**1. What is JavaScript?**

JavaScript is a very powerful client-side scripting language. JavaScript is used mainly for enhancing the interaction of a user with the webpage.

In other words, you can make your webpage more lively and interactive, with the help of JavaScript. JavaScript is also being used widely in game development and Mobile application development.

**2) What are the different data types present in JavaScript?**

There are two types of data types in JavaScript:

* Primitive data types
* Non- Primitive data types

**3. What are Primitive data Types in JavaScript?**

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

Breaking within a string statement can be done by using a backslash, ‘\,’ at the end of the first line.

Example:

document. Write ("This is \a program,");

**6). Which company developed JavaScript?**

Netscape is the software company that developed JavaScript.

**7). 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.

**8). What Is a Prompt Box?**

An input container with a label and a text field that lets a user enter input.

**9). What’s the Difference Between Alert & Confirmation Boxes?**

An alert box shows a single ‘OK’ button to the user, while confirmation boxes show two buttons with ‘OK’ and ‘Cancel’.

**10). Is JavaScript Case-Sensitive?**

Yes. A variable named say Hello and a variable named say hello will be treated as two separate variables.

**11). How Can You Handle Exceptions With Javascript?**

You use *try…catch…finally* syntax.

* try statement: defines the code you want to run
* catch statement: handle errors
* finally statement: code you want to run regardless of any errors

**12). How Do You Run a JavaScript File?**

Add a *script* element in the *head* or *body* elements of your HTML file, then link to an external script using the script element's *src attribute*.

**13). How Do You Create an Array in Javascript?**

i). Create an instance with the new keyword:

let myArray = newArray('value1', 'value2',..., 'valueN');

or

ii). Use the Array.of() method:

let myArray = Array.of('value1', 'value2',..., 'valueN');

or

iii). Use an array literal:

let myArray = [value1, value2,...., valueN];

**14). How Do You Check if a Value Is in an Array in Javascript?**

i). Using *.includes()*

let myArray = [1, 2, 3, 4, 5,];

console.log(myArray.includes(1)); // true

console.log(myArray.includes(7)); // false

or

ii) . Using .indexOf()

let myArray = [1, 2, 3, 4, 5];

console.log(myArray.indexOf(1) !== -1); // true

console.log(myArray.indexOf(7) !== -1) // false

**15). What Is a Class in JavaScript?**

A blueprint for creating objects, which provides a way to define object methods and properties. You define these with the *class* keyword.

**16. Is JavaScript a Class-Based Language?**

No, it’s a prototype-based language. JavaScript uses prototypes to define object properties, and objects are not instances of classes, but rather they inherit properties and methods from other objects creating a prototype chain.

In a class-based language, objects are instances of classes, and classes define object properties and methods.

**17). How Do You Create a New Object in JavaScript?**

**i).Using object literals:**

       const user = {

    name: "Jane Doe",

  age: 25

};

Or

**ii). Using the new keyword :**

const user = newObject();

user.name = "Jane Doe";

user.age = 25;

**18). What Is a Constructor in JavaScript?**

A constructor is used to create an object instance of a class. To call the constructor, you use the *new* keyword.

function User(name) {

  this.name = name;

}

let firstUser = new User('John Doe');

**19. How Can You Return a Character From a Specified Index?**

Use the *charAt()* method. In the example below, we can retrieve the first char at index 0 to fetch *J*.

let userName = "John";

console.log(userName.charAt(0)); // "J"

**20. What Are the Advantages of Using External JavaScript Files?**

It separates HTML and JS code, improves the readability of your code, makes it easy to reuse your code, and improves page load speed with cached JS files.

**21. What Is the “This” Keyword in JavaScript?**

This allows you to refer to the object which was used to make the function call.

**22. What Is the For...In Loop in JavaScript?**

This is used to iterate over the properties of a JavaScript object. Every property returns a key, which is used to access the value.

**23. What’s the Difference Between Var, Let, & Const Keywords?**

* **Var**: Declared variables are functionally scoped, or globally scoped when declared outside a function. You can reassign new values and redeclare them.
* **Let**: Declared variables are block-scoped, meaning they can only be accessed within their declaration scope. It’s possible to reassign different values to them, but you cannot redeclare them.
* **Const**: These are constants, meaning they are block-scoped, and you cannot reassign new values or redeclare them.

**24. What Is Garbage Collection in JavaScript?**

This type of *automatic memory management* monitors memory allocation to reclaim and free up allocated memory blocks that are no longer needed by the JavaScript program.

**25. List the Error Types in JavaScript.**

* EvalError: Error regarding the global function *eval()*
* InternalError: Internal error in the JS engine (stack overflow, etc)
* RangeError: Numeric variable or parameter is outside of valid range
* ReferenceError: Invalid variable reference
* SyntaxError: Syntax error while parsing code in *eval()*
* TypeError: Parameter or variable not of a valid type
* URIError: Invalid parameters passed to *decodeURI()* or *encodeURI()*

**26. 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 :

function named(){

// write code here

}

**27. 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.

**28. 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.

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

|  |  |
| --- | --- |
| **Built-in Built-in Method Method** | **Values** |
| **CharAt()** | It returns the character at the specified index. |
| **Concat()** | It joins two or more strings. |
| **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. |
| Date() | Returns the present date and time |
| concat() | Joins two strings and returns the new string |

 30**. 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 :**

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

**31. 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.

**32 . 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.

**33. 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**

**34).Explain Inline,Internal and External Script ?**

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.

**35. 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.

**36. 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.

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

JavaScript has a collection of many frameworks that aim towards fulfilling the different aspects of the web application development process. Some of the prominent frameworks are:

* React - Frontend development of a web application
* Angular - Frontend development of a web application
* Node - Backend or server-side development of a web application

**38. 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-

parseInt("4F", 16)

**39 . 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.

**40. What is the difference between Document and Window in JavaScript?**

|  |  |
| --- | --- |
| Window | Document |
| Window in JavaScript is a global object that holds the structure like variables, functions, location, history, etc. | The document comes under the windows object and can also be considered as its property. |

**41. What is the difference between Session storage and Local storage?**

|  |  |
| --- | --- |
| Session storage | Local storage |
| The data stored in session storage gets expired or deleted when a page session ends. | Websites store some data in local machine to reduce loading time; this data does not get deleted at the end of a browsing session. |

**42. What are object prototypes?**

Following are the different object prototypes in javascript that are used to inherit particular properties and methods from the Object.prototype.

1. Date objects are used to inherit properties from the Date prototype
2. Math objects are used to inherit properties from the Math prototype
3. Array objects are used to inherit properties from the Array prototype.

**43. Which method is used to retrieve a character from a certain index?**

We can retrieve a character from a certain index with the help of charAt() function method.

**44. What is BOM?**

BOM is the Browser Object Model where users can interact with browsers that is a window, an initial object of the browser. The window object consists of a document, history, screen, navigator, location, and other attributes. Nevertheless, the window’s function can be called directly as well as by referencing the window.

 45**) 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.

**46) 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. |

**47) 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.

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

**48. Difference between client-side and server-side**

**Client-side JavaScript:**

* Client-side JavaScript is made up of fundamental language and predefined objects that perform JavaScript in a browser.
* Also, it is automatically included in the HTML pages where the browser understands the script.

**Server-side JavaScript:**

* Server-side JavaScript is quite similar to Client-side javascript.
* Server-side JavaScript can be executed on a server.
* The server-side JavaScript is deployed once the server processing is done.

**49). What is Primitive data types ?**

The primitive data types are capable of displaying one value at a time. It consists of Boolean, Undefined, Null, Number, and String data types.

**50). What are the pop-up boxes available in JavaScript?**

Pop-up boxes available in JavaScript are

Alert Box,

Confirm Box, and

Prompt Box.

**51). How do you empty an array in JavaScript?**

There are a few ways in which we can empty an array in JavaScript:

* By assigning array length to 0:

var arr = [1, 2, 3, 4];

arr.length = 0;

* By assigning an empty array:

var arr = [1, 2, 3, 4];

arr = [];

* By popping the elements of the array:

var arr = [1, 2, 3, 4];

while (arr.length > 0) {

  arr.pop();

}

* By using the splice array function:

var arr = [1, 2, 3, 4];

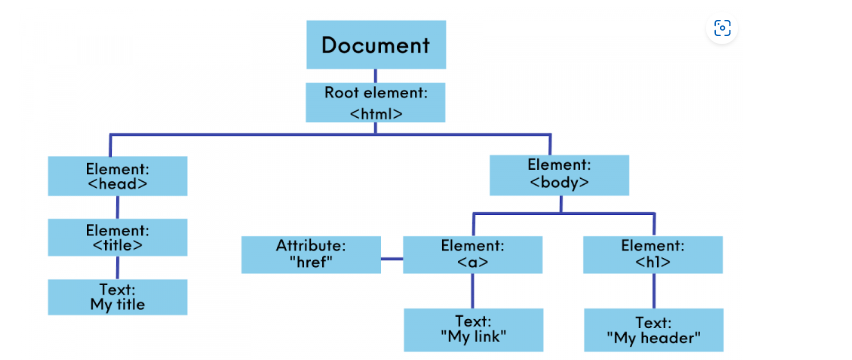
arr.splice(0, arr.length);

**52). Can you draw a simple JavaScript DOM (Document Object Model)?**

 When a web page is loaded, the browser creates a **D**ocument **O**bject **M**odel of the page.

The **HTML DOM** model is constructed as a tree of **Objects**.

With the object model, JavaScript gets all the power it needs to create dynamic HTML.



**53) 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.

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

**54) 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:

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

**55) 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:

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

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

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

function display()

{

  document.writeln(10+20+"30");

}

display();

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

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

function display()

{

  document.writeln("10"+20+30);

}

display();

**58) 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.

**59) 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.

**60) 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.

**61). Is javascript a statically typed or a dynamically typed language?**

JavaScript is a dynamically typed language. In a dynamically typed language, the type of a variable is checked during run-time in contrast to a statically typed language, where the type of a variable is checked during compile-time.

**62). what are Arrays in JavaScript ?**

Arrays are the next item on our JavaScript cheat sheet. Arrays are used in a variety of programming languages. They are a method of categorising variables and attributes. Arrays can be defined as a collection of objects of the same type. In JavaScript, here's how one can make an array of cars:

**var** cars = ["Mercedes", "Tesla","Volvo"];

**63).Explain pop(),push() methods ?**

* **pop():** This method is used for removing the last element of an array.
* **push():**This method is used for adding a new element at the very end of an array.

64). **Explain concat(),reverse(),shift(),slice() methods ?**

* **concat():**This method is used for joining various arrays into a single array.
* **reverse():** This method is used for reversing the order of the elements in an array.
* **shift():**This method is used for removing the first element of an array.
* **slice():**This method is used for pulling a copy of a part of an array into a new array.

65). **Explain splice(),toString(),unshift() methods ?**

* **splice()**: This method is used for adding elements in a particular way and position.
* **toString():**This method is used for converting the array elements into strings.
* **unshift():**This method is used for adding new elements at the beginning of the array.

66). **Explain valueOf(),indexOf(),lastIndex() methods ?**

* **valueOf():**This method is used for returning the primitive value of the given object.
* **indexOf():**This method is used for returning the first index at which a given element is found in an array.
* **lastIndexOf():**This method is used for returning the final index at which a given element appears in an array.

**67). Explain join,sort methods ?**

* **join():**This method is used for combining elements of an array into one single string and then returning it.
* **sort():**This method is used for sorting the array elements based on some condition.

**68. What are JavaScript Functions** ?

JavaScript Functions can be defined as chunks of code written in JavaScript to perform a single task.

A function in JavaScript looks like this:

function nameOfTheFunction(parameterOne, parameterTwo, parameterThree, parameterFour,....,parameterN) {   
// Job or Task of the function    
}

**69) Explain prompt(),alert(),console.log(),document.write() functions ?**

**Functions For Throwing Data As Output**: The output of data is a common application for functions. You have the following options for outputting data:

* **prompt()**: This function is used for creating a dialogue box for taking input from the user.
* **alert()**: This function is used for outputting information in an alert box in the browser window
* **console.log()**: This function is used for writing data to the browser's console and is used for the purpose of debugging code by developers.
* **document.write()**: This function is used for writing straight to our HTML document
* **confirm()**: This function is used for opening up a yes or no dialogue box and for returning a boolean value depending upon the user's click

**70). Explain parseInt(),parseFloat(),isNaN(),eval(),isFinite() functions ?**

**Global Functions:**Every browser that can run JavaScript has a set of global functions built-in. Some of them are as follows:

* **parseFloat():** This function is used for parsing the argument passed to it and returning a floating-point number.
* **parseInt():**This function is used for parsing the argument passed to it and returning an integral number.
* **isNaN():**This function is used for determining if a given value is Not a Number or not.
* **Number():**This function is used for returning a number converted from what is passed as an argument to it.
* **eval():**This function is used for evaluating JavaScript programs presented as strings.
* **isFinite():**This function is used for determining if a passed value is finite or not

**71). Explain the working of timers in JavaScript?**

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

72. **What is negative infinity?**

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

**73. 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");

**74)** **What is AJAX?**

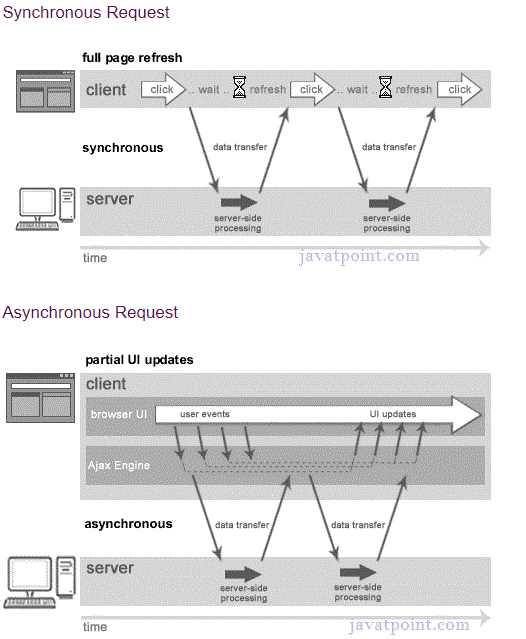
AJAX stands for Asynchronous JavaScript and XML. It is a group of related technologies used to display data asynchronously. In other words, it sends and retrieves data without reloading the web page.

### 75) What are the advantages of AJAX?

* Quick Response
* Bandwidth utilization
* The user is not blocked until data is retrieved from the server.
* It allows us to send only important data to the server.
* It makes the application interactive and faster.

### 76) What is the difference between synchronous and asynchronous requests?

Synchronous request blocks the user until a response is retrieved whereas asynchronous doesn't block the user.



**77). What are the technologies used by AJAX?**

* HTML/XHTML and CSS - These technologies are used for displaying content and style.
* DOM - It is used for dynamic display and interaction with data.
* XML - It is used for carrying data to and from server
* XMLHttpRequest - It is used for asynchronous communication between client and server.
* JavaScript - It is used mainly for client-side validation

**78) What is the purpose of XMLHttpRequest?**

* It sends data in the background to the server.
* It requests data from the server.
* It receives data from the server.
* It updates data without reloading the page.

### 79) What are the properties of XMLHttpRequest?

The important properties of the XMLHttpRequest object are given below.

* onReadyStateChange - It is called whenever readystate attribute changes.
* readyState - It represents the state of the request.
* responseText - It returns response as text.
* responseXML - It returns response as XML.
* status - It returns the status number of a request.
* statusText - It returns the details of status.

**80) What are the important methods of XMLHttpRequest?**

* abort() - It is used to cancel the current request.
* getAllResponseHeaders() - It returns the header details.
* getResponseHeader() - It returns the specific header details.
* open() - It is used to open the request.
* send() - It is used to send the request.
* setRequestHeader() - It adds request header.

**81) What are the types of open() method used for XMLHttpRequest?**

* open(method, URL) - It opens the request specifying get or post method and URL.
* open(method, URL, async) - It is same as above but specifies asynchronous or not.
* open(method, URL, async, username, password) - It is same as above but specifies the username and password.

### 82). What is JSON in AJAX?

JSON stands for JavaScript Object Notation. In AJAX, it is used to exchange data between a browser and a server. It is easy to understand, and data exchange is faster than XML. It supports array, object, string, number, and values.

|  |  |
| --- | --- |
| **JavaScript** | **AJAX** |
| JavaScript is an object-based scripting language. | AJAX is a group of inter-related technologies like JavaScript, XML, HTML, CSS etc. |
| It requests the server and waits for the response. | It sends a request to the server and doesn't wait for the response |
| It consumes more bandwidth as it reloads the page. | It doesn't reload the page so consumes less bandwidth. |

### 83). What is the difference between JavaScript and AJAX?

**84) Mention what is JSON?**

JSON is a simple data exchange format. JSON means JavaScript Object Notation; it is language and platform independent.

**85) Explain how to transform JSON text to a JavaScript object?**

One of the common use of JSON is to collect JSON data from a web server as a file or HTTP request, and convert the JSON data to a JavaScript, ant then it avails the data in a web page.

**86) Why must one use JSON over XML?**

* It is faster and lighter than XML as on the wire data format
* XML data is typeless while JSON objects are typed
* JSON types: Number, [Array](https://career.guru99.com/top-50-array-interview-questions-answers/), Boolean, String
* [XML](https://career.guru99.com/xml-interview-questions/) data are all string
* Data is readily available as JSON object is in your JavaScript
* Fetching values is as simple as reading from an object property in your JavaScript code

### 87) Mention what is the file extension of JSON?

File extension of JSON is .json

### 88). Mention which function is used to convert a JSON text into an object?

To convert JSON text into an object “eval” function is used.

**89). Mention what are the data types supported by JSON?**

Data types supported by JSON includes

* Number
* String
* Boolean
* Array
* Object
* Null

### 90) Mention what is the role of JSON.stringify?

**JSON.stringify()** converts an object into a JSON text and saves that JSON text in a string.

### 91) Show how to parse JSON in JQuery?

To parse JSON in JQuery, we will see the example

var json = '{"name": "Guru 99", "Description ": "Learn PHP Interactively with PHP Beginner Tutorials"}'

obj = $.parseJSON(json);

//alert(obj.name);