**JavaScript Evidence**

1. Calculate the odd-even number by given input during runtime? (More imp.)
2. Calculate the largest number among the three numbers? (More imp.)
3. Calculate the Maximum and Minimum number among n numbers? (Most imp.)
4. Giving input in runtime and check whether it is prime on not? (Most imp.) If it is a non-prime number, displays all of its divisors.
5. Calculate factorial value of a number? (Most imp.)
6. Display the table number of a given number. (Less imp.)
7. Create an array which has five elements and display its value and finally display its total? (Less imp.)
8. Create a numeric array and display the sorted value of that array. (Most imp.)
9. Open/Create a new window and close it? (Less imp.)
10. Retrieve email address from a text field and validate whether “@” and “.” symbol available or not and also Retrieve password from another text field and validate whether it contains 7 characters or not? (Most imp.)
11. Change background accepting input from drop down list? (Less imp.)
12. Accept input from different form elements and display it on a new window? (Most imp.)
13. Make a digital clock? (Less imp.)
14. Find out 1 1 2 3 5 8 13 21 34 55? Or display a Fibonacci series. (More imp.)
15. Calculate the percentage of discount and the amount to be paid from the following table? (Less imp.)

|  |  |
| --- | --- |
| Range (Tk.) | Discount (%) |
| 100-300 | 10 |
| 301-700 | 15 |
| 701-1000 | 20 |
| >1000 | 25 |

**Descriptive on JavaScript.**

1. ***What do you mean by webpage?***

**Answer:** ***Webpage:***

Webpage generally written in HTML and they make up the World Wide Web. WebPages can either be static or dynamic. Static pages show the same content each time they are view. Dynamic pages have content that can be change each time they are access. Dynamic pages can be written in PHP, JSP, ASP

***Web site:***

Web site is a collection of WebPages.

1. **Write down some features of JavaScript?**
2. **What is the difference between prompt() and alert()?**

**Answer:**

|  |  |
| --- | --- |
| ***prompt ()*** | ***alert()*** |
| The prompt() method displays a dialog box that prompts the visitor for input. | The alert() method displays an alert box with a specified message and an OK button. |
| This method returns the string the visitor has entered. |  |
| **Syntax**  prompt(*msg,defaultText*) | Syntax alert(*message*) |
| The prompt() method is supported in all major browsers. | The alert() method is supported in all major browsers. |

1. What do you mean by escape sequence? Write down some escape sequence.

**Answer:**

**Escape sequence:**

In JavaScript we can add special characters to a text string by using the backslash sign is called escape sequence. Some escape codes include the following:

|  |  |
| --- | --- |
| **Code** | **Outputs** |
| \' | single quote |
| \" | double quote |
| \\ | backslash |
| \n | new line |
| \r | carriage return |
| \t | tab |
| \b | backspace |
| \f | form feed |

1. ***What do you mean by object? Write down some built-in object?***

**Answer:**

**Object:** JavaScript is an Object Oriented Programming (OOP) language. An OOP language allows us to define our own objects and make our own variable types.

Properties are the values associated with an object.

Methods are the actions that can be performed on objects.

* The String object is used to manipulate a stored piece of text.
* The Date object is used to work with dates and times.
* The Array object is used to store multiple values in a single variable.
* The Boolean object is used to convert a non-Boolean value to a Boolean value (true or false).
* The Math object allows you to perform mathematical tasks.
* RegExp, is short for regular expression.
* The Navigator object contains information about the visitor's browser.

1. ***What do you mean by array? Declare an array.***

**Answer:**

**Array:** The Array object is used to store multiple values in a single variable.

Var a= new array()

document.write(a.length)

1. ***Write down of toString(), shor() and join()?***

**Answer:**

|  |  |
| --- | --- |
| **Method** | **Description** |
| [concat()](http://www.w3schools.com/jsref/jsref_concat_array.asp) | Joins two or more arrays, and returns a copy of the joined arrays |
| [join()](http://www.w3schools.com/jsref/jsref_join.asp) | Joins all elements of an array into a string |
| [pop()](http://www.w3schools.com/jsref/jsref_pop.asp) | Removes the last element of an array, and returns that element |
| [push()](http://www.w3schools.com/jsref/jsref_push.asp) | Adds new elements to the end of an array, and returns the new length |
| [reverse()](http://www.w3schools.com/jsref/jsref_reverse.asp) | Reverses the order of the elements in an array |
| [shift()](http://www.w3schools.com/jsref/jsref_shift.asp) | Removes the first element of an array, and returns that element |
| [slice()](http://www.w3schools.com/jsref/jsref_slice_array.asp) | Selects a part of an array, and returns the new array |
| [sort()](http://www.w3schools.com/jsref/jsref_sort.asp) | Sorts the elements of an array |
| [splice()](http://www.w3schools.com/jsref/jsref_splice.asp) | Adds/Removes elements from an array |
| [toString()](http://www.w3schools.com/jsref/jsref_tostring_array.asp) | Converts an array to a string, and returns the result |
| [unshift()](http://www.w3schools.com/jsref/jsref_unshift.asp) | Adds new elements to the beginning of an array, and returns the new length |
| [valueOf()](http://www.w3schools.com/jsref/jsref_valueof_array.asp) | Returns the primitive value of an array |

1. **What is difference between ‘==’ and ‘===’?**
2. ***What do mean by local and global variable?***

**Answer:**

**Local JavaScript Variable:**

* A variable declared within a JavaScript function becomes **LOCAL** and can only be accessed within that function. (the variable has local scope).
* We can have local variables with the same name in different functions, because local variables are only recognized by the function in which they are declared.
* Local variables are destroyed when we exit the function.
* We will learn more about functions in a later chapter of this tutorial.

**Global JavaScript Variable:**

* Variables declared outside a function become **GLOBAL**, and all scripts and functions on the web page can access it.
* Global variables are destroyed when we close the page.
* If we declare a variable, without using "var", the variable always becomes **GLOBAL**.

1. ***What do you mean function and return statement?***

**Answer:**

***Function:***

* The heart and soul of JavaScript belongs in functions.
* A function contains code that will be executed by an event or by a call to the function.
* We may call a function from anywhere within a page (or even from other pages if the function is embedded in an external .js file).
* Functions can be defined both in the <head> and in the <body> section of a document.
* A function with no parameters must include the parentheses () after the function name.

***Return Statement:***

* The return statement is used to specify the value that is returned from the function.
* So, functions that are going to return a value must use the return statement.

1. ***What is the difference between setInterval() and settimeout()?***

**Answer:**

***setInterval():***

* The setInterval() method calls a function or evaluates an expression at specified intervals (in milliseconds).
* The setInterval() method will continue calling the function until [clearInterval()](http://www.w3schools.com/jsref/met_win_clearinterval.asp) is called, or the window is closed.
* The ID value returned by setInterval() is used as the parameter for the clearInterval() method.
* 1000 ms = 1 second.

***Settimeout()***

* The setTimeout() method calls a function or evaluates an expression after a specified number of milliseconds.
* 1000 ms = 1 second.

1. **What do you mean by ceil() and floor()?**

**Answer:**

***ceil():***

The ceil() method rounds a number UPWARDS to the nearest integer, and returns the result.

Syntax: Math.ceil(x)

***floor():***

The floor() method rounds a number DOWNWARDS to the nearest integer, and returns the result.

Syntax: Math.floor(x)

1. ***What do you mean by event and event handler?***

**Answer:**

By using JavaScript, we have the ability to create dynamic web pages. Events are actions that can be detected by JavaScript.

Examples of events:

* A mouse click
* A web page or an image loading
* Mousing over a hot spot on the web page
* Selecting an input field in an HTML form
* Submitting an HTML form
* A keystroke

**Event handler:**  
Event handlers are the way Javascript deals with events. Javascript contains a variety of event handlers for various purposes.

**Javascript event handlers:**

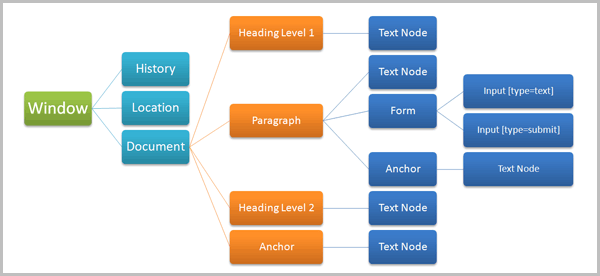
* onmouseover - Is used when the mouse cursor is moved over an object.
* onmouseout - Is used when the mouse cursor is moved off an object.
* onload - Is used when an object has finished loading.
* onclick - Is used when an object is clicked.
* onfocus - Is used when an object is made active.
* onselect - Is used when the contents of an object are selected.

1. **What do you mean by DOM?**

**Answer:**

***DOM:***

DOM stands for Document Object Model. The DOM is a [W3C](http://www.w3.org) standard by which HTML documents and the elements within them are accessed through Javascript. For example, you can access and change the value in a textbox or change the title of a webpage with the DOM.



A basic outline of the typical DOM hierarchy (Simplified)

1. ***What is the function of typeOf(), parseInt and eval()?***

**Answer:**

***typeOf():***

JavaScript **typeof** operator tells us what **type of** data we are dealing with:

var BooleanValue = true;

var NumericalValue = 354;

var StringValue = "This is a String";

alert(typeof BooleanValue) // displays "boolean"

alert(typeof NumericalValue) // displays "number"

alert(typeof StringValue) // displays "string"

***parseInt():***The parseInt() function parses a string and returns an integer.

Syntax: parseInt(*string, radix*)

var iNum1 = parseInt(“1234blue”); //returns 1234

var iNum2 = parseInt(“0xA”); //returns 10

var iNum3 = parseInt(“22.5”); //returns 22

var iNum4 = parseInt(“blue”); //returns NaN

var iNum1 = parseInt(“AF”, 16); //returns 175

var iNum2 = parseInt(“010”, 8); //returns 8

document.write("<BR>"+parseInt("50"))  
document.write("<BR>"+parseInt("50.12345"))

document.write("<BR>"+parseInt("32.00000000"))  
document.write("<BR>"+parseInt("71.348 92.218 95.405"))  
document.write("<BR>"+parseInt(" 37 aardvarks"))  
document.write("<BR>" + parseInt("Awarded the best wine of 1992"))

***Eval():***The eval() function evaluates or executes an argument.

eval is a top-level function and is not associated with any object

Syntax: eval(*string*)

eval(new String("2 + 2")); // returns a String object containing "2 + 2"

eval("2 + 2"); // returns 4

1. ***What do you mean by cookies?***

**Answr:**

***Cookies:***

A cookie is a variable that is stored on the visitor's computer. Each time the same computer requests a page with a browser, it will send the cookie too. With JavaScript, we can both create and retrieve cookie values.

*Examples of cookies:*

* ***Name cookie -*** The first time a visitor arrives to our web page, he or she must fill in her/his name. The name is then stored in a cookie. Next time the visitor arrives at your page, he or she could get a welcome message like "Welcome John Doe!" The name is retrieved from the stored cookie
* ***Password cookie -*** The first time a visitor arrives to your web page, he or she must fill in a password. The password is then stored in a cookie. Next time the visitor arrives at your page, the password is retrieved from the cookie
* ***Date cookie -*** The first time a visitor arrives to your web page, the current date is stored in a cookie. Next time the visitor arrives at your page, he or she could get a message like "Your last visit was on Tuesday August 11, 2005!" The date is retrieved from the stored cookie

1. **Write down structure of a ternary operator with an example?**

**Answer:**

The ternary operator is the question mark symbol (?), it works the same way as the if-else structure.

1. **What do you mean by ‘with’ statement?**

Instead of having to list all of the properties of an object by represent the basic object,

We can state the bulk of the object in a with statement and the properties within the context of the with statement.

For example:

1. **What is difference between substring() and charAt()?**

#### charAt

charAt() gives you the character at a certain position. For instance, when you do

var b = 'I am a JavaScript hacker.'

document.write(b.charAt(5))

gives 'a', because that's the character at the sixth position (remember, first character is 0!).

#### substring

substring is used to take a part of a string. Syntax is substring(first\_index,last\_index). So for instance

var a = 'Hello world!';

document.write(a.substring(4,8));

gives 'o wo', from the first 'o' (index 4) to the second one (index 7) Note that the 'r' (index 8) is not part of this substring.

You can also do

var a = 'Hello world!';

document.write(a.substring(4));

This gives the whole string from the character with index 4 on: 'o world!'.

#### substr

substr() // is not supported by Netscape 2 and 3, Explorer 3, Hotjava 3, Opera 3 and WebTV.

There is also a method substr() that works slightly differently. Instead of the second number being an index number, it gives the number of characters. So

var a = 'Hello world!';

document.write(a.substr(4,8));

starts at the character with index 4 ('o') and then gives 8 characters, so the output is

o world!

Since substr() is not supported by the Version 3 browsers I never use it.

1. **What do you mean by history object? Write down it’s method?**

The History object contains an array of previously visited URLs by the visitor. To simulate the browser's back button, for example, you can use the History object:

<a href="javascript:history.go(-1)">Go back</a>

Property: length

Method: 1. Back() <a href="javascript:history.go(-3)">Go back 3 pages</a>

<a href="javascript:history.go('javascriptkit.com')">Go to nearest URL in history list with "javascriptkit.com" in the URL.</a>

2. forward()

3. go()