Descriptive questions & answers for Java Script

1. **"JavaScript is called interpreted language"- why?**

   Answer: JavaScript is called interpreted language because JavaScript has no compilation step like other languages (C++ or Java), instead, an interpreter in the browser reads over the JavaScript code, interprets each line, and runs it.

2. **Why JavaScript is considered as a "weakly typed" or "untyped" language?**

   Answer: JavaScript is considered as a **"weakly typed"** or **"untyped"** language because it will figure out what type of data someone has and make the necessary adjustments so that there is no need to redefine different types of data. That's why new programmers regard JavaScript as **smart** as it saves time in learning several different conversion steps and data type declarations.

3. **What do you understand by Undefined and null values?**

   Answer: **Undefined** value is the value that returns when someone attempts to use a variable that has not been defined or one that is declared but forgot to provide with a value. A non-existent property of an object also returns undefined if it is addressed.

   For example: | **var a;** | | **var a = undefined;** | | **var a = {};** |

   On the other hand, a variable can be declared or defined as **null**, if there is absolutely nothing in it but someone doesn't want it to be undefined. Null is not the same as zero (0) in JavaScript. It represents the null, empty, or non-existent reference.

   Example: Let assign the value of null to b: | **var b = null;** |

4. **What are global and local variables?**

   Answer: Variables in JavaScript have scope, which refers to the regions of the script where the variables can be used. A **global variable**, has global scope and is defined in the entire script.

   Whereas, **local variables** are local to the functions in which they are defined. As a general rule, we should avoid naming any two variables, whether global or local, with the same name or identifier.

5. **What is an array?**

Answer: An array is a data structure that contains a group of elements. Typically, these elements are all of the homogenous data, such as an integer or string. Arrays are commonly used in computer programs to organize data so that a related set of values can be easily sorted or searched.

Example:

var variable name = new Array(value);

var myArray = new Array("Value 1", "Value 2", "Value 3", "Value 4");

6. **What are the three parts of "for loop"?**

Answer: The three parts of "for loop" are: initialization, condition, and increment or decrement.

Example: If we want to print 1 to 10 using for loop, the format will be like:

for(i=1; i<=10; i++);

7. **What are the three main event categories?**

Answer: The three main event categories are:

- Keyboard & mouse events;
- Load events;
- Form-related events.

8. **What are the benefits of using "With" statement?**

Answer: One can gain the following benefits by using the **"With"** statement:

- It doesn't need to evaluate a complex expression multiple times or assign the result to a temporary variable to refer to its members multiple times.
- It makes the code more readable by eliminating repetitive qualifying expressions.

Syntax:

with(object) {

   // Calling the functions or methods of the object

}

9. What is Document Object Model (DOM)?

Answer: It is a w3c standard for accessing documents. The Document Object Model (DOM) is a cross-platform and language-independent application programming interface that defines the logical structure of documents and the way a document is accessed and manipulated. With the DOM, programmers can create and build documents, navigate their structure, and add, modify, or delete elements and content. Anything found in an HTML document can be accessed, changed, deleted, or added using the DOM.

10. What is the event and event handler?

Answer: In programming, **an event** is an action that occurs as a result of the user or another source, such as key strokes, mouse activity, action selections, timer expirations, and so forth. An HTML event can be something the browser does, or something a user does. Often, when events happen, you may want to do something. JavaScript lets you execute code when events are detected.

And **an event handler** is a software routine that is used to deal with the event, allowing a programmer to write code that will be executed when the event occurs. With Web sites, event handlers make Web content dynamic. JavaScript is a common method of scripting event handlers for Web content.

Example of Events & Event Handlers:

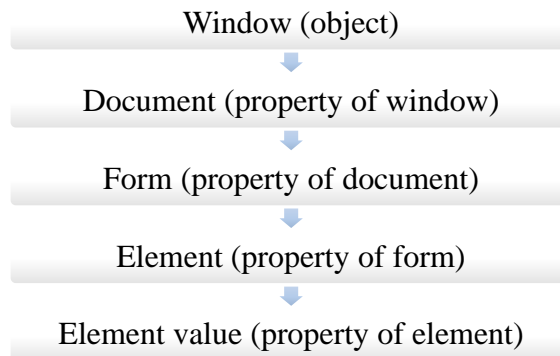| Events	Mouse events	Key events	Form events	Page/Window/Image events
Event Handlers	onClick onMouseDown onMouseUp onMouseMove onMouseOver onMouseOut	onKeyDown onKeyUp onKeyPress	onBlur onChange onFocus onReset onSubmit	onAbort onLoad onUnload onError onResize

11. Reserved word? Write down five reserved words.

Answer: In JavaScript reserved word means some specific words those cannot be used as variables, labels, or function names.

Example: break, double, throw, try, public, default, class, package, int, private, native, transient, implements, delete, final, new, this etc.

12. Write down the hierarchy of objects of an html form and the object-property chain.

Answer: <u>The Hierarchy of Objects in an HTML Form:</u>

Window (object)

Document (property of window)

Form (property of document)

Element (property of form)

Element value (property of element)

Object-Property Chain:	
Object	**Properties**
Window object	Document, Form, Element, Element value
Document object	Form, Element, Element value
Form object	Element, Element value
Element object	Element value
Element value object	Property of element

13. Write some features of JavaScript.

Answer: Features of JavaScript are given below:

- JavaScript is object-based language as it provides predefined objects.
- JavaScript is the engine that makes things move on a page, by working with dynamic design elements.
- It gives the user more control over the browser.
- It is light weighted and case sensitive.
- It has no compilation step like other languages (C++ or Java).
- JavaScript allows designers to release those aspects of design creatively that cannot be expressed in static HTML.

- It is a ranged language and ranges from extremely simple built in functions and statements that can make the page jump to fairly sophisticated coding structures.
- It lives right among the HTML tags and most JavaScript programs are relatively small.
- It captures user events that cause actions to happen on a web page.
- The JavaScript Date object simplifies the process of computing and working with dates and times.
- JavaScript is an object-based and an interpreter-based scripting language.
- An important part of JavaScript is the ability to create new functions within scripts by declaring a function using function keyword.

14. What are primitive data types?

Answer: In JavaScript data types are divided into two basic categories: Primitive; & Compound. Primitive data constitutes with boolean values, numbers, strings, the null & undefined values. Primitive data types are stored in a fixed chunk of memory, depending on the type of data. Primitive data types have a finite and known amount of space in memory and can be stored with the variables.

Example: var x = 18; var y = "Apple";

15. What is variable?

Answer: Variables can be treated as containers that can hold different types of values those can be changed anytime. JavaScript uses reserved keyword **var** to declare a variable. A JavaScript variable must be identified with unique names those are called identifiers. Identifiers are case-sensitive and can be short names (like x and y) or more descriptive names (age, sum, totalVolume). After variable declaration one can assign a value using equal to (=) operator before using the variable.

Example: var variable name = (value);

Rules of naming for variables:

The general rules for constructing names for variables (unique identifiers) are:

- Names can contain letters, digits, underscores, and dollar signs;
- Names must begin with a letter;
- Names can also begin with $ and _ (but it is not considered as good practice);

- Names are case sensitive (y and Y are different variables);
- Reserved words (like JavaScript keywords) cannot be used as names.

16. What are the two purposes of plus operator?

Answer: The basic arithmetic operators in JavaScript are fairly self-explanatory, with a few exceptions like: add (+). It has two different uses; First, the add operator adds value in math operations. Second, it joins strings or strings with other literals which is called concatenation. Mathematical addition is fairly straightforward, but concatenation is not. When the add operation joins a string and a number, it concatenates them, and converts the number into string.

Example of Add operation: document.write(200+50); Ans: 250

Example of Concatenation: document.write(200+"50"); Ans: 20050

17. What is the use of "type of" operator?

Answer: The **"type of"** operator is a unary operator that is placed before its single operand, which can be of any type. Its value is always a string indicating the data type of the operand. The operand can be either a literal or a data structure such as a variable, a function, or an object. There are six possible values that **"type of"** returns: object, boolean, function, number, string, and undefined.

The typeof operator is used in either of the following ways:

typeof operand;

typeof(operand);

18. Which are called LIFO based and FIFO based method?

Answer: **LIFO** based method is known as pop() method of an array object. pop() operation removes the last element of an array, and returns it. That is why it is called LIFO-last in, first off.

Example:

var num = new Array("one","two","three");

num.pop(); would return three and removes it from the array.

FIFO based method is known as shift() method of an array object. shift() operation removes the first element of an array, and returns it. That is why it is called FIFO-first in, first off.

Example:

var num = new Array("one","two","three");

num.shift(); would return one and removes it from the array.

19. Which operator can be used as an alternative of conditional statement?

Answer: A conditional operator works as an alternative of if/else statement. It evaluates a true/false condition and then either runs the code for option A or option B. The default pattern of the conditional operator (?:) is;

BooleanExpression ? expression1 : expression2

The conditional operator (?:) evaluates a true/false condition (BooleanExpression) first. When that condition is true, it executes expression1. And when it evaluates to false, the conditional operator runs expression2. This way the logic of the conditional operator is: "if this is true, do the first; otherwise, do the second."

20. What are the three-basic structure of JavaScript?

Answer: Like every other programming language, JavaScript has also three basic structures. These are as follows:

1. Sequential structures
2. Branch structures
3. Loop structures

21. What are the three categories of operators?

Answer: The three categories of operators are:

1. **Binary operator:** A binary operator requires two operands, one before the operator and one after the operator:

   operand1 operator operand2

   For example; x + y

   In JavaScript most of the operators are binary operators like; assignment operators, compound operators, comparison operators, arithmetic operators, strict equality operators, and so on.

2. **Unary operator:** A unary operator requires a single operand, either before or after the operator:

   operator operand   or   operand operator

For example; x++ or ++x

Increment (++) and Decrement (--) operators are unary operator.

3. **Ternary operator:** The ternary operator is the only JavaScript operator that takes three operands. The operator can have one of two values based on a condition. It is also known as conditional operator. The syntax is:

> condition ? option1 : option2

That means, if the condition is true, the operator will execute the option1. Otherwise, it will execute the option2.

22. Which loop checks the condition at the end of the statement?

Answer: The do/while loop checks the condition at the end of the statement but executes statements in the first iteration of the loop. The general format of do/while loop looks like the following:

Do{

Statements

Counter increment/decrement

}while(termination condition)

23. Where return statement is used?

Answer: The return statement can be used only as part of a function. The role of the return statement is to provide the value of the expressions within the function. When a function is created in JavaScript, it is essential to provide a return statement in the script to use the function as data in another expression. The script of the return statement will be as following:

function doIt() {

      var a = "Good";

      var b = "Luck";

      var c = a + b;

      return c;

   }

   document.write(doIt());

The screen will provide "GoodLuck" on the screen.

24. How can you fire a function?

Answer: One can fire a function with event handlers. Event handlers can be divided into three main general categories:

1. Keyboard & mouse events like; onKeyPress, onClick etc.
2. Load events like; onLoad, onError etc.
3. Form-related events like; onFocus, onSubmit etc.

25. What are the differences between substring() and substr()?

Answer: substring() methods extracts the characters from a string, between two specified indices where **substr()** methods extracts the characters from a string, beginning at a specified start position, and through the specified number of character.

Example:

```
var x = "Hello World";
var y = x.substr(6,5);
document.write(y+"<br/>"); The output will be = World
var z = x.substring(6,10);
document.write(z+"<br/>"); The output will be = Worl
```

26. What do you mean by indexOf()?

Answer: The **indexOf()** method returns the position of the first occurrence of a specified text in a string where the counts of positions starts with zero (0).

Example:

```
var x = "Hello World";
var y = x.indexOf("W");
document.write(y);
The output will be = 6.
```

27. Why object-oriented programming is essential in JavaScript?

Answer: It is essential because;

- Objects created for Object Oriented Programs can easily be reused in other programs.
- Objects have the ability to hide certain parts of themselves from programmers. This prevents programmers from tampering with values they shouldn't. For

example, a programmer (or another program) cannot set the width of a window to -400.

- In response to large programs, Object Oriented Programs are actually easier to program than non-Object-Oriented ones as OOP follows an extensive planning phase.
- An Object-Oriented Program is much easier to modify and maintain than a non-Object-Oriented Program. So, although a lot of work is spent before the program is written, less work is needed to maintain it over time.
- Instances can become templates for other instances, if you want to "clone" an object, just use the existing object as the prototype for the new object. No need to write lots of complex cloning logic for different classes.

28. What is the benefit of preloading image?

Answer: The benefits are as following:

- Reduces loading time;
- Lessen unexpected behaviour;
- Increase responsiveness;
- Put a great impression towards the viewers of the content.

In JavaScript, one can easily preload images by following two easy steps; **First:** Create a JavaScript Image Object, and **Second;** Load an Image File into the Object. The format will look like;

var streetLight = new Image();

streetLight.src = "light.jpg";

29. What are the functions of open and close method?

Answer: The open and the close method are the functions of window object. The **window.open()** method opens a new window while **window.close()** method closes the current window.

30. What are the events in HTML and JavaScript?

Answer: There are four types events in HTML and JavaScript. These are:

1. **Mouse events:** onClick, onMouseDown, onMouseUp, onMouseMove, onMouseOver, onMouseOut

2. **Key events:** onKeyDown, onKeyUp, onKeyPress
3. **Form events:** onBlur, onChange, onFocus, onReset, onSubmit
4. **Page/Window/Image events:** onAbort, onload, onUnload, onError, onResize

31. What are the three cross-browser methods of history object?

Answer: The three cross-browser methods of history object are: **back(), forward(), and go(n).**

The back() method loads the previous URL in the history list and The forward() method loads the next URL in the history list. The back() method and The forward() method works as the same as clicking the "Back button" and the "Forward button" respectively in your browser. But these methods won't work if the previous page and the next page does not exist in the history list. The format for these two methods will be as following: **window.history.back(); and window.history.forward();**

The go() method loads a specific URL from the history list. Suppose, you want to go to "two pages back" than the format will be as following: **window.history.go(-2);**

32. What are the two methods of location object?

Answer: Two methods of location objects are:

- **reload():** This method is used to reload the current document. By default, the reload() method reloads the page from the cache, but you can force it to reload the page from the server by setting the parameter to true, like: **location.reload(true);**

- **replace():** This method is used to replace the current document with a new one. The replace() method removes the URL of the current document from the document history, so that it is not possible to use the "back" button to navigate back to the original document. The format is: **location.replace();**

33. What do you mean by cookie?

Answer: Cookies are data, stored in small text files, in a computer. Cookies were invented to remember information about the user:

When a user visits a web page, his name can be stored in a cookie.

Next time the user visits the page, the cookie "remembers" his name.