) What is the difference between Bootstrap and Foundation?**

**Answer:** Bootstrap uses very fewer preprocessors as it supports less and it allows the designing and development for both mobile and desktop. On the other hand, Foundation supports sass processors and it is used only for mobile UI designing.

**Q**#**48) What are Glyphicons in Bootstrap?**

**Answer:** By this, we can use the icon simply anywhere in your code.

<span class=”glyphicon glyphicon-search”></span>

**Q**#**49) What is a transition plugin in Bootstrap?**

**Answer:** It provides simple transition effects like sliding or fading in modals.

**Q**#**50) Explain the concept of creating a vertical or basic form in Bootstrap.**

**Answer:** For this first we have to add a role form to the parent <form> element then we have to wrap labels and controls in a <div> with class.form-group and then we have to add a class of .form-control to all text url <input>,<textarea> and <select> elements.

**1) Explain what is Bootstrap?**

Bootstrap is a HTML, CSS, and JS framework for building the rich web applications with minimal effort. This framework emphasis more on building mobile web applications.

**2) Explain why to choose Bootstrap for building the websites?**

There are few reason why we choose Bootstrap for building websites

* Mobile Support: For mobile devices it provides full support in one single file rather than in separate file. It supports the responsive design including adjusting the CSS based on the different types of device, size of the screen etc. It reduces extra effort for developers.
* Easy to learn: Writing application in bootstrap is easy if you know CSS and HTML
* Browser Support: It supports all the popular browsers like Firefox, Opera, Safari, IE etc.

**3) What are the key components of Bootstrap?**

The key components of Bootstrap are

* CSS : It comes with plenty of CSS files
* Scaffolding : It provides a basic structure with Grid system , link styles and background
* Layout Components : List of layout components
* JavaScript Plugins: It contains many jQuery and JavaScript plugins
* Customize: To get your own version of framework you can customize your components

**4) Explain what are class loaders in Bootstrap?**

Class loader is a part of JRE (Java Runtime Environment) which loads Java classes into Java virtual environment. Class loaders also does the process of converting a named class into its equivalent binary form.

**5) What are the types of layout available in Bootstrap?**

In Bootstrap there are two types of Layout available

* Fluid Layout: Fluid layout is used when you want to create a app that is 100% wide and use up all the width of the screen
* Fixed Layout: For a standard screen you will use fixed layout (940 px) option



**6) Explain what is Bootstrap Grid System?**

For creating page layout through a series of rows and columns that house your content Bootstrap Grid Sytem is used.

**7) What are offset columns in Bootstrap?**

For more specialized layouts offsets are a useful feature. For more spacing they can be used by pushing column over.

For example, .col-xs=\* classes do not support offset but they are easily replicated using an empty cell

**8) What is column ordering in Bootstrap?**

Column ordering is one of the feature available in bootstrap and you can easily write columns in an order and show them in another one. With **.col-md-push-\*** and **.col-md-pull-\***

the order of the column can be easily changed.

**9) What function you can use to wrap a page content?**

To wrap a page content you can use **.container**and using that you can also center the content.

**10) Explain what pagination in bootstrap is and how they are classified?**

Pagination is the handling of an unordered list by bootstrap. To handle pagination bootstrap provides following classes

* .pagination: To get pagination on your page you have to add this class
* .disabled, .active: Customize links by .disabled for unclickable links and .active to indicate the current page
* .pagination-Ig, .pagination-sm: Use these classes to get different size item

**11) What is the use of Jumbotron in Bootstrap?**

In bootstrap, Jumbotron is generally used for content that you want to highlight like some slogan or marketing headline etc. in other words it is used to enlarge the size of the headings and to add a margin for landing page content

To use the Jumbotron in Bootstrap

* Create a container <div> with the class of .jumbotron

**12) What is the difference between Bootstrap and Foundation?**

|  |  |
| --- | --- |
| Bootstrap | Foundation |
| – Bootstrap offers unlimited number of UI elements | – In Foundation UI element options are very limited in numbers |
| – Bootstraps uses pixels | – Foundation use REMs |
| – Bootstrap encourages to design for both desktop and mobile. | – Foundation encourages to design mobile first |
| – Bootstrap support LESS as its preprocessor | – Foundation support Sass and Compass as its preprocessor |

**13) In Bootstrap what are the two ways you can display the code?**

In bootstrap you can display code in two ways

* <code> tag : If you are going to display code inline, you should use <code> tag
* <pre> tag: If you want to display the code as a standalone block element or it has multiple lines then you should use <pre> tag

**14) Explain what are the steps for creating basic or vertical forms?**

The steps for creating basic or vertical forms are

* Add a role ***form*** to the parent <form> element
* Wrap labels and controls in a <div> with class ***.form-group***. To achieve optimum spacing this is needed
* Add a class of ***.form-control*** to all texturl <input> , <textarea> , and <select> elements

**15) Explain what is Modal plugin used for in Bootstrap?**

A modal is a child window that is layered over its parent window. Using a custom Jquery Plugin, Bootstrap Modal are created. To enrich user experience and to add functionality to users, modal windows are created with the help of Modal plugin.

**16) Explain what is Bootstrap Container?**

Bootstrap container is a class which is useful and creates a centred area within the page where our site content can be put within. The advantage of the bootstrap .container is that it is responsive and will place all our other HTML code.

**17) Explain what is Bootstrap collapsing elements?**

Bootstrap collapsing elements enables you to collapse any particular element without writing any JavaScript code or the accordion markup. In Bootstrap to apply collapsing elements you have to add data-toggle= “collapse” to the controller element along with a data-target or href to automatically assign control of a collapsible element. Likewise, you can use .collapse (options), .collapse (‘show’) or .collapse (‘hide’)

**18) Explain what is list group in Bootstrap and what is the use of it?**

List groups are components to display both simple and complex element with custom content

For example, a simple list group is created using class **.list-group** to address the list, and class .list-group-item to address individual item.

**19) How you can add badge to list group in Bootstrap?**

To add badge to list group in Bootstrap you have to simply add **<span class = “badge”>** within the <li> element.

**20) Explain what media object in Bootstrap is and what are their types?**

Media objects in Bootstrap enables to put media object like image, video or audio to the left or right of the content blocks. Media element can be created using the class **.media** and the source is specified in using the class **.media-object.**Media-objects are of two types,

They are of two types

* .media
* .media-list

**21) Explain what is Bootstrap well?**

Bootstrap well is a container <div> that makes the content to appear sunken or an inset effect on the page. In order to create a well, wrap the content that you would like to appear in the well with a <div> containing the class of .well.

**22) Explain how you can create Nav elements in Bootstrap?**

Bootstrap offers various options for styling navigation elements all of them use the same markup and base class .nav.

To create Tabular Navigation or Tabs

* Start with a basic unordered list with the base class of **.nav**
* Then add class **.nav-tabs**

**23) Explain what is the use of Bootstrap Carousel plugin?**

The Carousel plugin is used to add a slider to your site. It is useful in condition where you want to display huge amount of contents within a small space on the web pages. Some of the standard carousel includes

* .carousel (options)
* .carousel (‘cycle’)
* .carousel (‘pause’)
* .carousel (‘number’)
* .carousel (‘prev’)
* .carousel (‘next’)

**1. Explain Bootstrap?**

Bootstrap is a platform for web development that is based on a front-end framework and creates exceptional responsive designs. It is fast, easy and has multiple templates designed using HTML, and CSS. These templates are used for forms, tables, buttons, typography, models, tables, navigation, carousels and images. Bootstrap also has Javascript plugins, which are optional. Bootstrap is preferred for developing mobile web applications.

**2. Explain why you prefer Bootstrap for website development?**

Bootstrap has features that are way better than other web development platforms. It provides an extensive browser support for almost every known browser such as Opera, Chrome, Firefox, Safari etc. With adequate knowledge of CSS and HTML, web development becomes easy on Bootstrap. Also, it supports mobile applications with the help of responsive design. It can adjust CSS as per the device, screen size etc. Instead of creating multiple files, it creates only a single file, which reduces any extra effort by the developer.

**3. What are the key components of Bootstrap?**

In total, there are five key components of Bootstrap i.e. **CSS** (multiple CSS files), **Scaffolding (**essential for the basic system that consist of Grid system, background and link styles), **Layout Components:** (shares a list of all layouts), **JavaScript Plugins** (includes jQuery and JavaScript plugins) and **Customization**(Allows customization of all components for a desired framework)

**4. How many types of layout are available in Bootstrap?**

There are two major layouts for Bootstrap i.e. Fluid Layout and Fixed Layout. Fluid layout is necessary for creating an app that is 100 % wider and covers all the screen width. Fixed Layout is used only for a standard screen (940px). Both layouts can be used for creating a responsive design.

**5. Why do we use Jumbotron in Bootstrap?**

Jumbotron has a very basic function in bootstrap i.e. highlighting a content. It could either be a slogan or probably a headline. It increases the heading size and gives a margin for the content of the landing page. In order to implement Jumbotron in a Bootstrap use:

Create a container <div> with the class of .jumbotron  
**For instance:** use <div class="container"> if you wish that Jumbotron should not reach the screen’s edge.

Jumbotron can have any valid HTML along with other functions and classes.

**6. Explain the two codes that are used for code display in Bootstrap?**

There are two simple ways to display a code in Bootstrap:

* <code> tag: In case you wish to display an inline code, simply add ‘<code> tag’
* <pre> tag: In case you have a code with several lines or even a block element, you can display it using ‘<pre> tag’

**7. what do you mean by Bootstrap collapsing elements?**

Bootstrap collapsing elements is a way in which you can collapse any specific element without developing any JavaScript code or even an accordion markup. Now, in order to implement this element, you need to add data-toggle=”collapse”  in controller element. To automatically assign the control of the collapsible element, you can use data-target or href along with data-toggle=”collapse”. There are other options also available such as .collapse (options), .collapse (‘hide’), .collapse (‘show’)

**8. Why do we use Bootstrap Carousel plugin?**

The Carousel plugin is responsible for adding a slider on the site. The developers usually use it when the web space is limited but the content is huge. Some of the common carousels includes:

* .carousel (options)
* .carousel (‘pause’)
* .carousel (‘prev’)
* .carousel (‘next’)
* .carousel (‘number’)
* .carousel (‘cycle’)

**9. What is the role of the media objects in Bootstrap and how many types are available?**

In Bootstrap, it is possible to add images, audio, and video using media object. It can either be placed on the left side or the right side of the content. If the developer wishes to create a media element then they can use **.media** a class where the source will be specified as a **.media-object class. There are usually two types of**Media-objects i.e. .media and .media-list.

**10. What is the role of pagination in bootstrap and what are their classifications?**

When an unordered list has to be handled by bootstrap, it uses pagination. To handle the list, different classes are available such as ‘.pagination’, ‘.disabled’, ‘.active’, ‘.pagination-lg’. The function ‘.pagination’ to paginate the webpage. The function ‘.disabled’ is used to make the links unclickable. The function ‘.active’ is used for indicating the current webpage. The function ‘.pagination-lg’ and ‘.pagination-sm’ can be used for getting different size items.

**11. How can you differentiate between Bootstrap and Foundation?**

Bootstrap supports LESS preprocessor and allows the designing and development for both mobile and desktop. It uses only pixels and has a number of UI elements. Foundation, on the other hand, supports Sass preprocessor and allows mostly mobile designing. It uses REMs and has only a few options when it comes to UI elements.

**12. What is the step-wise procedure for creating basic or vertical forms?**

There are a few steps, which can be followed for creating vertical or basic.

* Firstly, a role form can be added to the parent <form> element.
* Secondly, add appropriate spacing by wrapping labels and control in <div> and using the function ‘class .form-group’.
* Lastly, apply the function ‘class .form-control’ to different elements such as text url <input> , <textarea> and <select>

**13. what do you mean by Bootstrap well?**

Bootstrap well is nothing but a container that makes the content appear sunken. Sometimes it may also give an inset effect on the webpage. Thus, a developer can create a well and also wrap the content in the well with the help of <div> and class .well. The content would appear as per your wish.

**14. What is the procedure to create Nav elements in Bootstrap?**

There are several styling navigation elements available on bootstrap and every style uses the same function i.e. ‘class .nav’. In order to create tabs or a tabular navigation, you can begin with a simple or rather basic unordered list using the function class .nav. To add the tabs the function ‘class .nav-tabs can be used.

**15. What is a list group in Bootstrap and where does it finds its application?**

When there is a need for displaying a simple element as well as a complex element along with a custom content, a list group can be used.  You can use the function ‘class .list-group’ to create a simple list that helps in addressing the entire list and ‘class.list-group-item’ to refer only to an individual item in that list.

**16. What do you mean by Modal plugin that is used for in Bootstrap?**

When there is a need to cover a parent window, a child window can be created using a modal. Developers generally use a custom Jquery Plugin to create a Modal on Bootstrap. Modal windows are best when it comes to improving the experience of a user and also adding more functionality. The modal windows can be simply created with the help of a modal plugin.

**17. what is a Bootstrap Container?**

Bootstrap container is again a class that is used for creating an area in the center of the webpage to place the content. There are row elements which act as containers for columns thereby making a grid system. Rows are used for creating horizontal column groups. Using the function .container,  you can get a responsive design, which will include all the other HTML code.

**18. What do you mean by column ordering in Bootstrap?**

Column ordering is one of the most interesting features that can be found in bootstrap. By using appropriate functions, the columns can be written easily and also in a defined order. You can also show them in another column. In order to change or alter the order of the column easily, the functions .col-md-push-\* and .col-md-pull-\* can be used.

**19. What is the most appropriate function to wrap the page content?**

In order to wrap the content of a page, the function ‘.container’ can be used. Using the same function, the text can be centered.

Thus, these interview questions cover mostly all the important elements and functions of bootstrap, which an interviewer can ask you. Prepare these sets of questions so that you can stand out of the competition.

Question 1: What is Bootstrap?

**Answer**  
Bootstrap is the most popular and powerful HTML, CSS, and JavaScript front-end framework for faster and easier use, responsive layout, and mobile-first web development. It includes design templates for UI components like Forms, Buttons, Tables, Navigation, Dropdowns, Alerts, Tabs, Accordion, Carousel and many others with optional JavaScript plugins.  
  
**Features of Bootstrap**

* **Easy to use**  
  Bootstrap is very easy to use. Anyone with the basic knowledge of HTML and CSS can start development with Bootstrap.
* **Open Source**  
  Bootstrap is completely free to download and use.
* **Browser compatibility**Bootstrap is compatible with all modern browsers like Google Chrome, Mozilla Firefox, Internet Explorer, Safari, and Opera.
* **Responsive features**  
  By using Bootstrap we can easily create responsive designs for web pages. Bootstrap's responsive CSS makes our web pages appear more appropriately on different devices as it adjusts to phones, tablets, and desktops.

For more details visit the following link,

* [Bootstrap For Beginners - Part One (Introduction And Implementation)](http://www.c-sharpcorner.com/article/bootstrap-for-beginners-part-one-introduction-and-impleme/)

Question 2: Why choose Bootstrap for building websites?

**Answer**  
Bootstrap is used for building websites for the following reasons,

* Mobile First Approach: Since Bootstrap 3, the framework consists of Mobile first styles throughout the entire library instead of in separate files.
* Browser Support: It is supported by all popular browsers.
* Easy to Get Started: With just the knowledge of HTML and CSS anyone can get started with Bootstrap.
* Responsive Design: Bootstrap's responsive CSS adjusts to Desktops, Tablets and Mobiles.
* Provides a clean and uniform solution for building an interface for developers.
* It contains beautiful and functional built-in components that are easy to customize. It also provides web based customization.

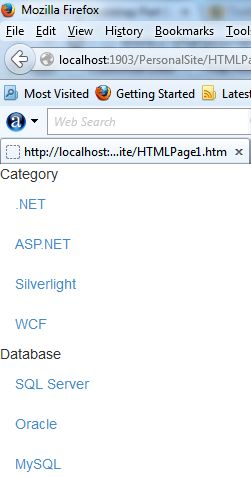
For more details visit the following link,

* [Getting Started With Bootstrap](http://www.c-sharpcorner.com/UploadFile/b69ed7/getting-started-with-bootstrap/)

Question 3: Explain the use of Bootstrap CSS class="navnav-list".

Answer  
  
You need to add the nav-list class in addition to the nav class that will make it look like a list. And adding class *nav-header* to any element of the nav class will make it look like a heading to the section of links. The HTML File looks as in the following,

1. <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
2. <html>
4. <head>
5. <title></title>
6. <script src="Scripts/jquery-1.4.1.min.js" type="text/javascript"></script>
7. <link href="Bootstrap/css/bootstrap.css" rel="stylesheet" type="text/css" />
8. <script src="Bootstrap/js/bootstrap.min.js" type="text/javascript"></script>
9. </head>
11. <body>
12. <ul class="navnav-list">
13. <li class="nav-header">Category</li>
14. <li><a href="#">.NET</a></li>
15. <li><a href="#">ASP.NET</a></li>
16. <li><a href="#">Silverlight</a></li>
17. <li><a href="#">WCF</a></li>
18. <li class="nav-header">Database</li>
19. <li><a href="#">SQL Server</a></li>
20. <li><a href="#">Oracle</a></li>
21. <li><a href="#">MySQL</a></li>
22. </ul>
23. </body>
25. </html>

Now see how the list is rendering.  
  
  
  
For more details visit the following link -

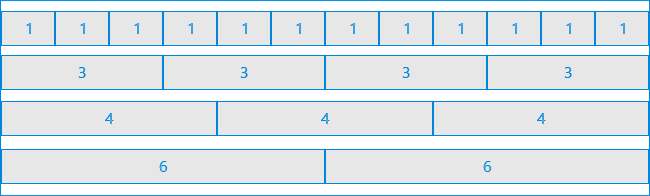
* [An Introduction to Bootstrap](http://www.c-sharpcorner.com/UploadFile/rohatash/an-introduction-to-bootstrap/)

Question 4: Explain Bootstrap Grid System.

**Answer**  
Bootstrap grid system provides the fastest and easiest way to create layouts of web pages. It introduces the responsive mobile first fluid grid system that appropriately scales up to 12 columns as the device or view port size increases. It includes predefined grid classes for making grid layouts for various types of devices, like phones, tablets, desktop, and so on, as well as powerful mixings for generating more semantic layouts.  
  
**The Grid Sizes**

It has four tiers of classes: xs, sm, md and lg. You can use any combination of these classes to create more dynamic and flexible layouts. The following is the breakdown of the various sizes of these classes.

|  |  |  |
| --- | --- | --- |
| Classes | Devices | Size |
| .col-xs-\* | Extra Small Phones | Less than 768px |
| .col-sm-\* | Small Devices Tablets | 768px and Up |
| .col-md-\* | Medium Devices Desktops | 992px and Up |
| .col-lg-\* | Large Devices Large Desktops | 1200px and Up |

Structure of Bootstrap Grid Layout,  
  
  
  
**Working of Bootstrap Grid System**  
  
Grid systems are used for creating page layouts through a series of rows and columns . Here's how the Bootstrap grid system works,

* Rows must be placed within a .container and .container-fluid class for proper alignment and padding.
* Use rows to create horizontal groups of columns.
* Content should be placed within columns and only columns may be immediate children of rows.
* Predefined grid classes like .row and .col-xs-4 are available for quickly making grid layouts. LESS mixings can also be used for more semantic layouts.
* Grid columns are created by specifying the number of twelve available columns you wish to span. For example, three equal columns would use three .col-xs-4.

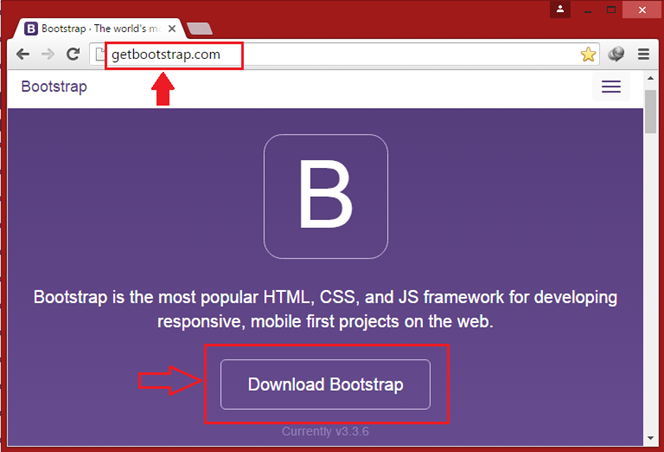
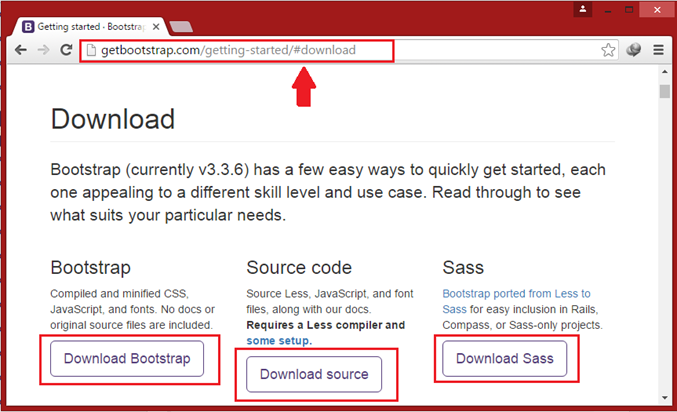
For more details visit the following link,

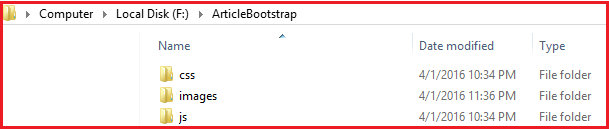
* [Getting Started With Bootstrap](http://www.c-sharpcorner.com/UploadFile/b69ed7/getting-started-with-bootstrap/)

Question 5: How can you get Bootstrap?

**Answer**There are two ways to get Bootstrap on our own web site.

* [Download](http://getbootstrap.com/)Bootstrap.
* Include Bootstrap from a CDN.

**Download Bootstrap**  
We can download the latest version of Bootstrap. When we click on this link we will see screen like this.  
  
  
  
Now if we click on the Download Bootstrap button, another page will open like this:  
  
  
  
There are two versions available for download, compiled Bootstrap and Bootstrap source files.

* **Download Bootstrap**  
  By clicking on this option, we can download the precompiled and minified versions of Bootstrap CSS, JavaScript, and fonts.  
    
  After downloading, we will unzip the folder and we will find the following subfolders inside that.  
    
  
* **Download Source**  
  By clicking on this option, we can get the latest Bootstrap LESS and JavaScript source code.  
    
  For better understanding we'll focus on the compiled Bootstrap files. As the files are compiled and minified we don't have to bother every time including separate files for individual functionality. It will also increase the performance of our website.

**Bootstrap CDN**  
If we don't want to download, we can include it from a CDN (Content Delivery Network). MaxCDN provides CDN support for Bootstrap's CSS and JavaScript and also includes jQuery. We can use Bootstrap CDN links,

1. <!-- Latest compiled and minified CSS -->
2. <link rel="stylesheet" href="http://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css">
3. <!-- jQuery library -->
4. <script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.0/jquery.min.js"></script>
5. <!-- Latest compiled JavaScript -->
6. <script src="http://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/js/bootstrap.min.js"></script>

For more details visit the following link,

* [Bootstrap For Beginners - Part One (Introduction And Implementation)](http://www.c-sharpcorner.com/article/bootstrap-for-beginners-part-one-introduction-and-impleme/)

Question 6: What do you understand about Responsive websites?

Answer  
  
The first understanding of Responsive websites that comes to a beginner's mind is that responsive means a website that responds to a user. But actually, responsiveness here means that a website is designed in such a way that it can be used on any platform whether it is a large desktop, laptop, tablet or mobile.  
  
So a responsive website actually is one that responds to the changing width and height of a device or screen.  
  
Benefits of a responsive website

* We all know that in today's world, all of the business work can be done using a small device like a tablet or a mobile. So it makes it easier for the user to access the website using a device, thus, increasing the user base and traffic on a website.
* When the user base increases, it also adds to the increase in the business of the website and the product sale will also increase.
* Makes it easy for an organization to analyze the user base and its productivity.
* Increase in the visibility in search engines. This is the major part of the success of an organization.
* Save time and cost on mobile development.

For more details visit the following link -

* [Basics of Bootstrap](http://www.c-sharpcorner.com/UploadFile/da55bf/basics-of-bootstrap675/)

Question 7: How can you create a web page using Bootstrap?

**Answer**Example:  
  
**Using Downloaded Bootstrap**  
  
Now we will create an HTML file that displays a "Hello C# Corner" message in our web browser.  
  
**Step 1**  
Creating a Basic HTML file.  
  
First we will open any code editor and create a new html file, and we will write the following code and then save the file name as "hello.html" on the particuler location. We will include HTML5 doctype at the beginning of the page, with the lang attribute and the correct character set.

1. <!DOCTYPE html>
2. <html lang="en">
4. <head>
5. <meta charset="utf-8">
6. <title>Bootstrap Part1</title>
7. </head>
9. <body>
10. <h1>Hello C# Corner</h1>
11. </body>
13. </html>

**Step 2**  
  
Bootstrap 3 is mobile-first.  
  
Using Bootstrap 3 we will design our HTML page to be responsive to mobile devices. To ensure proper rendering and enable touch zooming, we will add the following <meta> tag inside the <head> element.

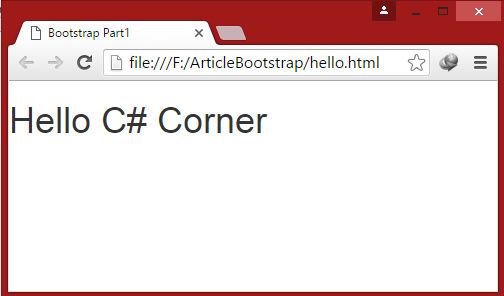
1. <meta name="viewport" content="width=device-width,initial-scale=1">

The width=device-width is used to set the width of the page to follow the screen-width of the device. The initial-scale=1 is used to set the initial zoom level when the page is first loaded by the browser.  
  
Now the code will look like this.

1. <!DOCTYPE html>
2. <html lang="en">
4. <head>
5. <meta charset="utf-8">
6. <title>Bootstrap Part1</title>
7. <meta name="viewport" content="width=device-width,initial-scale=1">
8. </head>
10. <body>
11. <h1>Hello C# Corner</h1>
12. </body>
14. </html>

**Step 3**Making an HTML File a Bootstrap Template.  
  
For making this file a Bootstrap Template, after Downloading Bootstrap by the above procedure we will include Bootstrap CSS and JS files. We should include JS files at the bottom of the HTML page before closing <body> tag. by the following code.

1. <!DOCTYPE html>
2. <html lang="en">
4. <head>
5. <meta charset="utf-8">
6. <title>Bootstrap Part1</title>
7. <meta name="viewport" content="width=device-width,initial-scale=1">
8. <link rel="stylesheet" type="text/css" href="css/bootstrap.min.css">
9. </head>
11. <body>
12. <h1>Hello C# Corner</h1>
13. <script src="js/jquery-2.1.4.min.js"></script>
14. <script src="js/bootstrap.min.js"></script>
15. </body>
17. </html>

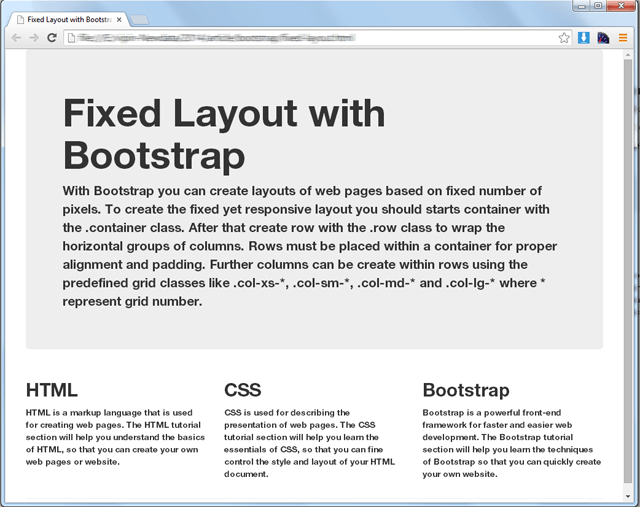
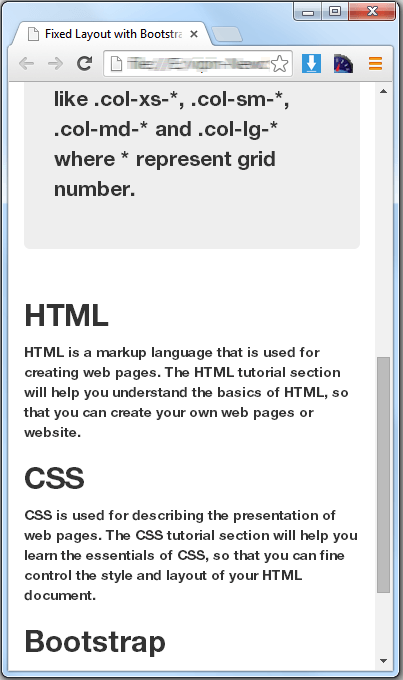
Now we will open the file in a browser by double clicking on it and see the output.  
  
**Output**  
  
  
  
For more details visit the following link -

* [Bootstrap For Beginners - Part One (Introduction And Implementation)](http://www.c-sharpcorner.com/article/bootstrap-for-beginners-part-one-introduction-and-impleme/)

Question 8: How can you create Fixed Layout with Bootstrap?

**Answer**With Bootstrap you can create layouts of web pages based on a fixed number of pixels. To create the fixed yet responsive layout you should start the container with the .container class. Then create the row with the .row class to wrap the horizontal groups of columns. Rows must be placed within a container for proper alignment and padding. Further columns can be created within rows using the predefined grid classes like*.col-xs-\*, .col-sm-\*, .col-md-\** and *.col-lg-\**where \* represents a grid number.  
  
The following code creates a fixed width responsive layout that is 970px wide on a medium device like desktop and laptop with a screen width ≥992px and 1170px wide on large devices like large desktops with a screen width ≥1200px. However the layout width will be automatically calculated for devices that have a screen width <768px like tablets and cell phones.  
  
**HTML Code**

1. <!DOCTYPE html >
2. <html>
4. <head>
5. <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
6. <title>Fixed Layout with Bootstrap</title>
7. <!-- compiled and minified CSS -->
8. <link rel="stylesheet" href="css/bootstrap.min.css">
9. <!-- Optional theme -->
10. <link rel="stylesheet" href="css/bootstrap-theme.min.css">
11. </head>
13. <body>
14. <div class="container">
15. <div class="jumbotron">
16. <h1>Fixed Layout with Bootstrap </h1>
17. <p>With Bootstrap you can create layouts of web pages based on fixed number of pixels. To create the fixed yet responsive layout you should starts container with the .container class. After that create row with the .row class to wrap the horizontal groups of columns. Rows must be placed within a container for proper alignment and padding. Further columns can be create within rows using the predefined grid classes like .col-xs-\*, .col-sm-\*, .col-md-\* and .col-lg-\* where \* represent grid number.</p>
18. </div>
19. <div class="row">
20. <div class="col-sm-4">
21. <h2>HTML</h2>
22. <p>HTML is a markup language that is used for creating web pages. The HTML tutorial section will help you understand the basics of HTML, so that you can create your own web pages or website.</p>
23. </div>
24. <div class="col-sm-4">
25. <h2>CSS</h2>
26. <p>CSS is used for describing the presentation of web pages. The CSS tutorial section will help you learn the essentials of CSS, so that you can fine control the style and layout of your HTML document.</p>
27. </div>
28. <div class="col-sm-4">
29. <h2>Bootstrap</h2>
30. <p>Bootstrap is a powerful front-end framework for faster and easier web development. The Bootstrap tutorial section will help you learn the techniques of Bootstrap so that you can quickly create your own website.</p>
31. </div>
32. </div>
33. <hr>
34. </div>
36. <!-- compiled and minified JavaScript -->
37. <script type="text/javascript" src="js/bootstrap.min.js"></script>
38. </body>
40. </html>

**Result**On a tablet, desktop and larger device,  
  
  
  
**Mobile Device**  
  
  
  
For more details visit the following link -

* [Getting Started With Bootstrap](http://www.c-sharpcorner.com/UploadFile/b69ed7/getting-started-with-bootstrap/)

Question 9: How can we create a dropdown menu in Bootstrap?

Answer  
  
To make the menu more attractive with a DropDown list using Bootstrap open up the bootstrap.css file and check out the following Bootstrap CSS class.

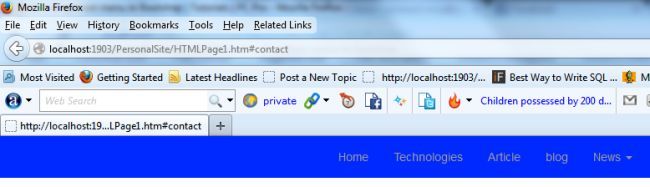
1. Caret Class
2. DropDown-Menu Class

**1. Using bootstrap CSS class="Caret"**Now first we use the caret class with li tag. The Caret class shows a caret sign. Now open the bootstrap.css file and find the .navbar-inverse class. It looks like this,

1. .caret
2. {
3. **display**: inline - block;
4. **width**: 0;
5. **height**: 0;
6. margin - left: 2 px;
7. vertical - align: middle;
8. border - top: 4 px solid;
9. border - right: 4 px solid transparent;
10. border - left: 4 px solid transparent;
11. }

The HTML file looks as in the following,

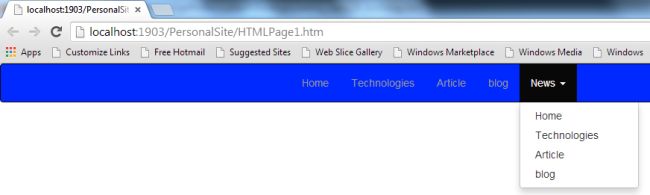
1. <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
2. <html>
4. <head>
5. <title></title>
6. <script src="Scripts/jquery-1.4.1.min.js" type="text/javascript"></script>
7. <link href="Bootstrap/css/bootstrap.css" rel="stylesheet" type="text/css" />
8. <script src="Bootstrap/js/bootstrap.min.js" type="text/javascript"></script>
9. </head>
11. <body>
12. <div class="navbar-inverse">
13. <div class="Container">
14. <ul class="navnavbar-nav">
15. <li><a href="#">Home</a></li>
16. <li><a href="#about">Technologies</a></li>
17. <li><a href="#contact">Article</a></li>
18. <li><a href="#contact">blog</a></li>
19. <li><a href="#contact">News <b class ="caret"></b></a></li>
20. </ul>
21. </div>
22. </div>
23. </body>
25. </html>

The HTML will render without Bootstrap as in the following:  
 **2. Using bootstrap CSS class="DropDown-Menu"**  
You now need to add the DropDown-Menu class. Now open the bootstrap.css file and find the DropDown-Menu class. It looks like this,

1. .dropdown - menu
2. {
3. **position**: absolute;
4. top: 100 % ;
5. left: 0;
6. z - index: 1000;
7. **display**: none;
8. **float**: left;
9. min - **width**: 160 px;
10. **padding**: 5 px 0;
11. **margin**: 2 px 0 0;
12. font - **size**: 14 px;
13. list - style: none;
14. background - **color**: #ffffff;
15. **border**: 1 px solid# cccccc;
16. **border**: 1 px solid rgba(0, 0, 0, 0.15);
17. border - radius: 4 px; -
18. webkit - box - shadow: 0 6 px 12 px rgba(0, 0, 0, 0.175);
19. box - shadow: 0 6 px 12 px rgba(0, 0, 0, 0.175);
20. background - **clip**: padding - box;
21. }

The HTML file looks as in the following,

1. <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
2. <html>
4. <head>
5. <title></title>
6. <script src="Scripts/jquery-1.4.1.min.js" type="text/javascript"></script>
7. <link href="Bootstrap/css/bootstrap.css" rel="stylesheet" type="text/css" />
8. <script src="Bootstrap/js/bootstrap.min.js" type="text/javascript"></script>
9. <script src="Bootstrap/js/bootstrap.min.js" type="text/javascript"></script>
10. <script src="Bootstrap/js/bootstrap.js" type="text/javascript"></script>
11. </head>
13. <body>
14. <div class="navbarnavbar-inverse">
15. <div class="Container">
16. <ul class="navnavbar-nav">
17. <li><a href="#">Home</a></li>
18. <li><a href="#about">Technologies</a></li>
19. <li><a href="#contact">Article</a></li>
20. <li><a href="#contact">blog</a></li>
21. <li class="dropdown"><a href="#" class="dropdown-toggle">News <b class="caret"></b></a>
22. <ul class="dropdown-menu">
23. <li><a href="#">Home</a></li>
24. <li><a href="#">Technologies</a></li>
25. <li><a href="#">Article</a></li>
26. <li><a href="#">blog</a></li>
27. </ul>
28. </li>
29. </ul>
30. </div>
31. </div>
32. <script src="Bootstrap/js/jquery.js" type="text/javascript"></script>
33. <script src="Bootstrap/js/dropdown.js" type="text/javascript"></script>
34. <script type="text/javascript">
35. $(document).ready(function() {
36. $('.dropdown-toggle').dropdown();
37. });
38. </script>
39. </body>
41. </html>

The HTML will be rendered with Bootstrap as in the following,  
  
  
  
For more details visit the following link -

* [Creating DropDown Menu in Bootstrap](http://www.c-sharpcorner.com/UploadFile/rohatash/creating-dropdown-menu-in-bootstrap/)

Question 10: How can you create a menu using navbar-inverse in Bootstrap?

Answer   
  
First create a list using ul and li tags. The HTML file looks as in the following,

1. <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
2. <html>
4. <head>
5. <title></title>
6. <script src="Scripts/jquery-1.4.1.min.js" type="text/javascript"></script>
7. <link href="Bootstrap/css/bootstrap.css" rel="stylesheet" type="text/css" />
8. <script src="Bootstrap/js/bootstrap.min.js" type="text/javascript"></script>
9. </head>
11. <body>
12. <ul>
13. <li><a href="#">Home</a></li>
14. <li><a href="#about">Technologies</a></li>
15. <li><a href="#contact">Article</a></li>
16. <li><a href="#contact">blog</a></li>
17. <li><a href="#contact">News</a></li>
18. </ul>
19. </body>
21. </html>

HTML will render without Bootstrap as in the following,  
  
  
  
**Using bootstrap CSS class="navbar-inverse"**You now need to add the navbar-inverse class.  
  
Now open the bootstrap.css file and find the .navbar-inverse class. It looks like this,

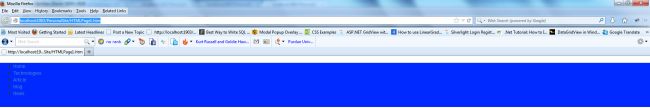
1. .navbar - inverse
2. {
3. background - **color**: #222222;
4. **border-color**: # 080808;
5. }

Now we want to change the background-color with "*background-color: #0029ff*".

1. .navbar-inverse {
2. **background-color**: #0029ff;
3. **border-color**: #080808;
4. }

The HTML file looks  as in the following,

1. <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
2. <html>
4. <head>
5. <title></title>
6. <script src="Scripts/jquery-1.4.1.min.js" type="text/javascript"></script>
7. <link href="Bootstrap/css/bootstrap.css" rel="stylesheet" type="text/css" />
8. <script src="Bootstrap/js/bootstrap.min.js" type="text/javascript"></script>
9. </head>
11. <body>
12. <div class="navbar-inverse">
13. <ul>
14. <li><a href="#">Home</a></li>
15. <li><a href="#about">Technologies</a></li>
16. <li><a href="#contact">Article</a></li>
17. <li><a href="#contact">blog</a></li>
18. <li><a href="#contact">News</a></li>
19. </ul>
20. </div>
21. </body>
23. </html>

The HTML will be rendered with Bootstrap as in the following:  
  
  
  
For more details visit the following link -

* [Creating Menu in Bootstrap](http://www.c-sharpcorner.com/UploadFile/rohatash/creating-menu-in-bootstrap/)

Question 11: What are the advantages of Bootstrap?

**Answer**There are many advantages of using Bootstrap,

* Bootstrap Css library takes care of our UI to make  interactive mobile + desktop enabled websites, developers can even create a fully furnished jaw dropping website easily.
* The speed of development can be increased with the help of Bootstrap.
* With the increase in mobile first, website designers need to work a lot in order to make the UI adapt to all kinds of devices, whereas with the help of bootstrap this overhead has been reduced to quite an extent.
* Responsive Grid: As you go in depth with Bootstrap and read about Grid you will see that 12 columns are grids and are responsive and you can make them self-adjusting according to the device.
* There is a huge list of Components --  Dropdown menu, badges etc. A few have been discussed in this article.
* Easy to read documentation: I personally believe if you want to learn Bootstrap from scratch <http://getbootstrap.com/> documentation is easy and great to start with. I personally appreciate their efforts we can go back to the site whenever we something is  needed in our View.
* Themes: Bootstrap provides free colorful themes that can be used in our Web Application.
* Support

For more details visit the following link,

* [Guide To The Basic Pillars Of Bootstrap](http://www.c-sharpcorner.com/article/guide-to-the-basic-pillars-of-bootstrap/)

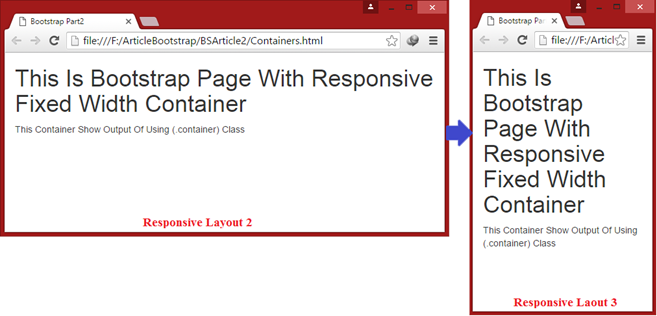
Question 12: What are Containers in Bootstrap?

Answer  
  
In Bootstrap we can contain elements to wrap site contents.  
  
There are two Bootstrap Container classes,

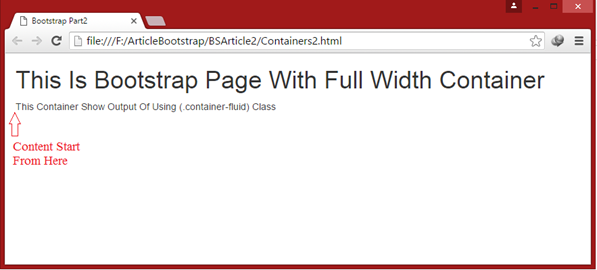
1. The .container class used to provide a responsive fixed width container.
2. The .container-fluid class used to provide a full width container.

We have to note that we cannot put a container inside another container since it is not nestable.  
  
**Example 1  
  
Using .container class (responsive fixed width container)**  
  
In this example we will create a simple Bootstrap page. Using "*.container*" class we will create a fixed width container that is responsive for different devices. In this we will write some text. We will have some space on the left and right side of the page by writing the following code.

1. <!DOCTYPE html>
3. <html lang="en">
5. <head>
6. <meta charset="utf-8">
7. <title>Bootstrap Part2</title>
8. <meta name="viewport" content="width=device-width,initial-scale=1">
9. <link rel="stylesheet" type="text/css" href="css/bootstrap.min.css">
10. </head>
12. <body>
13. <div class="container">
14. <h1>This Is Bootstrap Page With Responsive Fixed Width Container</h1>
15. <p>This Container Show Output Of Using (.container) Class</p>
16. </div>
17. <script src="js/jquery-2.1.4.min.js"></script>
18. <script src="js/bootstrap.min.js"></script>
19. </body>
21. </html>

**Output**  
  
  
  
  
**Example 2**  
  
**Using .container-fluid class (full width container)**  
  
In this example we will create a simple Bootstrap page like Example 1. Using "*.container-fluid*"class we will create a full width container. In this we will write some text. Now we will create a page by writing the following code.

1. <!DOCTYPE html>
3. <html lang="en">
5. <head>
6. <meta charset="utf-8">
7. <title>Bootstrap Part2</title>
8. <meta name="viewport" content="width=device-width,initial-scale=1">
9. <link rel="stylesheet" type="text/css" href="css/bootstrap.min.css">
10. </head>
12. <body>
13. <div class="container-fluid">
14. <h1>This Is Bootstrap Page With Full Width Container</h1>
15. <p>This Container Show Output Of Using (.container-fluid) Class</p>
16. </div>
17. <script src="js/jquery-2.1.4.min.js"></script>
18. <script src="js/bootstrap.min.js"></script>
19. </body>
21. </html>

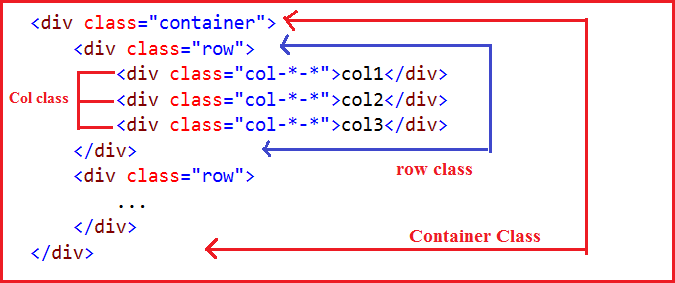
**Output**  
  
  
For more details visit the following link -

* [Bootstrap For Beginners - Part Two (Bootstrap Containers)](http://www.c-sharpcorner.com/article/bootstrap-for-beginners-part-two-bootstrap-containers/)

Question 13: Explain the basic structure of the Bootstrap grid.

**Answer**Now for creating rows and columns using 12 column responsive grid systems.

* First we need to create a container that acts as a wrapper for our rows and columns by using the .container class.
* Then we have to create rows inside the container by using (<div class="row">) ,
* Then we need to add the desired number of columns inside any row by using classes *.col-xs-\*, .col-sm-\*, .col-md-\** and *.col-lg-\** the number of columns should always add up to 12 for each row. In these columns we write our contents.

The following is a Basic Structure,  
  


1. <div class="container">
2. <div class="row">
3. <div class="col-\*-\*">col1</div>
4. <div class="col-\*-\*">col2</div>
5. <div class="col-\*-\*">col3</div>
6. </div>
7. <div class="row">
8. ...
9. </div>
10. </div>

Now we will create some examples for Grid Layouts by which we can easily make responsive website layouts.  
  
**Example 1  
  
Two Column Layouts**  
  
In this example we will create two column layouts for different devices. In mobile the column will automatically arrange horizontally according to screen size. We know that grid system works on 12 columns, so for creating two column layouts we keep the sum of the grid column numbers equal to 12 in each row so columns will be in one line. In this example we will add three rows and inside each row we will add two columns. Let's create the example.  
  
**Step 1**  
  
First we will create a Bootstrap Template, HTML page named "TwoColLayout.html" by using the following code.

1. <!DOCTYPE html>
3. <html lang="en">
5. <head>
6. <meta charset="utf-8">
7. <title>Bootstrap Part3</title>
8. <meta name="viewport" content="width=device-width,initial-scale=1">
9. <link rel="stylesheet" type="text/css" href="css/bootstrap.min.css">
10. </head>
12. <body>
13. <div class="container">
14. <h1>Two Column Layouts</h1>
15. </div>
16. <script src="js/jquery-2.1.4.min.js"></script>
17. <script src="js/bootstrap.min.js"></script>
18. </body>
20. </html>

**Step 2**  
Now we will add rows and columns for creating Two Column Layouts by the following code; in this we will give style "*background-color*" for each column so output shows clearly.

1. <!DOCTYPE html>
3. <html lang="en">
5. <head>
6. <meta charset="utf-8">
7. <title>Bootstrap Part3</title>
8. <meta name="viewport" content="width=device-width,initial-scale=1">
9. <link rel="stylesheet" type="text/css" href="css/bootstrap.min.css">
10. </head>
12. <body>
13. <div class="container">
14. <h1>Two Column Layouts</h1>
15. <!--First row with column number(4,8)-->
16. <div class="row">
17. <div class="col-sm-4" style="background-color:#8dc1aa">column1: col-sm-4</div>
18. <div class="col-sm-8" style="background-color:#808080">column2: col-sm-8</div>
19. </div>
20. <!--Second row with column number(6,6)-->
21. <div class="row">
22. <div class="col-sm-6" style="background-color:#cfa6e2">column3: col-sm-6</div>
23. <div class="col-sm-6" style="background-color:#faa76c">column4: col-sm-6</div>
24. </div>
25. <!--Third row with column number(3,9)-->
26. <div class="row">
27. <div class="col-sm-3" style="background-color:#c8fcfc">column5: col-sm-3</div>
28. <div class="col-sm-9" style="background-color: #79ad96">column6: col-sm-9</div>
29. </div>
30. </div>
31. <script src="js/jquery-2.1.4.min.js"></script>
32. <script src="js/bootstrap.min.js"></script>
33. </body>
35. </html>

**Output**  
See the output for different devices according to screen size.  
  
  
  
For more details visit the following link -

* [Bootstrap For Beginners - Part Three (Bootstrap Grids)](http://www.c-sharpcorner.com/article/bootstrap-for-beginners-part-threebootstrap-grids/)

Question 14: Explain Bootstrap buttons and its styles.

**Answer**Buttons play various roles in websites and applications and are used to submit, reset, show and hide on click of button, link button etc.  
  
By using Bootstrap we can easily create and customize the buttons.  
  
**Bootstrap Button Styles**  
In Bootstrap different classes are used for styling the buttons and styles can be applied to any element like *<a>*, *<button>* and *<input>*.  
  
Bootstrap provides seven styles of buttons,  
  
Button   
  
To apply the button styles, Bootstrap provides the following classes,

* .btn-default - Default gray color Standard button
* .btn-primary - To indicate primary action button
* .btn-info - Used for informational alert messages
* .btn-success - Indicates a successful or positive action
* .btn-danger - Indicates a dangerous or negative action.
* .btn-warning - Indicates caution should be taken with this action
* .btn-link - Button looks like a link while maintaining button behavior

**Example**Bootstrap Button Styles,  
  
In this example we will create buttons and by using above classes we will apply styles on buttons through the following code.

1. <!DOCTYPE html>
2. <html lang="en">
4. <head>
5. <meta charset="utf-8">
6. <title>Bootstrap Part4</title>
7. <meta name="viewport" content="width=device-width,initial-scale=1">
8. <link rel="stylesheet" type="text/css" href="css/bootstrap.min.css"> </head>
10. <body>
11. <div class="container">
12. <h1>Button Styles</h1>
13. <!--Button Style1:Default--><button type="button" class="btn btn-default">Default Button</button>
14. <!--Button Style2:Primary--><button type="button" class="btn btn-primary">Primary Button</button>
15. <!--Button Style3:Info--><button type="button" class="btn btn-info">Info Button</button>
16. <!--Button Style4:Success--><button type="button" class="btn btn-success">Success Button</button>
17. <!--Button Style5:Danger--><button type="button" class="btn btn-danger">Danger Button</button>
18. <!--Button Style6:Warning--><button type="button" class="btn btn-warning">Warning Button</button>
19. <!--Button Style7:Link--><button type="button" class="btn btn-link">Link Button</button> </div>
20. <script src="js/jquery-2.1.4.min.js"></script>
21. <script src="js/bootstrap.min.js"></script>
22. </body>
24. </html>

**Output**  
  
  
For more details visit the following link -

* [Bootstrap For Beginners - Part Four (Bootstrap Buttons)](http://www.c-sharpcorner.com/article/bootstrap-for-beginners-part-four-bootstrap-buttons/)

Question 15: Explain the Bootstrap basic table?

**Answer**The following table elements are used with Bootstrap,

* <table> - It is used for wrapping element for displaying data in a tabular format
* <thead> - It is used for container element for table header rows (<tr>) to label table columns.
* <tbody> - It is used for container element for table rows (<tr>) in the body of the table.
* <tr> - It is used for container element for a set of table cells (<td> or <th>) that appears on a single row.
* <th> - Special table cell for column (or row) labels. It must be used within a <thread>
* <td> - It is used for default table cell.
* <caption> - It is used for description or summary of what the table holds.

**Bootstrap Basic Table**  
We can create a basic Bootstrap table with basic styling that has a small cell padding and only horizontal dividers by adding Bootstrap class "*.table*" to the *<table>* element.  
  
**Example**Bootstrap Basic Table using .table class.  
  
In this example we will create a simple table with Bootstrap class and using html table elements we will create an employee table with Fields Sr. No., Employee Name, Email, City by writing the following code.

1. <!DOCTYPE html>
2. <html lang="en">
4. <head>
5. <meta charset="utf-8">
6. <title>Bootstrap Part5</title>
7. <meta name="viewport" content="width=device-width,initial-scale=1">
8. <link rel="stylesheet" type="text/css" href="css/bootstrap.min.css">
9. </head>
11. <body>
12. <div class="container">
13. <h1>Bootstrap Basic Table</h1>
14. <!--Bootstrap Basic Table using .table class-->
15. <table class="table">
16. <thead>
17. <tr>
18. <th>Sr.No.</th>
19. <th>Emolpyee Name</th>
20. <th>Email</th>
21. <th>City</th>
22. </tr>
23. </thead>
24. <tbody>
25. <tr>
26. <td>1</td>
27. <td>Shaili Dashora</td>
28. <td>abc@mail.com </td>
29. <td>Chittorgarh</td>
30. </tr>
31. <tr>
32. <td>2</td>
33. <td>Sourabh Somani</td>
34. <td>xyz@mail.com </td>
35. <td>Banglore</td>
36. </tr>
37. <tr>
38. <td>3</td>
39. <td>Shobhna Singvi</td>
40. <td>pqr@mail.com</td>
41. <td>Mumbai</td>
42. </tr>
43. </tbody>
44. </table>
45. </div>
46. <script src="js/jquery-2.1.4.min.js"></script>
47. <script src="js/bootstrap.min.js"></script>
48. </body>
50. </html>

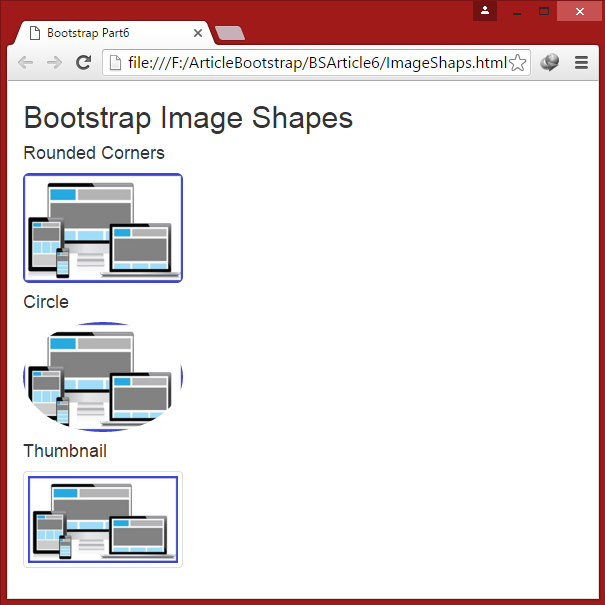
**Output**  
Basic Table Layout,  
  
  
  
For more details visit the following link -

* [Bootstrap For Beginners - Part Five (Bootstrap Tables)](http://www.c-sharpcorner.com/article/bootstrap-for-beginners-part-five-bootstrap-tables/)

Question 16: How can you give a style to images in Bootstrap?

**Answer**By using the Bootstrap classes we can easily style images like making images round, cornered, or circular, or we can give a thumbnail effect on images. We have the following classes for image shapes.

* Rounded Corners - The .img-rounded class used for adding rounded corners to an image.
* Circle - The .img-circle class used for shaping the image to a circle.
* Thumbnail - The .img-thumbnail class used for shaping the image to a thumbnail.

**Example - Bootstrap Image Shapes**  
  
In this example we will give different shapes of images using the above Bootstrap classes, we will put an image inside images folder where our html page exists. After that, we will give shapes on image: Rounded, Circle, or Thumbnail by using the following code.  
  
**Output**  
  
Bootstrap Image Shapes,  
  
  
  
For more details visit on to following link,

* [Bootstrap For Beginners - Part Six (Bootstrap Images)](http://www.c-sharpcorner.com/article/bootstrap-for-beginners-part-six-bootstrap-images/)

Question 18: What is panel in Bootstrap?

**Answer**A panel in bootstrap is a bordered box with some padding around its content. To create a basic panel, we use .panel class to the <div> element, and content inside the panel has a .panel-body class.  
  
**Example 1 - Creating Panel**  
In this example we will create a basic panel by using .panel class to the <div> element and we will write content in the panel by using .panel-body class to the <div> element. In this example we will use .panel-default class for styling of the panel by writing the following code.

1. <!DOCTYPE html>
3. <html lang="en">
5. <head>
6. <meta charset="utf-8">
7. <title>Bootstrap Part7</title>
8. <meta name="viewport" content="width=device-width,initial-scale=1">
9. <link rel="stylesheet" type="text/css" href="css/bootstrap.min.css">
10. </head>
12. <body>
13. <div class="container">
14. <h2>Bootstrap Panel</h2>
15. <div class="panel panel-default">
16. <div class="panel-body">Demo Panel</div>
17. </div>
18. </div>
19. <script src="js/jquery-2.1.4.min.js"></script>
20. <script src="js/bootstrap.min.js"></script>
21. </body>
23. </html>

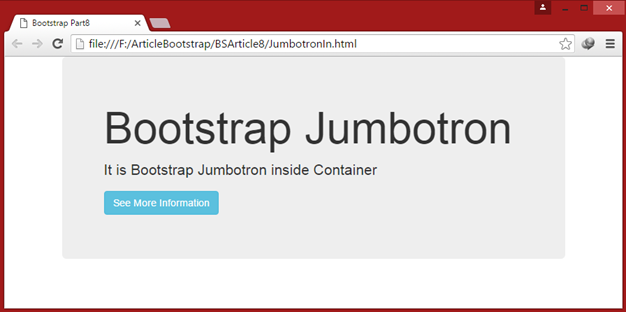
**Output**  
  
  
For more details visit the following link -

* [Bootstrap For Beginners - Part Seven (Bootstrap Panels)](http://www.c-sharpcorner.com/article/bootstrap-for-beginners-part-seven-bootstrap-panels/)

Question 19: What is Jumbotron in Bootstrap?

**Answer**A Jumbotron provides some different look for a heading or we can say it is used to show some special content or information on the webpages, it is displayed as a gray box with rounded corners. It also increase the font sizes of the text inside it. To create a Jumbotron we use <div> element with class.jumbotron.  
  
**Creating Jumbotron Inside Container**  
We place the Jumbotron component by using .jumbotron class inside the <div> with class .container by which it is not extended to the edge of the screen.  
  
**Example - Jumbotron Inside Container**  
  
In this example we will create a <div> with .container class, inside it we will add one <div> with class.jumbotron and inside jumbotron we can put any HTML, Bootstrap elements for Heading, Descriptions, etc. We will create jumbotron by writing the following code. 

1. <!DOCTYPE html>
3. <html lang="en">
5. <head>
6. <meta charset="utf-8">
7. <title>Bootstrap Part8</title>
8. <meta name="viewport" content="width=device-width,initial-scale=1">
9. <link rel="stylesheet" type="text/css" href="css/bootstrap.min.css">
10. </head>
12. <body>
13. <div class="container">
14. <!--Creating Jumbotron inside container-->
15. <div class="jumbotron">
16. <h1>Bootstrap Jumbotron</h1>
17. <p>It is Bootstrap Jumbotron inside Container</p>
18. <p><a href="#" class="btn btn-info" role="button">See More Information</a></p>
19. </div>
20. </div>
21. <script src="js/jquery-2.1.4.min.js"></script>
22. <script src="js/bootstrap.min.js"></script>
23. </body>
25. </html>

**Output**Jumbotron inside Container.  
  
  
  
For more details visit the following link -

* [Bootstrap For Beginners - Part Eight (Bootstrap Jumbotron)](http://www.c-sharpcorner.com/article/bootstrap-for-beginners-part-eight-bootstrap-jumbotron/)

Question 20: Explain the Bootstrap progress bar.

**Answer**Now we will create different types of progress bars and see how they are useful for webpages.  
  
**Basic Progress Bar**  
We can create a default progress bar, by adding .progress class to a <div> element.  
  
**Example - Creating Default Progress Bar**  
  
In this example to create a basic progress bar we will add a <div> with class .progress inside the container, now inside this <div>.  
  
We will add one more <div> with class .progress-bar, we will also add width in style attribute that indicates the progress of task or action.  
  
We will add .sr-only class for Progress Bar,

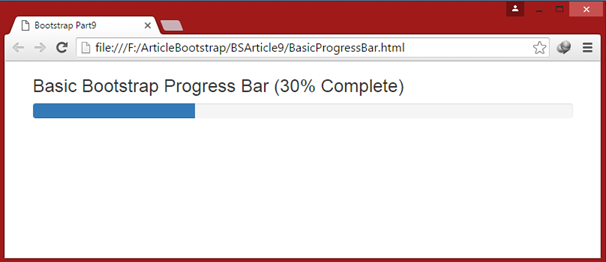
* **.sr-only -** It is used to hide an element to all devices except screen readers

We will add following properties for Progress Bar,

* **aria-valuenow -** It is used to define the current value for a range for Progress Bar
* **aria-valuemin -** It is used to define the minimum allowed value for a range
* **aria-valuemax -**It is used to define the maximum allowed value for a range

Let's create an example for Default Progress Bar by writin the following code.

1. <!DOCTYPE html>
3. <html lang="en">
5. <head>
6. <meta charset="utf-8">
7. <title>Bootstrap Part9</title>
8. <meta name="viewport" content="width=device-width,initial-scale=1">
9. <link rel="stylesheet" type="text/css" href="css/bootstrap.min.css">
10. </head>
12. <body>
13. <div class="container">
14. <h3>Basic Bootstrap Progress Bar (30% Complete)</h3>
15. <!-- A <div> element with class .progress -->
16. <div class="progress">
17. <!-- A <div> element with class .progress-bar -->
18. <div class="progress-bar" role="progressbar" aria-valuenow="30" aria-valuemin="0" aria-valuemax="100" style="width:30%">
19. <span class="sr-only">30% Complete Process</span>
20. </div>
21. </div>
22. </div>
23. <script src="js/jquery-2.1.4.min.js"></script>
24. <script src="js/bootstrap.min.js"></script>
25. </body>
27. </html>

**Output**  
  
  
For more details visit the following link -

* [Bootstrap For Beginners - Part Nine ( Bootstrap Progress Bars )](http://www.c-sharpcorner.com/article/bootstrap-for-beginners-part-nine-bootstrap-progress-bar/)

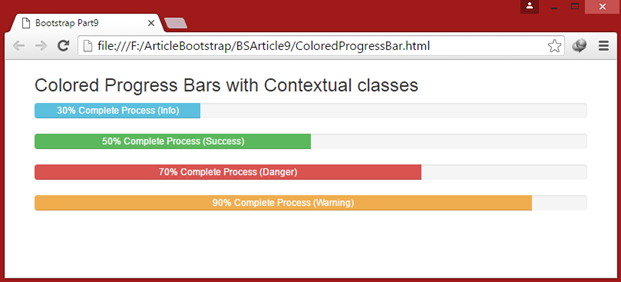
Question 21: What contextual classes are used with progress bars in Bootstrap?

**Answer**The contextual classes used with progress bars are:

* .progress-bar-info
* .progress-bar-success
* .progress-bar-danger
* .progress-bar-warning

**Creating Progress Bars with Contextual classes**  
  
In this example to create Colored Progress Bars we will add <div> with class .progress inside the container, now inside this <div> we will add one more <div> with class .progress-bar and we will add Contextual classes here like .progress-bar-info,.progress-bar-success,.progress-bar-danger,.progress-bar-warning. We will also add width in style attribute that indicates the progress of the task or action. We will also add aria attributes for progress bar, by writing the following code.

1. <!DOCTYPE html>
3. <html lang="en">
5. <head>
6. <meta charset="utf-8">
7. <title>Bootstrap Part9</title>
8. <meta name="viewport" content="width=device-width,initial-scale=1">
9. <link rel="stylesheet" type="text/css" href="css/bootstrap.min.css">
10. </head>
12. <body>
13. <div class="container">
14. <h3>Colored Progress Bars with Contextual classes</h3>
15. <div class="progress">
16. <!-- A <div> element with class .progress-bar and .progress-bar-info -->
17. <div class="progress-bar progress-bar-info" role="progressbar" aria-valuenow="30" aria-valuemin="0" aria-valuemax="100" style="width:30%">
18. 30% Complete Process (Info)
19. </div>
20. </div>
21. <div class="progress">
22. <!-- A <div> element with class .progress-bar and .progress-bar-success -->
23. <div class="progress-bar progress-bar-success" role="progressbar" aria-valuenow="50" aria-valuemin="0" aria-valuemax="100" style="width:50%">
24. 50% Complete Process (Success)
25. </div>
26. </div>
27. <div class="progress">
28. <!-- A <div> element with class .progress-bar and .progress-bar-danger -->
29. <div class="progress-bar progress-bar-danger" role="progressbar" aria-valuenow="70" aria-valuemin="0" aria-valuemax="100" style="width:70%">
30. 70% Complete Process (Danger)
31. </div>
32. </div>
33. <div class="progress">
34. <!-- A <div> element with class .progress-bar and .progress-bar-warning -->
35. <div class="progress-bar progress-bar-warning" role="progressbar" aria-valuenow="90" aria-valuemin="0" aria-valuemax="100" style="width:90%">
36. 90% Complete Process (Warning)
37. </div>
38. </div>
39. </div>
40. <script src="js/jquery-2.1.4.min.js"></script>
41. <script src="js/bootstrap.min.js"></script>
42. </body>
44. </html>

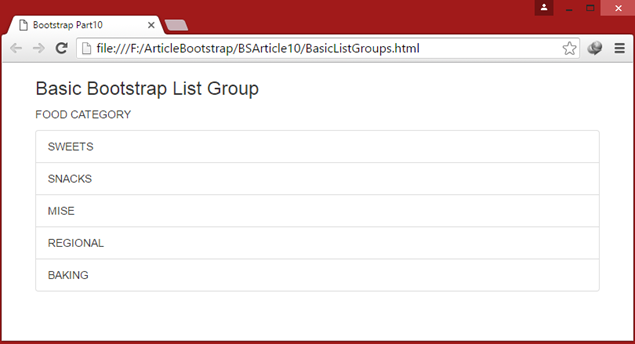
**Output**  
  
  
For more details visit the following link -

* [Bootstrap For Beginners - Part Nine ( Bootstrap Progress Bars )](http://www.c-sharpcorner.com/article/bootstrap-for-beginners-part-nine-bootstrap-progress-bar/)

Question 22: Explain Bootstrap Basic Buttons.

**Answer**The basic list group is like an unordered list with list items. For creating a Basic List Group use class .list-group to <ul> element, and into this <li> elements.  
  
**Example - Basic List Groups**  
  
In this example first we will create a <div> with class .container, then for Basic List Group, we will add <ul> element with class .list-group and inside this we will add <li> elements with class .list-group-item, and in this we will add elements of list.  
  
Let’s create a Basic List Group by writing the following code and see the output. 

1. <!DOCTYPE html>
3. <html lang="en">
5. <head>
6. <meta charset="utf-8">
7. <title>Bootstrap Part10</title>
8. <meta name="viewport" content="width=device-width,initial-scale=1">
9. <link rel="stylesheet" type="text/css" href="css/bootstrap.min.css">
10. </head>
12. <body>
13. <div class="container">
14. <h3>Basic Bootstrap List Group</h3>
15. <p>FOOD CATEGORY</p>
16. <ul class="list-group">
17. <li class="list-group-item">SWEETS</li>
18. <li class="list-group-item">SNACKS</li>
19. <li class="list-group-item">MISE</li>
20. <li class="list-group-item">REGIONAL</li>
21. <li class="list-group-item">BAKING</li>
22. </ul>
23. </div>
24. <script src="js/jquery-2.1.4.min.js"></script>
25. <script src="js/bootstrap.min.js"></script>
26. </body>
28. </html>

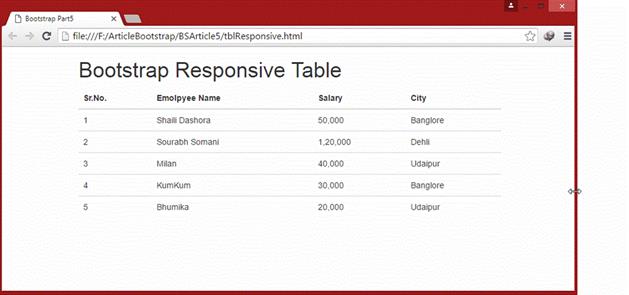
**Output**  
  
  
For more details visit the following link -

* [Bootstrap For Beginners - Part Ten (Bootstrap List Groups)](http://www.c-sharpcorner.com/article/bootstrap-for-beginners-part-ten-bootstrap-list-groups/)

Question 23: How can you create Responsive Tables with Bootstrap?

**Answer**In Bootstrap 3 we can create a responsive table by using .table-responsive class. The table will scroll horizontally on small devices (screen width under 768px). When viewing on larger than 768px wide, we will not see any difference in tables.  
  
**Example - Creating Responsive Table**  
  
In this example to make the table responsive we will place the table inside a <div> element and then we will apply the class .table-responsive on it by using the following code.

1. <!DOCTYPE html>
2. <html lang="en">
4. <head>
5. <meta charset="utf-8">
6. <title>Bootstrap Part5</title>
7. <meta name="viewport" content="width=device-width,initial-scale=1">
8. <link rel="stylesheet" type="text/css" href="css/bootstrap.min.css">
9. </head>
11. <body>
12. <div class="container">
13. <h1>Bootstrap Responsive Table</h1>
14. <!-- Div using .table-responsive class-->
15. <div class="table-responsive">
16. <!--Bootstrap Table using .table class-->
17. <table class="table">
18. <thead>
19. <tr>
20. <th>Sr.No.</th>
21. <th>Emolpyee Name</th>
22. <th>Salary</th>
23. <th>City</th>
24. </tr>
25. </thead>
26. <tbody>
27. <tr>
28. <td>1</td>
29. <td>Shaili Dashora</td>
30. <td>50,000</td>
31. <td>Banglore</td>
32. </tr>
33. <tr>
34. <td>2</td>
35. <td>Sourabh Somani</td>
36. <td>1,20,000</td>
37. <td>Dehli</td>
38. </tr>
39. <tr>
40. <td>3</td>
41. <td>Milan</td>
42. <td>40,000</td>
43. <td>Udaipur</td>
44. </tr>
45. <tr>
46. <td>4</td>
47. <td>KumKum</td>
48. <td>30,000</td>
49. <td>Banglore</td>
50. </tr>
51. <tr>
52. <td>5</td>
53. <td>Bhumika</td>
54. <td>20,000</td>
55. <td>Udaipur</td>
56. </tr>
57. </tbody>
58. </table>
59. </div>
60. </div>
61. <script src="js/jquery-2.1.4.min.js"></script>
62. <script src="js/bootstrap.min.js"></script>
63. </body>
65. </html>

**Output**Responsive Table Layout,  
  
  
  
  
  
For more details visit the following link -

* [Bootstrap For Beginners - Part Five (Bootstrap Tables)](http://www.c-sharpcorner.com/article/bootstrap-for-beginners-part-five-bootstrap-tables/)

Question 24: Explain Navigation Bar in Bootstrap.

**Answer**The Bootstrap navbar component is used to create a responsive navigation header for our website or application. A navigation bar can collapse or extend, according to the screen size of different devices.  
  
**Default Navigation Bar (navbar)**  
  
For creating default navbar we need to use <nav class="navbarnavbar-default"> and we can add responsive Default Navigation Bar to the top of the page in any web application.  
 **Example 1 - Creating Default Navbar**  
  
**Step 1**  
  
In this example we will create a default navbar. For this first we will create a basic Bootstrap HTML page by using the following code. 

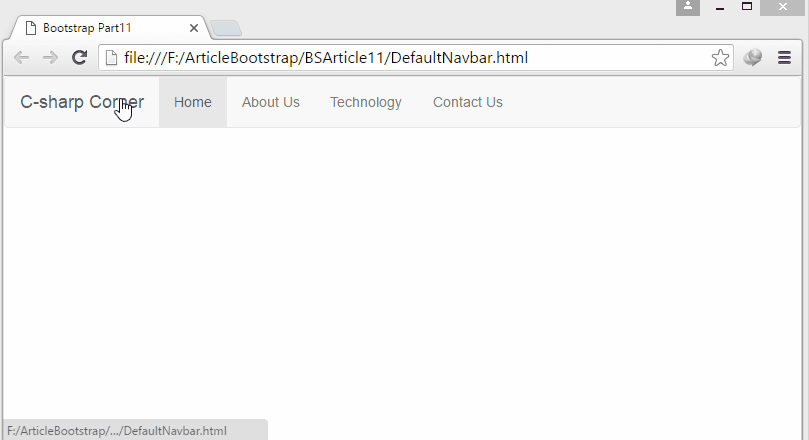
1. <!DOCTYPE html>
3. <html lang="en">
5. <head>
6. <meta charset="utf-8">
7. <title>Bootstrap Part11</title>
8. <meta name="viewport" content="width=device-width,initial-scale=1">
9. <link rel="stylesheet" type="text/css" href="css/bootstrap.min.css">
10. </head>
12. <body>
14. <script src="js/jquery-2.1.4.min.js"></script>
15. <script src="js/bootstrap.min.js"></script>
16. </body>
18. </html>

**Step 2**Now to create default navbar,

* In the <body> of the HTML page we will add <nav> tag with classes .navbar, .navbar-default.
* Inside this we will add <div> with class .container-fluid.
* In this we will add one <div> with header class .navbar-header, now we will add <a> element with class navbar-brand. By using this text becomes slightly larger in size.
* Now for adding links to the navbar we will add unordered list (ul) with the classes .nav, .navbar-nav and provide menu links using <li> element.

Let's create default navbar by using the following code,

1. <!DOCTYPE html>
3. <html lang="en">
5. <head>
6. <meta charset="utf-8">
7. <title>Bootstrap Part11</title>
8. <meta name="viewport" content="width=device-width,initial-scale=1">
9. <link rel="stylesheet" type="text/css" href="css/bootstrap.min.css">
10. </head>
12. <body>
13. <!--Default Navbar Start Here-->
14. <!--<nav> tag start with classes .navbar and .navbar-default-->
15. <nav class="navbar navbar-default">
16. <div class="container-fluid">
17. <!--Navbar Header Start Here-->
18. <div class="navbar-header">
19. <a class="navbar-brand" href="#">C-sharp Corner</a>
20. </div>
21. <!--Navbar Header End Here-->
22. <!--Menu Start Here-->
23. <ul class="nav navbar-nav">
24. <li class="active"><a href="#">Home</a></li>
25. <li><a href="#">About Us</a></li>
26. <li><a href="#">Technology</a></li>
27. <li><a href="#">Contact Us</a></li>
28. </ul>
29. <!--Menu End Here-->
30. </div>
31. </nav>
32. <!--<nav> tag end-->
33. <!--Default Navbar End Here-->
34. <script src="js/jquery-2.1.4.min.js"></script>
35. <script src="js/bootstrap.min.js"></script>
36. </body>
38. </html>

**Output**Default Navigation Bar (navbar).  
  
  
  
For more details visit the following link -

* [Bootstrap For Beginners: Navigation Bar - Part Eleven](http://www.c-sharpcorner.com/article/bootstrap-for-beginners-navigation-bar-part-eleven2/)

Question 25: What are the Responsive Utilities Classes in Bootstrap?

**Answer**

You can use the following responsive classes to enable the element's visibility depending on the device screen sizes.

|  |  |
| --- | --- |
| Class | Description |
| .visible-xs-\* | The elements visible only on extra small devices having screen width less than 768px. Hidden on others. |
| .visible-sm-\* | The elements visible only on small devices having screen width greater than or equal to 768px. Hidden on others. |
| .visible-md-\* | The elements visible only on medium devices having screen width greater than or equal to 992px. Hidden on others. |
| .visible-lg-\* | The elements visible only on larger devices having screen width greater than or equal to 1200px. Hidden on others. |

**Note***You can also mix these classes to make the elements visible on multiple devices.*  
**HTML Code**

1. <!DOCTYPE html >
2. <html>
4. <head>
5. <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
6. <title>Responsive</title>
7. <!-- compiled and minified CSS -->
8. <link rel="stylesheet" href="css/bootstrap.min.css">
9. <!-- Optional theme -->
10. <link rel="stylesheet" href="css/bootstrap-theme.min.css">
11. </head>
13. <body>
14. <p class="visible-xs-block">This paragraph is visible only on extra small devices.</p>
15. <p class="visible-sm-block">This paragraph is visible only on small devices.</p>
16. <p class="visible-md-block">This paragraph is visible only on medium devices.</p>
17. <p class="visible-lg-block">This paragraph is visible only on large devices.</p>
18. <!-- compiled and minified JavaScript -->
19. <script type="text/javascript" src="js/bootstrap.min.js"></script>
20. </body>
22. </html>

Similarly you can use these hidden utility classes to hide the elements depending on the devices.

|  |  |
| --- | --- |
| **Class** | **Description** |
| .hidden-xs-\* | Hide the elements only on extra small devices having screen width less than 768px. Visible on others. |
| .hidden-sm-\* | Hide the elements only on small devices having screen width greater than or equal to 768px. Visible on others. |
| .hidden-md-\* | Hide the elements only on medium devices having screen width greater than or equal to 992px. Visible on others. |
| .hidden-lg-\* | Hide the elements only on larger devices having screen width greater than or equal to 1200px. Visible on others. |

**Code**

1. <p class="hidden-xs">This paragraph is hidden only on extra small devices.</p>
2. <p class="hidden-sm">This paragraph is hidden only on small devices.</p>
3. <p class="hidden-md">This paragraph is hidden only on medium devices.</p>
4. <p class="hidden-lg">This paragraph is hidden only on large devices.</p>

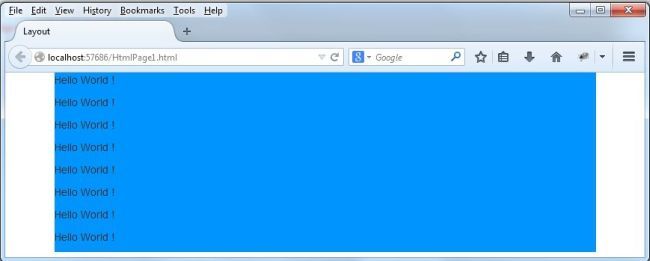
For more details visit the following link -

* [Getting Started With Bootstrap](http://www.c-sharpcorner.com/UploadFile/b69ed7/getting-started-with-bootstrap/)

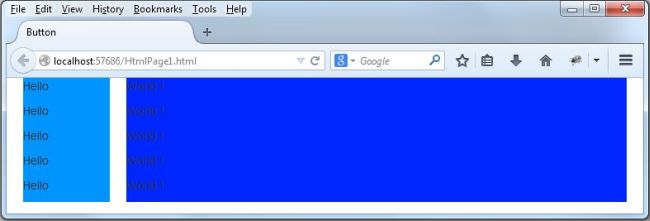
**Question 26: Explain Layout in Bootstrap.**

**Answer**Twitter Bootstrap Layout has two different layouts. The first is fixed layout and the second is fluid layout. Bootstrap has added the responsive features for phone, tablets and websites that are a perfect view for large-screen desktops to small-screen desktops.   
  
**Fixed/ Grid Layout**If we are looking for a webpage or app layout on a fixed number of size (pixels), then we choose fixed layout. 

1. <body>
2. <div class="container">
3. <p> Hello World !</p>
4. <p> Hello World !</p>
5. <p> Hello World !</p>
6. <p> Hello World !</p>
7. <p> Hello World !</p>
8. <p> Hello World !</p>
9. <p> Hello World !</p>
10. <p> Hello World !</p>
11. </div>
12. </body>

  
  
**Fluid Layout**  
If we want to create a layout that is not fixed, not on the basis of percentage to keep it flexible, then we choose fluid layout.

1. <div class="container-fluid">
2. <div class="row-fluid">
3. <div class="span2">
4. <p>Hello </p>
5. <p>Hello </p>
6. <p>Hello </p>
7. <p>Hello </p>
8. <p>Hello </p>
9. </div>
10. <div class="span10">
11. <p>World !</p>
12. <p>World !</p>
13. <p>World !</p>
14. <p>World !</p>
15. <p>World !</p>
17. </div>
18. </div>
19. </div>

  
  
For more details visit the following link -

* [Twitter Bootstrap 3 Layout and Buttons](http://www.c-sharpcorner.com/UploadFile/736ca4/twitter-bootstrap-3-layout-and-buttons/)

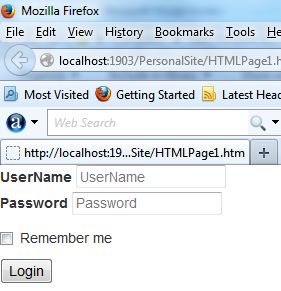
Question 27: How can you create forms using Twitter Bootstrap?

Answer  
  
To make the forms more attractive using Bootstrap, open up the bootstrap.css file and check out the following Bootstrap CSS class.

1. Vertical Form (default form layout)
2. Horizontal Form
3. Inline Form

**1. Using default form layout**Now first we use the form tag with label and TextBox. The HTML file looks as in the following,

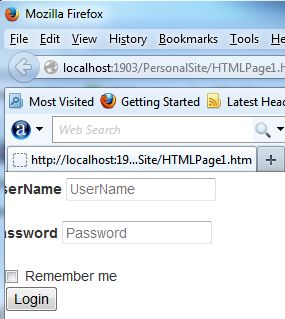
1. <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
2. <html>
4. <head>
5. <title></title>
6. <link href="Bootstrap/css/bootstrap.css" rel="stylesheet" type="text/css" />
7. <script src="dist/js/bootstrap.js" type="text/javascript"></script>
8. </head>
10. <body>
11. <form>
12. <div>
13. <label for="inputUserName">UserName</label>
14. <input type="email" id="inputEmail" placeholder="UserName">
15. </div>
16. <div>
17. <label for="inputPassword">Password</label>
18. <input type="password" id="inputPassword" placeholder="Password">
19. </div>
20. <div class="checkbox">
21. <label><input type="checkbox"> Remember me</label>
22. </div>
23. <button type="submit">Login</button>
24. </form>
25. </body>
27. </html>

The HTML will render without Bootstrap as in the following,  
  
  
  
**2. Using bootstrap CSS class="Horizontal Form"**In the horizontal form layout the label in the right is floated to the left to make them appear on the same line as form controls. To make a form that uses the Horizontal layout,

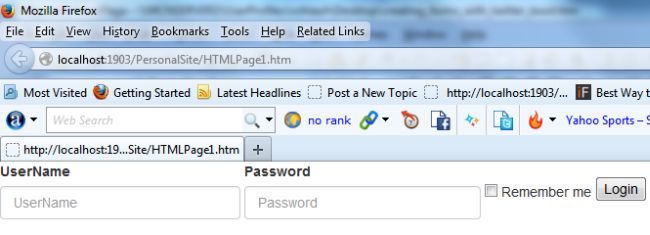
* Add .form-horizontal to the <form> element
* Wrap labels and form controls in .control-group
* Add .control-label to the <label> element

The list groups class is used to display lists of items in a beautiful manner. To do that you can use a list-group-item class with every item.  
  
The HTML file looks as in the following,

1. <form class="form-horizontal">
2. <div class="form-group">
3. <label for="inputUserName" class="control-label">
4. UserName</label>
5. <input type="email" id="inputUserName" placeholder="UserName">
6. </div>
7. <div class="form-group">
8. <label for="inputPassword" class="control-label">
9. Password</label>
10. <input type="password" id="inputPassword" placeholder="Password">
11. </div>
12. <div class="checkbox">
13. <label>
14. <input type="checkbox">
15. Remember me</label>
16. </div>
17. <button type="submit">
18. Login</button>
19. </form>

The HTML will render without Bootstrap as in the following,  
  
  
  
**3. Using bootstrap CSS class="Inline Form"**  
To make a form where all the elements are inline just add Inline Form class with the form tag. You can do this by simply applying the Bootstrap's class Inline Form:  
  
The HTML file looks as in the following,

1. <form class="form-inline">
2. <div class="form-group">
3. <label for="inputUserName">
4. UserName</label>
5. <input type="email" id="inputEmail" class="form-control" placeholder="UserName">
6. </div>
7. <div class="form-group">
8. <label for="inputPassword">
9. Password</label>
10. <input type="password" id="inputPassword" class="form-control" placeholder="Password">
11. </div>
12. <div class="checkbox">
13. <label>
14. <input type="checkbox">
15. Remember me</label>
16. </div>
17. <button type="submit">
18. Login</button>
19. </form>

The HTML will render without Bootstrap as in the following,  
  
  
  
For more details visit the following link -

* [Creating Forms with Twitter Bootstrap](http://www.c-sharpcorner.com/UploadFile/rohatash/creating-forms-with-twitter-bootstrap679/)

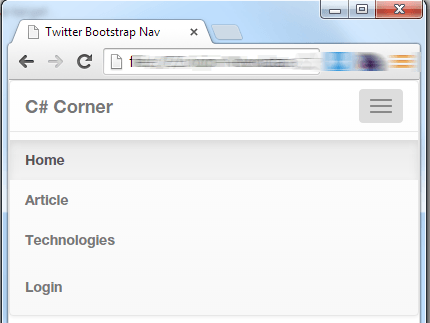
Question 28: How can you create a simple Navbar with Twitter Bootstrap?

**Answer**You can create various variations of navbars quickly and without too much effort. The procedure is listed below:,

* Create nav container with class .navigation and *.navbar-default*.
* For better mobile display you need to create a container with the class*.navbar-header*. Inside this container, you can create a button to toggle the menu items using data-target.
* Now, you can create your menu item list container. This container has the same id as *data-target*and the class is .collapse and*.navbar-collapse*.
* Create menu list with ul li structure with the class .nav and .navbar-nav.

The following example will you show how to create a static navbar with basic navigation.  
  
**HTML Code**

1. <nav role="navigation" class="navbar navbar-default">
2. <!-- Brand and toggle get grouped for better mobile display -->
3. <div class="navbar-header">
4. <button type="button" data-target="#navbarCollapse" data-toggle="collapse" class="navbar-toggle"> <span class="sr-only">Toggle navigation</span> <span class="icon-bar"></span> <span class="icon-bar"></span> <span class="icon-bar"></span> </button>
5. <a href="#" class="navbar-brand">C# Corner</a> </div>
6. <!-- Collection of nav links and other content for toggling -->
7. <div id="navbarCollapse" class="collapse navbar-collapse">
8. <ul class="nav navbar-nav">
9. <li class="active"><a href="#">Home</a></li>
10. <li><a href="#">Article</a></li>
11. <li><a href="#">Technologies</a></li>
12. </ul>
13. <ul class="nav navbar-nav navbar-right">
14. <li><a href="#">Login</a></li>
15. </ul>
16. </div>
17. </nav>

**Result**  
  
  
  
**Mobile Display**  
  
  
  
For more details visit the following link -

* [Getting Started With Bootstrap: Part 3](http://www.c-sharpcorner.com/UploadFile/b69ed7/getting-started-with-bootstrap-part-3311/)

Question 29: Explain the history of Bootstrap and what languages are required to start Bootstrap.

**Answer**  
History of Bootstrap - Bootstrap began as an internal project at Twitter hence sometimes Bootstrap is known as “*Twitter Bootstrap*”. Two guys named Mark Otto and Jacob Thornton got management approval to open source it.  
  
Language Required when starting Bootstrap: Knowledge of HTML, CSS and JavaScript.  
  
Now Start With Bootstrap: The coding starts with*<!DOCTYPE html>*, it shows the version of HTML that we use.

1. <!DOCTYPE html>
2. <html>
3. …………..
4. ………..
5. </html>

To ensure proper rendering and touch zooming, add the viewport meta tag to <Head>.

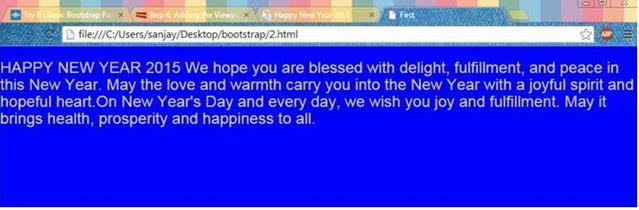
1. <meta name="viewport" content="width=device-width, initial-scale=1">

You can disable zooming capabilities on mobile devices by adding user-scalable=no to the viewport meta tag.

1. <meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1, user-scalable=no">

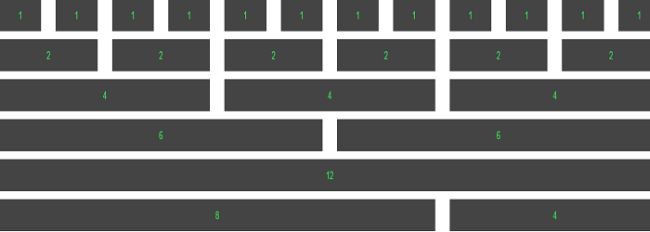
First program with Bootstrap,

1. <!DOCTYPE html>
2. <html>
4. <head>
5. <title>First</title>
6. <link rel="stylesheet" href="css/bootstrap.min.css">
7. <style>
8. body {
9. background-color: #00F;
10. }
12. h3 {
13. color: #CCC;
14. }
15. </style>
16. <meta name="viewport" content="width= device-width,initial-scale=1">
17. <script src="js/bootstrap.min.js"></script>
18. </head>
20. <body>
21. <h3>HAPPY NEW YEAR 2015 We hope you are blessed with delight, fulfillment, and peace in this New Year. May the love and warmth carry you into the New Year with a joyful spirit and hopeful heart.On New Year's Day and every day, we wish you joy and fulfillment. May it brings health, prosperity and happiness to all.</h3>
22. </body>
24. </html>

**Output**  
  
  
For more details visit the following link -

* [Journey With Bootstrap: Day 1](http://www.c-sharpcorner.com/UploadFile/2cb323/journey-with-bootstrap-day-1/)

Question 30: How many types of screens can be in Grid System?

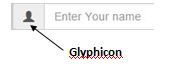
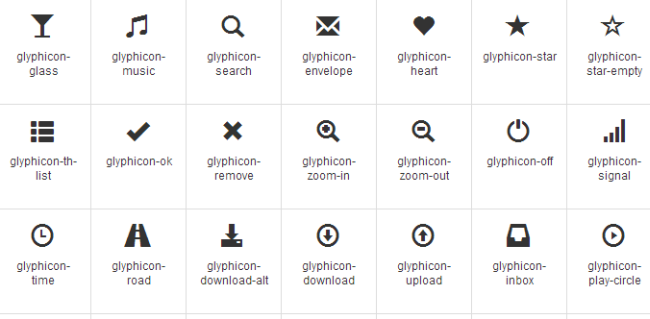
Answer  
  
Bootstrap has the cool feature of being able to resize automatically based on the screen size of devices. We can organize content in multiple columns when using large screen devices but when we are using small screen devices we need to customize the columns to prevent loss of content and also to look good of our applications or websites.  
  
  
  
We have four types of screens

* col-xs extra small screen size in devices such as smartphones (<768 px).
* col-sm small screen size in devices such as tablets (>=768 px).
* col-md medium screen size in devices such as laptops (>=992 px).
* col-lg large screen size in devices such as computers (>=1200 px).

For more details visit the following link -

* [Journey With Bootstrap: Day 2 ( Grid System)](http://www.c-sharpcorner.com/UploadFile/2cb323/journey-with-bootstrap-day-2-grid-system/)

Question 31: What is Glyphicons in Bootstrap?

Answer  
  
Glyphicons are icons we use to make our web page more attractive. For example, when we open a signup page then we see a user icon before a text box, this icon is glyphicon.   
  
  
  
**Syntax**  
  
The syntax to add Glyphicons to a web page is,  
  
*<span class="glyphiconglyphicon-name"></span>*  
  
Some glyphicons are,  
  
  
  
The following is an example of using it.

1. <!DOCTYPE html>
2. <html>
4. <head>
5. <title>
6. Glyphicons
7. </title>
8. <link rel="stylesheet" href="js/css/bootstrap.min.css">
9. <script src="js/bootstrap.min.js"></script>
10. <meta name="viewport" content="width= device-width, initial-scale=1">
11. <style>
12. body {
13. text-align: center;
14. color: #006;
15. background-color: #CCC;
16. }
17. </style>
18. </head>
20. <body>
21. User Name <span class="glyphicon glyphicon-user"></span><br> Contact No <span class="glyphicon glyphicon-phone"></span><br> Address <span class="glyphicon glyphicon-home"></span><br> Download
22. <a href="#">
23. <span class="glyphicon glyphicon-download"></span></a><br>
24. <button type="button" class="btn btn-default btn-md">
25. <span class="glyphicon glyphicon-cloud-upload">Upload</span>
26. </button>
27. </body>
29. </html>

**Output**  
  
  
For more details visit the following link -

* [Journey With Bootstrap: Day 3 (Glyphicons)](http://www.c-sharpcorner.com/UploadFile/2cb323/journey-with-bootstrap-day-3-glyphicons959/)

Question 32: Give the syntax of buttons in Bootstrap.

Answer  
  
Syntax of Button  
  
*<button type="button" class="btn btn-size">   
<span class="glyphicon glyphicon-name"></span></button>*

* Default
* Info
* Primary
* Danger
* success
* warning

**Default Button Syntax**  
*<button type=”button” class=”btn btn-default btn-size”>   
Default   
</button>*  
**Info Button Syntax**  
*<button type=”button” class=”btn btn-Info btn-size”>   
Info   
</button>*  
**Primary Button Syntax***<button type=”button” class=”btn btn-primary btn-size”>   
Primary   
</button>*  
**Danger Button Syntax***<button type=”button” class=”btn btn-Danger btn-size”>   
Danger   
</button>*  
  
**Success Button Syntax***<button type=”button” class=”btn btn-Success btn-size”>   
Success   
</button>*  
  
**Warning Button Syntax**  
*<button type=”button” class=”btn btn-Warning btn-size”>   
Warning   
</button>*  
  
For more details visit the following link -

* [Journey With Bootstrap : Day 4 (Dropdown Buttons)](http://www.c-sharpcorner.com/UploadFile/2cb323/journey-with-bootstrap-day-4-dropdown-buttons/)

Question 33: Explain media objects in Bootstrap.

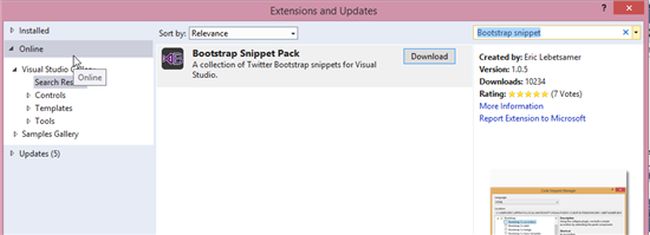
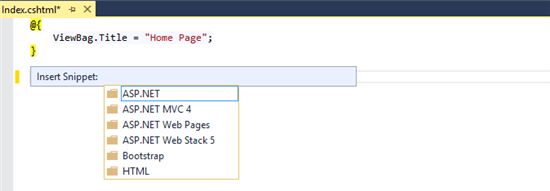
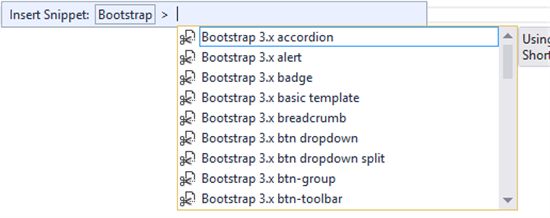
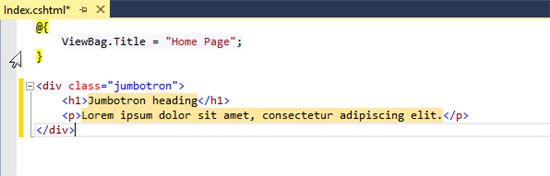
**Answer**If you want to create a layout like blog comments that contain a left- or right-aligned image along with the textual content, you can do that easily through the newly introduced Bootstrap media component, like this.  
 **HTML Code**

1. <div class="media">
2. <a href="#" class="pull-left">
3. <img src="image.png" class="media-object">
4. </a>
5. <div class="media-body">
6. <h4 class="media-heading">Vipin Kumar <small><i>Posted on September 02, 2014</i></small></h4>
7. <p> The first part Getting started with Bootstrap provided a basic introduction of bootstrap and bootstrap layout.</p>
8. </div>
9. </div>

**Result**  
**Note***You can also create a list of media objects or nested media objects using the media list component. It can be useful for comment threads or article lists.*  
  
For more details visit the following link -

* [Getting Started With Bootstrap: Part 2](http://www.c-sharpcorner.com/UploadFile/b69ed7/getting-started-with-bootstrap437/)

Question 34: What is Bootstrap Snippet Pack?

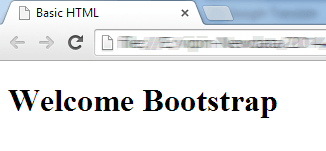
**Answer**  
  
Bootstrap Snippet Pack has a collection of Bootstrap snippets for Visual Studio.  
  
Very often, web developers need to minimize their development environment then go to the Bootstrap website to find a piece of HTML for a specific component.  
  
Doing this all day reduces the developer productivity since he must switch back and forth between the browser and the development environment.  
  
Fortunately, with Bootstrap Snippet Pack, a code snippet can be retrieved via Visual Studio.  
  
To install the Bootstrap Snippet Pack from Visual Studio, go to Tools, then Extensions and Updates > Online > Search Bootstrap Snippet Pack.  
  
  
  
Once installed from Visual Studio editor, hit CTL+K, CTL+X to bring in the snippet tool.  
  
  
  
Then select Bootstrap and the component needed.  
  
  
  
The Bootstrap Snippet Pack will automatically generate the required HTML.  
  
  
  
For more details visit the following link -

* [Visual Studio Tools For Bootstrap](http://www.c-sharpcorner.com/UploadFile/a5470d/visual-studio-tools-for-bootstrap/)

Question 35: How can you make an HTML File a Bootstrap Template?

**Answer**  
  
For making this file a Bootstrap Template, after Downloading Bootstrap by the above procedure we will include Bootstrap CSS and JS files. We should include JS files at the bottom of the HTML page before closing <body> tag with the following code.

1. <!DOCTYPE html>
2. <html lang="en">
4. <head>
5. <meta charset="utf-8">
6. <title>Bootstrap Part1</title>
7. <meta name="viewport" content="width=device-width,initial-scale=1">
8. <link rel="stylesheet" type="text/css" href="css/bootstrap.min.css">
9. </head>
11. <body>
12. <h1>Hello C# Corner</h1>
13. <script src="js/jquery-2.1.4.min.js"></script>
14. <script src="js/bootstrap.min.js"></script>
15. </body>
17. </html>

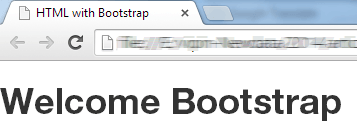
Now we will open the file in a browser by double clicking on it and see the output.  
 **Output**  
  
  
For more details visit the following link -

* [Bootstrap For Beginners - Part One (Introduction And Implementation)](http://www.c-sharpcorner.com/article/bootstrap-for-beginners-part-one-introduction-and-impleme/)

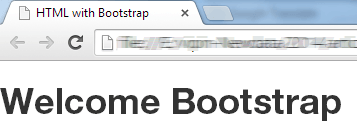
Question 36: What is Twitter Bootstrap? Explain with an example.

**Answer**  
  
Twitter Bootstrap is a very popular and powerful front-end framework currently. It is self-generated for faster and easier web development. Twitter Bootstrap comes prepared with HTML, CSS and JavaScript for various web and user interface components. Bootstrap provides iconography, dropdowns, navigation, alerts, popovers and much more. This is covered in detail in the section Layout Components.  
  
**Creating Your First Web Page with Twitter Bootstrap**  
  
You have learned the structure and the purposes of Bootstrap files, now we'll create a basic Bootstrap template that includes everything we specified in the file structure. You will have made an HTML file that displays a "Welcome Bootstrap" message in your web browser.  
  
**Step 1**  
  
Creating a Basic HTML File.  
  
Open up your code editor and create a new HTML file. Start with an empty window and type the following code,

1. <!DOCTYPE html >
2. <html>
4. <head>
5. <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
6. <title>Basic HTML</title>
7. </head>
9. <body>
10. <h1>Welcome Bootstrap</h1>
11. </body>
13. </html>

**Result**  
  
  
  
**Step 2**  
  
Making this HTML File a Bootstrapped Template  
  
To make this HTML file a Bootstrapped template, just include the appropriate Bootstrap CSS and JavaScript files.  
  
**Note**  
  
*You should include JavaScript files at the bottom of the page; before closing the <body> tag (in other words </body>) to improve the performance of your web pages.*

1. <!DOCTYPE html >
2. <html>
4. <head>
5. <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
6. <title>HTML with Bootstrap</title>
7. <!-- compiled and minified CSS -->
8. <link rel="stylesheet" href="css/bootstrap.min.css">
9. <!-- Optional theme -->
10. <link rel="stylesheet" href="css/bootstrap-theme.min.css">
11. </head>
13. <body>
14. <h1>Welcome Bootstrap</h1>
15. <!-- compiled and minified JavaScript -->
16. <script type="text/javascript" src="js/bootstrap.min.js"></script>
17. </body>
19. </html>

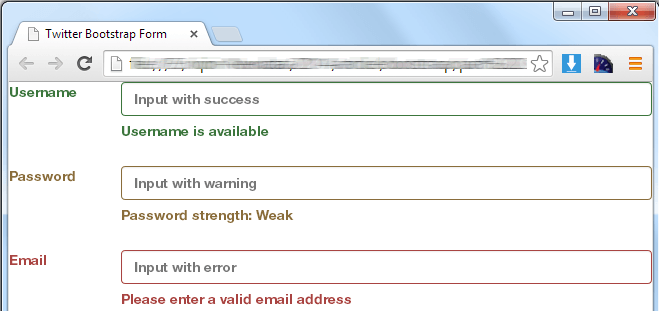
**Result**  
  
  
For more details visit the following link -

* [Getting Started With Bootstrap](http://www.c-sharpcorner.com/UploadFile/b69ed7/getting-started-with-bootstrap/)

Question 37: Explain Validation States with Twitter Bootstrap?

**Answer**  
  
Bootstrap provides an easy-to-use and powerful mechanism for styling input controls to present various validation states. Bootstrap includes validation styles for error, warning and success messages. To use, just add the appropriate class to the surrounding .form-group.  
  
The following shows Twitter Bootstrap Form Validation States.  
  
**HTML Code**

1. <form class="form-horizontal">
2. <div class="form-group has-success">
3. <label class="col-xs-2 control-label" for="inputSuccess">Username</label>
4. <div class="col-xs-10">
5. <input type="text" id="inputSuccess" class="form-control" placeholder="Input with success">
6. <span class="help-block">Username is available</span>
7. </div>
8. </div>
9. <div class="form-group has-warning">
10. <label class="col-xs-2 control-label" for="inputWarning">Password</label>
11. <div class="col-xs-10">
12. <input type="password" id="inputWarning" class="form-control" placeholder="Input with warning">
13. <span class="help-block">Password strength: Weak</span>
14. </div>
15. </div>
16. <div class="form-group has-error">
17. <label class="col-xs-2 control-label" for="inputError">Email</label>
18. <div class="col-xs-10">
19. <input type="email" id="inputError" class="form-control" placeholder="Input with error">
20. <span class="help-block">Please enter a valid email address</span>
21. </div>
22. </div>
23. </form>

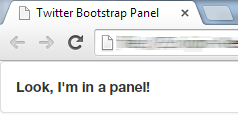
**Result**For more details visit the following link –

* [Getting Started With Bootstrap: Part 2](http://www.c-sharpcorner.com/UploadFile/b69ed7/getting-started-with-bootstrap437/)

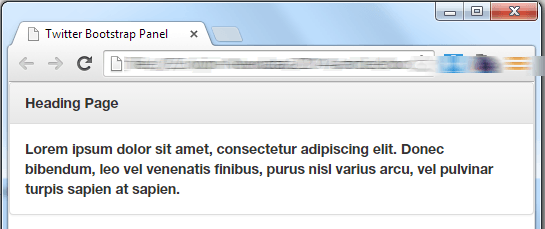
Question 38: How do you create panels with Twitter Bootstrap?

**Answer**  
Sometimes you might need to put your content in a box for a certain reason. In such a condition we can use panel components. In the most basic form the panel component applies a border and padding around the content. The following example will you show how to create simple panels.  
  
**HTML Code**

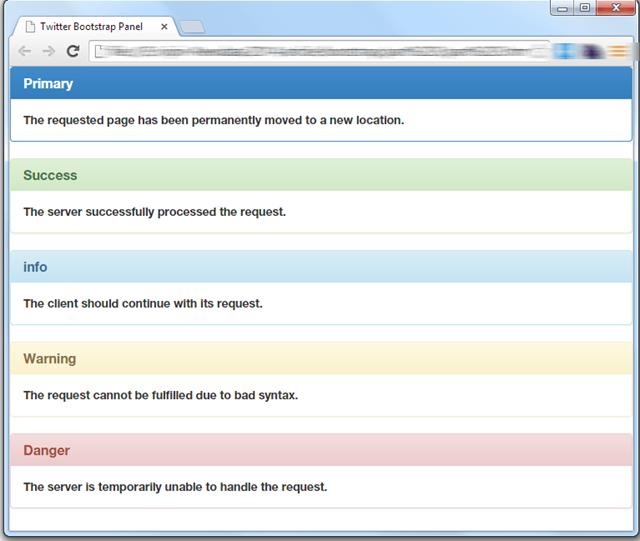
1. <div class="panel panel-default">
2. <div class="panel-body">Look, I'm in a panel!</div>
3. </div>

**Result**  
  
  
  
You can also add a heading to your panel with the .panel-heading class.  
  
**HTML Code**

1. <div class="panel panel-default">
2. <div class="panel-heading">Heading Page</div>
3. <div class="panel-body">Lorem ipsum dolor sit amet, consectetur adipiscing elit. Donec bibendum, leo vel venenatis finibus, purus nisl varius arcu, vel pulvinar turpis sapien at sapien. </div>
4. </div>

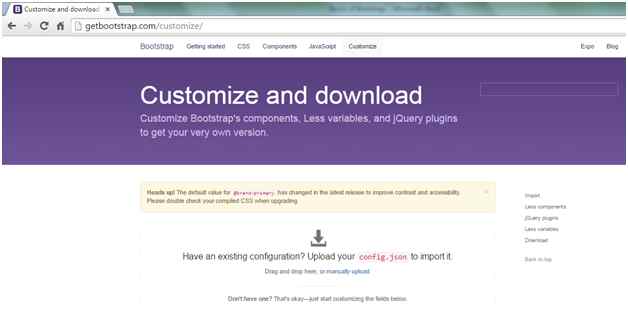
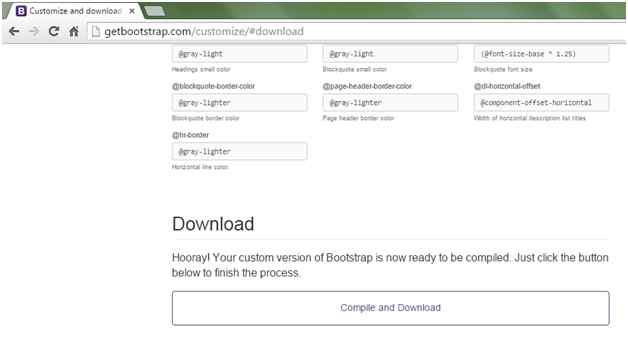
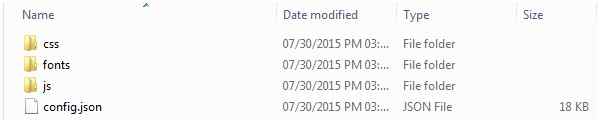
**Result**  
  
  
**Panels with Contextual States**  
  
In Twitter Bootstrap you can also add contextual classes like .panel-primary, .panel-success, .panel-info, .panel-warning, or .panel-danger on the panel components to make it more meaningful and to draw the attention of the user.  
  
**HTML Code**

1. <div class="panel panel-primary">
2. <div class="panel-heading">
3. <h3 class="panel-title">Primary</h3>
4. </div>
5. <div class="panel-body">The requested page has been permanently moved to a new location.</div>
6. </div>
7. <div class="panel panel-success">
8. <div class="panel-heading">
9. <h3 class="panel-title">Success</h3>
10. </div>
11. <div class="panel-body">The server successfully processed the request.</div>
12. </div>
13. <div class="panel panel-info">
14. <div class="panel-heading">
15. <h3 class="panel-title">info</h3>
16. </div>
17. <div class="panel-body">The client should continue with its request.</div>
18. </div>
19. <div class="panel panel-warning">
20. <div class="panel-heading">
21. <h3 class="panel-title">Warning</h3>
22. </div>
23. <div class="panel-body">The request cannot be fulfilled due to bad syntax.</div>
24. </div>
25. <div class="panel panel-danger">
26. <div class="panel-heading">
27. <h3 class="panel-title">Danger</h3>
28. </div>
29. <div class="panel-body">The server is temporarily unable to handle the request.</div>
30. </div>

**Result**  
For more details visit the following link -

* [Getting Started With Bootstrap: Part 3](http://www.c-sharpcorner.com/UploadFile/b69ed7/getting-started-with-bootstrap-part-3311/)

Question 39: How can you customize the CSS of Bootstrap?

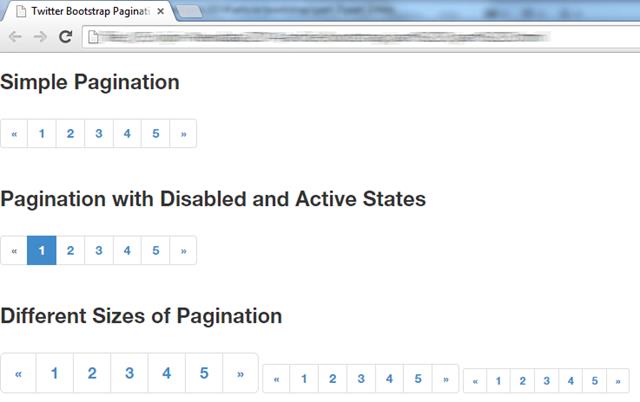
**Answer**Customize:  
  
  
  
  
  
Here you can customize you CSS of Bootstrap like color, border, font size, font style and many more. And at the bottom of this page, there is a compile and download button that will download the Bootstrap files to your local machine.  
  
When you download the file, extract this file and you will see this.  
  
  
  
Just copy them to the root directory of your web application.  
  
For more details visit the following link -

* [Basics of Bootstrap](http://www.c-sharpcorner.com/UploadFile/da55bf/basics-of-bootstrap675/)

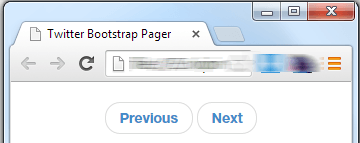
Question 40: What is Twitter Bootstrap Pagination?

Answer - Pagination is the process of organizing content by dividing it into separate pages. The following example shows you how to create various types of pagination.  
  
**HTML Code**

1. <h3> Simple Pagination </h3>
2. <ul class="pagination">
3. <li><a href="#">«</a></li>
4. <li><a href="#">1</a></li>
5. <li><a href="#">2</a></li>
6. <li><a href="#">3</a></li>
7. <li><a href="#">4</a></li>
8. <li><a href="#">5</a></li>
9. <li><a href="#">»</a></li>
10. </ul>
11. <h3> Pagination with Disabled and Active States </h3>
12. <ul class="pagination">
13. <li class="disabled"><a href="#">«</a></li>
14. <li class="active"><a href="#">1</a></li>
15. <li><a href="#">2</a></li>
16. <li><a href="#">3</a></li>
17. <li><a href="#">4</a></li>
18. <li><a href="#">5</a></li>
19. <li><a href="#">»</a></li>
20. </ul>
21. <h3>Different Sizes of Pagination</h3>
22. <!-- Larger pagination -->
23. <ul class="pagination pagination-lg">
24. <li><a href="#">«</a></li>
25. <li><a href="#">1</a></li>
26. <li><a href="#">2</a></li>
27. <li><a href="#">3</a></li>
28. <li><a href="#">4</a></li>
29. <li><a href="#">5</a></li>
30. <li><a href="#">»</a></li>
31. </ul>
32. <!-- Default pagination -->
33. <ul class="pagination">
34. <li><a href="#">«</a></li>
35. <li><a href="#">1</a></li>
36. <li><a href="#">2</a></li>
37. <li><a href="#">3</a></li>
38. <li><a href="#">4</a></li>
39. <li><a href="#">5</a></li>
40. <li><a href="#">»</a></li>
41. </ul>
42. <!-- Smaller pagination -->
43. <ul class="pagination pagination-sm">
44. <li><a href="#">«</a></li>
45. <li><a href="#">1</a></li>
46. <li><a href="#">2</a></li>
47. <li><a href="#">3</a></li>
48. <li><a href="#">4</a></li>
49. <li><a href="#">5</a></li>
50. <li><a href="#">»</a></li>
51. </ul>

**Result**  
  
  
**Twitter Bootstrap Pager**  
  
In Bootstrap you can create a pager (Previous and Next links) on your website. For creating a pager with Bootstrap you can add a Bootstrap class .pager in the ul li structure.   
  
**HTML Code**

1. <ul class="pager">
2. <li><a href="#">Previous</a></li>
3. <li><a href="#">Next</a></li>
4. </ul>

**Result**  
  
  
**Note***You can create an alignment of the pager using the .previous and .next class and you can disable the links using the .disable class.*  
For more details visit the following link -

* [Getting Started With Bootstrap: Part 3](http://www.c-sharpcorner.com/UploadFile/b69ed7/getting-started-with-bootstrap-part-3311/)

Question 41: What is a Bootstrap Breadcrumb?

**Answer**A breadcrumb is a navigation scheme that indicates the user's location in a website or web application. The following example shows how to create breadcrumbs.  
  
**HTML Code**

1. <ul class="breadcrumb">
2. <li><a href="#">Home</a></li>
3. <li><a href="#">Products</a></li>
4. <li class="active">Accessories</li>
5. </ul>

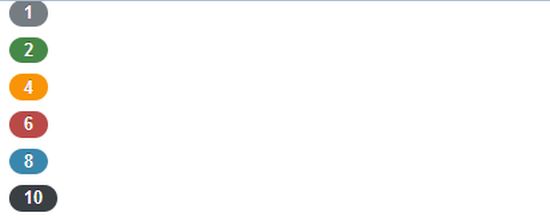
**Result**  
  
  
For more details visit the following link -

* [Getting Started With Bootstrap: Part 3](http://www.c-sharpcorner.com/UploadFile/b69ed7/getting-started-with-bootstrap-part-3311/)

Question 42: What are Bootstrap Badges?

Answer - Badges are small and simple components for displaying an indicator or to count some sort of number. This is very useful for mail count and notification etc.  
  
**Example**

1. <html>
3. <head>
4. <title>Example of Badges</title>
5. <link href="bootstrap.css" rel="stylesheet">
6. </head>
8. <body>
9. <div class="Container">
10. <div class="Row">
11. <div class="Span">
12. <p><span class="badge">1</span></p>
13. <p><span class="badge badge-success">2</span></p>
14. <p><span class="badge badge-warning">4</span></p>
15. <p><span class="badge badge-important">6</span></p>
16. <p><span class="badge badge-info">8</span></p>
17. <p><span class="badge badge-inverse">10</span></p>
18. </div>
19. </div>
20. </div>
21. </body>
23. </html>

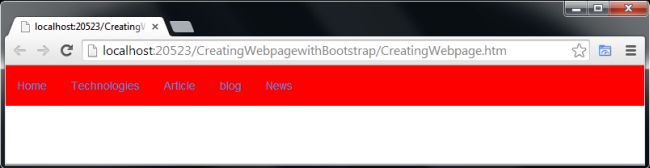
**Output**  
  
  
  
For more details visit the following link -

* [Twitter Bootstrap Inline and Badges in PHP](http://www.c-sharpcorner.com/UploadFile/d9da8a/twitter-bootstrap-inline-and-badges-in-php/)

Question 43: How can you create a Header using Bootstrap?

Answer  
  
Using Twitter Bootstrap, you may create static navigation menus. To create a menu using Bootstrap open up the bootstrap.css file and check out the CSS class navbar-nav. The HTML file looks as in the following,

1. <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
3. <html xmlns="http://www.w3.org/1999/xhtml">
5. <head>
6. <title></title>
7. <script src="Scripts/jquery-1.4.1.min.js" type="text/javascript"></script>
8. <link href="Bootstrap/css/bootstrap.css" rel="stylesheet" type="text/css" />
9. <script src="Bootstrap/js/bootstrap.min.js" type="text/javascript"></script>
10. </head>
12. <body>
13. <div class="Container">
14. <ul class="nav navbar-nav">
15. <li><a href="#">Home</a></li>
16. <li><a href="#about">Technologies</a></li>
17. <li><a href="#contact">Article</a></li>
18. <li><a href="#contact">blog</a></li>
19. <li><a href="#contact">News</a></li>
20. </ul>
21. </div>
22. </body>
24. </html>

The HTML will be rendered with Bootstrap as in the following,  
  
  
  
For more details visit the following link -

* [Creating a Webpage With Twitter BootStrap](http://www.c-sharpcorner.com/UploadFile/rohatash/creating-a-webpage-with-twitter-bootstrap/)

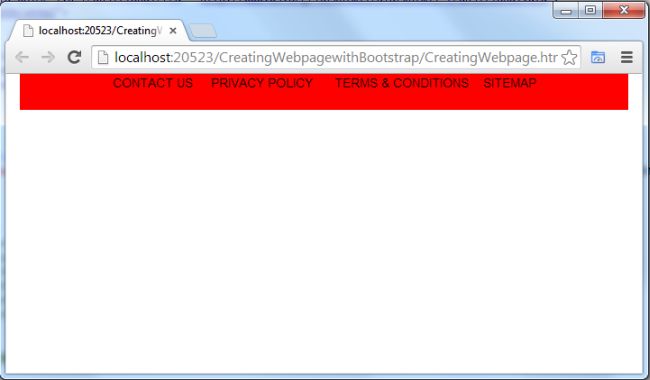
Question 44: How can you create a Footer using Bootstrap?

**Answer**  
  
Using Twitter Bootstrap, you may create static footer menus. To create a footer using Bootstrap open up the bootstrap.css file and check out the following CSS class and update it with the following code.

1. .container
2. {
3. **width**: 100 % ;
4. }
5. .text - muted
6. {
7. **color**: Black;
8. background - **color**: red;
9. **height**: 40 px;
10. text - align: center;
11. }

The HTML file looks as in the following,

1. <div id="footer">
2. <div class="container">
3. <p class="text-muted credit">
4. CONTACT US PRIVACY POLICY TERMS & CONDITIONS SITEMAP</p>
5. </div>
6. lt;/div>

The HTML will be rendered with Bootstrap as in the following,  
  
  
  
For more details visit the following link -

* [Creating a Webpage With Twitter BootStrap](http://www.c-sharpcorner.com/UploadFile/rohatash/creating-a-webpage-with-twitter-bootstrap/)

Question 45: How can you create Rating Control with Bootstrap?

**Answer**  
  
**Step 1**Download the CSS and JavaScript files from github.com.  
  
**Step 2**  
  
Open your Visual Studio, then add your downloaded file into your project then add index.aspx page and call your necessary files within the head tag from that downloaded folder.

1. <head runat="server">
2. <title></title>
3. <link href="Bootstrap/css/bootstrap.min.css" rel="stylesheet" />
4. <link href="Bootstrap/css/star-rating.css" rel="stylesheet" />
5. <script src="http://ajax.googleapis.com/ajax/libs/jquery/2.1.1/jquery.min.js"></script>
6. <script src="Bootstrap/js/star-rating.js"></script>
7. </head>

**Step 3**  
Now call your rating control.

1. <div class="row">
2. <div class="col-lg-12">
4. <input id="input-21a" value="0" type="number" class="rating" data-symbol="\*" min=0 max=5 step=0.5 data-size="xl">
5. <hr>
6. <input id="input-21b" type="number" class="rating" min=0 max=5 step=0.5 data-glyphicon="false" data-star-captions="{}" data-default-caption="{rating} Stars" data-size="lg">
7. <hr>
8. <input id="input-21c" value="0" type="number" class="rating" min=0 max=8 step=0.5 data-size="xl" data-stars="8">
9. <hr>
10. <input id="input-21d" value="2" type="number" class="rating" min=0 max=5 step=0.5 data-size="sm">
11. <hr>
12. <input id="input-21e" value="0" type="number" class="rating" min=0 max=5 step=0.5 data-size="xs">
13. <hr>
14. </div>
15. </div>

Here, value is your filled start on page load, min is the minimum star value, max is the maximum star or maximum rating, data-size is your rating control size.   
  
**Step 4**  
Now we can see how to get the rated value from this rating control using jQuery as in the following,

1. <script>
2. $(document).ready(function()
3. {
4. $("#input-21b").on("rating.change", function(event, value, caption)
5. {
6. alert("You rated: " + value + " = " + $(caption).text());
7. });
8. });
9. </script>

**Step 5**  
Let's see how to get the rated value from code behind in C#. Add one hidden field in your design page and assign your rated value into your hidden field.  
  
*<asp:HiddenField ID="hdfRatingValue" runat="server" />*  
  
Now assign your rated value into your hidden field. Let's see how to do this using jQuery.

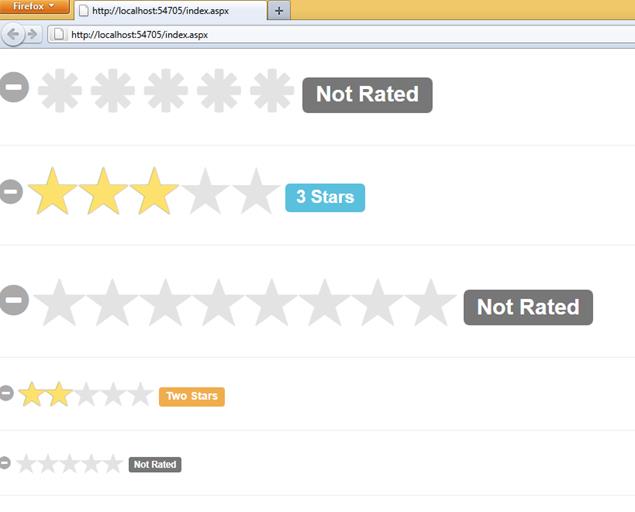
1. <script>
2. $(document).ready(function()
3. {
4. $("#input-21b").on("rating.change", function(event, value, caption)
5. {
7. var ratingValue = $('#<%=hdfRatingValue.ClientID%>').val();
8. ratingValue = value;
9. alert(ratingValue);
10. });
11. });
12. </script>

Now you can get the rating value from code behind. Type hdfRating.value and save your value to the database.  
  
**Full Example**

1. <%@ Page Language="C#" AutoEventWireup="true" CodeBehind="index.aspx.cs" Inherits="RatingControl.index" %>
3. <!DOCTYPE html>
5. <html xmlns="http://www.w3.org/1999/xhtml">
7. <head runat="server">
8. <title></title>
9. <link href="Bootstrap/css/bootstrap.min.css" rel="stylesheet" />
10. <link href="Bootstrap/css/star-rating.css" rel="stylesheet" />
11. <script src="http://ajax.googleapis.com/ajax/libs/jquery/2.1.1/jquery.min.js"></script>
12. <script src="Bootstrap/js/star-rating.js"></script>
14. <script>
15. $(document).ready(function() {
16. $("#input-21b").on("rating.change", function(event, value, caption) {
18. var ratingValue = $('#<%=hdfRatingValue.ClientID%>').val();
19. ratingValue = value;
20. alert(ratingValue);
21. });
22. });
23. </script>
24. </head>
26. <body>
27. <form id="form1" runat="server">
28. <div>
30. <div class="row">
31. <div class="col-lg-12">
33. <input id="input-21a" value="0" type="number" class="rating" data-symbol="\*" min=0 max=5 step=0.5 data-size="xl">
34. <hr>
35. <input id="input-21b" type="number" class="rating" min=0 max=5 step=0.5 data-glyphicon="false" data-star-captions="{}" data-default-caption="{rating} Stars" data-size="lg">
36. <hr>
37. <input id="input-21c" value="0" type="number" class="rating" min=0 max=8 step=0.5 data-size="xl" data-stars="8">
38. <hr>
39. <input id="input-21d" value="2" type="number" class="rating" min=0 max=5 step=0.5 data-size="sm">
40. <hr>
41. <input id="input-21e" value="0" type="number" class="rating" min=0 max=5 step=0.5 data-size="xs">
42. <hr>
43. </div>
44. </div>
46. </div>
47. <asp:HiddenField ID="hdfRatingValue" runat="server" />
48. </form>
49. </body>
51. </html>

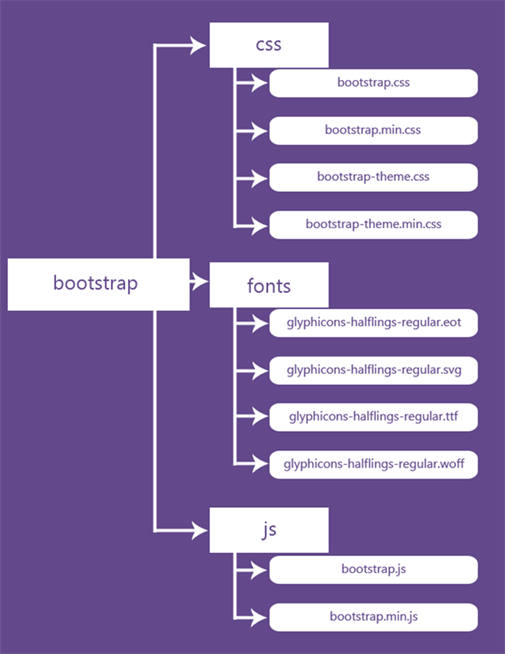
**CS Code**

1. **using** System;
2. **using** System.Collections.Generic;
3. **using** System.Linq;
4. **using** System.Web;
5. **using** System.Web.UI;
6. **using** System.Web.UI.WebControls;
8. **namespace** RatingControl
9. {
10. **public** partial **class** index: System.Web.UI.Page
11. {
12. **protected** **void** Page\_Load(**object** sender, EventArgs e)
13. {
15. }
16. **public** **void** saveRating()
17. {
18. **string** val = hdfRatingValue.Value;
19. }
20. }
21. }

**Output**  
  
For more details visit the following link -

* [How to Create Rating Control in ASP.Net With Bootstrap](http://www.c-sharpcorner.com/UploadFile/7dc621/how-create-rating-control-in-Asp-Net-with-bootstrap/)

Question 46: Explain File Structure of Bootstrap.

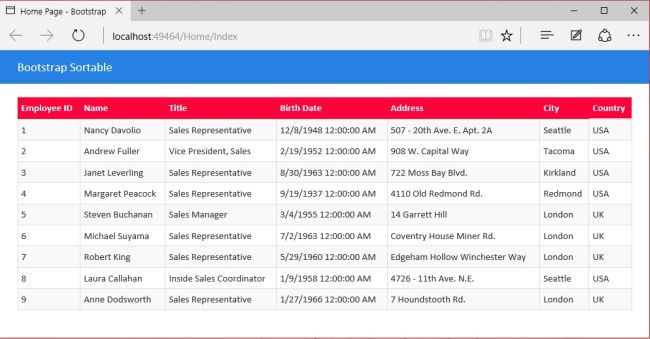
**Answer**  
  
After downloading the Bootstrap files, unzip the compressed folder to see the structure. You'll find the following file structure and contents.  
  
  
  
You can see that the compiled version of Bootstrap provides compiled CSS and JavaScript files (bootstrap.\*), as well as compiled and minified CSS and JavaScript (bootstrap.min.\*).  
  
There are four font files (glyphicons-halflings-regular.\*) inside the fonts folder. These fonts files includes 200 icons from the Glyphicon Halflings set.  
  
For more details visit the following link -

* [Getting Started With Bootstrap](http://www.c-sharpcorner.com/UploadFile/b69ed7/getting-started-with-bootstrap/)

Question 47: Explain Bootstrap Sortable.

**Answer**  
  
Bootstrap Sortable is a jQuery extension for Bootstrap that adds the capability of sorting rows of your Bootstrap tables.  
  
**Features**

* Sorts table data alphabetically.
* Sorts table data by numbers.
* Sorts table data by dates. (Requires Momont.js).
* Allows disabling sorting for a specific table column.

  
  
For more details visit the following link -

* [Bootstrap Sortable In ASP.NET MVC 5](http://www.c-sharpcorner.com/UploadFile/47548d/bootstrap-sortable-in-Asp-Net-mvc-5/)

Question 48: How can we make Customize Buttons?

**Answer**  
  
To make Customize Buttons more attractive than default buttons using Bootstrap, open the bootstrap.css file and check out the following Bootstrap CSS class.

1. btn-default
2. btn-success
3. btn-info
4. btn-warning
5. btn-danger
6. btn-sm

**1. Using Bootstrap CSS class="btn-default"**Now first we use the btn-default clsss. Now open the bootstrap.css file and find the btn-default class. It looks like this,

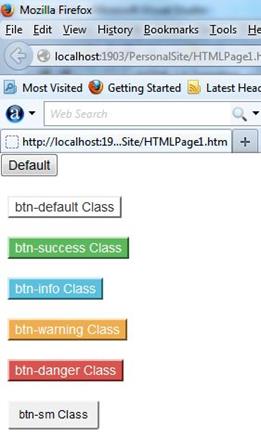
1. .btn-default {
2. **background-color**: #FFFFFF;
3. **border-color**: #CCCCCC;
4. **color**: #333333;

The HTML file looks as in the following,

1. <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
2. <html>
4. <head>
5. <title></title>
6. <script src="Scripts/jquery-1.4.1.min.js" type="text/javascript"></script>
7. <link href="Bootstrap/css/bootstrap.css" rel="stylesheet" type="text/css" />
8. <script src="Bootstrap/js/bootstrap.min.js" type="text/javascript"></script>
9. <script src="Bootstrap/js/bootstrap.min.js" type="text/javascript"></script>
10. <script src="Bootstrap/js/bootstrap.js" type="text/javascript"></script>
11. <style>
13. </style>
14. </head>
16. <body>
17. <button type="button">
18. Default</button>
19. <button type="button" class="btn-default">
20. btn-default Class</button>
21. </body>
23. </html>

The HTML will be rendered with Bootstrap as in the following:  
  
HTML will be rendered with Bootstrap  
  
**2. Using bootstrap CSS classbtn-info, btn-warning, btn-danger, btn-sm**  
The HTML file looks as in the following,

1. <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
2. <html>
4. <head>
5. <title></title>
6. <script src="Scripts/jquery-1.4.1.min.js" type="text/javascript"></script>
7. <link href="Bootstrap/css/bootstrap.css" rel="stylesheet" type="text/css" />
8. <script src="Bootstrap/js/bootstrap.min.js" type="text/javascript"></script>
9. <script src="Bootstrap/js/bootstrap.min.js" type="text/javascript"></script>
10. <script src="Bootstrap/js/bootstrap.js" type="text/javascript"></script>
11. <style>
13. </style>
14. </head>
16. <body>
17. <button type="button">
18. Default</button>
19. <br />
20. <br />
21. <button type="button" class="btn-default">
22. btn-default Class</button>
23. <br />
24. <br />
25. <button type="button" class="btn-success">
26. btn-success Class</button>
27. <br />
28. <br />
29. <button type="button" class="btn-info">
30. btn-info Class</button>
31. <br />
32. <br />
33. <button type="button" class="btn-warning">
34. btn-warning Class</button>
35. <br />
36. <br />
37. <button type="button" class="btn-danger">
38. btn-danger Class</button>
39. <br />
40. <br />
41. <button type="button" class="btn-sm">
42. btn-sm Class</button>
43. </body>
45. </html>

The HTML will be rendered with Bootstrap as in the following,  
  
  
  
For more details visit the following link -

* [Creating Customize Default Buttons Using Bootstrap](http://www.c-sharpcorner.com/UploadFile/rohatash/creating-customize-default-buttons-using-bootstrap/)

Question 49: Explain Alignment Classes in Bootstrap.

**Answer**  
  
Alignment classes are used to align the text.  
  
**Example**

1. <div class="container">
2. <divclassdivclass="row">
3. <div class="col-lg-12 h1 text-center">Alignment Classes</div>
4. </div>
5. <div class="row">
6. <!--Here style for paragraph -->
7. <div class="col-xs-12">
8. <p class="text-left">Left Text</p>
9. <p class="text-center">Centered Text</p>
10. <p class="text-right">Right Text</p>
11. <p class="text-justify">Justified Text</p>
12. <p class="text-nowrap">No Wrap Text</p>
13. </div>
14. </div>
15. </div>

  
  
**Block quotes -** Quoting block of another source content into your content  
  
Use a Footer tag for naming the source,

1. <divclassdivclass="container">
2. <divclassdivclass="row">
3. <divclassdivclass="col-lg-12 h1 text-center">Blockquotes and Footer</div>
5. </div>
6. <divclassdivclass="row">
7. <!--Here style for paragraph -->
8. <divclassdivclass="col-xs-12">
9. <blockquote>Learning Bootstrap Step by Step</blockquote>
10. <footer>- http://getbootstrap.com/css</footer>
11. </div>
13. </div>
14. </div>

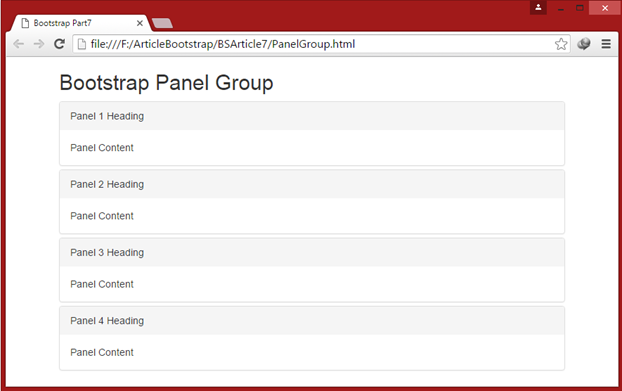
  
For more details visit the following link -

* [Guide To The Basic Pillars Of Bootstrap](http://www.c-sharpcorner.com/article/guide-to-the-basic-pillars-of-bootstrap/)

Question 50: How can you create a panel group?

**Answer**We can group many panels together, by wrapping a <div> with class .panel-group around them. The.panel-group class clears the bottom-margin of each panel.  
  
**Creating Panel Group**  
  
In this example we will create Panels and Group it by using .panel-group class by writing the following code.

1. <DOCTYPE html>
3. <html lang="en">
5. <head>
6. <meta charset="utf-8">
7. <title>Bootstrap Part7</title>
8. <meta name="viewport" content="width=device-width,initial-scale=1">
9. <link rel="stylesheet" type="text/css" href="css/bootstrap.min.css">
10. </head>
12. <body>
13. <div class="container">
14. <h2>Bootstrap Panel Group</h2>
15. <!--Panel Group Start-->
16. <div class="panel-group">
17. <!--Panel 1 Start-->
18. <div class="panel panel-default">
19. <div class="panel-heading">Panel 1 Heading</div>
20. <div class="panel-body">Panel Content</div>
21. </div>
22. <!--Panel 1 End-->
23. <!--Panel 2 Start-->
24. <div class="panel panel-default">
25. <div class="panel-heading">Panel 2 Heading</div>
26. <div class="panel-body">Panel Content</div>
27. </div>
28. <!--Panel 2 End-->
29. <!--Panel 3 Start-->
30. <div class="panel panel-default">
31. <div class="panel-heading">Panel 3 Heading</div>
32. <div class="panel-body">Panel Content</div>
33. </div>
34. <!--Panel 3 End-->
35. <!--Panel 4 Start-->
36. <div class="panel panel-default">
37. <div class="panel-heading">Panel 4 Heading</div>
38. <div class="panel-body">Panel Content</div>
39. </div>
40. <!--Panel 4 End-->
41. </div>
42. <!--Panel Group End-->
43. </div>
44. <script src="js/jquery-2.1.4.min.js"></script>
45. <script src="js/bootstrap.min.js"></script>
46. </body>
48. </html>

**Output**  


What is Twitter Bootstrap?

Bootstrap is a sleek, intuitive, and powerful mobile first front-end framework for faster and easier web development. It uses HTML, CSS and Javascript.

Why use Bootstrap?

Bootstrap can be used as −

* **Mobile first approach** − Since Bootstrap 3, the framework consists of Mobile first styles throughout the entire library instead of in separate files.
* **Browser Support** − It is supported by all popular browsers.



* **Easy to get started** − With just the knowledge of HTML and CSS anyone can get started with Bootstrap. Also the Bootstrap official site has a good documentation.
* **Responsive design** − Bootstrap's responsive CSS adjusts to Desktops,Tablets and Mobiles.
* Provides a clean and uniform solution for building an interface for developers.
* It contains beautiful and functional built-in components which are easy to customize.
* It also provides web based customization.
* And best of all it is an open source.

What does Bootstrap package includes?

Bootstrap package includes −

* **Scaffolding** − Bootstrap provides a basic structure with Grid System, link styles, background. This is is covered in detail in the section **Bootstrap Basic Structure**
* **CSS** − Bootstrap comes with feature of global CSS settings, fundamental HTML elements styled and enhanced with extensible classes, and an advanced grid system. This is covered in detail in the section **Bootstrap with CSS**.
* **Components** − Bootstrap contains over a dozen reusable components built to provide iconography, dropdowns, navigation, alerts, popovers, and much more. This is covered in detail in the section **Layout Components**.
* **JavaScript Plugins** − Bootstrap contains over a dozen custom jQuery plugins. You can easily include them all, or one by one. This is covered in details in the section **Bootstrap Plugins**.
* **Customize** − You can customize Bootstrap's components, LESS variables, and jQuery plugins to get your very own version.

What is Contextual classes of table in Bootstrap?

The Contextual classes allow you to change the background color of your table rows or individual cells.

|  |  |
| --- | --- |
| **Class** | **Description** |
| .active | Applies the hover color to a particular row or cell |
| .success | Indicates a successful or positive action |
| .warning | Indicates a warning that might need attention |
| .danger | Indicates a dangerous or potentially negative action |

What is Bootstrap Grid System?

Bootstrap includes a responsive, mobile first fluid grid system that appropriately scales up to 12 columns as the device or viewport size increases. It includes predefined classes for easy layout options, as well as powerful mixins for generating more semantic layouts.

What are Bootstrap media queries?

Media Queries in Bootstrap allow you to move, show and hide content based on viewport size.

Show a basic grid structure in Bootstrap.

Following is basic structure of Bootstrap grid −

<div class = "container">

<div class = "row">

<div class = "col-\*-\*"></div>

<div class = "col-\*-\*"></div>

</div>

<div class = "row">...</div>

</div>

<div class = "container">....

What are Offset columns?

Offsets are a useful feature for more specialized layouts. They can be used to push columns over for more spacing, for example. The .col-xs = \* classes don't support offsets, but they are easily replicated by using an empty cell.

How can you order columns in Bootstrap?

You can easily change the order of built-in grid columns with .col-md-push-\* and .col-md-pull-\* modifier classes where \* range from 1 to 11.

How do you make images responsive?

Bootstrap 3 allows to make the images responsive by adding a class .img-responsive to the <img> tag. This class applies max-width: 100%; and height: auto; to the image so that it scales nicely to the parent element.

Explain the typography and links in Bootstrap.

Bootstrap sets a basic global display (background), typography, and link styles −

* **Basic Global display** − Sets *background-color: #fff;* on the *<body>*element.
* **Typography** − Uses the *@font-family-base*, *@font-size-base*, and *@line-height-base* attributes as the typographic base
* **Link styles** − Sets the global link color via attribute *@link-color* and apply link underlines only on *:hover*.

What is Normalize in Bootstrap?

Bootstrap uses Normalize to establish cross browser consistency.

Normalize.css is a modern, HTML5-ready alternative to CSS resets. It is a small CSS file that provides better cross-browser consistency in the default styling of HTML elements.

What is Lead Body Copy

To add some emphasis to a paragraph, add class = "lead". This will give you larger font size, lighter weight, and a taller line height

Explain types of lists supported by Bootstrap.

Bootstrap supports ordered lists, unordered lists, and definition lists.

* **Ordered lists** − An ordered list is a list that falls in some sort of sequential order and is prefaced by numbers.
* **Unordered lists** − An unordered list is a list that doesn't have any particular order and is traditionally styled with bullets. If you do not want the bullets to appear then you can remove the styling by using the class *.list-unstyled*. You can also place all list items on a single line using the class *.list-inline*.
* **Definition lists** − In this type of list, each list item can consist of both the <dt> and the <dd> elements. <dt> stands for *definition term*, and like a dictionary, this is the term (or phrase) that is being defined. Subsequently, the <dd> is the definition of the <dt>.

You can make terms and descriptions in <dl> line up side-by-side using class *dl-horizontal*.

What are glyphicons?

Glyphicons are icon fonts which you can use in your web projects. Glyphicons Halflings are not free and require licensing, however their creator has made them available for Bootstrap projects free of cost.

How do you use Glyphicons?

To use the icons, simply use the following code just about anywhere in your code. Leave a space between the icon and text for proper padding.

<span class = "glyphicon glyphicon-search"></span>

What is a transition plugin?

The transition plugin provides simple transition effects such as Sliding or fading in modals.

What is a Modal Plugin?

A modal is a child window that is layered over its parent window. Typically, the purpose is to display content from a separate source that can have some interaction without leaving the parent window. Child windows can provide information, interaction, or more.

How do you use the Dropdown plugin?

You can toggle the dropdown plugin's hidden content −

* **Via data attributes** − Add **data-toggle = "dropdown"** to a link or button to toggle a dropdown as shown below −

<div class = "dropdown">

<a data-toggle = "dropdown" href = "#">Dropdown trigger</a>

<ul class = "dropdown-menu" role = "menu" aria-labelledby = "dLabel">

...

</ul>

</div>

* If you need to keep links intact (which is useful if the browser is not enabling JavaScript), use the **data-target** attribute instead of **href="#"** −

<div class = "dropdown">

<a id = "dLabel" role = "button" data-toggle = "dropdown" data-target = "#" href = "/page.html">

Dropdown

<span class = "caret"></span>

</a>

<ul class = "dropdown-menu" role = "menu" aria-labelledby = "dLabel">

...

</ul>

</div>

* **Via JavaScript** − To call the dropdown toggle via JavaScript, use the following method −

$('.dropdown-toggle').dropdown()

What is Bootstrap caraousel?

The Bootstrap carousel is a flexible, responsive way to add a slider to your site. In addition to being responsive, the content is flexible enough to allow images, iframes, videos, or just about any type of content that you might want.

What is button group

Button groups allow multiple buttons to be stacked together on a single line. This is useful when you want to place items like alignment buttons together.

Which class is used for basic button group

.btn-group class is used for a basic button group. Wrap a series of buttons with class .btn in .btn-group.

Which class is used to draw a toolbar of buttons

.btn-toolbar helps to combine sets of <div class = "btn-group"> into a <div class = "btn-toolbar"> for more complex components.

Which classes can be applied to button group instead of resizing each button

.btn-group-lg, .btn-group-sm, .btn-group-xs classes can be applied to button group instead of resizing each button.

Which class make a set of buttons appear vertically stacked rather than horizontally

.btn-group-vertical class make a set of buttons appear vertically stacked rather than horizontally.

What are input groups

Input groups are extended [Form Controls](https://www.tutorialspoint.com/bootstrap/bootstrap_forms.htm). Using input groups you can easily prepend and append text or buttons to the text-based inputs.

By adding prepended and appended content to an input field, you can add common elements to the user's input. For example, you can add the dollar symbol, the @ for a Twitter username, or anything else that might be common for your application interface.

To prepend or append elements to a **.form-control** −

* Wrap it in a <div> with class **.input-group**
* As a next step, within that same <div> , place your extra content inside a <span> with class **.input-group-addon**.
* Now place this <span> either before or after the <input> element.

How will you create a tabbed navigation menu

To create a tabbed navigation menu −

* Start with a basic unordered list with the base class of .nav.
* Add class .nav-tabs.

How will you create a pills navigation menu

To create a pills navigation menu −

* Start with a basic unordered list with the base class of .nav.
* Add class .nav-pills.

How will you create a vertical pills navigation menu

You can stack the pills vertically using the class .nav-stacked along with the classes: .nav, .nav-pills.

What is bootstrap navbar

The navbar is one of the prominent features of Bootstrap sites. Navbars are responsive 'meta' components that serve as navigation headers for your application or site. Navbars collapse in mobile views and become horizontal as the available viewport width increases. At its core, the navbar includes styling for site names and basic navigation.

How to create a navbar in bootstrap

To create a default navbar −

* Add the classes .navbar, .navbar-default to the <nav> tag.
* Add role = "navigation" to the above element, to help with accessibility.
* Add a header class .navbar-header to the <div> element. Include an <a> element with class navbar-brand. This will give the text a slightly larger size.
* To add links to the navbar, simply add an unordered list with the classes of .nav, .navbar-nav.

What is bootstrap breadcrumb

Breadcrumbs are a great way to show hierarchy-based information for a site. In the case of blogs, breadcrumbs can show the dates of publishing, categories, or tags. They indicate the current page's location within a navigational hierarchy.

A breadcrumb in Bootstrap is simply an unordered list with a class of .breadcrumb. The separator is automatically added by CSS (bootstrap.min.css).

Which class is used for basic pagination

.pagination class is uesed to add the pagination on a page.

How will you customize links of pagination

You can customize links by using .disabled for unclickable links and .active to indicate the current page.

What are bootstrap labels

Bootstrap labels are great for offering counts, tips, or other markup for pages. Use class .label to display labels.

What are bootstrap badges

Badges are similar to labels; the primary difference is that the corners are more rounded. Badges are mainly used to highlight new or unread items. To use badges just add <span class = "badge"> to links, Bootstrap navs, and more.

What is Bootstrap Jumbotron

As the name suggest this component can optionally increase the size of headings and add a lot of margin for landing page content. To use the Jumbotron −

* Create a container <div> with the class of .jumbotron.
* In addition to a larger <h1>, the font-weight is reduced to 200px.

What is Bootstrap page header

The page header is a nice little feature to add appropriate spacing around the headings on a page. This is particularly helpful on a web page where you may have several post titles and need a way to add distinction to each of them. To use a page header, wrap your heading in a <div> with a class of .page-header.

How to create thumbnails using Bootstrap

To create thumbnails using Bootstrap −

* Add an <a> tag with the class of .thumbnail around an image.
* This adds four pixels of padding and a gray border.
* On hover, an animated glow outlines the image.

How to customize thumbnails using Bootstrap

it's possible to add any kind of HTML content like headings, paragraphs, or buttons into thumbnails. Follow the steps below −

* Change the <a> tag that has a class of .thumbnail to a <div>.
* Inside of that <div>, you can add anything you need. As this is a <div>, we can use the default span-based naming convention for sizing.
* If you want to group multiple images, place them in an unordered list, and each list item will be floated to the left.

What are bootstrap alerts?

Bootstrap Alerts provide a way to style messages to the user. They provide contextual feedback messages for typical user actions.

You can add an optional close icon to alert.

How will you create a bootstrap alert?

You can add a basic alert by creating a wrapper <div> and adding a class of .alert and one of the four contextual classes (e.g., .alert-success, .alert-info, .alert-warning, .alert-danger).

How will you create a Bootstrap Dismissal Alert?

To build a dismissal alert −

* Add a basic alert by creating a wrapper <div> and adding a class of .alert and one of the four contextual classes (e.g., .alert-success, .alert-info, .alert-warning, .alert-danger).
* Also add optional .alert-dismissable to the above <div> class.
* Add a close button.
* Use the <button> element with the data-dismiss = "alert" data attribute.

How will you create a progress bar using bootstrap?

To create a basic progress bar −

* Add a <div> with a class of .progress.
* Next, inside the above <div>, add an empty <div> with a class of .progress-bar.
* Add a style attribute with the width expressed as a percentage. Say for example, style = "60%"; indicates that the progress bar was at 60%.

How will you create a alternate progress bar using bootstrap?

To create a progress bar with different styles −

* Add a <div> with a class of .progress.
* Next, inside the above <div>, add an empty <div> with a class of .progress-bar and class progress-bar-\* where \* could be success, info, warning, danger.
* Add a style attribute with the width expressed as a percentage. Say for example, style = "60%"; indicates that the progress bar was at 60%.

How will you create a striped progress bar using bootstrap

To create a striped progress bar −

* Add a <div> with a class of .progress and .progress-striped.
* Next, inside the above <div>, add an empty <div> with a class of .progress-bar and class progress-bar-\* where \* could be success, info, warning, danger.
* Add a style attribute with the width expressed as a percentage. Say for example, style = "60%"; indicates that the progress bar was at 60%.

How will you create a animated progress bar using bootstrap?

To create an animated progress bar −

* Add a <div> with a class of .progress and .progress-striped. Also add class .active to .progress-striped.
* Next, inside the above <div>, add an empty <div> with a class of .progress-bar.
* Add a style attribute with the width expressed as a percentage. Say for example, style = "60%"; indicates that the progress bar was at 60%.

How will you create a stacked progress bar using bootstrap

You can even stack multiple progress bars. Place the multiple progress bars into the same .progress to stack them.

What are bootstrap media objects

These are abstract object styles for building various types of components (like blog comments, Tweets, etc.) that feature a left-aligned or right-aligned image alongside the textual content. The goal of the media object is to make the code for developing these blocks of information drastically shorter.

The goal of media objects (light markup, easy extendability) is achieved by applying classes to some of the simple markup.

What is the purpose of .media class in bootstrap?

This class allows to float a media object (images, video, and audio) to the left or right of a content block.

What is the purpose of .media-list class in bootstrap

If you are preparing a list where the items will be part of an unordered list, use a class. useful for comment threads or articles lists.

What are bootstrap panels

Panel components are used when you want to put your DOM component in a box. To get a basic panel, just add class .panel to the <div> element. Also add class .panel-default to this element.

How will you create a bootstrap panel with heading

here are two ways to add panel heading −

* Use .panel-heading class to easily add a heading container to your panel.
* Use any <h1>-<h6> with a .panel-title class to add a pre-styled heading.

How will you create a bootstrap panel with footer

You can add footers to panels, by wrapping buttons or secondary text in a <div> containing class .panel-footer.

What contextual classes are available to style the panels

Use contextual state classes such as, panel-primary, panel-success, panel-info, panel-warning, panel-danger, to make a panel more meaningful to a particular context.

Can you put a table within bootstrap panel

Yes! To get a non-bordered table within a panel, use the class .table within the panel. Suppose there is a <div> containing .panel-body, we add an extra border to the top of the table for separation. If there is no <div> containing .panel-body, then the component moves from panel header to table without interruption.

Can you put a listgroup within bootstrap panel

Yes! You can include list groups within any panel. Create a panel by adding class .panel to the <div> element. Also add class .panel-default to this element. Now within this panel include your list groups.

What is bootstrap well

A well is a container in <div> that causes the content to appear sunken or an inset effect on the page. To create a well, simply wrap the content that you would like to appear in the well with a <div> containing the class of .well.

What is Scrollspy plugin

The Scrollspy (auto updating nav) plugin allows you to target sections of the page based on the scroll position. In its basic implementation, as you scroll, you can add .active classes to the navbar based on the scroll position.

What is affix plugin

The affix plugin allows a <div> to become affixed to a location on the page. You can also toggle it's pinning on and off using this plugin. A common example of this are social icons. They will start in a location, but as the page hits a certain mark, the <div> will be locked in place and will stop scrolling with the rest of the page.

**Part 1 – Bootstrap Interview Questions (Basic)**

This first part covers basic Bootstrap Interview Questions and Answers

**1. Why use Bootstrap?**

Answer:

* It has the support of all the web browsers like Internet Explorer, Google Chrome, Firefox, Opera, Safari etc.
* It is a very powerful mobile first front-end framework, Mobile first styles are installed with a bootstrap3 framework.
* It is very easy to start as one needs to have an idea of [HTML and CSS](https://www.educba.com/html-vs-css/) only to work with it, prior knowledge in HTML and CSS is fairly sufficient to start with Bootstrap.   Bootstrap official site also holds a good documentation.
* Bootstrap supports the build of highly reactive websites.
* The customisation can be easily done with the presence of built-in components.
* Allows a various variety of templates for use.
* Bootstrap is an open source framework.

**2. Components of Bootstrap package?**

Answer:  
This is the basic Bootstrap Interview Questions which is asked in an interview. Following is the components of Bootstrap are as follows:

**Scaffolding**: Grid System, link styles, background form the basic structure of bootstrap.

 Popular Course in this category

[**Bootstrap Training Course**5 Online Courses | 22+ Hours | Verifiable Certificate of Completion | Lifetime Access   
4.5 (1,352 ratings)](https://www.educba.com/software-development/courses/bootstrap-training-course/?btnz=edu-blg-inline-banner1)

[Course Price   
₹4999 ~~₹12999~~   
**View Course**](https://www.educba.com/software-development/courses/bootstrap-training-course/?btnz=edu-blg-inline-banner1)

Related Courses

[Programming Languages Course](https://www.educba.com/software-development/courses/programming-languages-course/?btnz=edu-blg-inline-banner1)[C Programming Course](https://www.educba.com/software-development/courses/c-programming-course/?btnz=edu-blg-inline-banner1)[Selenium Training Certification](https://www.educba.com/software-development/courses/selenium-training-certification/?btnz=edu-blg-inline-banner1)

**CSS:** Bootstrap withholds global CSS settings, HTML, extensible classes, and advanced grid system.

 **JavaScript Plug-in:** Bootstrap contains over a dozen custom jQuery plugins.

**Customize**:   Bootstrap’s components can be easily customized with jQuery plugins to get your very own version.

Let us move to the next Bootstrap Interview Questions And Answer.

**3. Foundation vs Bootstrap, key differences?**

Answer:

|  |  |
| --- | --- |
| **Bootstrap** | **Foundation** |
| – Numerous User interface components are supported by Bootstrap | – In Foundation UI element options are very limited in numbers |
| – Bootstraps uses pixels | – Foundation use REMs |
| – Both Desktop and mobile sites can be designed primarily with bootstrap | – Foundation encourages to design mobile first |
| – Bootstrap support LESS as its preprocessor | – Foundation support [Sass](https://www.educba.com/sass-interview-questions/) and Compass as its preprocessor |

**4. Explain the Bootstrap framework structure?**

Answer:  
Bootstrap is a responsive front-end [mobile development framework](https://www.educba.com/mobile-apps-development-tools/). Sites can be effectively created with the grid system of the Bootstrap framework. its holds a highly responsive design and very much customizable, Mobile sites can be built on top of the Bootstrap framework using below  programming entities,

* HTML5 markup
* CSS Styling
* Javascript
* Fonts

Here on mobile measurements content can be divided into 12 columns, then it can further be reduced to 6,4,3, etc. Ease of style and making define the Bootstrap buttons with default classes available for images as well.

**5. What are Breadcrumbs in bootstrap?**

Answer:  
Breadcrumbs depict the navigational hierarchy of a site, separators in the hierarchy are automatically added by means of CSS.

Bootstrap breadcrumb used for,

* Documentation page
* Magazine
* ERP system

A breadcrumb in Bootstrap is simply an unordered list with a class of .breadcrumb.

**Ex:**

*.breadcrumb > li + li:before {*

*color: #CCCCCC;*

*content: “/ “;*

*padding: 0 5px;*

*}*

**Part 2 –Bootstrap Interview Questions (Advanced)**

Let us now have a look at the advanced Bootstrap Interview Questions.

**6. Why do we use Jumbotron in Bootstrap?**

Answer:  
Jumbotron is a user-attractive function of Bootstrap. it is a brilliant way to highlight the key contents of a page. the content could be a slogan or probably a headline or any other catchy element of the page. Jumbotron works by increasing the content of the landing page and also increasing the heading size of the content. In order to implement Jumbotron in a Bootstrap use:

**Create a container <div> with the class of. Jumbotron**

***Ex:*** The example shows wrapping up of heading contents inside a div class Jumbotron container.

*<div class=”jumbotron”>*

*<h1>Learn to Create Websites</h1>*

*<p>In today’s world internet is the most popular way…</p>*

*<p><a href=”#” class=”btn btn-primary btn-lg”>Learn more</a></p>*

*</div>*

**7. Explain types of lists supported by Bootstrap?**

Answer:  
Lists supported by Bootstrap are as following,

**Definition lists −**  Evey list is by default introduced with a <dt> and <dd> tag ement

* <dt> : definition term
* <dd> : corresponds to the defenition of <dt>

**Ordered lists** − lies in a sequential order and prefixed by numerical values.

**Unordered lists −**  styled traditionally, does not follow any specific order. bullets can be easily customized.  class, list-inline can be used to hold all list items within a single line.

Let us move to the next Bootstrap Interview Questions And Answer.

**8. Navigation bar in Bootstrap?**

Answer:  
This is the advanced Bootstrap Interview Questions which is asked in an interview. It resembles the navigation header in the top of the page, the navigation bar can be created with Nav tag. Navbars and their contents are fluid by default. The horizontal width can be scaled and controlled using Navigation containers. Navbars are concealed by default while printing.

**Ex:**  *<nav class=”navbar navbar-expand-lg navbar-light bg-light”>*

*<div class=”dropdown-menu” aria-labelledby=”navbarDropdown”>*

*<a class=”dropdown-item” href=”#”>Action</a>*

*<a class=”dropdown-item” href=”#”>Another action</a>*

*<div class=”dropdown-divider”></div>*

*<a class=”dropdown-item” href=”#”>Something else here</a>*

*</div>*

*</nav>*

**9. Would you tell me something about Glyphicons and how are they written in our code?**

Answer:  
Glyphicons are symbols or definitive icon fonts which provide easy orientation in your web projects. they are preferably addended in buttons, form inputs, toolbars and navigation components.  Glyphicons Halflings are not free to use entities and requires licensing. [Bootstrap includes more than 250 glyphs](https://www.educba.com/install-bootstrap/) from the Glyphicons Halflings set.

**Ex:** <p>Search icon: <span class=”glyphicon glyphicon-search”></span></p>

**10. Describe a Bootstrap panel , Bootstrap well and Bootstrap carousel?**

Answer:  
Bootstrap well: Adds a greyish rounded background around the content, it is a container class. The .well-sm class for small wells and .well-lg class for large wells allows alternate the size of the well. Default well size is normal.

**Ex**: *<div class=”well”>Basic Well</div>*

**Bootstrap panel:** There could be situations where some site contents need to be wrapped up in a box to enhance the visibility for the user. Bootstrap panels lock hands on these needs. They can be created with the .panel class and content inside the panel has a .panel-body class.

**Ex :**    *<div class=”panel panel-default”>*

*<div class=”panel-body”>Look, I’m in a panel!</div>*

*</div>*

**Bootstrap carousel:**  Adds a slider through which images, videos, and contents can be populated and sided for the users. It is a flexible, handy and responsive way to add a carousel to our site. Carousel plug-in is a component which is used for the same.

**Bootstrap Interview Questions and Answers**

**What is Twitter Bootstrap?**

Bootstrap is a sleek, intuitive, and powerful mobile first front-end framework for faster and easier web development. It uses HTML, CSS, and [Javascript training Bangalore](https://tekslate.com/javascript-training/).

**What Bootstrap Package Includes?**

**Scaffolding**: Bootstrap provides a basic structure with Grid System, link styles, background. This is is covered in detail in the section Bootstrap Basic Structure

**CSS**: Bootstrap comes with the feature of global CSS settings, fundamental HTML elements styled and enhanced with extensible classes, and an advanced grid system. This is covered in detail in the section Bootstrap with CSS.

**Components:** Bootstrap contains over a dozen reusable components built to provide iconography, dropdowns, navigation, alerts, popovers, and much more. This is covered in detail in the section Layout Components.

**JavaScript Plugins:** Bootstrap contains over a dozen custom [jQuery](https://en.wikipedia.org/wiki/JQuery) plugins. You can easily include them all, or one by one. This is covered in details in the section Bootstrap Plugins.

**Customize:** You can customize Bootstrap's components, LESS variables, and jQuery plugins to get your very own version.

**What is Bootstrap Grid System?**

Bootstrap includes a responsive, mobile first fluid grid system that appropriately scales up to 12 columns as the device or viewport size increases. It includes predefined classes for easy layout options, as well as powerful mixins for generating more semantic layouts.

**What are Glyphicons?**

Glyphicons are icon fonts which you can use in your web projects. Glyphicons Halflings are not free and require licensing, however their creator has made them available for Bootstrap projects free of cost.

**Define Bootstrap Breadcrumb?**

Breadcrumbs are a great way to show hierarchy-based information for a site. In the case of blogs, breadcrumbs can show the dates of publishing, categories, or tags. They indicate the current page's location within a navigational hierarchy.

A breadcrumb in Bootstrap is simply an unordered list with a class of .breadcrumb. The separator is automatically added by CSS (bootstrap.min.css) through the following class:

.breadcrumb > li + li:before {

color: #CCCCCC;

content: "/ ";

padding: 0 5px;

}

**Define  Bootstrap thumbnails?**

A lot of sites need a way to lay out images, videos,text, etc in a grid, and Bootstrap has an easy way to do this with thumbnails.

To create thumbnails using Bootstrap:

* Add an tag with the class of .thumbnail around an image.
* This adds four pixels of padding and a gray border.
* On hover, an animated glow outlines the image.

**What are Bootstrap Plugins?**

Bootstrap comes bundled with 12 jQuery plugins that extend the features and can add more interaction to your site. To get started with Bootstrap’s JavaScript plugins, you don’t need to be an advanced JavaScript developer. By utilizing Bootstrap Data API, most of the plugins can be triggered without writing a single line of code.

**Define UI Bootstrap?**

UI Bootstrap incorporates Bootstrap components written in AngularJS by the AngularUI team. It aims to provide AngularJS directives with the markup and CSS of Bootstrap. It has dependencies on AngularJS and Bootstrap CSS.

***Aspired to become an Bootstrap?***

***Explore the post to discover the know-hows on*** [***Bootstrap Training***](https://tekslate.com/bootstrap-training)***.***

**Define Kickstrap?**

Kickstrap is a product where AngularJS is baked together with Bootstrap in conjunction with JavaScript Package Manager (JSPM). You can leverage the advantage of running an authenticated database-driven web application without the native backend. Kickstrap uses Firebase adhering to a Backend as a Service (BaaS) model.

**What are different button styles in Bootstrap?**

There are seven styles to add a button in Bootstrap. Use the following classes to achieve the different button styles:

* .btn-default
* .btn-primary
* .btn-success
* .btn-info
* .btn-warning
* .btn-danger
* .btn-link

**What are Bootstrap Form Rules?**

* Always use <form role="form"> (helps improve accessibility for people using screen readers)
* Wrap labels and form controls in <div class="form-group"> (needed for optimum spacing)
* Add class .form-control to all textual <input>, <textarea>, and <select> elements

**What are Bootstrap Alerts ?**

Bootstrap Alerts are used to provide an easy way to create predefined alert messages. Alert adds a style to your messages to make it more appealing to the users.

There are four classes that are used within <div> element for alerts.

* .alert-success
* .alert-info
* .alert-warning
* .alert-danger

**What is Bootstrap pager ?**

The bootstrap pager is a form of pagination. It is used to create previous and next buttons (links).The **".pager"** class is used within the <ul> element to create the previous / next buttons.

**What is the role of pagination in bootstrap and what are their classifications?**

When an unordered list has to be handled by bootstrap, it uses pagination. To handle the list,                      different classes are available such as ‘.pagination’, ‘.disabled’, ‘.active’, ‘.pagination-lg’. The function         ‘.pagination’ to paginate the webpage. The function ‘.disabled’ is used to make the links unclickable.          The function ‘.active’ is used for indicating the current webpage. The function ‘.pagination-lg’ and              ‘.pagination-sm’ can be used for getting different size items.

What is the procedure to create Nav elements in Bootstrap?

There are several styling navigation elements available on bootstrap and every style uses the same             function i.e. ‘class .nav’. In order to create tabs or a tabular navigation, you can begin with a simple          or rather basic unordered list using the function class .nav. To add the tabs the function ‘class .nav-             tabs can be used.

1. **Question 1. What Do You Understand By Bootstrap?**

**Answer :**

Bootstrap, originally named Twitter Blueprint, was developed by Mark Otto and Jacob Thornton at Twitter, is free and open source front end Javascript web framework for designing the rich, sleek, intuitive web applications with minimal effort. This framework is more useful for building mobile based web applications. It uses HTML, Javascript and CSS based design templates and other interface components. It also has optional JavaScript extensions.

1. **Question 2. Why Would You Use Bootstrap? Tell Me About Its Benefits?**

**Answer :**

**The major benefits of using Bootstrap are-**

* + It is a free and open-source Web Designing Framework
  + It has support of all the web browsers like Internet Explorer, Google Chrome, Firefox, Opera, Safari etc.
  + It is a very powerful mobile first front-end framework
  + It is very easy to start as one needs to have idea of HTML and CSS only to work with it
  + We can design responsive website through it which adjust to desktop, tablet and mobile
  + It comprises functional built-in components which are easy to customize

[Python Interview Questions](https://www.wisdomjobs.com/e-university/python-interview-questions.html)

1. **Question 3. Who Developed The Bootstrap?**

**Answer :**

Mark Otto and Jacob Thornton at Twitter

1. **Question 4. Which Components Are There In Bootstrap?**

**Answer :**

**Main components of Bootstrap are-**

**Reusable Components :** It contains plenty of reusable layout components

**CSS :** It contains lots of CSS files

**jQuery & JavaScript Plugins:** It comprises many jQuery and JavaScript plugins

**Scaffolding :** It comes with a basic structure with link styles, grid system and background

**Customization:** You have a freedom to customize components as per your wish

[Python Tutorial](https://www.wisdomjobs.com/e-university/python-tutorial-176.html)

1. **Question 5. Tell Me About Different Layouts Available In Bootstrap?**

**Answer :**

**Two layouts are available in Bootstrap-**

* + **Fluid Layout:**Its design automatic adjusts to the browser size.
  + **Fixed Layout:**Its design doesn't adjust automatically to different browsers but it can be responsive.

[HTML 4 Interview Questions](https://www.wisdomjobs.com/e-university/html-4-interview-questions.html)

* **Question 6. Could You Explain Me About Responsive Layout?**

**Answer :**

Responsive layout's design automatic adjusts to the browser size but when resizing, the number of columns changes according to the space available.

* **Question 7. Would You Differentiate Between Fluid Layout & Responsive Layout?**

**Answer :**

Yes, sure. The main difference is that Fluid Layout are built using widths percentages and they are based on proportionally laying out the website so elements take up the same percent of space on different screen sizes. Responsive Layout is built on a fluid grid and here CSS Media Queries are used to present different layouts based on different screen sizes.

[HTML 4 Tutorial](https://www.wisdomjobs.com/e-university/html-4-tutorial-182.html) [HTML Interview Questions](https://www.wisdomjobs.com/e-university/html-interview-questions.html)

* **Question 8. What Do You Understand By A List Group In Bootstrap And Its Use?**

**Answer :**

List group is a powerful and flexible component to display simple and complex elements together with custom contents. A simple list group is created using class .list-group to address the list with usage of <ul> element, and class .list-group-item to address individual item with usage of <li> elements.

* **Question 9. Would You Tell Me Something About Glyphicons And How Are They Written In Our Code?**

**Answer :**

Glyphicons are symbols and icon fonts which are created for providing simplicity and easy orientation in your web projects. Glyphicons Halflings are normally not available for free use and they require licensing but their creator has made them cost free for Bootstrap projects. Bootstrap includes more than 250 glyphs from the Glyphicon Halflings set. To use the icons, we need to write <span class = "glyphicon glyphicon-search"></span> code anywhere in our program.

[HTML 5 Interview Questions](https://www.wisdomjobs.com/e-university/html-5-interview-questions.html)

* **Question 10. How Would You Define Bootstrap Pagination?**

**Answer :**

Whenever a web site has lots of pages, we need to do pagination to each page. In Bootstrap, we can create a basic pagination by adding the .pagination class to an <ul> element.

[HTML Tutorial](https://www.wisdomjobs.com/e-university/html-tutorial-206.html)

* **Question 11. Describe Bootstrap Panel?**

**Answer :**

Whenever there is a need of putting the contents in a bordered box with some padding around, panel components are used. They can be created with the .panel class and content inside the panel has a .panel-body class. For creating a basic panel, add class .panel to the <div> element and add class .panel-default to this element.

[Web Designing Interview Questions](https://www.wisdomjobs.com/e-university/web-designing-interview-questions.html)

* **Question 12. What Is Normalize In Bootstrap?**

**Answer :**

Bootstrap uses Normalize to establish cross browser consistency.

Normalize.css is a modern, HTML5-ready alternative to CSS resets. It is a small CSS file that provides better cross-browser consistency in the default styling of HTML elements.

[Python Interview Questions](https://www.wisdomjobs.com/e-university/python-practice-tests-176-326595)

* **Question 13. What Is Lead Body Copy?**

**Answer :**

To add some emphasis to a paragraph, add class = "lead". This will give you larger font size, lighter weight, and a taller line height.

[HTML 5 Tutorial](https://www.wisdomjobs.com/e-university/html-5-tutorial-207.html)

* **Question 14. Explain Types Of Lists Supported By Bootstrap?**

**Answer :**

Bootstrap supports ordered lists, unordered lists, and definition lists.

**Ordered lists :** An ordered list is a list that falls in some sort of sequential order and is prefaced by numbers.

**Unordered lists :** An unordered list is a list that doesn't have any particular order and is traditionally styled with bullets. If you do not want the bullets to appear then you can remove the styling by using the class .list-unstyled. You can also place all list items on a single line using the class .list-inline.

**Definition lists :** In this type of list, each list item can consist of both the <dt> and the <dd> elements. <dt> stands for definition term, and like a dictionary, this is the term (or phrase) that is being defined. Subsequently, the <dd> is the definition of the <dt>.

You can make terms and descriptions in <dl> line up side-by-side using class dl-horizontal.

* **Question 15. What Are Glyphicons?**

**Answer :**

Glyphicons are icon fonts which you can use in your web projects. Glyphicons Halflings are not free and require licensing, however their creator has made them available for Bootstrap projects free of cost.

[Java Script Interview Questions](https://www.wisdomjobs.com/e-university/java-script-interview-questions.html)

* **Question 16. What Is A Transition Plugin?**

**Answer :**

The transition plugin provides simple transition effects such as Sliding or fading in modals.

[Web Designing Tutorial](https://www.wisdomjobs.com/e-university/web-designing-tutorial-208.html)

* **Question 17. What Is A Modal Plugin?**

**Answer :**

A modal is a child window that is layered over its parent window. Typically, the purpose is to display content from a separate source that can have some interaction without leaving the parent window. Child windows can provide information, interaction, or more.

[PHP Interview Questions](https://www.wisdomjobs.com/e-university/php-interview-questions.html)

* **Question 18. What Is Bootstrap Carousel?**

**Answer :**

The Bootstrap carousel is a flexible, responsive way to add a slider to your site. In addition to being responsive, the content is flexible enough to allow images, iframes, videos, or just about any type of content that you might want.

[HTML 4 Interview Questions](https://www.wisdomjobs.com/e-university/html-4-interview-questions.html)

* **Question 19. What Is Button Group?**

**Answer :**

Button groups allow multiple buttons to be stacked together on a single line. This is useful when you want to place items like alignment buttons together

[Java Script Tutorial](https://www.wisdomjobs.com/e-university/java-script-tutorial-209.html)

* **Question 20. Which Class Is Used For Basic Button Group?**

**Answer :**

.btn-group class is used for a basic button group. Wrap a series of buttons with class .btn in .btn-group.

[. NET Interview Questions](https://www.wisdomjobs.com/e-university/dot-net-interview-questions.html)

* **Question 21. Which Class Is Used To Draw A Toolbar Of Buttons?**

**Answer :**

.btn-toolbar helps to combine sets of <div class = "btn-group"> into a <div class = "btn-toolbar"> for more complex components.

* **Question 22. Which Classes Can Be Applied To Button Group Instead Of Resizing Each Button?**

**Answer :**

.btn-group-lg, .btn-group-sm, .btn-group-xs classes can be applied to button group instead of resizing each button.

[PHP Tutorial](https://www.wisdomjobs.com/e-university/php-tutorial-223.html)

* **Question 23. What Is Twitter Bootstrap?**

**Answer :**

Bootstrap is a sleek, intuitive, and powerful mobile first front-end framework for faster and easier web development. It uses HTML, CSS and Javascript.

[ASP.NET Interview Questions](https://www.wisdomjobs.com/e-university/asp-dot-net-interview-questions.html)

* **Question 24. What Does Bootstrap Package Includes?**

**Answer :**

**Bootstrap package includes:**

* + **Scaffolding** − Bootstrap provides a basic structure with Grid System, link styles, background. This is is covered in detail in the sectionBootstrap Basic Structure
  + **CSS** − Bootstrap comes with feature of global CSS settings, fundamental HTML elements styled and enhanced with extensible classes, and an advanced grid system. This is covered in detail in the section Bootstrap with CSS.
  + **Components** − Bootstrap contains over a dozen reusable components built to provide iconography, dropdowns, navigation, alerts, popovers, and much more. This is covered in detail in the section Layout Components.
  + **JavaScript Plugins** − Bootstrap contains over a dozen custom jQuery plugins. You can easily include them all, or one by one. This is covered in details in the section Bootstrap Plugins.
  + **Customize** − You can customize Bootstrap's components, LESS variables, and jQuery plugins to get your very own version.

[HTML Interview Questions](https://www.wisdomjobs.com/e-university/html-interview-questions.html)

* **Question 25. What Is Bootstrap Grid System?**

**Answer :**

Bootstrap includes a responsive, mobile first fluid grid system that appropriately scales up to 12 columns as the device or viewport size increases. It includes predefined classes for easy layout options, as well as powerful mixins for generating more semantic layouts.

[ASP.NET Tutorial](https://www.wisdomjobs.com/e-university/asp-dot-net-tutorial-263.html)

* **Question 26. What Are Bootstrap Media Queries?**

**Answer :**

Media Queries in Bootstrap allow you to move, show and hide content based on viewport size.  
Show a basic grid structure in Bootstrap.  
**Following is basic structure of Bootstrap grid:**  
<div class = "container">  
   <div class = "row">  
      <div class = "col-\*-\*"></div>  
      <div class = "col-\*-\*"></div>      
   </div>  
   <div class = "row">...</div>  
</div>  
<div class = "container">....

[Web Developer Guide Interview Questions](https://www.wisdomjobs.com/e-university/web-developer-guide-interview-questions.html)

* **Question 27. What Are Offset Columns?**

**Answer :**

Offsets are a useful feature for more specialized layouts. They can be used to push columns over for more spacing, for example. The .col-xs = \* classes don't support offsets, but they are easily replicated by using an empty cell.

[HTML 5 Interview Questions](https://www.wisdomjobs.com/e-university/html-5-interview-questions.html)

* **Question 28. How Can You Order Columns In Bootstrap?**

**Answer :**

You can easily change the order of built-in grid columns with .col-md-push-\* and .col-md-pull-\* modifier classes where \* range from 1 to 11.

[Bootstrap Tutorial](https://www.wisdomjobs.com/e-university/bootstrap-tutorial-1185.html)

* **Question 29. How Do You Make Images Responsive?**

**Answer :**

Bootstrap 3 allows to make the images responsive by adding a class .img-responsive to the <img> tag. This class applies max-width: 100%; and height: auto; to the image so that it scales nicely to the parent element.

[WordPress Interview Questions](https://www.wisdomjobs.com/e-university/wordpress-interview-questions.html)

* **Question 30. Explain The Typography And Links In Bootstrap?**

**Answer :**

**Bootstrap sets a basic global display (background), typography, and link styles:**  
**Basic Global display** − Sets background-color: #fff; on the <body> element.  
**Typography** − Uses the @font-family-base, @font-size-base, and @line-height-base attributes as the typographic base  
**Link styles** − Sets the global link color via attribute @link-color and apply link underlines only on :hover.

* **Question 31. How Will You Create A Bootstrap Panel With Footer?**

**Answer :**

You can add footers to panels, by wrapping buttons or secondary text in a <div> containing class .panel-footer.

[Web Developer Guide Tutorial](https://www.wisdomjobs.com/e-university/web-developer-guide-tutorial-1203.html)

* **Question 32. What Contextual Classes Are Available To Style The Panels?**

**Answer :**

Use contextual state classes such as, panel-primary, panel-success, panel-info, panel-warning, panel-danger, to make a panel more meaningful to a particular context.

[Javascript Objects Interview Questions](https://www.wisdomjobs.com/e-university/javascript-objects-interview-questions.html)

* **Question 33. Can You Put A Table Within Bootstrap Panel?**

**Answer :**

Yes! To get a non-bordered table within a panel, use the class .table within the panel. Suppose there is a <div> containing .panel-body, we add an extra border to the top of the table for separation. If there is no <div> containing .panel-body, then the component moves from panel header to table without interruption.

[Web Designing Interview Questions](https://www.wisdomjobs.com/e-university/web-designing-interview-questions.html)

* **Question 34. Can You Put A Listgroup Within Bootstrap Panel?**

**Answer :**

Yes You can include list groups within any panel. Create a panel by adding class .panel to the element. Also add class .panel-default to this element. Now within this panel include your list groups.

* **Question 35. What Is Bootstrap Well?**

**Answer :**

A well is a container in that causes the content to appear sunken or an inset effect on the page. To create a well, simply wrap the content that you would like to appear in the well with a containing the class of .well.

* **Question 36. What Is Scrollspy Plugin?**

**Answer :**

The Scrollspy (auto updating nav) plugin allows you to target sections of the page based on the scroll position. In its basic implementation, as you scroll, you can add .active classes to the navbar based on the scroll position.

[Java Script Interview Questions](https://www.wisdomjobs.com/e-university/java-script-interview-questions.html)

* **Question 37. What Is Affix Plugin?**

**Answer :**

The affix plugin allows a to become affixed to a location on the page. You can also toggle it's pinning on and off using this plugin. A common example of this are social icons. They will start in a location, but as the page hits a certain mark, the will be locked in place and will stop scrolling with the rest of the page.

* **Question 38. How To Create A Navbar In Bootstrap?**

**Answer :**

**To create a default navbar:**

* + Add the classes .navbar, .navbar-default to the tag.
  + Add role = "navigation" to the above element, to help with accessibility.
  + Add a header class .navbar-header to the

element. Include an element with class navbar-brand. This will give the text a slightly larger size.

* + To add links to the navbar, simply add an unordered list with the classes of .nav, .navbar-nav.
* **Question 39. What Is Bootstrap Breadcrumb?**

**Answer :**

Breadcrumbs are a great way to show hierarchy-based information for a site. In the case of blogs, breadcrumbs can show the dates of publishing, categories, or tags. They indicate the current page's location within a navigational hierarchy.  
A breadcrumb in Bootstrap is simply an unordered list with a class of .breadcrumb. The separator is automatically added by CSS (bootstrap.min.css).

* **Question 40. Which Class Is Used For Basic Pagination?**

**Answer :**

.pagination class is used to add the pagination on a page.

[PHP Interview Questions](https://www.wisdomjobs.com/e-university/php-interview-questions.html)

* **Question 41. How Will You Customize Links Of Pagination?**

**Answer :**

You can customize links by using .disabled for unclickable links and .active to indicate the current page.

* **Question 42. What Are Bootstrap Labels?**

**Answer :**

Bootstrap labels are great for offering counts, tips, or other markup for pages. Use class .label to display labels.

[. NET Interview Questions](https://www.wisdomjobs.com/e-university/dot-net-interview-questions.html)

* **Question 43. What Are Bootstrap Badges?**

**Answer :**

Badges are similar to labels; the primary difference is that the corners are more rounded. Badges are mainly used to highlight new or unread items. To use badges just add <span class = "badge"> to links, Bootstrap navs, and more.

* **Question 44. What Is Bootstrap Jumbotron?**

**Answer :**

As the name suggest this component can optionally increase the size of headings and add a lot of margin for landing page content.

**To use the Jumbotron:**

* + Create a containerwith the class of .jumbotron.
  + In addition to a larger
* **Question 45. What Is Bootstrap Page Header?**

**Answer :**

The page header is a nice little feature to add appropriate spacing around the headings on a page. This is particularly helpful on a web page where you may have several post titles and need a way to add distinction to each of them. To use a page header, wrap your heading in a <div> with a class of .page-header.

1. What is Bootstrap?

Bootstrap is a mobile first front-end framework for easier web development with supporting all virtual devices. It uses [HTML](https://www.dotnettricks.com/learn/html), [CSS](https://www.dotnettricks.com/learn/stylesheet), [JavaScript](https://www.dotnettricks.com/learn/javascript).

1. Why we should use Bootstrap?

Every browser can support bootstrap features. With the knowledge of HTML, CSS, and JavaScript anyone can start working with [Bootstrap](https://en.wikipedia.org/wiki/Bootstrap_(front-end_framework)). Bootstrap responsive [CSS](https://www.dotnettricks.com/learn/stylesheet/css-inline-block-and-none-display-style) adjusts to Desktops, Mobiles, and Tablets. It provides web-based customization. Bootstrap is open source. It can be used for mobile first design styles using the entire library instead of separate and different files.

1. What are Responsive websites?

Responsive means a website can be used on any platform with all types of screen size. A platform like it may be a desktop, laptop, tablet or smartphones. It automatically adjusts with changing the width and height of the screen.

In today’s era, people, as well as business, follow the smartphones for their personal and professional use. So, bootstrap is the best choice to make the website one time and it supports all kinds of devices.

1. What are the features of Bootstrap v4?

The initial release of [bootstrap v4.x](https://www.dotnettricks.com/learn/bootstrap/introduction-to-bootstrap) was 4.0 and the final release in 2018-2019 is 4.2 ;

* + The toast component is added for showing notifications
  + The spinner component is added for loading
  + The utility class .text-decoration-none is added
  + The bootstrap-grid.css file updates the margin and padding of the grid system
  + The .modal-xl class is used for modals
  + The .font-weight-lighter and .font-weight-bolder are added as utility classes
  + The validation styles for file inputs is added with new features
  + The print feature of dark tables is updated with new quality
  + The new utility class .dropdown-item-text is added for dropdown items
  + The Tooltips supports Shadow DOM
  + The double transitions issue on carousels is improved now

1. What is a Source map file?

A source map file allows the web debuggers to refer to the original context from where the code was generated. The file has the .map extension.

1. How to make images responsive using bootstrap3?

We can make responsive images using .img-responsive class. Let’s see the code,

* + <div class="container">
  + <img class="img-responsive" src="images/Satya.jpg" alt="Responsive"/><div>

1. How to create responsive images in Bootstrap 4?

Bootstrap 4 works with .img-fluid to build responsive images. Let’s see the code,

* + <div class="container">
  + <img src="/Satya/Pics/Name.jpg" class="img-fluid" alt="">
  + </div>

1. How to make videos responsive in bootstrap3?

We can create responsive using a .embed-responsive class. Then we will add the .embed-responsive-item class to an <iframe> tag. Let’s see the code,

* + <div class="container">
  + <div class="embed-responsive">
  + <iframe class="embed-responsive-item"src="https://www.youtube.com/channel/UCrxtdnFenZ5aDVQmHv2iGGg/videos?view\_as=subscriber&disable\_polymer=1"></iframe>
  + </div>
  + </div>

1. How to make tab component using bootstrap?

We can create nav components such as tabs using Twitter Bootstrap. You can use .nav-tabs class with the base class .nav.

* + <ul class="nav nav-tabs">
  + <li class="active"><a href="#">Profile</a></li>
  + <li><a href="#">Details</a></li>
  + <li><a href="#">Contact</a></li>
  + </ul>

1. How many types of classes are used in Bootstrap3 to create a responsive page layout?

There are 4 types of bootstrap grid classes. we can create grid column layouts extra small devices like mobile phones to large devices like large desktop screens.

.col-xs-\* : This class is used for extra small device like smart phone (Size<768px)

.col-sm-\*: This class is used for small device like tablets (size>=768px)

.col-md-\*: This class is used for medium device like laptops and small size desktop CRT screen (size >=992px)

.col-lg-\* : This class is used for large device like flat screen or large size desktop (size>=1200px)

1. How to hide elements in Bootstrap3 based on device?

This can be possible using hidden utility classes. See

**hidden-xs-\* :** Hide the elements only on extra small de

**hidden-sm-\* :** Hide the elements only on small devices

**hidden-md-\*:** Hide the elements only on medium devi

**hidden-lg-\*:** Hide the elements only on larger devices.

1. How to visible elements based on device screen?

This can be possible using hidden utility classes. See the below classes,

**visible-xs-\*:** Visible the elements only on extra small de

**visible-sm-\*:** Visible the elements only on small devices.

**visible-md-\*:** Visible the elements only on medium devi

**visible-lg-\*:** Visible the elements only on larger devices.

1. How to show certain elements for printing purpose?

visible-print class is visible for printing purpose. Let’s see the code,

* + <div class="container">
  + <div class="row">
  + <div class="col-xs-10 visible-print">
  + *<!--It is visible when printing.-->*
  + </div>
  + </div>
  + </div>

1. What is the grid size in Bootstrap 4?

[Bootstrap 4](https://www.dotnettricks.com/learn/bootstrap/introduction-to-bootstrap) supports 5 tier grid system. The below lists for different screen sizes of device,

**Extra small <576px :**

* + Width container will be auto.
  + Here class prefix is .col-.
  + A number of columns will be 12.
  + It can be nestable.
  + It supports column ordering.

**Small =576px :**

* + Width container will be 540px.
  + Here class prefix is .col-sm-.
  + A number of columns will be 12.
  + It can be nestable.
  + It supports column ordering.

**Medium =768px :**

* + Width container will be 720px.
  + Here class prefix is .col-md-.
  + A number of columns will be 12.
  + It can be nestable.
  + It supports column ordering.

**Large =992px:**

* + Width container will be 960px.
  + Here class prefix is .col-lg-.
  + A number of columns will be 12.
  + It can be nestable.
  + It supports column ordering.

**Extra large =1200px :**

* + Width container will be 1140px.
  + Here class prefix is .col-xl-.
  + A number of columns will be 12.
  + It can be nestable.
  + It supports column ordering.
* How to create a bootstrap tooltip using jQuery?

Using data-toggle attribute that means include data-toggle="tooltip" to the element and initialize it using [jQuery.](https://www.dotnettricks.com/learn/jquery) For setting the tooltip message include the title attribute to the element.

Let’s see the code,

* + <button class="btn btn-primary" tooltip="The title message is shown"
  + data-toggle="tooltip">
  + Click Here
  + </button>
  + <script type="text/javascript">
  + $(document).ready(function () {
  + $('[data-toggle="tooltip"]').tooltip();
  + });
  + </script>

1. How to set alert using JavaScript?

To create an alert, I can use the alert class. Types of alert contextual classes are mentioned below,

* + **alert-danger:** This can be used for error purpose.
  + **alert-success:** This can be used for success purpose.
  + **alert-danger:**This can be used for information purpose.
  + **alert-warning:** This can be used for warning purpose.

First, create a button element for click event and then create div element with ID and message information.

* + <button id="btnAlert" class="btn btn-primary">
  + Alert
  + </button>
  + <div id="dvAlert" class="alert alert-danger collapse">
  + <a href="#" class="close" data-dismiss="alert">×</a>
  + <strong>Error!</strong> Check once using unit test.
  + </div>
  + Next step is to mention div id inside button click event using jquery.
  + <script type="text/javascript">
  + $(document).ready(function () {
  + $('#btnAlert').click(function () {
  + $('#dvAlert').show('fade');
  + });
  + });
  + </script>

1. How to toggle a drop-down menu using JavaScript?

Using .dropdown-toggle class of an anchor link or a button using [JavaScript](https://www.dotnettricks.com/learn/javascript/datatypes).

* + <button class="btn btn-primary dropdown-toggle" type="button" data-toggle="dropdown">
  + Dropdown menu
  + <span class="caret"></span>
  + </button>
  + <script>
  + $(document).ready(function () {
  + $(".dropdown-toggle").dropdown();
  + });
  + </script>

1. What is the relationship between Sass and bootstrap?

SASS is a stylesheet language and [CSS](https://www.dotnettricks.com/learn/stylesheet/html-color-codes) pre-processors. It should be compiled into [CSS](https://www.dotnettricks.com/learn/stylesheet/css-to-force-long-text-and-urls-to-wrap-on-all-browser) and that can be understood by the Web browser. The files are written in SASS that have the. scss extension. The complete Bootstrap 4 [CSS](https://www.dotnettricks.com/learn/stylesheet/css-to-show-only-horizontal-and-vertical-scroll-bar-in-div) source is written in the SASS language.

The Bootstrap 4 CSS files are compiled using a SASS compiler i.e. A.K.A pre-processor. SASS is the best way to way to customize Bootstrap.

1. What is the difference between Bootstrap v3 and v4?

Bootstrap 4 is the latest version of Bootstrap. It is the most popular framework including HTML, CSS, and JavaScript. New components are added in Bootstrap 4. The final release in 2018-2019 is 4.2:

**COMPONENTS**

**BOOTSTRAP v3**

**BOOTSTRAP v4**

font size

The global font size is 14px

The global font size is 16px

primary CSS unit

The primary CSS unit is px

The primary CSS unit is rem

source CSS files

The source CSS files are named as LESS

The source CSS files are named as SCSS

grid system

It has 4 tier grid system(These are xs, sm, md, lg)

It has 5 tier grid system(These are xs, sm, md, lg, xl)

offset columns

It implements col-\*-offset-\* classes to offset columns

It uses offset-\*-\* classes to offset columns

inverse tables

The inverse table is not supported

Uses inverse tables by .table-dark

condensed tables

The .table-condensed is used for condensed tables

The .table-sm is used for condensed tables

contextual table classes

It doesn't contain .table- prefix for its contextual classes

It contains .table- prefix for its contextual classes

radio buttons

It uses .radio, .radio-inline to display radio buttons

It uses .form-check-input, .form-check-inline to display checkboxes

checkboxes

It uses .checkbox, .checkbox-inline to display checkboxes

It uses .form-check, .form-check-label to display radio buttons

input controls size

It uses .input-lg and .input-sm to customize the size of an input controls

It uses form-control-lg and .form-control-sm to customize the size of an input controls

label size

It doesn't contain any classes for changing form label size

It uses .col-form-label-sm and .col-form-label-lg to customize the size of a label

help text

It uses .help-block class to show help text

It uses the .form-text class to show help text

form legends

It has no classes for making the form legends

It uses .col-form-label on legend elements

validation styles

It contains validation styles for validating states on form controls(like .has-warning)

It doesn't contain validation styles

custom forms

It doesn't support custom forms

It supports custom forms

buttons

It contains .btn-default and .btn-info classes but .btn-secondary is not found

It contains the .btn-secondary, .btn-light, and .btn-dark classes

outline button

It doesn't contain the outline button

It supports .btn-outline-\* classes for adding styles buttons using an outline colour

button sizes

It uses .btn-xs for button sizes

The .btn-sm and .btn-lg are used but .btn-xs class is no longer available

responsive images

It uses .img-responsive class for responsive images

It uses .img-fluid class for responsive images

image alignment

It uses .pull-right, .pull-left for right and left image alignment

It uses .pull-\*-right and .pull-\*-left for right and left image alignment

block level images

It uses .center-block to align block level images

It uses .m-x-auto to align block level images

Dropdown header

It uses li tags to build dropdowns with .dropdown-header

It uses to h1 - h2 tags to build dropdowns with .dropdown-header

Divider in dropdown

It uses .divider class to the li element to make lists of build dropdowns

It uses .divider class to the div element to make lists of build dropdowns

disable menu items

It uses the .disabled class to the li element to disable menu items

It uses .disabled class to the <a> element to disable menu items

Nav inline

There is no .nav-inline class

It uses the .nav-inline class for specifying navs to be displayed inline

navbar alignment

It uses .navbar-right, .navbar-left for navbar alignment

It uses .mr-auto for navbar alignment

Fixed navbar

It uses .navbar-fixed-top and .navbar-fixed-bottom for setting navbar in top or bottom

It uses .fixed-top and .fixed-bottom for setting navbar in top or bottom

glyphicons

It supports glyphicons

It doesn’t support glyphicons

carousel item

It uses .item class for carousel item

It uses .carousel-item class

affix

It supports affix

It doesn’t support it

Thumbnails

It supports thumbnails

It doesn’t support it

Wells and Panels

It supports both

It doesn’t support it

cards

It doesn’t support it

It supports cards instead of affix, thumbnails, wells and panels

1. What are cards in Bootstrap 4?

Bootstrap 4 uses cards by replacing panels, wells, and thumbnails. Bootstrap 4 uses .card and .card-body classes to an element to make container. It adds .card-title to heading elements and It uses .card-text to text elements.

Let’ see code,

* + <div class="container-fluid">
  + <div class="card card-body">
  + <h4 class="card-title">Name</h4>
  + <p class="card-text">My name is Satyaprakash</p>
  + <a href="#" class="btn btn-primary">More Detilas>></a>
  + </div>
  + </div>

**1. What is jQuery?**

jQuery is not a programming language but a well written JavaScript code. It is a JavaScript code, which do document traversing, event handling, Ajax interactions and Animations.

**2. Why jQuery is needed?**

jQuery is needed for the following list:

* Used to develop browser compatible web applications
* Improve the performance of an application
* Very fast and extensible
* UI related functions are written in minimal lines of codes

**3. Whether jQuery HTML work for both HTML and XML documents?**

No, jQuery HTML only works for HTML documents not for XML Documents.

**4. What are the methods used to provide effects?**

Some of the effects methods are:

* Show()
* Hide()
* Toggle()
* FadeIn() and
* FadeOut()

**5. What is the advantage of using minimized version of jQuery?**

Efficiency of web page increases when minimized version of jQuery is used.min.js file will be more than 50% less than the normal js file. Reduction in the file size makes the web page faster.



*Jquery*

**6. Is jQuery is a JavaScript or JSON library file?**

jQuery is a library of JavaScript file and it consists of DOM, event effects and the Ajax functions. jQuery is said to be a single JavaScript file.

**7. Which operating system is more compatible with jQuery?**

Mac, Windows and Linux are more compatible with the jQuery.

**8. How can we include jQuery library in ASP.Net project?**

Download the jQuery library from jQuery.com and include that reference in the asp.net page.

**9. Which command will give a version of jQuery?**

The command $.ui.version returns jQuery UI version.

**10. In what scenarios jQuery can be used?**

jQuery can be used in following scenarios:

* Apply CSS static or dynamic
* Calling functions on events
* Manipulation purpose
* Mainly for Animation effects

**11. What is the difference between find and children methods?**

Find method is used to find all levels down the DOM tree but children find single level down the DOM tree.

**12. What is jQuery connect?**

A ‘ jQuery connect’  is a plugin used to connect or bind a function with another  function. Connect is used to execute function from any other function or plugin is executed.

**13. How to use connect?**

Connect can be used by downloading jQuery connect file from jQuery.com and then include that file in the HTML file. Use $.connect function to connect a function to another function.

**14. What are the features of jQuery, has been used in web applications?**

jQuery uses features like Sliding, File uploading and accordian in web applications.

**15. What are the browser related issues for jQuery?**

Browser compatibility of jQuery plugin is an issue and needs lot of time to fix it.

**16. Whether we need to add jQuery file in both Master and Content page?**

jQuery file should be added to the Master page and can use access from the content page directly without having any reference to it.

**17. What are the basic selectors in jQuery?**

Following are the basic selectors in jQuery:

* Element ID
* CSS Name
* Tag Name
* DOM hierarchy

**18. Can we call C# code behind using jQuery?**

Yes, we can call C# code from jQuery as it supports .net application.

**19. What is the use jQuery.data method?**

jQuery.data methods is used to associate the data with the DOM nodes and the objects. This data method makes the jQuery code clear and concise.

**20. What is the use of each function in jQuery?**

Each function is used to iterate each and every element of an object. It is used to loop DOM elements, arrays and the object properties.

**21. What is the difference between size and length of jQuery?**

Size and length both returns the number of element in an object. But length is faster than the size because length is a property and size is a method.

**22. Can we add more than one ‘document.ready’ function in a page?**

Yes, we can add more than one document.ready function in a page. But, body.onload can be added once in a page.

**23. What is the use of jQuery load method?**

jQuery load method is a powerful AJAX method which is used to load the data from a server and assign the data into the element without loading the page.

**24. Whether our own specific characters are used in place of $ in jQuery?**

Yes, We can use our own variable in place of $ by using the method called no Conflict () method.

var sample = $.noConflict()

**25. What are the four parameters used for jQuery Ajax method?**

The four parameters are

* URL – Need to specify the URL to send the request
* type – Specifies type of request(Get or Post)
* data – Specifies data to be sent to server
* Cache – Whether the browser should cache the requested page

**26. What is the use of jQuery filter?**

The jQuery filter is used to filter the certain values from the object list based on the criteria. Example is to filter certain products from the master list of products in a cart website.

**27. Which program is useful for testing jQuery?**

QUnit is used to test jQuery and it is very easy and efficient.

**28. What is CDN?**

CDN is abbreviated as Content Distribution network and it is said to be a group of companies in different location with network containing copies of data files to maximize bandwidth in accessing the data.

**29. What are the two types of CDNs?**

There are two types of CDNs:

* Microsoft – Load jQuery from Ajax CDN
* Google – Load jQuery from Google libraries API

**30. Which sign is used as a shortcut for jQuery?**

Dollar ($) sign is used as a shortcut for jQuery.

**31. Is jQuery is a client or server scripting?**

jQuery is a client scripting.

**32. What is the script build up by jQuery?**

jQuery is a Javascript file and it is single javascript file that contains common DOM, event effects and Ajax functions.

**33. How can we debug jQuery?**

There are two ways to debug jQuery:

Debugger keyword

* Add the debugger to the line from where we have to start debugging and then run Visual Studio in Debug mode with F5 function key.
* Insert a break point after attaching the process

**34. What are all the ways to include jQuery in a page?**

Following are the ways to include jQuery in a page:

* Local copy inside script tag
* Remote copy of jQuery.com
* Remote copy of Ajax API
* Local copy of script manager control
* Embedded script using client script object

**35. What is the use of jQuery.ajax method ()?**

jQuery.ajax method is used for asynchronous HTTP requests.

**36. Where can we download JQuery?**

jQuery javascript can be downloaded from jQuery official website – www.jquery.com

**37. Is jQuery is a replacement of JavaScript?**

No, jQuery is not a replacement of JavaScript.

**38. What is called chaining?**

Chaining is used to connect multiple events and functions in a selector.

**39. What are the advantages of jQuery?**

Following are the advantages of jQuery:

* Just a JavaScript enhancement
* Coding is simple, clear, reusable
* Removal of writing more complex conditions and loops

**40. Whether C# code behind can be called from jQuery?**

Yes, we can call C# code behind from jQuery.

**41. What is the use of jQuery.data() method?**

jQuery data method is used to associate data with DOM nodes and JavaScript objects. This method will make a code very concise and neat.

**42. What is the difference between onload() and document.ready()?**

In a page, we can have only one onload function but we can have more than one document.ready function. Document.ready function is called when DOM is loaded but onload function is called when DOM and images are loaded on the page.

**43. What is the use of jQuery each function?**

jQuery each function is used to loop through each and every element of the target jQuery object. It is also useful for multi element DOM, looping arrays and object properties.

**44. How method can be called inside code behind using jQuery?**

$.ajax can be called and by declaring WebMethod inside code behind using jQuery.

**45. Which is the fastest selector in jQuery?**

ID and Element are the fastest selectors in jQuery.

**46. What is the slowest selector in jQuery?**

Class selectors are the slowest selectors in jQuery.

**47. Where jQuery code is getting executed?**

jQuery code is getting executed on a client browser.

**48. What is the method used to define the specific character in place of $ sign?**

‘NoConflict’ method is used to reference a jQuery and save it in a variable. That variable can be used instead of Sign.

**49. Why jQuery is better than JavaScript?**

jQuery is a library used for developing Ajax application and it helps to write the code clean and concise. It also handles events, animation and Ajax support applications.

**50. What are the types of selectors in jQuery?**

There are three types of selectors in jQuery:

* CSS Selector
* XPath Selector
* Custom Selector

Question 1: What is JQuery?

JQuery is a cross-browser lightweight JavaScript library. In simple words jQuery has been designed to make navigation to any element easier, or adding/invoking event handlers on your HTML page and also simplify the way you access the elements in your web pages, provide help in working with client-side events, enable visual effects like animation, and make it easier to use Ajax in your applications.  
  
A quick look at what is available in jQuery:

* Cross-browser support and detection.
* AJAX functions
* CSS functions
* DOM manipulation
* DOM transversal
* Attribute manipulation
* Event detection and handling.
* JavaScript animation
* Hundreds of plugins for pre-built user interfaces, advanced animations, form validation, etc.
* Expandable functionality using custom plugins.

For more follow the link: [Introduction To jQuery](http://www.c-sharpcorner.com/UploadFile/036f9e/introduction-of-jquery/)

Question 2. What is JQuery.noConflict?

jQuery no-conflict is an option given by jQuery to overcome the conflicts between the different js frameworks or libraries. When we use jQuery no-conflict mode, we are replacing the $ to a new variable and assigning to jQuery some other JavaScript libraries. Also use the $ (Which is the default reference of jQuery) as a function or variable name what jQuery has. And in our development life, we are not at all strict to only jQuery.  
  
Learn more here: [JQuery No-Conflict And Using Different Versions Of jQuery](http://www.c-sharpcorner.com/UploadFile/65794e/jquery-no-conflict-and-using-different-versions-of-jquery/)

Question 3. What is a CDN?

Content Delivery Network (CDN) in simple terms is a collection of servers spread across the globe. In other words, a CDN is a network of servers in which each request will go to the closest server.  
  
**Need For a CDN**  
  
For any web application, data can be categorized into either static or dynamic content. Dynamic content is the one that generally comes from a database. Static content is like CSS, images, JavaScript, flash files, video files and so on.  
  
Now one may ask, how are requests served when a user enters a URL into the browser? Interesting, let's have a look at it. Before knowing a CDN and its usage, it is very important to understand this process: [What and Why of CDN](http://www.c-sharpcorner.com/UploadFile/41e70f/what-and-why-of-cdn/)

Question 4. What are selectors in jQuery and how many types of selectors are there?

The basic operation in jQuery is selecting an element in DOM. This is done with the help of $() construct with a string parameter containing any CSS selector expression. $() will return zero or more DOM elements on which we can apply an effect or style.  
  
**$(document).ready()** indicates that code in it needs to be executed once the DOM got loaded. It won't wait for the images to load for executing the jQuery script. We created an anonymous function inside ready() function to hide div1.   
  
We can rewrite $(document).ready() as jQuery (document).ready(), since $ is an alias for jQuery. Always use jQuery in place of $, if you are using more than one JavaScript library to resolve conflicts with jQuery library. The methods called on $(), will implicitly be applied on all the elements returned by it without need of explicit looping. Let's say, $('.myclass').hide() will hide all elements with class as myclass with implicit looping.

As we discussed earlier, $() accepts a string parameter having tag name [like div, p] or Element Id or class name as shown in the following table:

|  |  |  |
| --- | --- | --- |
| Selector | jQuery Syntax | Description |
| Tag Name | $('div') | All div tags in the document |
| ID | $('#TextId') | Selects element with ID as TextId. It starts with # followed Element Id. |
| Class | $('.myclass') | Selects all elements with class as myclass. It starts with '.' followed by class name |

For more details read this article: [Selectors in jQuery](http://www.c-sharpcorner.com/uploadfile/satisharveti/selectors-in-jquery/)

Question 5. What is the use of jQuery .each() function?

The "j**Query.each()**" function is a general function that will loop through a collection (object type or array type). Array-like objects with a length property are iterated by their index position and value. Other objects are iterated on their key-value properties. The "jQuery.each()" function however works differently from the $(selector).each() function that works on the DOM element using the selector. But both iterate over a jQuery object.   
  
**Callback method**  
  
In the "jQuery.each()" method we're able to pass in an arbitrary array or object in which for each item will have the callback function executed.   
  
The "$.each()" function loops over any type of collection, either an array or an object collection. The "jQuery..each()" function has a callback function in which we pass the indexing value and the corresponding value of each item of the array each time. We can access the value of the current index position in the loop using the "this" keyword and append it in any DOM element.  
  
When you pass the array-like object to the .each() function, the callback can accept two arguments: index of the item, where index is the numerical zero-based index in the array of the current items and item is the value of the current array.  
  
**For example:**If we pass an array to each function, it iterates over items in the array and accesses both the current item and its index position.  
  
**Syntax**

1. jQuery.each(collection, callback(indexInArray, valueOfElement))
3. < script type = "text/javascript" >
4. $(document).ready(function() {
6. var arr = ["Goergie", "Johnson", "Agile", "Harrison", "Gaurav"];
8. $.each(arr, function(index, value) {
9. alert('Position is : ' + index + ' And Value is : ' + value);
10. });
12. });
13. < /script>

For more details follow the link: [$.each() Function in jQuery](http://www.c-sharpcorner.com/UploadFile/99bb20/each-function-in-jquery/)

Question 6. What is difference between prop and attr?

*jQuery.attr()*  
  
Gets the value of an attribute for the first element in the set of matched elements.  
  
Whereas:  
  
*jQuery. prop ()*  
  
Gets the value of a property for the first element in the set of matched elements.  
  
**What Attributes actually are**  
  
Attributes carry additional information about an HTML element and come in name=”value” pairs. You can set an attribute for an HTML element and define it when writing the source code.   
  
**For example**

1. <input id="txtBox" value="Jquery" type="text" **readonly**="readonly" />

As shown above, “**id**”, "**type**” and “**value**" are attributes of the input elements.   
  
For more details follow the link: [Difference Between prop and attr in jQuery](http://www.c-sharpcorner.com/UPLOADFILE/97FC7A/DIFFERENCE-BETWEEN-PROP-AND-ATTR-IN-JQUERY/)

Question 7. What is jQuery UI?

jQuery UI enable our applications to have a cool user interface and animation in a faster way. It is the set of plug-ins that include interface interactions, effects, animations, widgets and themes built on the JavaScript Library. jQuery is used to create cohesive and consistent APIs. It is a method that we can use to extend jQuery prototype objects. By that prototype object you can enable all jQuery objects to inherit any method that you add.  
  
**Interactions**  
We can use interactions for basic mouse-based behaviours to any element. Examples of Interactions are the following:

* Draggable
* Droppable
* Resizable
* Selectable
* Sortable

Learn more here: [Getting Started With jQuery UI Plugin](http://www.c-sharpcorner.com/UPLOADFILE/22D000/GETTING-STARTED-WITH-JQUERY-UI-PLUGIN/)

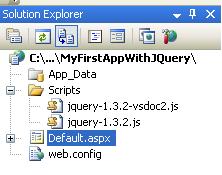
Question 8. What are the methods used to provide effects?

jQuery provides many amazing effects, we can apply these effects quickly and with simple configuration. The effect may be hiding, showing, toggling, fadeout, fadein, fadeto and so on toggle(), Show() and hide() methods. Similarly we can use other methods as in the following:

* **animate( params, [duration, easing, callback] )** This function makes custom animations for your HTML elements.
* **fadeIn( speed, [callback] )**This function fades in all the matched elements by adjusting their opacity and firing an optional callback after completion.
* **fadeOut( speed, [callback] )** This function is used to fade out all the matched elements by adjusting their opacity to 0, then setting the display to "none" and firing an optional callback after completion.
* **fadeTo( speed, opacity, callback )**This function fade the opacity of all the matched elements to a specified opacity and firing an optional callback after completion.
* **stop( [clearQueue, gotoEnd ])**This function stops all the currently running animations.

For More Info follow the link: [jQuery Effects Methods](http://www.c-sharpcorner.com/UploadFile/79037b/jquery-effects-methods/)

Question 9. How we can use jQuery in ASP.NET?

As you know jQuery is a fast, lightweight JavaScript library that is CSS3 compliant and supports many browsers. The jQuery framework is extensible and handles the DOM manipulations, CSS, AJAX, Events and Animations, very nicely.   
  
**Some differences between JavaScript and jQuery**  
  
JavaScript is a language whereas jQuery is a library written using JavaScript.  
  
Let us go through an example, which will help you in understanding the use of jQuery with ASP.NET application.  
  
Make a folder with the name Scripts inside your application. Right click on **Scripts folder > Add Existing Item > Browse**to the path where you downloaded the jQuery library (**jquery-1.3.2.js**) and the intellisense documentation **(jquery-1.3.2-vsdoc2.js).** Select the files and click **Add**. The structure will look similar to the following image:  
  
  
  
In this example, I am going to display an alert on **asp:Button** click using jQuery.  
  
**Here's the Default.aspx code**

1. <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN""http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
2. <html xmlns="http://www.w3.org/1999/xhtml">
4. <head runat="server">
5. <title>My First Application With JQuery</title>
7. <script src="Scripts/jquery-1.3.2.js" type="text/javascript"></script>
9. <script type="text/javascript">
10. $(document).ready(function() {
11. $("#Button1").click(function() {
12. alert("Welcome jQuery !");
13. });
14. });
15. </script>
17. </head>
19. <body>
20. <form id="form1" runat="server">
21. <div>
22. <asp:Button ID="Button1" runat="server" Text="Click Me" />
23. </div>
24. </form>
25. </body>
27. </html>

For more code examples follow the link: [An Introduction to jQuery with ASP.NET](http://www.c-sharpcorner.com/uploadfile/rahul4_saxena/an-introduction-to-jquery-with-Asp-Net/) Or [Consuming ASP.NET Web Service through jQuery](http://www.c-sharpcorner.com/UploadFile/sapnabeniwal/consuming-asp-net-web-service-through-jquery/)

Question 10. How can we use hide() method on a button click using jQuery?

In jQuery the hide () method is very useful. By using this method you can hide HTML elements with the hide() method. In this example, we create a div element which contains text. When we click on the Button the text we use in the div will be hidden.  
  
**Example**  
We are showing you the complete code for the .aspx page below.

1. <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN""http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
2. <html xmlns="http://www.w3.org/1999/xhtml">
4. <head>
5. <title>here</title>
6. <script src="Scripts/jquery-1.4.1.min.js" type="text/javascript"></script>
7. <script type="text/javascript">
8. $(document).ready(function() {
9. $("button").click(function() {
10. $("#div1").hide();
11. });
12. });
13. </script>
14. </head>
16. <body>
17. <h2>
18. This **is** a heading</h2>
19. <div id="div1">
20. jQuery **is** great library **for** developing ajax based application.
21. <br> jQuery **is** great library **for** the JavaScript programmers, which simplifies the development of web 2.0 applications.
22. <br />
23. <br />
24. <br />
25. </div>
26. <button>
27. Hide</button>
28. </body>
29. <html>

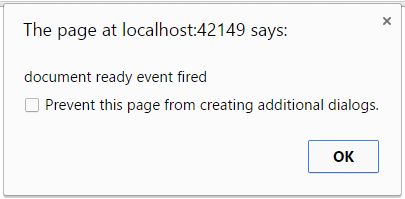
For more follow the link: [html() and hide() method in jQuery](http://www.c-sharpcorner.com/UploadFile/rohatash/html-and-hide-method-in-jquery/)

Question 11. What is the difference between $(window).load and $(document).ready function in jQuery?

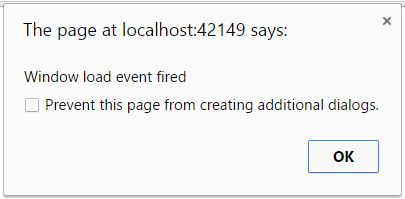
$(window).load is an event that fires when the DOM and all the content (everything) on the page is fully loaded. This event is fired after the ready event.  
  
Let's look at an example.

1. <script type="text/javascript" lang="ja">
2. $(window).load(function() {
3. alert("Window load event fired");
4. });
6. $(document).ready(function() {
7. alert("document ready event fired");
8. });
9. </script>

In the preceding JavaScript, we created an anonymous function that contains an alert message. So, when the preceding two events are fired an alert window will pop-up.  
  
Run the application and let's see which event is fired first.  
  
The document ready function will be fired first.



Then the window load event will be fired.



**When to use $(window).load instead of $(document).ready**  
  
In most cases, the script can be executed as soon as the DOM is fully loaded, so ready() is usually the best place to write your JavaScript code. But there could be some scenario where you might need to write scripts in the load() function. For example, to get the actual width and height of an image.  
  
As we know the $(window).load event is fired once the DOM and all the CSS, images and frames are fully loaded. So, it is the best place to write the jQuery code to get the actual image size or to get the details of anything that is loaded just before the load event is raised.  
  
Follow the link for more details: [Basics of jQuery: Part 1](http://www.c-sharpcorner.com/UploadFile/219d4d/basics-of-jquery-part-1/)

Question 12. How to handle Controls attribute Using jQuery?

For handle Controls attribute using jQuery we used **.addClass(),** **.removeClass(),** **.css(), .toggleClass(),**etc to manage all css and html attributes of any html control.  
  
You can follow the link: [Handle Controls Attribute Using jQuery](http://www.c-sharpcorner.com/UploadFile/abhikumarvatsa/handle-controls-attribute-using-jquery/)

Question 13: What is chaining in jQuery?

Chaining is a  powerful feature of jQuery. Chaining means specifying multiple functions and/or selectors to an element.  
  
Chaining reduces the code segment and keeps it very clean and easy to understand. Generally chaining uses the jQuery built in functions that makes compilation a bit faster.  
  
By using chaining we can write the above code as follows:

1. $(document).ready(function() {
2. $("#div2").html($("#txtBox").prop("readonly")) + '</br>';
3. $("#div3").html($("#txtBox").attr("readonly"));
4. });

The code segment above is described by the following image:  
  
  
  
Follow the link for more details: [jQuery Interview Questions and Answers With Practices: Part 2](http://www.c-sharpcorner.com/UploadFile/97fc7a/jquery-interview-question-and-answer-with-practices-part-2/) Or [jQuery - "Write Less Do More": Day 1](http://www.c-sharpcorner.com/UploadFile/79037b/jquery-write-less-do-more-day-1/)

Question 14: How to work with parent(), children() and siblings() methods in jQuery?

The parent() function returns the parent of the selected element by calling the jQuery parent() function. The siblings() function returns all the siblings of given HTML elements.

Learn more here: [Getting Started With jQuery Traversing](http://www.c-sharpcorner.com/UploadFile/22d000/getting-started-with-jquery-traversing/)

Question 15. What is jQuery Datepicker in jQuery?

As per jQueryUI Documents, the jQuery UI Datepicker is a highly configurable plugin that adds datepicker functionality to your pages. You can customize the date format and language, restrict the selectable date ranges and add in buttons and other navigation options easily.  
  
By default, the datepicker calendar opens in a small overlay when the associated text field gains focus. For an inline calendar, simply attach the datepicker to a div or span.  
  
You must use the following jQuery reference in your **HTML** Code, otherwise it will not work.

1. <head>
2. <link rel="stylesheet" href="//code.jquery.com/ui/1.11.4/themes/smoothness/jquery-ui.css">
3. <script src="//code.jquery.com/jquery-1.10.2.js"></script>
4. <script src="//code.jquery.com/ui/1.11.4/jquery-ui.js"></script>
5. </head>

Display month & year menus in jQuery Datepicker.  
  
**HTML**

1. Date : <input id="Datepicker" type="text" />

**jQuery**  
  
The following code will show the **Month** & **Year** menus in jquery Datepicker. The "**yearRange**" will specify the range of the year you want.

1. $(function() {
3. $('#Datepicker').datepicker({
4. dateFormat: 'dd/mm/yy',
5. changeMonth: **true**,
6. changeYear: **true**,
7. yearRange: '1950:2100'
9. });
11. })

For more details follow the link: j[Query Datepicker - Part 2](http://www.c-sharpcorner.com/UploadFile/2a6dc5/jquery-datepicker-part-2/)

Question 16. How to use scrolling an ASP.NET Multiline Textbox using jQuery?

We will write the jQuery code which will be inside the <script></script> tag and always placed between head section or body section. Its your choice that where you want to place it. Let's see the jQuery code here:  
  
  
 **Explanation**  
Here we will explain the jQuery code which is given above. When the user clicks on the button (btn), we toggle the click behavior. On the first click, we cancel the postback by using e.preventDefault() and then call a function called s\_roll() passing in the textarea and the scrollHeight. The code $txt[0].scrollHeight is for scrolling downwards.

1. e.preventDefault();
2. s\_roll($txt, $txt[0].scrollHeight);

Again when the user clicks the button (btn) again, the postback is cancelled and the scrollHeight is set to 0. And by using it the multiline textbox will be scrolling upwards.

1. e.preventDefault();
2. s\_roll($txt, 0);

Here the scrollArea() function accepts the textarea that is to be scrolled as well as the scrollHeight. We then animate the scrollTop property to scroll upwards/downwards depending on the height parameter. The duration of the animation is set to 1000 milliseconds which provides a smooth scrolling effect and you can change according to your requirement. The function is given below which will animate it.

Learn more here: [Advanced Typing Scroller Using jQuery](http://www.c-sharpcorner.com/UploadFile/sapnabeniwal/advanced-typing-scroller-using-jquery/)

Question 17. What is Ajax in jQuery?

AJAX stands for “Asynchronous JavaScript and XML”. AJAX is about exchanging data with a server, without reloading the whole page. It is a technique for creating fast and dynamic web pages.  
  
In .NET, we can call server side code using two ways:

1. ASP .NET AJAX
2. jQuery AJAX

In this article we will focus on jQuery Ajax.  
  
**$.ajax () Method**  
  
JQuery’s core method for creating Ajax requests. Here are some jQuery AJAX methods:

* **$.ajax()**-Performs an async AJAX request.
* **$.get()**-Loads data from a server using an AJAX HTTP GET request.
* **$.post()** - Loads data from a server using an AJAX HTTP POST request.

To know more [click](https://learn.jquery.com/ajax/jquery-ajax-methods/).  
  
**$.ajax () Method Configuration option**  
  
Options that we use:

* async
* type
* url
* data
* datatype
* success
* error

Let’s have a detailed overview:  
  
**async**  
  
Set to false if the request should be sent synchronously. Defaults to true.   
  
Follow the link for more details: [ASP.NET MVC Application - Using jQuery, AJAX](http://www.c-sharpcorner.com/UploadFile/302f8f/Asp-Net-mvc-using-jquery-ajax/)

Question 18. Define slideToggle() effect?

The slide methods do the up and down element. To implement slide up and down on element jQuery here are the three methods:

* slideDown()
* slideUp()
* lideToggle()

And how to use them:  
  
**1. slideDown() Method**  
This function is used to slide and hide an element on down side:

1. <script type="text/javascript">
2. $(document).ready(function() {
3. $("#btnSlideDown").click(function() {
4. $("#login\_wrapper").slideDown();
5. **return** **false**;
6. });
7. });
8. </script>

**2. slideUp() Method**  
This function is used to slide and show element up side:

1. <script type="text/javascript">
2. $(document).ready(function() {
3. $("#btnSlideUp").click(function() {
4. $("#login\_wrapper").slideUp();
5. **return** **false**;
6. });
7. });
8. </script>

**3. slideToggle() Method**  
  
This method is between slideUp() method and slideDown() method. It shows/hides an element in up/down side:

1. <script type="text/javascript">
2. $(document).ready(function() {
3. $("#btnSlideToggle").click(function() {
4. $("#login\_wrapper").slideToggle();
5. **return** **false**;
6. });
7. });
8. </script>

For more details follow the link: [jQuery Effects using Slide methods](http://www.c-sharpcorner.com/UploadFile/3d39b4/jquery-effects-using-slide-methods/)

Question 19: What are the advantages of jQuery?

In JavaScript we write more code because it doesn't have more functions like animation effects functions and event handling. So if you use JavaScript, developers write more code and they often feel embrace when they execute the code on the browser and get a problem related to cross-browser support. To solve these types of problems, John has created a JavaScript library with a nice motto, "*write less and do more"* in 2006; that is called jQuery. So you can use all the functions and other capabilities available in JavaScript. It saves developer's time, testing efforts, lines of code and improves their productivity and efficiency of development. The following are some important points to use jQuery.

* Fully documented
* Lot of plugins
* Small size
* Everything works in IE 6.0+, FF 2.0+, Safari 3.0+, Chrome and Opera 9.0+

To know more about jQuery follow the link: [jQuery Overview](http://www.c-sharpcorner.com/UploadFile/rohatash/jquery-overview/)

Question 20: Define bind() and unbind elements in jQuery?

The jQuery bind() method attaches an event handler to elements, whereas unbind() detaches an existing event handler from elements. Use basic HTML code to create the HTML elements.

Learn more here: [jQuery Unbind() method to remove an attached event handler](http://www.c-sharpcorner.com/UploadFile/88d8c0/unbind-method-for-group-of-html-div-elements-using-jquery/)

Question 21: How to remove a DOM Element using jQuery?

Sometimes we get a requirement to delete all child nodes and remove DOM elements using jQuery to make a dynamic web page. In this scenario jQuery provides a couple of methods to remove DOM elements. Here are the options:

* empty()
* remove()
* html()

For more details follow the below given link: [Remove DOM Elements Dynamically in jQuery](http://www.c-sharpcorner.com/UploadFile/manas1/remove-dom-elements-dynamically-in-jquery/) Or [Remove a DOM Element Using jQuery](http://www.c-sharpcorner.com/UploadFile/65794e/remove-a-dom-element-using-jquery/)

Question 22: What are the features of jQuery used in web applications?

jQuery has some important features such as event handling, Ajax support and animation effects functions. Here is the list of important features supported by jQuery.

1. **HTML/DOM Manipulation:**JavaScript do not have any features related to the DOM, but JavaScript in the browser does include some intelligence about the DOM.  
     
   **Example:** GetElementById() function  
     
   DOM is an important feature of jQuery. It defines the logical structure of documents and the way a document is accessed and manipulated. jQuery has the extra intelligence regarding  DOM.
2. **Event Handling:**jQuery introduced a feature called Event handling. Before starting event handling you need to understand event. Events are actions. This means that you can write code that runs when a user clicks on a certain part of the page, or when mouse is moved over a form element. jQuery contains many events, such as a user clicking on a button, moving a mouse over an element and so on.
3. **Ajax Support:** For example, when you select an item from a DropDownList or other control on the same page then that can cause loss of data. Ajax is used to update the part of the web page without reloading the page. For example, if you create a search functionality in your website like Google Search. When you enter text into the Search TextBox then without reloading the page you see the related text. You can do it easily using Ajax Methods.
4. **Animations in jQuery:** The jQuery comes with plenty of built-in animation effects that you can use in your websites. For example, animation, show, hide and so on. In jQuery the animate() method is very useful. By using this method we can change the size of elements.  
   * Animation method
   * Show method
   * Hide method

For more details follow link: [jQuery Overview](http://www.c-sharpcorner.com/UploadFile/rohatash/jquery-overview/)

Question 23. What is the use of jQuery filter?

JQuery supports various types of filters, such as:

1. .eq()
2. .first()
3. .last()
4. .filter()
5. .has()
6. .not()

For more details follow link: [Filter in jQuery](http://www.c-sharpcorner.com/UploadFile/jayendra/filter-in-jquery/)

Question 24. What is the use of jQuery.ajax method()?

The ajax() method is used to do an AJAX (asynchronous HTTP) request. It provides more control of the data sending and on response data. It allows the handling of errors that occur during a call and the data if the call to the ajax page is successful.  
  
Here is the list of some basic parameters required for jQuery.ajax Method:

* **type:**Specifies the type of request (GET or POST).
* **url:**Specifies the URL to send the request to. The default is the current page.
* **contentType:** The content type used when sending data to the server. The default is "*application/x-www-form-urlencoded*".
* **dataType:** The data type expected of the server response.
* **data:**Specifies data to be sent to the server.
* **success(result,status,xhr):**A function to run when the request succeeds.
* **error(xhr,status,error):** A function to run if the request fails.

For more details follow link: [Introduction to jQuery.ajax Call in ASP.NET](http://www.c-sharpcorner.com/UploadFile/0cefc4/introduction-to-jquery-ajax-call-in-Asp-Net/)

Question 25. What is an attribute in jQuery?

There are many important properties of DOM or HTML elements such as for the <img> tag the src, class, id, title and other properties. jQuery provides ways to easily manipulate an elements attribute and gives us access to the element so that we can also change its properties.

1. attr( properties ) - Set a key/value object as properties to all matched elements.
2. attr( key, fn ) - Set a single property to a computed value, on all matched elements.
3. removeAttr( name ) - Remove an attribute from each of the matched elements.
4. hasClass( class ) - Returns true if the specified class is present on at least one of the set of matched elements.
5. removeClass( class ) - Removes all or the specified class(es) from the set of matched elements.
6. toggleClass( class ) - Adds the specified class if it is not present, removes the specified class if it is present.
7. html( ) - Gets the HTML contents (innerHTML) of the first matched element.
8. html( val ) - Sets the HTML contents of every matched element.
9. text( ) - Gets the combined text contents of all matched elements.
10. text( val ) - Sets the text contents of all matched elements.
11. val( ) - Gets the input value of the first matched element.

Learn more here: [jQuery Attribute Basics](http://www.c-sharpcorner.com/UploadFile/79037b/jquery-attributes-basics/)

Question 26. What are jQuery Events?

When we design dynamic web pages, we need to apply some events such as Mouse Click, for forms submit the form after a button click, change a color after a click, etc. So in layman language, events are actions that are used for dynamic web pages. When we perform these actions on an HTML page, we can do whatever we want.

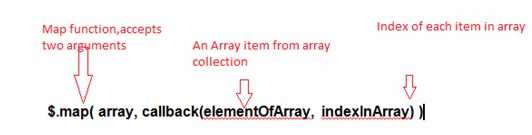
We use some event handlers to perform the action. Some important handlers are bind(), unbind(), blur(), off(), hover(), on(), one(), ready(), trigger() etc.

Learn more here: [Overview of jQuery Events](http://www.c-sharpcorner.com/UploadFile/79037b/overview-of-jquery-events/)

Question 27. What is the jQuery Unbind() method?

The jQuery bind() method attaches an event handler to elements, whereas unbind() detaches an existing event handler from elements. Use basic HTML code to create the HTML elements.  
  
For more details follow link: [jQuery Unbind() method to remove a attached event handler](http://www.c-sharpcorner.com/UploadFile/88d8c0/unbind-method-for-group-of-html-div-elements-using-jquery/)

Question 28. What is the jQuery Animation?

In short, the *.animate* method is used to perform a custom animation of a set of CSS properties. The .*animate()* method comes in two flavours. The first takes four arguments and the second takes two arguments.  
  
  
  
For more details follow link: [jQuery .animate() Method Part 1](http://www.c-sharpcorner.com/UploadFile/abhikumarvatsa/jquery-animate-method-part-1/)

Question 29. How can you find browser and browser version in jQuery?

Using $.browser property of jQuery  returns the browser information.  
  
*Using $.browser is not recommended by jQuery itself, so this feature has been moved to the jQuery.migrate plugin which is available for downloading if the user want. It is a vulnerable practice to use the same. Use it only if needed. It is always better to not use browser specific codes.*  
  
For more details follow link: [Find Browser And Browser Version Using jQuery](http://www.c-sharpcorner.com/UploadFile/65794e/find-browser-and-browser-version-using-jquery/)

Question 30. What is $.each() function in jQuery?

The "jQuery.each()" function is a general function that will loop through a collection (object type or array type). Array-like objects with a length property are iterated by their index position and value. Other objects are iterated on their key-value properties. The "jQuery.each()" function however works differently from the $(selector).each() function that works on the DOM element using the selector. But both iterate over a jQuery object.  
 **For example:** If we pass an array to the each() function, it iterates over items in the array and accesses both the current item and its index position.  
  
For more details follow link: [$.each() function in jQuery](http://www.c-sharpcorner.com/UploadFile/99bb20/each-function-in-jquery/)

Question 31: What is the difference between Map and Grep function in jQuery?

In $.map() you need to loop over each element in an array and modify its value whilst the $. Grep() method returns the filtered array using some filter condition from an existing array.  
  
The basic structure of Map() is: **$.map ( array, callback(elementOfArray, indexInArray) )**

Learn more here: [Diifference Between Map and Grep Function in jQuery](http://www.c-sharpcorner.com/UploadFile/97fc7a/diiffernece-between-map-and-grep-function-in-jquery/)

Question 32: What are jQuery plugins?

Plugins are a piece of code. In jQuery plugins it is a code written in a standard JavaScript file. These JavaScript files provide useful jQuery methods that can be used along with jQuery library methods. 

Any method you use in plugins must have a semicolon (;) at the end. The method must return an object (jQuery), unless explicitly noted otherwise. Use each to iterate over the current set of matched elements. It produces clean and compatible code that way. Prefix the filename with jQuery, follow that with the name of the plugin and conclude with .js. (For example, jquery.plug-in.js). Always attach the plugin to jQuery directly instead of $, so users can use a custom alias via the noConflict() method (via the jQuery Team).

Learn more here: [jQuery Plugins](http://www.c-sharpcorner.com/UploadFile/79037b/jquery-plugins/)

Question 33: Define jQuery .animate() method?

In jQuery the animate() method is very useful. By using this method we can change the size of elements. In this example we will create a div element which contains an Image; when we move the mouse over the image, the image size will change. First of all you add an image to the application, add a new form to the application and add the following HTML code to the aspx page.

1. <div style="height: 100px; width: 100px; position: relative">
2. <img src="animate.gif" id="img" />
3. </div>

Now add the following code in the head section.

1. <script type="text/javascript">
2. $(document).ready(function() {
4. $("div").mouseover(function() //mouseover function will execute when mouse pointer will reach on <div>element
5. {
6. $("img").animate({
7. height: 300
8. }, "slow"); //image height will change by using animate method
9. $("img").animate({
10. width: 300
11. }, "slow");
12. $("img").animate({
13. height: 100
14. }, "slow");
15. $("img").animate({
16. width: 100
17. }, "slow");
18. });
19. });
20. </script>

In the above code we create a mouseover function.

1. $("img").animate({ height: 300 }, "slow"); //image height will change by using animate method
2. $("img").animate({ width: 300 }, "slow");
3. $("img").animate({ height: 100 }, "slow");
4. $("img").animate({ width: 100 }, "slow");

For more follow the link: [CSS() and Animate() Method in jQuery](http://www.c-sharpcorner.com/UploadFile/rohatash/css-and-animate-method-in-jquery/)

Question 34: What is the difference between bind() and live() method in jQuery ?

The binding of event handlers are the most confusing part of jQuery for many developers working on jQuery projects. Many of them unsure of which is better to use. In this article we will see the main differences between Bind and Live methods in jQuery.  
  
**Bind() Method**  
The bind method will only bind event handlers for currently existing items. That means this works for the current element.  
  
**Example**

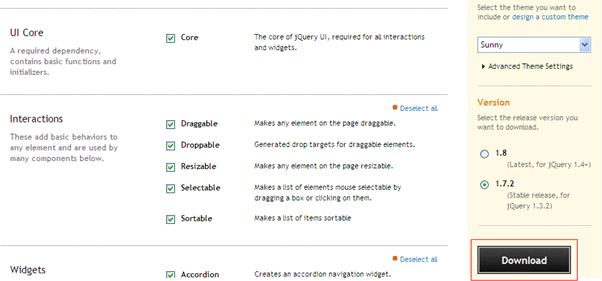
1. $(document).ready(function () {
2. $('P').bind('click', function () {
3. alert("Example of Bind Method");
4. e.preventDefault();
5. });
6. });

**Live() Method**  
The Live method can bind event handlers for currently existing items or future items.  
  
**Example**

1. $(document).ready(function() {
2. $('P').live('click', function() {
3. alert("Example of live method");
4. e.preventDefault();
5. });
6. $('body').append('<p>Adding Future items</p>');
8. });

Follow the link: [Difference Between Bind and Live Methods in jQuery](http://www.c-sharpcorner.com/UploadFile/rohatash/difference-between-bind-and-live-methods-in-jquery/)

Question 35: What is jQuery.holdReady() function?

jQuery.holdReady() function is what we can hold or release the execution of jQuery's ready event. This method should be called before we run the ready event. To delay the ready event, we need to call jQuery.holdReady(true);  
  
When we want to release the ready event then we need to call jQuery.holdReady(false);  
  
This function is helpful when we want to load any jQuery plugin before the execution of the ready event or want to perform certain events/functions before document.ready() loads. For example, some information.  
  
**For example**  
  


Learn more hereL [jQuery Interview Questions and Answers With Practices: Part 2](http://www.c-sharpcorner.com/UploadFile/97fc7a/jquery-interview-question-and-answer-with-practices-part-2/) or [HoldReady Functions in jQuery 1.9.1](http://www.c-sharpcorner.com/UploadFile/97fc7a/holdready-functions-in-jquery-1-9-1/)

Question 36: What is resize() function in jQuery?

This method in jQuery is used for changing of the size of the element. You can use by .resize() function. For more visit the following link: [Changing Size of The Text Using jQuery](http://www.c-sharpcorner.com/UploadFile/sapnabeniwal/changing-size-of-the-text-using-jquery/)

Question 37: Define Add or Remove class in jQuery?

*addclass* will be used for adding a new CSS class after replacing the old class and *removeClass* will work for removing the selected class.

1. $(document).ready(function() {
2. $('.button').click(function() {
3. **if** (**this**.id == "add") {
4. $('#animTarget').addClass("myClass", "fast")
5. } **else** **if** (**this**.id == "toggle") {
6. $('#animTarget').toggleClass("myClass", 1000, "easeOutSine")
7. } **else** **if** (**this**.id == "switch") {
8. $("#animTarget").switchClass("myClass", "switchclass", "fast")
9. } **else** {
10. $('#animTarget').removeClass("myClass", "fast")
11. }
12. })
13. });

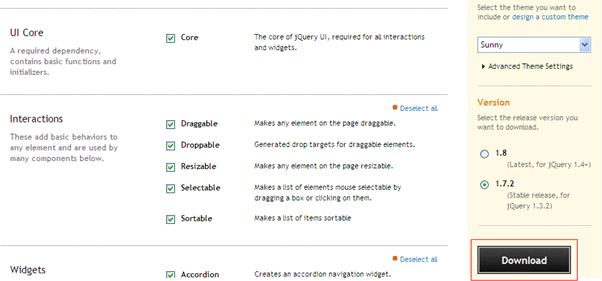
Learn more here: [How To Work With jQuery UI Effects](http://www.c-sharpcorner.com/UploadFile/22d000/how-to-work-with-jquery-ui-effects/)

Question 38: What is the usage of Draggable, Droppable, Resizable, Selectable in jQuery UI?

There are only 5 plugins available in the interaction section; that is Draggable, Droppable, Resizable, Selectable and Sortable. Interaction Plugins handles complex behaviors such as drag and drop, resizing, selection and sorting.  
  
Graphical representation of jQuery UI subordinates:  
  
  
  
**Draggable:** It enables draggable functionality on any DOM element. Move the draggable object by clicking on it with the mouse and dragging it anywhere within the viewport.  
  
**Droppable:** It enables any DOM element to be droppable, a target for draggable elements.  
  
**Resizable:** It enables any DOM element to be resizable. With the cursor, grab the right or bottom border and drag to the desired width or height.  
  
**Selectable:**It enables a DOM element (or group of elements) to be selectable. Draw a box with your cursor to select items. Hold down the Ctrl key to make multiple non-adjacent selections.  
  
**Sortable:** It enables a group of DOM elements to be sortable. Click on and drag an element to a new spot within the list, and the other items will adjust to fit. By default, sortable items share draggable properties.

Learn more here: [jQuery UI: Interaction Plugins](http://www.c-sharpcorner.com/UploadFile/abhikumarvatsa/jquery-ui-interaction-plugins/)

Question 39: What is the history of jQuery UI and how to use it?

jQuery UI is really very easy to learn and it provides abstractions for low-level interaction and animation, advanced effects and high-level, theme-able widgets, built on top of the jQuery JavaScript Library which you can use to build highly interactive web applications. The whole jQuery UI is categorized into four groups; they are core, interactions, widgets and effects.  
  
  
  
The components of jQuery UI are:

* **Core:** It's a perquisite for other widgets and effects to work properly.
* **Interactions:** It allows us to add behavior like Draggable, Droppable, Sortable, etc on the UI elements.
* **Widgets:**It provides UI controls like tabs, dialog, slider, etc.
* **Effects:** It provides ready to use effects like clip, bounce, explode, etc.

Learn more here: [Introduction to jQuery U](http://www.c-sharpcorner.com/uploadfile/satisharveti/introduction-to-jquery-ui/)

Question 40: What $(document).ready(function()) is and when to use it?

$(document).ready(function()) is a jQuery event that fires as soon as the DOM is fully loaded and ready to be manipulated by script. This is the earliest point in the page load process where the script can safely access elements in the page's HTML DOM. This event is fired before all the images and CSS are fully loaded.

Learn more here: [Basics of jQuery: Part 1](http://www.c-sharpcorner.com/UploadFile/219d4d/basics-of-jquery-part-1/)

Question 41: Define jQuery UI Autocomplete ?

Autocomplete is one of the best widgets ON websites and is used in nearly all websites. jQuery has a powerful widget, autocomplete, and in this article I will try to explain how to use jQuery Autocomplete in websites. All the way and all other features of autocomplete. We can make autocomplete, using AJAX, to call to build a list (server-side) and then bind that list into a text box using JavaScript. However there are other alternatives to make autocomplete rather then this in an easy way. The most robust and efficient tool of autocomplete is **jQuery-ui autocomplete** and this tool is free and there is no need to license it.  
  
**Points to remember**

* The Autocomplete widget requires some functional CSS, otherwise it won't work. If you build a custom theme, use the widget's specific CSS as a starting point.
* This widget manipulates its element's value programmatically, therefore a native change event may not be fired when the element's value changes.

Learn more here: [jQuery UI Autocomplete](http://www.c-sharpcorner.com/UploadFile/736ca4/autocomplete-in-jquery/)

Question 42: What is jQuery UI Sortable and how to use it?

The jQuery UI is a library provided by jQuery for a better user interface. Using sortable we can reorder the DOM elements in the defined area. Users have to click on the item and drag that item to a new place. The other items will be automatically arranged. Accordingly, use the following procedure to enable sortable elements:

1. Include the jQuery js file.
   1. <script src="http://ajax.aspnetcdn.com/ajax/jquery/jquery-1.10.0.min.js"></script>
2. Include the jQuery UI js file.
   1. <script src="http://ajax.aspnetcdn.com/ajax/jquery.ui/1.10.3/jquery-ui.min.js" type="text/javascript"></script>

Learn more here: [jQuery UI Sortable](http://www.c-sharpcorner.com/UploadFile/f18cf0/jquery-ui-sortable/)

Question 43: Why to use jQuery $ sign?

The basic operation in jQuery is selecting an element in DOM. This is done with the help of $() construct with a string parameter containing any CSS selector expression. $() will return zero or more DOM elements on which we can apply an effect or a style.

Learn more here: [Selectors in jQuery](http://www.c-sharpcorner.com/uploadfile/satisharveti/selectors-in-jquery/)

Question 44: What is slice() method in jQuery?

This method selects a subset of the matched elements by giving a range of indices. In other words, it gives the set of DOM elements on the basis of it's parameter (start, end).  
  
**Syntax:**.slice( start, end[Optional] )   
  
**Start:**This is the first and mandatory parameter of the slice method. This specifies from where to start to select the elements.   
  
**End:**This is an optional parameter. It specifies the range of the selection. This indicates where to stop the selection of elements, excluding end element. 

**Note:**The Negative Indices started from -1. This last element is denoted by index -1 and so on.

Learn more here: [.slice() Method in jQuery](http://www.c-sharpcorner.com/UploadFile/99bb20/slice-method-in-jquery/)

Question 45: What is jQuery Effects - Fading?

The fade methods define visibility of content in UI, in other words how the web page is hidden/shown. To use the fade methods of jQuery I need a jQuery library in my project, so I directly used the [Google AJAX Libraries](http://code.google.com/apis/ajaxlibs/) content delivery network to serve jQuery from Google. Doing so has several advantages over hosting jQuery on our server, decreased latency, increased parallelism, and better caching. We add the script to our project.

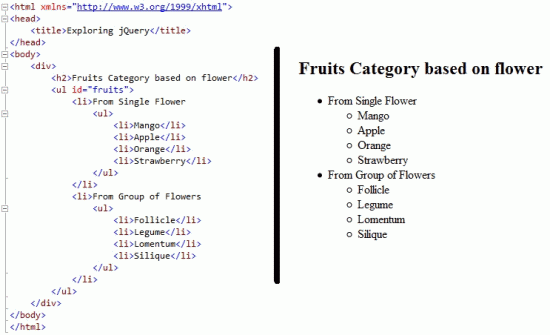
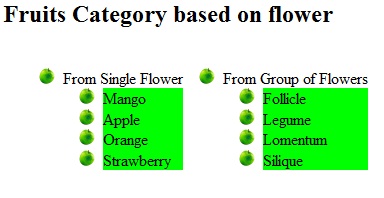
1. <script type="text/javascript" src="//ajax.googleapis.com/ajax/libs/jquery/1.10.1/jquery.min.js"></script>

Our UI design is ready so now implement the fade methods. Here are the four types of jQuery fade methods:

1. fadeIn()
2. fadeOut()
3. fadeToggle()
4. fadeTo()

Learn more here: [jQuery Effects Using Fade Methods](http://www.c-sharpcorner.com/UploadFile/3d39b4/jquery-effects-using-fade-methods/)

Question 46: How to work with jQuery css() method?

The following code is very simple, no styles are applied. Now, we want to achieve the following style by using jQuery and some CSS.  
  
   
  


Learn more here:  [JQuery and CSS Selectors: Part 2](http://www.c-sharpcorner.com/UploadFile/abhikumarvatsa/jquery-and-css-selectors-part-2/)

Question 47: What is queue() in Jquery? Use of queue() in jquery?

Delay comes under the custom effect category in jQuery. Its sole use is to delay the execution of subsequent items in the execution queue.   
  
*delay( duration [, queueName ] )*

*queueName* is a name of the queue in which the delay time is to be inserted. By default it is a "fx" queue. A "fx" queue is also known as an effects queue.

Learn more here: [Timers in jQuery: Delay Method](http://www.c-sharpcorner.com/UploadFile/4aac15/timers-in-jquery-delay-method)

Question 48: How jQuery selectors are executed?

A selector starts with $(). In the parentheses may be an element, a class or an ID. For example:

1. *<div class=”leftBorder”> C# Corner</div>*
2. *<div ID=”leftPanel”>C# Corner</div>*

For the preceding code, jQuery syntax (for selectors ) will be:

1. $(“div”).action
2. $(“.leftBorder”).action
3. $(“#leftPanel”).action

So here we used the following three things: HTML tag name, class name and ID name. There are jQuery selectors. The factory function $() is a synonym of the jQuery() function.

Learn more here: [jQuery Selectors Basics](http://www.c-sharpcorner.com/UploadFile/79037b/jquery-selectors-basics/)

Question 49: What are the advantages of Ajax?

Ajax stands for Asynchronous JavaScript and XML; in other words Ajax is the combination of various technologies such as JavaScript, CSS, XHTML, and DOM, etc.  
  
AJAX allows web pages to be updated asynchronously by exchanging small amounts of data with the server behind the scenes. This means that it is possible to update parts of a web page, without reloading the entire page.  
  
We can also define Ajax is a combination of client side technologies that provides asynchronous communication between the user interface and the web server so that partial page rendering occurs instead of complete page post back.  
 **Advantages of AJAX based application**

* Improved application performance by reducing the amount of data downloaded from the server.
* Rich, responsive and Slick UI with no page flickers.
* Eliminates frequent page refresh which usually happens in a typical request/response model (Everything is updated on fly).
* Easy to implement as there are variety of AJAX implementations available around.
* AJAX mechanism works behind the scene nothing much required from user perspective.
* Works with all the web browsers.
* Avoids the round trips to the server.
* Rendering of webpage faster.
* Decreases the consumption of server resources.
* Response time of application is faster.
* Rendering of data is dynamic.

Learn more here: [Introduction to Ajax and Ajax Control Toolkit](http://www.c-sharpcorner.com/UploadFile/0c1bb2/introduction-to-ajax-and-ajax-control-toolkit-part-1/)

Question 50: How can you use array with jQuery?

Arrays are zero indexed, ordered lists of values. They are really handy for storing a set of values of the same data type.  
  
*var names = [“Name1”,”Name2”] //Recommended*  
  
Both of the preceding approaches are kind of static declarations. Now let's do some dynamic programming with Arrays.

1. var namearray = [];
2. namearray.push(“Name1”) //Index 0
3. namearray.push(“Name2”) //Index 1
4. namearray.push(“Name3”) //Index 2

Here, .push() is a jQuery function used in conjunction with Arrays that adds an element at the end of the array. Items can be inserted by specifying the index as well, as follows:

1. namearray[0] = “Name1”;
2. namearray[1] = “Name2”;
3. namearray[2] = “Name3”;

Now let’s print the values of the array:

1. Console.log(namearray);

The statement above will produce the output as [ *"Name1", "Name2",”Name3”*].  
  
We can see that we just printed the array object but not the individual values, so to extract individual values the following statement can be executed:

1. Console.log(namearray[0]) //Name1;
2. Console.log(namearray[1]) //Name2;

How to print an array of values using a for loop in jQuery:

1. var myArray = ["Name1", "Name2", "Name3"];
2. **for** (var i = 0; i < myArray.length; i = i + 1) {
3. console.log(myArray[i]);
4. }

How to print an array of values using $.each() in jQuery:

1. $.each(myArray, function (index, value) {
2. console.log(index + ": " + value);
3. });

Top 30 jQuery Interview Questions

Q1) What is jQuery?

[jQuery](https://mindmajix.com/jquery-tutorial) can simply be interpreted as most advanced JavaScript framework which is used for performing high-level applications which include feature-rich, fast,and  lightweight framework which assists in to through the HTML DOM, add Ajax interaction,build animations, manipulation of the page content, modify the alignment and render cool UI effect. This is a client-side scripting language.

Q2) Can differentiate between jQuery and JavaScript?

The difference between the jQuery and the JavaScript can simply be explained.   
JavaScript can simply be interpreted as a high-level application language whereas, jQuery can be interpreted as a library which is built within the JavaScript language & helps in the better functioning of JavaScript.

Q3) Can a jQuery library be used for server scripting?

jQuery can be interpreted as an internally built library which is being used in the JavaScript.   
And jQuery is strictly been designed with the functionality of client-side scripting. jQuery is not compatible with server-side scripting.

Q4) Explain the basic requirement, to begin with, jQuery?

In order to get begin with the jQuery, all that one needs to do at the beginning is to make a reference to its complete library. After this, you will be having access to all the application files which are present within the library. jQuery latest version can now get downloaded from jQuery.com.

Q5) Why exactly is the need for the use of jQuery?

jQuery is a highly programmed internal library for the better functioning of JavaScript. It handles a number of applications and its prominence can never be ignored. The following list will be depicting the reasons for why is the need to use jQuery?  
**\***jQuery comes very handily for the purpose of designing user-friendly web applications.  
**\*** By making an effective use of jQuery the performance of an application can be greatly enhanced.  
**\*** Very fast and extensible.  
**\*** Only minimal lines of codes will be sufficient for writing UI related functions.

Q6) State different jQuery methods that are used to provide effects?

jQuery has a set of methods implying which we can expect to provide different effects within the system. Some of the jQuery methods include    
**\*** FadeOut()  
**\*** Toggle()  
**\*** Hide()  
**\*** Show()  
**\*** FadeIn() and

Q7) Describe the minimized version of the jQuery and its benefits.

With the minimized version of the jQuery, the efficiency of the webpage can be greatly improved. The effectiveness of the webpage which is having a minimized version of jQuery will be a lot better than the effectiveness of the webpage which is having a normal js file.

This is the reason why most of the web pages are nowadays are found to have a minimized version of the jQuery.

Q8) Explain about different scenarios where jQuery can be effectively implemented?

jQuery is a highly advanced library application which extensively is used in following scenarios:  
**\*** Apply CSS static or dynamic functions.  
**\*** Calling functions on events.  
**\*** For the purpose of manipulation.  
**\*** For creating different animation effects.

Having the presence of all the advanced library applications jQuery is best at maintaining different applications with a higher degree of effectiveness.

Q9) Explain about the jQuery connect?

jQuery connect can simply be interpreted as a plug-in which is used to connect or bind one function to the another. This will be helpful for executing the fro any other function or the plug-in is executed.

Q10) Which parameters are being used for the jQuery Ajax method?

JQuery Ajax method makes use of four different parameters which include  
**\*** URL – The URL must be clearly specified in order to send the request.   
**\*** type – Specifies the type of request(Get or Post)  
**\*** data – Specifies data to be sent to the server  
**\*** Cache – This tells if the browser should index the specific page or not.

[Checkout jQuery Tutorials](https://mindmajix.com/jquery-tutorial)

Q11) Explain the ways through which we can include jQuery on a page?

Including jQuery in a page can be done with the help of the following ways.   
**\*** Local copy inside script tag  
**\***Remote copy of [jQuery.com](http://jquery.com/)  
**\*** By keeping a remote copy of Ajax API  
**\*** Making a local copy of script manager control  
**\*** Embedded script using client script object

Q12) Differentiate the concepts of ID selector and class selector in jQuery?

Just like they are used in the CSS, both ID selector and Class selector are used for the same functioning .

In order explain the functioning of both the concepts in simpler words, ID selector uses ID while class selector makes use of a class to select the different elements.

In the case where you are needed to select only one element then we have to go with the ID selector. And if you want to select a group of elements, then we have to go with the class selector.

jQuery Practical Interview Questions & Answers

Q13) Explain the exact purpose of the animate function in jQuery.

The animate function is extensively being used in the jQuery. It is an advanced concept which is well received by the audience. Well, the main feature of the animation function in jQuery is to apply the custom animation effect to different elements.

The syntax for this function is as follows  
Syntax:

$(selector).animate({params}, [duration], [easing], [callback])

Here,  
**\*** "param" defines the CSS properties where are intended to apply the animation.  
**\*** "duration" it simply depicts for what exact period of time the animation will run. Its values are:   "slow", "fast", "normal" or it can even be milliseconds.  
**\*** "easing" is the string which specifies the function for the transition.  
**\*** "callback" it is the function with the help of which we can call an action upon the completion of the animation.

Q14) What is JQuery.noConflict?

In order to help the system to overcome the conflicts between the different libraries and frameworks the jQuery no-conflict option given by the jQuery. This concept is really advanced which has been developed over the recent times. In the case of any conflicts which arise between the libraries and frameworks using this jQuery.noConflict function will help in overcoming all the conflicts.

Q15) Explain the concept of fade effect in jQuery?

1. In jQuery there are three different applications applying which we can avail the fade effect.

 Fade effect can be availed by using the functions which are fadeIn, fadeOut and fadeTo.

2. The opacity of elements gets changed with animation through the effect of these methods.   
The syntax for the fading effect includes   
Syntax:

$(selector).fadeIn(speed,callback)  
$(selector).fadeOut(speed,callback)  
$(selector).fadeTo(speed,opacity,callback)

 “speed” this can be one among the following values: “slow”, “fast”, “normal” or milliseconds.  
“opacity” this is used to specify the capacity of fading to give opacity.  
“callback” it is the function with the help of which we can call an action upon the completion of fading.

Q16) Explain the Struts2 jQuery plug-in and its advantages.

The struts2 jQuery plug-in is used to:  
\* Easy integration of Ajax and widgets will be provided to the user.  
 \* The extent of coding needed to be done will be reduced.

For ex:

Result Div

The advantages of using truts2 jQuery are:

1. It greatly reduces the extent of coding needed to be done for a particular action.  
2. It also supports ajax form validation.  
3. The readability levels of the code will be very high and so it will become much easier to perform the debugging.

Q17) Explain how the jQuery stores data related to an element?

Information of an element can be stored in the simple javascript by simply adding a domain object model property to the element. This results in the rising of issues such as leakage of memory in the browsers.   
But in the case of jQuery the user does not need to be bothered by the memory management issues.

For ex : Storing and retrieving data related to an element:

$(‘#myDiv’).data(‘keyName’, { foo : ‘bar’ });

$(‘#myDiv’).data(‘keyName’); // { foo : ‘bar’ }

Q18) Describe the procedure of extracting a query string with regular expressions?

The procedure of extracting a query string with regular expressions can be done in two ways.  
1. Approach of String-based:   
It is considered to be the simplest way in use for extracting a query string with regular expressions. This method is carried ahead by making the utilisation of the .replace() method to function.

For ex :

var data = string.replace("http://localhost/view.php?", "");

The above procedure will be fine but it has some flexible issues.

2. Regular expression approach:  
This is the most powerful method for extracting a query string and the pattern must be used that seeks about the question mark among the string. When done, the JS regular expressions are delimited

Q19) Differentiate the concepts of .js and .min.js?

jQuery library has two different versions Development and Production. The other name for the deployment version is minified version.

Considering the functionality, both the files they are much similar to each other. Being smaller in size, the  .min.js   gets loaded quickly saving the bandwidth.

Q20) What is a CDN?

The term CDN relates to the concept of Content Delivery Network. In general, CDN can be interpreted as a large system of servers that are deployed in multiple data centers across the internet.

CDN is mainly aimed at delivering the content to the end-users by enabling them to have high availability and high performance.

Q21) Explain the advantages of the CDN.

The benefits of CDN are quite effective in relation to the working functionality of the jQuery.  
Some of the benefits of the CDN include

**\***CDN greatly diminishes the load from your server.  
**\*** Saving the bandwidth CDN helps the jQuery to load faster than the usual time  
**\*** The most prominent feature of the CDN is that it if any user visits a website which is running on jQuery it caches it.

Q22) What is an event?PreventDefault?

The event.preventDefault() method function is to stop the default action of an element from taking place  or to halt the default action from happening.

Q23) Differentiate onload() and document.ready().

The main differentiating factor of onload() and document.ready() is that

At max, a page supports the existence of only one onload function, & all the other onload functions will be getting terminated

While coming to the document.ready function, a page can have more than one document.ready function.   
   
Also, the document.ready function is called when the DOM gets loaded and whereas the onload function gets called only when the DOM images get loaded on the page.

Q24)  Why is jQuery better than JavaScript?

In general, jQuery is considered to be far better than the JavaScript. jQuery can be interpreted as an application for developing Ajax application and helps in simplifying the code & the code readability will also be good.

jQuery also supports the creation of animations and delivers such high-level functionalities.

Related Page: [AngularJS Vs. JQuery](https://mindmajix.com/angularjs-vs-jquery)

Q25) Explain the types of selectors in jQuery?

There are three types of selectors in jQuery:  
**\*** CSS Selector  
**\***XPath Selector  
**\*** Custom Selector

Q26) Explain the concept of Grouping?

The concept of grouping can simply be explained as whenever the same declaration gets shared by more than one selector they get separated by a common separated list.

This helps in greatly reducing the size of the CSS and helps it in making it become more readable.

Q27) Explain the concept of finish method in jQuery?

In order to stop all the queued animations and to help them place in their final state the .finish() method is used. This method was introduced in jQuery 1.9.

Q28) Differentiate between calling stop (true, true) and finish method?

The .finish() method will be quite similar to .stop(true, true). It helps in clearing the queue & thereby helps the animations to enter into their end state. In .finish() it also causes the CSS property of all queued animations to jump to their end values, as well.

Q29) Which is the latest version of the jQuery library?

The latest version (when this post is written) of jQuery is 1.10.2 or 2.0.3. jQuery 2.x has the same API as jQuery 1.x but does not support Internet Explorer 6, 7, or 8.

Explore jQuery Sample Resumes! Download & Edit, Get Noticed by Top Employers![**Download Now!**](https://mindmajix.com/jquery-sample-resumes)

Q30) Explain the concept of writing the browser specific code using jQuery?

We can write browser specific code by simply making use of the jQuery.browser property. This property contains flags for the user agent, read from navigator.userAgent.

This property was excluded from jQuery 1.9.

Q31) What is the procedure to check the data type of any variable in jQuery?

By using $.type(Object) which returns the built-in JavaScript type for the object.

**1. What is jQuery Selectors? Give some examples**

* jQuery Selectors are used to select one or a group of HTML elements from your web page.
* jQuery support all the CSS selectors as well as many additional custom selectors.
* jQuery selectors always start with dollar sign and parentheses: $()

There are three building blocks to select the elements in a web document.

1. Select elements by tag name

**Example**

**$(div)**  
It will select all the div elements in the document.

2. Select elements by ID

**Example**

**$(“#abc”)**  
It will select single element that has an ID of abc.

3. Select elements by Class

**Example**

**$(“.xyzClass”)**  
It will select all the elements having class xyzClass.

**2. What is jQuery?**

**jQuery**is a lightweight JavaScript library which gives a quick and simple method for **HTML DOM traversing and manipulation**, its event handling, its client-side animations, and so on. One of the best features of jQuery is that jQuery supports an efficient way to implement [AJAX](https://www.onlineinterviewquestions.com/ajax-interview-questions/)applications because of its lightweight nature and make normalize and efficient web programs.

**3. What are the advantages of JQuery ?**

**There are many advantages of using JQuery. Some of them are :**

* It is more like a JavaScript enhancement so there is no overhead in learning a new syntax.
* It has the ability to keep the code simple, readable, clear and reusable.
* Cross-browser support (IE 6.0+, FF 1.5+, Safari 2.0+, Opera 9.0+)
* It would eradicate the requirement for writing complex loops and DOM scripting library calls.
* Event detection and handling.
* Tons of plug-ins for all kind of needs.

**4. What are the various ajax functions available in Jquery ?**

Ajax allows the user to exchange data with a server and update parts of a page without reloading the entire page. Some of the functions of ajax are as follows:

**$.ajax()**: This is considered to be the most low level and basic of functions. It is used to send requests . This function can be performed without a selector.

**$.ajaxSetup()**: This function is used to define and set the options for various ajax calls.

**For example.**

$.ajaxSetup({

"type":"POST",

"url":"ajax.php",

"success":function(data){

$("#bar")

.css("background","yellow")

.html(data);

}

});

**Shorthand ajax methods**: They comprise of simply the wrapper function that call $.ajax() with certain parameters already set.

**$.getJSON()**: this is a special type of shorthand function which is used to accept the url to which the requests are sent. Also optional data and optional callback functions are possible in such functions.

**5. Explain .empty() vs .remove() vs .detach() in Jquery**

* **.empty()** method is used to remove all the child elements from matched elements.
* **.remove()** method is used to remove all the matched element. This method will remove all the jQuery data associated with the matched element.
* **.detach()** method is same as .remove() method except that the .detach() method doesn’t remove jQuery data associated with the matched elements.

**.remove()** is faster than .empty() or .detach() method.

**Syntax:**

$(selector).empty();

$(selector).remove();

$(selector).detach();

**6. Explain width() vs css(‘width’) in JQuery**

In jQuery, there is two way to change the width of an element. One way is using .css(‘width’) and other way is using .width().

**For example**

$('#mydiv').css('width','300px');

$('#mydiv').width(100);

* The difference in .css(‘width’) and .width() is the data type of value we specify or return from the both functions.
* In .css(‘width’) we have to add “px” in the width value while in .width() we don’t have to add.
* When you want to get the width of “mydiv” element then .css(‘width’) will return ‘300px’ while .width() will return only integer value 300.

**7. Explain bind() vs live() vs delegate() methods in Jquery**

**The bind()** method will not attach events to those elements which are added after DOM is loaded while **live()** and **delegate()** methods attach events to the future elements also.

The difference between **live()** and **delegate()** methods is **live()** function will not work in chaining. It will work only on an selector or an element while **delegate()** method can work in chaining.

**For example**

$(document).ready(function(){

$("#myTable").find("tr").live("click",function(){

alert($(this).text());

});

});

Above code will not work using live() method. But using delegate() method we can accomplish this.

$(document).ready(function(){

$("#dvContainer")children("table").delegate("tr","click",function(){

alert($(this).text());

});

});

**8. What is the use of param() method in JQuery .**

The param() method is used to represent an array or an object in serialize manner.  
While making an ajax request we can use these serialize values in the query strings of URL.

**Syntax:**

$.param(object | array, boolValue)

“object | array” specifies an array or an object to be serialized.

“boolValue” specifies whether to use the traditional style of param serialization or not.

**Example**

personObj=new Object();

empObject.name="Ravi";

empObject.age="28";

empObject.dept="IT";

$("#clickme").click(function(){

$("span").text($.param(empObject));

});

It will set the text of span to “name=Ravi&age=28&dep=IT”

**9. What is difference between $(this) and this in jQuery ?**

$(document).ready(function(){

$('#clickme').click(function(){

alert($(this).text());

alert(this.innerText);

});

});

**this**and **$(this)** references the same element but the difference is that “this” is used in traditional way but when “this” is used with $() then it becomes a jQuery object on which we can use the functions of jQuery.?

In the example given, when only “this” keyword is used then we can use the jQuery text() function to get the text of the element, because it is not jQuery object. Once the “this” keyword is wrapped in $() then we can use the jQuery function text() to get the text of the element.

**10. How to read, write and delete cookies in jQuery ?**

To deal with cookies in jQuery we have to use the Dough cookie plugin.

Dough is easy to use and having powerful features.

* **Create cookie:**  
  $.dough(“cookie\_name”, “cookie\_value”);
* **Read Cookie:**  
  $.dough(“cookie\_name”);
* **Delete cookie:**  
  $.dough(“cookie\_name”, “remove”);

**11. What is the difference between jquery.size() and jquery.length?**

jQuery **.size()** method returns number of element in the object. But it is not preferred to use the size() method as jQuery provide **.length** property and which does the same thing. But the .length property is preferred because it does not have the overhead of a function call.

**12. How can events be prevented from stopping to work after an ajax request?**

There are two ways to handle this issue:  
**Use of event delegation**: The event delegation technique works on principle by exploiting the event bubbling. It uses event bubbling to capture the events on elements which are present anywhere in the domain object model. In jquery the user can make use of the live and die methods for the events delegation which contains a subset of event types.  
**For example.** handling even delegation, handling of clicks on any <a> element:

$('#mydiv').click(function(e){

if( $(e.target).is('a') )

fn.call(e.target,e);

});

$('#mydiv').load('my.html')

**Event rebinding usage**: When this method is used it requires the user to call the bind method and the added new elements.

$('a').click(fn);

$('#mydiv').load('my.html',function(){

$('#mydiv a').click(fn);

});

**13. Why jQuery?**

jQuery is very compact and well-written JavaScript code that increases the productivity of the developer by enabling them to achieve critical UI functionality by writing very less amount of code.

It helps to

* Improve the performance of the application
* Develop most browser compatible web page
* Implement UI related critical functionality without writing hundreds of lines of codes
* Fast
* Extensible – jQuery can be extended to implement customized behavior

Other advantages of jQuery are

* No need to learn fresh new syntax’s to use jQuery, knowing simple JavaScript syntax is enough
* Simple and Cleaner code, no need to write several lines of codes to achieve complex functionality.

**14. What are the different type of selectors in Jquery?**

There are 3 types of selectors in Jquery

* CSS Selector
* XPath Selector
* Custom Selector

**15. What are features of JQuery or what can be done using JQuery?**

Features of Jquery

* One can easily provide effects and can do animations.
* Applying / Changing CSS.
* Cool plugins.
* Ajax support
* DOM selection events
* Event Handling

**16. How is body onload() function is different from document.ready() function used in jQuery?**

**Document.ready()** function is different from body onload() function because off 2 reasons.

* We can have more than one document.ready() function in a page where we can have only one onload function.
* Document.ready() function is called as soon as DOM is loaded where body.onload() function is called when everything gets loaded on the page that includes DOM, images and all associated resources of the page.

Q1. What is jQuery?

Ans: jQuery is **fast, lightweight and feature-rich**client side JavaScript Library/Framework which helps in to traverse HTML DOM, make animations, add Ajax interaction, manipulate the page content, change the style and provide cool UI effect. It is one of the most popular client side library and as per a survey it runs on every second website.

Q2. Why do we use jQuery?

Ans: Due to following advantages.

* Easy to use and learn.
* Easily expandable.
* Cross-browser support (IE 6.0+, FF 1.5+, Safari 2.0+, Opera 9.0+)
* Easy to use for DOM manipulation and traversal.
* Large pool of built in methods.
* AJAX Capabilities.
* Methods for changing or applying CSS, creating animations.
* Event detection and handling.
* Tons of plug-ins for all kind of needs.

Q3. How JavaScript and jQuery are different?

Ans: JavaScript is a language While jQuery is a library built in the JavaScript language that helps to use the JavaScript language.

Q4. Is jQuery replacement of Java Script?

Ans: **No.** jQuery is not a replacement of JavaScript. jQuery is a different library which is written on top of JavaScript. jQuery is a lightweight JavaScript library that emphasizes interaction between JavaScript and HTML.

Q5. Is jQuery a library for client scripting or server scripting?

Ans. Client side scripting.

Q6. Is jQuery a W3C standard?

Ans: No. jQuery is not a W3C standard.

Q7. What is the basic need to start with jQuery?

Ans: To start with jQuery, one need to make reference of it's library. The latest version of jQuery can be downloaded from [jQuery.com](http://jquery.com/).

Q8. Which is the starting point of code execution in jQuery?

Ans: The starting point of jQuery code execution is $(document).ready() function which is executed when DOM is loaded.

Q9. What does dollar sign ($) means in jQuery?

Ans: Dollar Sign is nothing but it's an alias for JQuery. Take a look at below jQuery code.

Hide   Copy Code

$(document).ready(function(){

});

Over here $ sign can be replaced with "jQuery" keyword.

Hide   Copy Code

jQuery(document).ready(function(){

});

Q10. Can we have multiple document.ready() function on the same page?

Ans: **YES**. We can have any number of document.ready() function on the same page.

Q11. Can we use our own specific character in the place of $ sign in jQuery?

Ans: **Yes.** It is possible using jQuery.noConflict().

Q12. Is it possible to use other client side libraries like MooTools, Prototype along with jQuery?

Ans: **Yes.**

Q13. What is jQuery.noConflict?

Ans: As other client side libraries like MooTools, Prototype can be used with jQuery and they also use $() as their global function and to define variables. This situation creates conflict as $() is used by jQuery and other library as their global function. To overcome from such situations, jQuery has introduced jQuery.noConflict().

Hide   Copy Code

jQuery.noConflict();

// Use jQuery via jQuery(...)

jQuery(document).ready(function(){

jQuery("div").hide();

});

You can also use your own specific character in the place of $ sign in jQuery.

Hide   Copy Code

var $j = jQuery.noConflict();

// Use jQuery via jQuery(...)

$j(document).ready(function(){

$j("div").hide();

});

Q14. Is there any difference between body onload() and document.ready() function?

Ans: document.ready() function is different from body onload() function for 2 reasons.

1. We can have more than one document.ready() function in a page where we can have only one body onload function.
2. document.ready() function is called as soon as DOM is loaded where body.onload() function is called when everything gets loaded on the page that includes DOM, images and all associated resources of the page.

Q15. What is the difference between .js and .min.js?

Ans: jQuery library comes in 2 different versions Development and Production/Deployment. The deployment version is also known as minified version. So .min.js is basically the minified version of jQuery library file. Both the files are same as far as functionality is concerned. but .min.js is quite small in size so it loads quickly and saves bandwidth.

Q16. Why there are two different version of jQuery library?

Ans: jQuery library comes in 2 different versions.

1. Development
2. Production/Deployment

The development version is quite useful at development time as jQuery is open source and if you want to change something then you can make those changes in development version. But the deployment version is minified version or compressed version so it is impossible to make changes in it. Because it is compressed, so its size is very less than the production version which affects the page load time.

Q17. What is a CDN?

Ans: A content delivery network or content distribution network (CDN) is a large distributed system of servers deployed in multiple data centers across the Internet. The goal of a CDN is to serve content to end-users with high availability and high performance.

Q18. Which are the popular jQuery CDN? and what is the advantage of using CDN?

Ans: There are 3 popular jQuery CDNs.

1. 1. Google.
2. 2. Microsoft
3. 3. jQuery.

Advantage of using CDN.

* It reduces the load from your server.
* It saves bandwidth. jQuery framework will load faster from these CDN.
* The most important benefit is it will be cached, if the user has visited any site which is using jQuery framework from any of these CDN

Q19. How to load jQuery from CDN?

Ans: Below is the code to load jQuery from all 3 CDNs.  
**Code to load jQuery Framework from Google CDN**

Hide   Copy Code

<script type="text/javascript"

src="http://ajax.googleapis.com/ajax/libs/jquery/1.9.1/jquery.min.js">

</script>

**Code to load jQuery Framework from Microsoft CDN**

Hide   Copy Code

<script type="text/javascript"

src="http://ajax.microsoft.com/ajax/jquery/jquery-1.9.1.min.js">

</script>

**Code to load jQuery Framework from jQuery Site(EdgeCast CDN)**

Hide   Copy Code

<script type="text/javascript"

src="http://code.jquery.com/jquery-1.9.1.min.js">

</script>

Q20. How to load jQuery locally when CDN fails?

Ans: It is a good approach to always use CDN but sometimes what if the CDN is down (rare possibility though) but you never know in this world as anything can happen.  
  
Below given jQuery code checks whether jQuery is loaded from Google CDN or not, if not then it references the jQuery.js file from your folder.

Hide   Copy Code

<script type="text/javascript">

if (typeof jQuery == 'undefined')

{

document.write(unescape("%3Cscript src='Scripts/jquery.1.9.1.min.js' type='text/javascript'%3E%3C/script%3E"));

}

</script>

It first loads the jQuery from Google CDN and then check the jQuery object. If jQuery is not loaded successfully then it will references the jQuery.js file from hard drive location. In this example, the jQuery.js is loaded from Scripts folder.

Q21. What are selectors in jQuery and how many types of selectors are there?

Ans: To work with an element on the web page, first we need to find them. To find the html element in jQuery we use selectors. There are many types of selectors but basic selectors are:

* Name: Selects all elements which match with the given element Name.
* #ID: Selects a single element which matches with the given ID
* .Class: Selects all elements which match with the given Class.
* Universal (\*): Selects all elements available in a DOM.
* Multiple Elements E, F, G: Selects the combined results of all the specified selectors E, F or G.
* Attribute Selector: Select elements based on its attribute value.

Q22. How do you select element by ID in jQuery?

Ans: To select element use ID selector. We need to prefix the id with "#" (hash symbol). For example, to select element with ID "txtName", then syntax would be,

Hide   Copy Code

$('#txtName')

Q23. What does $("div") will select?

Ans: This will select all the div elements on page.

Q24. How to select element having a particular class (".selected")?

Ans: $('.selected'). This selector is known as class selector. We need to prefix the class name with "." (dot).

Q25. What does $("div.parent") will select?

Ans: All the div element with parent class.

Q26. What are the fastest selectors in jQuery?

Ans: ID and element selectors are the fastest selectors in jQuery.

Q27. What are the slow selectors in jQuery?

Ans: class selectors are the slow compare to ID and element.

Q28. How jQuery selectors are executed?

Ans: Your last selectors is always executed first. For example, in below jQuery code, jQuery will first find all the elements with class ".myCssClass" and after that it will reject all the other elements which are not in "p#elmID".

Hide   Copy Code

$("p#elmID .myCssClass");

Q29. Which is fast document.getElementByID('txtName') or $('#txtName').?

Ans: Native JavaScipt is always fast. jQuery method to select txtName "$('#txtName')" will internally makes a call to document.getElementByID('txtName'). As jQuery is written on top of JavaScript and it internally uses JavaScript only So JavaScript is always fast.

Q30. Difference between $(this) and 'this' in jQuery?

Ans: this and $(this) refers to the same element. The only difference is the way they are used. 'this' is used in traditional sense, when 'this' is wrapped in $() then it becomes a jQuery object and you are able to use the power of jQuery.

Hide   Copy Code

$(document).ready(function(){

$('#spnValue').mouseover(function(){

alert($(this).text());

});

});

In below example, this is an object but since it is not wrapped in $(), we can't use jQuery method and use the native JavaScript to get the value of span element.

Hide   Copy Code

$(document).ready(function(){

$('#spnValue').mouseover(function(){

alert(this.innerText);

});

});

Q31. How do you check if an element is empty?

Ans: There are 2 ways to check if element is empty or not. We can check using ":empty" selector.

Hide   Copy Code

$(document).ready(function(){

if ($('#element').is(':empty')){

//Element is empty

}

});

And the second way is using the "$.trim()" method.

Hide   Copy Code

$(document).ready(function(){

if($.trim($('#element').html())=='') {

//Element is empty

}

});

Q32. How do you check if an element exists or not in jQuery?

Ans: Using jQuery length property, we can ensure whether element exists or not.

Hide   Copy Code

$(document).ready(function(){

if ($('#element').length > 0){

//Element exists

}

});

Q33. What is the use of jquery .each() function?

Ans: The $.each() function is used to iterate over a jQuery object. The $.each() function can be used to iterate over any collection, whether it is an object or an array.

Q34. What is the difference between jquery.size() and jquery.length?

Ans: jQuery .size() method returns number of element in the object. But it is not preferred to use the size()method as jQuery provide .length property and which does the same thing. But the .length property is preferred because it does not have the overhead of a function call.

Q35. What is the difference between $('div') and $('<div/>') in jQuery?

Ans: $('<div/>') : This creates a new div element. However this is not added to DOM tree unless you don't append it to any DOM element.  
  
$('div') : This selects all the div element present on the page.

Q36. What is the difference between parent() and parents() methods in jQuery?

Ans: The basic difference is the parent() function travels only one level in the DOM tree, where parents() function search through the whole DOM tree.

Q37. What is the difference between eq() and get() methods in jQuery?

Ans: eq() returns the element as a jQuery object. This method constructs a new jQuery object from one element within that set and returns it. That means that you can use jQuery functions on it.  
  
get() return a DOM element. The method retrieve the DOM elements matched by the jQuery object. But as it is a DOM element and it is not a jQuery-wrapped object. So jQuery functions can't be used. Find out more [**here**](http://jquerybyexample.blogspot.com/2013/04/jquery-difference-between-eq-and-get-method.html).

Q38. How do you implement animation functionality?

Ans: The .animate() method allows us to create animation effects on any numeric CSS property. This method changes an element from one state to another with CSS styles. The CSS property value is changed gradually, to create an animated effect.  
  
Syntax is:

Hide   Copy Code

(selector).animate({styles},speed,easing,callback)

* styles: Specifies one or more CSS properties/values to animate.
* duration: Optional. Specifies the speed of the animation.
* easing: Optional. Specifies the speed of the element in different points of the animation. Default value is "swing".
* callback: Optional. A function to be executed after the animation completes.

Simple use of animate function is,

Hide   Copy Code

$("btnClick").click(function(){

$("#dvBox").animate({height:"100px"});

});

Q39. How to disable jQuery animation?

Ans: Using jQuery property "**jQuery.fx.off**", which when set to true, disables all the jQuery animation. When this is done, all animation methods will immediately set elements to their final state when called, rather than displaying an effect.

Q40. How do you stop the currently-running animation?

Ans: Using jQuery ".stop()" method.

Q41. What is the difference between .empty(), .remove() and .detach() methods in jQuery?

Ans: All these methods .empty(), .remove() and .detach() are used for removing elements from DOM but they all are different.  
  
.empty(): This method removes all the child element of the matched element where remove() method removes set of matched elements from DOM.  
  
.remove(): Use .remove() when you want to remove the element itself, as well as everything inside it. In addition to the elements themselves, all bound events and jQuery data associated with the elements are removed.  
  
.detach(): This method is the same as .remove(), except that .detach() keeps all jQuery data associated with the removed elements. This method is useful when removed elements are to be reinserted into the DOM at a later time.  
  
Find out more [**here**](http://jquerybyexample.blogspot.com/2012/05/empty-vs-remove-vs-detach-jquery.html)

Q42. Explain .bind() vs .live() vs .delegate() vs .on()

Ans: All these 4 jQuery methods are used for attaching events to selectors or elements. But they all are different from each other.  
  
<a href="http://jquerybyexample.blogspot.com/2010/06/jquery-bind-function-exampledemo.html">.bind()</a>: This is the easiest and quick method to bind events. But the issue with bind() is that it doesn't work for elements added dynamically that matches the same selector. bind() only attach events to the current elements not future element. Above that it also has performance issues when dealing with a large selection.  
  
<a href="http://jquerybyexample.blogspot.com/2010/06/jquery-live-function-exampledemo.html">.live()</a>: This method overcomes the disadvantage of bind(). It works for dynamically added elements or future elements. Because of its poor performance on large pages, this method is deprecated as of jQuery 1.7 and you should stop using it. Chaining is not properly supported using this method.  
  
<a href="http://jquerybyexample.blogspot.com/2010/08/jquery-delegate-function-exampledemo.html">.delegate()</a>: The .delegate() method behaves in a similar fashion to the .live() method, but instead of attaching the selector/event information to the document, you can choose where it is anchored and it also supports chaining.  
  
.on(): Since live was deprecated with 1.7, so new method was introduced named ".on()". This method provides all the goodness of previous 3 methods and it brings uniformity for attaching event handlers.  
  
Find out more [**here**](http://www.elijahmanor.com/2012/02/differences-between-jquery-bind-vs-live.html)

Q43. What is wrong with this code line "$('#myid.3').text('blah blah!!!');"

Ans: The problem with above statement is that the selectors is having meta characters and to use any of the meta-characters ( such as !"#$%&'()\*+,./:;<=>?@[\]^`{|}~ ) as a literal part of a name, it must be escaped with with two backslashes: \\. For example, an element with id="foo.bar", can use the selector $("#foo\\.bar").  
So the correct syntax is,

Hide   Copy Code

$('#myid\\.3').text('blah blah!!!');

Q44. How to create clone of any object using jQuery?

Ans: jQuery provides clone() method which performs a deep copy of the set of matched elements, meaning that it copies the matched elements as well as all of their descendant elements and text nodes.

Hide   Copy Code

$(document).ready(function(){

$('#btnClone').click(function(){

$('#dvText').clone().appendTo('body');

return false;

});

});

Q45. Does events are also copied when you clone any element in jQuery?

Ans: As explained in previous question, using clone() method, we can create clone of any element but the default implementation of the clone() method doesn't copy events unless you tell the clone() method to copy the events. The clone() method takes a parameter, if you pass true then it will copy the events as well.

Hide   Copy Code

$(document).ready(function(){

$("#btnClone").bind('click', function(){

$('#dvClickme').clone(true).appendTo('body');

});

​

Q46. What is difference between prop and attr?

Ans: attr(): Get the value of an attribute for the first element in the set of matched elements. Whereas,.prop(): (Introduced in jQuery 1.6) Get the value of a property for the first element in the set of matched elements.  
  
Attributes carry additional information about an HTML element and come in name="value" pairs. Where Property is a representation of an attribute in the HTML DOM tree. once the browser parse your HTML code ,corresponding DOM node will be created which is an object thus having properties.  
  
attr() gives you the value of element as it was defines in the html on page load. It is always recommended to use prop() to get values of elements which is modified via javascript/jquery , as it gives you the original value of an element's current state. Find out more [here](http://techbrij.com/jquery-attr-vs-prop-difference).

Q47. What is event.PreventDefault?

Ans: The event.preventDefault() method stops the default action of an element from happening. For example, Prevents a link from following the URL.

Q48. What is the difference between event.PreventDefault and event.stopPropagation?

Ans: event.preventDefault(): Stops the default action of an element from happening.  
event.stopPropagation(): Prevents the event from bubbling up the DOM tree, preventing any parent handlers from being notified of the event. For example, if there is a link with a click method attached inside of a DIV or FORM that also has a click method attached, it will prevent the DIV or FORM click method from firing.

Q49. What is the difference between event.PreventDefault and "return false"?

Ans: e.preventDefault() will prevent the default event from occurring, e.stopPropagation() will prevent the event from bubbling up and return false will do both.

Q50. What is the difference between event.stopPropagation and event.stopImmediatePropagation?

Ans: event.stopPropagation() allows other handlers on the same element to be executed, while event.stopImmediatePropagation() prevents every event from running. For example, see below jQuery code block.

Hide   Copy Code

$("p").click(function(event){

event.stopImmediatePropagation();

});

$("p").click(function(event){

// This function won't be executed

$(this).css("background-color", "#f00");

});

If event.stopPropagation was used in previous example, then the next click event on p element which changes the css will fire, but in case event.stopImmediatePropagation(), the next p click event will not fire.

Q51. How to check if number is numeric while using jQuery 1.7+?

Ans: Using "<a href="http://jquerybyexample.blogspot.com/2011/11/jqueryisnumeric-in-jquery-17.html">isNumeric()</a>" function which was introduced with jQuery 1.7.

Q52. How to check data type of any variable in jQuery?

Ans: Using [$.type(Object)](http://jquerybyexample.blogspot.in/2012/04/common-utility-methods-of-jquery.html) which returns the built-in JavaScript type for the object.

Q53. How do you attach a event to element which should be executed only once?

Ans: Using jQuery one() method. This attaches a handler to an event for the element. The handler is executed at most once per element. In simple terms, the attached function will be called only once.

Hide   Copy Code

$(document).ready(function() {

$("#btnDummy").one("click", function() {

alert("This will be displayed only once.");

});

});​

Q54. Can you include multiple version of jQuery? If yes, then how they are executed?

Ans: Yes. Multiple versions of jQuery can be included in same page.

Q55. In what situation you would use multiple version of jQuery and how would you include them?

Ans: Well, it is quite possible that the jQuery plugins which are used are dependent on older version but for your own jQuery code, you would like to use newer version. So because of this dependency, multiple version of jQuery may required sometimes on single page.  
  
Below code shows how to include multiple version of jQuery.

Hide   Copy Code

<script type='text/javascript' src='js/jquery\_1.9.1.min.js'></script>

<script type='text/javascript'>

var $jq = jQuery.noConflict();

</script>

<script type='text/javascript' src='js/jquery\_1.7.2.min.js'></script>

By this way, for your own jQuery code use "$jq", instead of "$" as "$jq" refers to jQuery 1.9.1, where "$" refers to 1.7.2.

Q56. Is it possible to hold or delay document.ready execution for sometime?

Ans: Yes, its possible. With Release of jQuery 1.6, a new method "jQuery.holdReady(hold)" was introduced. This method allows to delay the execution of document.ready() event. document.ready() event is called as soon as your DOM is ready but sometimes there is a situation when you want to load additional JavaScript or some plugins which you have referenced.

Hide   Copy Code

​

$.holdReady(true);

$.getScript("myplugin.js", function() {

$.holdReady(false);

});

Q57. What is chaining in jQuery?

Ans: Chaining is one of the most powerful feature of jQuery. In jQuery, Chaining means to connect multiple functions, events on selectors. It makes your code short and easy to manage and it gives better performance. The chain starts from left to right. So left most will be called first and so on.

Hide   Copy Code

​$(document).ready(function(){

$('#dvContent').addClass('dummy');

$('#dvContent').css('color', 'red');

$('#dvContent').fadeIn('slow');

});​

The above jQuery code sample can be re-written using chaining. See below.

Hide   Copy Code

​$(document).ready(function(){

$('#dvContent').addClass('dummy')

.css('color', 'red')

.fadeIn('slow');

});​

Not only functions or methods, chaining also works with events in jQuery. Find out more [**here**](http://jquerybyexample.blogspot.com/2012/06/what-is-chaining-in-jquery.html).

Q58. How does caching helps and how to use caching in jQuery?

Ans: Caching is an area which can give you awesome performance, if used properly and at the right place. While using jQuery, you should also think about caching. For example, if you are using any element in jQuery more than one time, then you must cache it. See below code.

Hide   Copy Code

$("#myID").css("color", "red");

//Doing some other stuff......

$("#myID").text("Error occurred!");

​

Now in above jQuery code, the element with #myID is used twice but without caching. So both the times jQuery had to traverse through DOM and get the element. But if you have saved this in a variable then you just need to reference the variable. So the better way would be,

Hide   Copy Code

var $myElement = $("#myID").css("color", "red");

//Doing some other stuff......

$myElement.text("Error occurred!");

​

So now in this case, jQuery won't need to traverse through the whole DOM tree when it is used second time. So in jQuery, Caching is like saving the jQuery selector in a variable. And using the variable reference when required instead of searching through DOM again.

Q59. You get "jquery is not defined" or "$ is not defined" error. What could be the reason?

Ans: There could be many reasons for this.

* You have forgot to include the reference of jQuery library and trying to access jQuery.
* You have include the reference of the jQuery file, but it is after your jQuery code.
* The order of the scripts is not correct. For example, if you are using any jQuery plugin and you have placed the reference of the plugin js before the jQuery library then you will face this error.

Find out more [**here**](http://jquerybyexample.blogspot.com/2013/01/jquery-is-not-defined-error-reasons.html).

Q60. How to write browser specific code using jQuery?

Ans: Using jQuery.browser property, we can write browser specific code. This property contains flags for the useragent, read from navigator.userAgent. This property was removed in jQuery 1.9.

Q61. Can we use jQuery to make ajax request?

Ans: Yes. jQuery can be used for making ajax request.

Q62. What are various methods to make ajax request in jQuery?

Ans: Using below jQuery methods, you can make ajax calls.

* load() : Load a piece of html into a container DOM
* $.getJSON(): Load JSON with GET method.
* $.getScript(): Load a JavaScript file.
* $.get(): Use to make a GET call and play extensively with the response.
* $.post(): Use to make a POST call and don't want to load the response to some container DOM.
* $.ajax(): Use this to do something on XHR failures, or to specify ajax options (e.g. cache: true) on the fly.

Find out more [**here**](http://net.tutsplus.com/tutorials/javascript-ajax/5-ways-to-make-ajax-calls-with-jquery/).

Q63. Is there any advantage of using $.ajax() for ajax call against $.get() or $.post()?

Ans: By using jQuery post()/ jQuery get(), you always trust the response from the server and you believe it is going to be successful all the time. Well, it is certainly not a good idea to trust the response. As there can be n number of reason which may lead to failure of response.  
  
Where jQuery.ajax() is jQuery's low-level AJAX implementation. $.get and $.post are higher-level abstractions that are often easier to understand and use, but don't offer as much functionality (such as error callbacks). Find out more [**here**](http://jquerybyexample.blogspot.com/2011/11/avoid-jquerypost-use-jqueryajax.html).

Q64. What are deferred and promise object in jQuery?

Ans: Deferred and promise are part of jQuery since version 1.5 and they help in handling asynchronous functions like Ajax. Find out more [**here**](http://www.bitstorm.org/weblog/2012-1/Deferred_and_promise_in_jQuery.html).

Q65. Can we execute/run multiple Ajax request simultaneously in jQuery? If yes, then how?

Ans: Yes, it is possible to execute multiple Ajax request simultaneously or in parallel. Instead of waiting for first ajax request to complete and then issue the second request is time consuming. The better approach to speed up things would be to execute multiple ajax request simultaneously.  
  
Using jQuery .when() method which provides a way to execute callback functions based on one or more objects, usually Deferred objects that represent asynchronous events. Find out more [**here**](http://jquerybyexample.blogspot.com/2013/05/jquery-execute-multiple-ajax-request-simultaneously-in-parallel.html).

Q66. Can you call C# code-behind method using jQuery? If yes,then how?

Ans: Yes. We can call C# code-behind function via $.ajax. But for do that it is compulsory to mark the method as WebMethod.

Q67. Which is the latest version of jQuery library?

Ans: The latest version (when this post is written) of jQuery is 1.10.2 or 2.0.3. jQuery 2.x has the same API as jQuery 1.x, but does not support Internet Explorer 6, 7, or 8.

Q68. Does jQuery 2.0 supports IE?

Ans: No. jQuery 2.0 has no support for IE 6, IE 7 and IE 8.

Q69. What are source maps in jQuery?

Ans: In case of jQuery, Source Map is nothing but mapping of minified version of jQuery against the un-minified version. Source map allows to debug minified version of jQuery library. Source map feature was release with jQuery 1.9. Find out more [**here**](http://jquerybyexample.blogspot.com/2013/01/all-you-need-to-know-about-jquery-source-maps.html).

Q70. How to use migrate jQuery plugin?

Ans: with release of 1.9 version of jQuery, many deprecated methods were discarded and they are no longer available. But there are many sites in production which are still using these deprecated features and it's not possible to replace them overnight. So jQuery team provided with jQuery Migrate plugin that makes code written prior to 1.9 work with it.  
  
So to use old/deprecated features, all you need to do is to provide reference of jQuery Migrate Plugin. Find out more [**here**](http://jquerybyexample.blogspot.com/2013/01/migrate-older-jquery-code-to-jquery1-9.html).

Q71. Is it possible to get value of multiple CSS properties in single statement?

Ans: Well, before jQuery 1.9 release it was not possible but one of the new feature of jQuery 1.9 was .css()multi-property getter.

Hide   Copy Code

var propCollection = $("#dvBox").css([ "width", "height", "backgroundColor" ]);

In this case, the propCollection will be an array and it will look something like this.

Hide   Copy Code

{

width: "100px",

height: "200px",

backgroundColor: "#FF00FF"

}

Q72. How do you stop the currently-running animation, remove all queued animations, and complete all animations for the matched elements?

Ans: It can be done via calling .stop([clearQueue ] [, jumpToEnd ]) method and by passing both the parameters as true.

Q73. What is finish method in jQuery?

Ans: The .finish() method stops all queued animations and places the element(s) in their final state. This method was introduced in jQuery 1.9.

Q74. What is the difference between calling stop(true,true) and finish method?

Ans: The .finish() method is similar to .stop(true, true) in that it clears the queue and the current animation jumps to its end value. It differs, however, in that .finish() also causes the CSS property of all queued animations to jump to their end values, as well.

Q75. Consider a scenario where things can be done easily with javascript, would you still prefer jQuery?

Ans: No. If things can be done easily via CSS or JavaScript then You should not think about jQuery. Remember, jQuery library always comes with xx kilobyte size and there is no point of wasting bandwidth.

Q76. Can we use protocol less URL while referencing jQuery from CDNs?

Ans: Yes. Below code is completely valid.

Hide   Copy Code

<script type="text/javascript" src="//ajax.googleapis.com/ajax/libs/jquery/1.9.1/jquery.min.js"></script>

Q77. What is the advantage of using protocol less URL while referencing jQuery from CDNs?

Ans: It is quite useful when you are moving from HTTP to HTTPS url. You need to make sure that correct protocol is used for referencing jQuery library as pages served via SSL should contain no references to content served through unencrypted connections.  
  
"protocol-less" URL is the best way to reference third party content that’s available via both HTTP and HTTPS. When a URL’s protocol is omitted, the browser uses the underlying document’s protocol instead. Find out more [**here**](http://jquerybyexample.blogspot.com/2013/06/use-protocol-less-url-referencing-jquery.html).

Q78. What is jQuery plugin and what is the advantage of using plugin?

Ans: A plug-in is piece of code written in a standard JavaScript file. These files provide useful jQuery methods which can be used along with jQuery library methods. jQuery plugins are quite useful as its piece of code which is already written by someone and re-usable, which saves your development time.

Q79. What is jQuery UI?

Ans: jQuery UI is a curated set of user interface interactions, effects, widgets, and themes built on top of the jQuery JavaScript Library that can be used to build interactive web applications.

Q80. What is the difference between jQuery and jQuery UI?

Ans: jQuery is the core library. jQueryUI is built on top of it. If you use jQueryUI, you must also include jQuery.

**Question 1:  What is jQuery?**  
**Answer**:  jQuery is a feature-rich JavaScript library that makes HTML document traversal and manipulation, event handling, animation, and Ajax much simpler and faster.  jQuery has an easy-to-use API that works across many browsers. UI related functions can be written with minimal lines of code using jQuery.

**Question 2: What is the difference between JavaScript and jQuery?**  
**Answer**: JavaScript is an interpreted programming language whereas jQuery is a library with APIs built for JavaScript. jQuery simplifies the use of the JavaScript language.

**Question 3: What are the *effects* methods used in jQuery?**  
**Answer**:  jQuery enables us to add effects on a web page. jQuery effects can be categorized into fading, sliding, hiding/showing and animation effects. jQuery provides many methods for effects on a web page

These are the *effects* methods used in jQuery:

* show() – It displays or shows the selected elements.
* hide() – It hides the matched or selected elements.
* toggle() – It shows or hides the matched elements. In other words, it toggles between the hide() and show() methods.
* fadeIn() – It shows the matched elements by fading it to opaque. In other words, it fades in the selected elements.
* fadeOut() – It shows the matched elements by fading it to transparent. In other words, it fades out the selected elements.

**Question 4:  What is jQuery Ajax?**  
**Answer**:  AJAX is an acronym standing for Asynchronous JavaScript and XML and this technology helps us to load data and exchange data with the server without a browser page refresh. JQuery is a great tool which provides a rich set of AJAX methods to develop next-generation web applications.

**Question 5:**What does ajax() method do?  
**Answer:**This method sends an asynchronous http request to the server.

**Question 6:**What does the ajax method load() do?  
**Answer:**The load() method sends an http request to load the html or text content from the server and adds them to the DOM element(s).

**Question 7:**What are the jQuery Ajax Events?   
**Answer:** The jQuery library also includes events which will be fired based on the *state* of the Ajax request, these are called Ajax events.

**Question 8:**What does the jQuery Ajax event method ajaxComplete() do ?   
**Answer**:  Whenever an Ajax request completes, jQuery triggers the *ajaxComplete* event. Any and all handlers that have been registered with the .*ajaxComplete*() method are executed at this time.

**Question 9:**What does the jQuery ajax event method ajaxStart() do?   
**Answer:**Whenever an Ajax request is about to be sent, jQuery checks whether there are any other outstanding Ajax requests. If none are in progress, jQuery triggers the *ajaxStart* event. Any and all handlers that have been registered with the .*ajaxStart*() method are executed at this time.

**Question10:**What are Events in jQuery?  
**Answer:**Responding to user actions on a webpage is called events. jQuery provides simple methods for attaching event handlers to selections. When an event occurs, the provided function is executed.

**Question 11: What are the categories in jquery Events?**  
**Answer**: The**common DOM events are as follows**

* Form
* Keyboard
* Mouse
* Browser
* Document Loading

**Question 12: What is the use of css() method in JQuery?**  
**Answer:** The jQuery CSS() method is used to *get (return*)or *set* style properties or values for selected elements. It facilitates you to get one or more style properties.

**Question 13: What is the difference between *find* and *children* methods?**  
**Answer:**Both the methods are used to filter the child of the matched elements. Find method is used to find all levels down the DOM tree but children method searches only a single level down the DOM tree.

**Question 14: What are the selectors in jQuery and what are the types of Selectors?**  
**Answer:** If you would like to work with an element on the web page, first you need to find or select it. Selectors find the HTML elements using jQuery.

There are many types of selectors in the jQuery library. Some basic selectors are:

* **Name**: It is used to select all elements which match with the given element Name.
* **#ID**: It is used to select a single element which matches with the given ID
* .**Class**: It is used to select all elements which match with the given Class.
* **Universal** (\*): It is used to select all elements available in a DOM.
* **Multiple Elements E, F, G**: It is used to select the combined results of all the specified selectors E, F or G.
* **Attribute Selector**: It is used to select elements based on its attribute value.

**Question 15:  What is the difference between the ID selector and class selector in jQuery?**  
**Answer**: ID selector and class selector are the same as they are in CSS. ID selector uses ID while the class selector uses a class to select elements. You use an ID selector to select just one element. If you want to select a group of elements, having the same CSS class, you can use the class selector.

**Question 16: What are the Advantages of jQuery Ajax methods?**  
**Answer:** The advantages of using jQuery Ajax methods are

* Cross-browser support
* Simple methods to use
* Ability to send GET and POST requests
* Ability to Load JSON, XML, HTML or Scripts

**Question 17: What is the difference between onload() and document.ready() methods?**  
**Answer:**Body.Onload() event will be called only after the DOM and associated resources like images get loaded, but jQuery’s document.ready() event will be called once the DOM is loaded and it does not wait for the resources such as images to be loaded.

**Question 18: What is jQuery connect?**  
**Answer**:  A ‘ jQuery connect’  is a plug-in used to connect or bind a function with another function. Connect is used to execute a function whenever a function from another object or plug-in is executed.

**Question 19: Is jQuery required for bootstrap?**  
**Answer:**[Bootstrap](https://hackr.io/tutorials/learn-bootstrap?ref=blog-post)uses jQuery for JavaScript plugins (like models, tooltips, etc). However, if you just use the CSS part of Bootstrap, you don’t need jQuery.

**Question 20: What is jQuery Mobile?**  
**Answer: j**Query Mobile is an HTML5-based user interface system designed to make responsive web sites and apps that are accessible on all smartphones, tablet and desktop devices.

**Question 21: What is the difference between jquery.min.js and jquery.js?**  
**Answer:** jquery.min.js is a compressed version of jquery.js(whitespaces and comments are removed, shorter variable names are used and so on) in order to preserve bandwidth. In terms of functionality, they are absolutely the same. It is recommended to use this compressed version in the production environment. The efficiency of web page increases when the minimized version of jQuery is used.

**Question 22: Is it possible that jQuery**[**HTML**](https://hackr.io/tutorials/learn-html-5?ref=blog-post)**works for both HTML and XML documents?**  
**Answer:**No, jQuery HTML only works for HTML documents. It doesn’t work for XML documents.

**Question 23: What is jQuery UI?**  
**Answer:**[jQuery UI](http://jqueryui.com/) is a set of user interface interactions, effects, widgets, and themes built on top of the jQuery JavaScript Library. jQuery UI works well for highly interactive web applications with many controls or for a simple page with a date picker control.

**Question 24: What is a Data Table plug-in for jQuery?**  
**Answer**: DataTables is a plug-in for the jQuery Javascript library. It is a highly flexible tool, built upon the foundations of progressive enhancement, which adds advanced features to any HTML table.

**Question 25: What is Qunit?**  
**Answer**: QUnit is a powerful, easy-to-use JavaScript unit testing framework. It’s used by the jQuery, jQuery UI, and jQuery Mobile projects and is capable of testing any generic JavaScript code.

**Question 26: What is the advantage of hosting a jQuery using a CDN?**  
**Answer**: CDN stands for Content Delivery Network or Content Distribution Network. It is a large distributed system of servers deployed in multiple data centers across the internet. It provides the files from servers at a higher bandwidth that leads to faster loading time.

Advantages of using CDN are-

* The jQuery library download time will be reduced. For example – Users in Europe will hit the CDN in Europe and users in the US will hit the US CDN. As a result, this will reduce the overall page load time.
* The jQuery library will already be cached in the user’s browser if the user visited another website that references the same jQuery library. In this case, the user need not download the jQuery library.

**Question 27: Explain the difference between the .detach() and remove() methods in jQuery.**  
**Answer:**The [detach()](https://api.jquery.com/detach/) and [remove()](https://api.jquery.com/remove/) methods are the same, except that .detach() retains all jQuery data associated with the removed elements and .remove() does not. detach() is therefore useful when removed elements may need to be reinserted into the DOM at a later time.

**Question 28: Can a jQuery library be used for server scripting?  
Answer**: jQuery is designed with the functionality for client-side scripting. jQuery is not compatible with server-side scripting.

Top Answers to JQuery Interview Questions

**1. Compare Jquery & AngularJS**

|  |  |  |
| --- | --- | --- |
| **Criteria** | **JQuery** | **AngularJS** |
| Availability of RESTful API | No | Yes |
| Support for MVC | No | Yes |
| Two-way data binding | No | Yes |

**2. What is $() in jQuery library?**

The $() function is an alias of jQuery() function, at first it looks weird and makes jQuery code cryptic, but once you get used to it, you will love it’s brevity. $() function is used to wrap any object into jQuery object, which then allows you to call various method defined jQuery object. You can even pass a selector string to $()function, and it will return jQuery object containing an array of all matched DOM elements. I have seen this jQuery asked several times, despite it’s quite basic, it is used to differentiate between developer who knows jQuery or not.

**3. What are the advantage of using jQuery?**

* Easy to use and learn.
* Easily expandable.
* Cross-browser support (IE 6.0+, FF 1.5+, Safari 2.0+, Opera 9.0+)
* Easy to use for DOM manipulation and traversal.
* Large pool of built in methods.
* AJAX Capabilities.
* Methods for changing or applying CSS, creating animations.
* Event detection and handling.
* Tons of plug-ins for all kind of needs.

[*Click here*](_blank)*,* to learn about jQuery online training course.

Learn JQuery in 20 hrs from experts

Top of Form



Bottom of Form

**4. Difference between ID selector and class selector in jQuery?**

If you have used CSS, then you might know the difference between ID and class selector, It’s same with jQuery. ID selector uses ID e.g. #element1 to select element, while class selector uses CSS class to select elements. When you just need to select only one element, use ID selector, while if you want to select a group of element, having same CSS class than use class selector. There is good chance that, Interview will ask you to write code using  ID and class selector. From syntax perspective, as you can see, another difference between ID and class  selector is that former uses “#” and later uses “.” character. More detailed analysis and discussion, see answer.

**5. Difference between $(this) and this keyword in jQuery?**

Could be a tricky questions for many jQuery beginners, but indeed it’s simplest one. $(this) returns a jQuery object, on which you can call several jQuery methods e.g. text() to retrieve text, val() to retrieve value etc, while this represent current element, and it’s one of the JavaScript keyword to denote current DOM element in a context. You can not call jQuery method on this, until it’s wrapped using $() function i.e. $(this).

**6. What is main advantage of loading jQuery library using CDN?**

This is slightly advanced jQuery question, and don’t expect that jQuery beginners can answer that. Well, apart from many advantages including reducing server bandwidth and faster download, one of the most important is that, if browser has already downloaded same jQuery version from same CDN, than it won’t download it again. Since now days, almost many public websites use jQuery for user interaction and animation, there is very good chance that browser already have jQuery library downloaded. Curious reader, please see the answer for in depth analysis.

**[Interested in Learning Jquery? Click Here](https://intellipaat.com/html-jquery-training/" \l "course-content)**

**7. How do CSS precedence/cascading rules work? How does the !important directive affect the rules?**

CSS style rules “cascade” in the sense that they follow an order of precedence. Global style rules apply first to HTML elements, and local style rules override them. For example, a style defined in a style element in a webpage overrides a style defined in an external style sheet. Similarly, an inline style that is defined in an HTML element in the page overrides any styles that are defined for that same element elsewhere.The !important rule is a way to make your CSS cascade but also have the rules you feel are most crucial always be applied. A rule that has the !important property will always be applied no matter where that rule appears in the CSS document. So if you wanted to make sure that a property always applied, you would add the !important property to the tag. So, to make the paragraph text always red, in the above example, you would write:p { color: #ff0000 !important; }p { color: #000000; }

**8. What is a class? What is an ID?**

**A class** is a style (i.e., a group of CSS attributes) that can be applied to one or more HTML elements. This means it can apply to instances of the same element or instances of different elements to which the same style can be attached. Classes are defined in CSS using a period followed by the class name. It is applied to an HTML element via the class attribute and the class name.The following snippet shows a class defined, and then it being applied to an HTML DIV element..test {font-family: Helvetica; font-size: 20; background: black;}<div class =”test”><p>test</p></div>Also, you could define a style for all elements with a defined class. This is demonstrated with the following code that selects all P elements with the column class specified.

p.column {font-color: black;}

**An ID selector** is a name assigned to a specific style. In turn, it can be associated with one HTML element with the assigned ID. Within CSS, ID selectors are defined with the # character followed by the selector name.

The following snippet shows the CSS example1 defined followed by the use of an HTML element’s ID attribute, which pairs it with the CSS selector.

#example1: {background: blue;}

<div id=”example1″></div>

[*Get certified*](_blank) in jQuery & HTML for boost up your career.

Become JQuery Certified in 20 hrs.

**GET CERTIFIED**

**9. What is Grouping?**

When more than one selector shares the same declaration, they may be grouped together via a comma-separated list; this allows you to reduce the size of the CSS (every bit and byte is important) and makes it more readable. The following snippet applies the same background to the first three heading elements.h1, h2, h3 {background: red;}

**10. What is HTML?**

HTML is short for HyperText Markup Language, and is the language of the World Wide Web. It is the standard text formatting language used for creating and displaying pages on the Web. HTML documents are made up of two things: the content and the tags that formats it for proper display on pages.

[*Click here*](https://intellipaat.com/html-training/)*,* to learn about HTML online training course.

**11. What is “Semantic HTML?”**

Semantic HTML is a coding style where the tags embody what the text is meant to convey. In Semantic HTML, tags like <b></b> for bold, and <i></i> for italic should not be used, reason being they just represent formatting, and provide no indication of meaning or structure. The semantically correct thing to do is use <strong></strong> and <em></em>. These tags will have the same bold and italic effects, while demonstrating meaning and structure (emphasis in this case).