

1) What is JavaScript?

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

2) What is the difference between JavaScript and Jscript?

Ans: Both JavaScript and Jscript are almost similar. JavaScript was developed by Netscape. Microsoft reverse engineered Javascript and called it JScript.

3) How do we add JavaScript onto a web page?

Ans: There are several way for adding JavaScript on a web page, but there are two ways which are commonly used by developers
If your script code is very short and only for single page, then following ways are the best:
a) You can place `<script type="text/javascript">` tag inside the `<head>` element.

```
<head>
<title>Page Title</title>
<script language="JavaScript"
type="text/javascript">
    var name = "Vikas Ahlawta"
    alert(name);
</script>
</head>
```

b) If your script code is very large, then you can make a JavaScript file and add its path in the following way:

```
<head>
<title>Page Title</title>
<script type="text/javascript"
src="myjavascript.js"></script>
</head>
```

4) Is JavaScript case sensitive?

Ans: Yes!

A function `getElementById` is not the same as `getElementbyID`.

5) What are the types used in JavaScript?

Ans: String, Number, Boolean, Function, Object, Null, Undefined.

6) What are the boolean operators supported by JavaScript? And Operator: `&&`

Or Operator: `||`

Not Operator: `!`

7) What is the difference between `"=="` and `"==="`?

Ans:

`"=="` checks equality only,

`"==="` checks for equality as well as the type.

8) How to access the value of a textbox using JavaScript?

Ans: ex:-

```
<!DOCTYPE html>
<html>
<body>
Full name: <input type="text" id="txtFullName"
name="FirstName" value="Vikas Ahlawat">
</body>
</html>
```

There are following ways to access the value of the above textbox:

```
var name =
document.getElementById('txtFullName').value;

alert(name);
or:
```

we can use the old way:

```
document.forms[0].mybutton.
```

```
var name =
document.forms[0].FirstName.value;
```

```
alert(name);
```

Note: This uses the "name" attribute of the element to locate it.

9) What are the ways of making comments in JavaScript?

Ans:

// is used for line comments

ex:- var x=10; //comment text

/*

*/ is used for block comments

ex:-

var x= 10; /* this is

block comment example.*/

10) How will you get the Checkbox status whether it is checked or not?

Ans:

var status =

document.getElementById('checkbox1').checked;

alert(status);

will return true or false.

11) How to create arrays in JavaScript?

Ans: There are two ways to create array in JavaScript like other languages:

a) The first way to create array

Declare Array:

var names = new Array();

Add Elements in Array:-

names[0] = "Vikas";

names[1] = "Ashish";

names[2] = "Nikhil";

b) This is the second way:

var names = new Array("Vikas", "Ashish", "Nikhil");

12) If an array with name as "names" contain three elements, then how will you print the third element of this array?

Ans: Print third array element

document.write(names[2]);

Note:- Array index starts with 0.

13) How do you submit a form using JavaScript?

Ans: Use document.forms[0].submit();

14) What does isNaN function do?

Ans: It returns true if the argument is not a number.

Example:

document.write(isNaN("Hello")+ "
");

document.write(isNaN("2013/06/23")+ "
");

document.write(isNaN(123)+ "
");

The output will be:

true

true

false

15) What is the use of Math Object in JavaScript?

Ans: The math object provides you properties and methods for mathematical constants and functions.

ex:-

var x = Math.PI; // Returns PI

var y = Math.sqrt(16); // Returns the square root of 16

var z = Math.sin(90); Returns the sine of 90

16) What do you understand by this keyword in JavaScript?

Ans: In JavaScript the this is a context-pointer and not an object pointer. It gives you the top-most context that is placed on the stack. The following gives two different results (in the browser, where by-default the window object is the 0-level context):

var obj = { outerWidth : 20 };

```
function say() {  
    alert(this.outerWidth);  
}
```

say();//will alert window.outerWidth

say.apply(obj);//will alert obj.outerWidth

17) What does "1"+2+4 evaluate to?

Ans: Since 1 is a string, everything is a string, so the result is 124.

18) What does 3+4+"7" evaluate to?

Ans: Since 3 and 4 are integers, this is number arithmetic, since 7 is a string, it is concatenation, so 77 is the result.

19) How do you change the style/class on any element using JavaScript?

Ans:

```
document.getElementById("myText").style.fontSize = "10";
```

-or-

Collapse | Copy Code

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

20) Does JavaScript support foreach loop?

Ans: JavaScript 1.6(ECMAScript 5th Edition) support foreach loop,

21) What looping structures are there in JavaScript?

Ans: for, while, do-while loops

22) What is an object in JavaScript, give an example?

Ans: An object is just a container for a collection of named values:

```
// Create the man object
var man = new Object();
man.name = 'Vikas Ahlawat';
man.living = true;
man.age = 27;
```

23) How you will add function as a property in a JavaScript object? Give an example.

Ans:

```
var man = new Object();
man.name = 'Vikas Ahlawat';
man.living = true;
man.age = 27;
man.getName = function() { return
man.name; }
console.log(man.getName()); // Logs 'Vikas
Ahlawat'.
```

24) What is the similarity between the 1st and 2nd statement?

1st:- var myString = new String('male'); // An object.

2nd:- var myStringLiteral = 'male'; // Primitive string value, not an object.

Ans: Both will call String() constructor function You can confirm it by running the following statement:

```
console.log(myString.constructor,
myStringLiteral.constructor);
```

25) What will be the output of the following statements?

```
var myString = 'Vikas' // Create a primitive
string object.
var myStringCopy = myString; // Copy its value
into a new variable.
```

```
var myString = null; // Manipulate the value
console.log(myString, myStringCopy);
```

Ans: // Logs 'null Vikas'

26) Consider the following statements and tell what would be the output of the logs statements?

```
var price1 = 10;
var price2 = 10;
var price3 = new Number('10'); // A complex
numeric object because new was used.
console.log(price1 === price2);
console.log(price1 === price3);
```

Ans:

```
console.log(price1 === price2); // Logs true.
console.log(price1 === price3); /* Logs false
because price3
contains a complex number object and price 1
is a primitive value. */
```

27) What would be the output of the following statements?

```
var object1 = { same: 'same' };
var object2 = { same: 'same' };
console.log(object1 === object2);
```

Ans: // Logs false, JavaScript does not care that they are identical and of the same object type.

When comparing complex objects, they are equal only when they reference the same object (i.e., have the same address). Two variables containing identical objects are not equal to each other since they do not actually point at the same object.

28) What would be the output of the following statements?

```
var object1 = { same: 'same' };  
var object2 = object1;  
console.log(object1 === object2);  
Ans: // Logs true
```

29) What is this?

```
var myArray = [[[]]];
```

Ans: Three dimensional array

30) Name any two JavaScript functions which are used to convert nonnumeric values into numbers?

Ans:

Number()

parseInt()

parseFloat()

```
var n1 = Number("Hello world!"); //NaN
```

```
var n2 = Number(""); //0
```

```
var n3 = Number("000010"); //10
```

```
var n4 = Number(true); //1
```

```
var n5 = Number(NaN); //NaN
```

31) Does JavaScript Support automatic type conversion, If yes give example.

Ans: Yes! Javascript support automatic type conversion. You should take advantage of it, It is most common way of type conversion used by Javascript developers.

Ex.

```
var s = '5';
```

```
var a = s*1;
```

```
var b = +s;
```

```
typeof(s); //"string"
```

```
typeof(a); //"number"
```

```
typeof(b); //"number"
```