**1) What is JavaScript?**

JavaScript is a text-based programming language used both on the client-side and server-side that allows you to make web pages interactive.

### 2) 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. |

### 3) ****What are the data types supported by JavaScript?****

### The data types supported by JavaScript are: Undefined, Null, Boolean, String, Symbol, Number, Object

### 4) ****What are the features of JavaScript?****

### **The primary features of Javascript are: it is Lightweight & it is an Interpreted programming language which is Good for the applications which are network-centric. On top of this as the language uses most of the keywords in JAVA it is complimentry to java language which most of the developers are familiar with. Javascript blends well with HTML as it uses the DOM tree to intrepret HTML for us to play around with elements.**

### 5) Is JavaScript a case-sensitive language?

### Yes, JavaScript is a case sensitive language. Here keywords, variable names, function names, and any other identifiers are to be used in the program consistentently and carefully.

### 6) List some advantages of javaScript?

### Few of the advantages in javascript are: In JS Server interaction is very less. The Feedback to the visitors is immediate. Interactivity is high as we can toggle the inputs inside browser developer tools and get an instant output.

### 7) Difference between “ == “ and “ === “ operators.

Both are comparison operators. The difference between both the operators is that “==” is used to compare values whereas, “ === “ is used to compare both values and types on LHS & RHS.

**8) Explain Hoisting in javascript.**

Hoisting is the default behaviour of javascript where all the variable and function declarations are moved on top.

Eg: function m1(){ var a; for(var i; i <=10; i++){ console.log(i); } console.log(i); //Variable access is permitted because it is hoisted }

**9) 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.

**10) How to create an array in JavaScript?**

There are 3 ways to create an array in JavaScript.

* By using array literal.
* By creating an instance of Array
* By using an Array constructor

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

var emp=["Shyam","Vimal","Ratan"];

**11) 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.

For example, a variable that is assigned a number type can be converted to a string type:

var a = 23;

var a = "Hello World!";

**12) What is NaN property in JavaScript?**

NaN property represents the “Not-a-Number” value. The typeof of NaN will return a Number.

To check if a value is NaN, we use the isNaN() function,Eg:

* isNaN("Hello") // Returns true
* isNaN(345) // Returns false
* isNaN('1') // Returns false, since '1' is converted to Number type which results in 0 ( a number)
* isNaN(true) // Returns false, since true converted to Number type results in 1 ( a number)
* isNaN(undefined) // Returns true

**13) Explain Higher Order Functions in javascript.**

Functions that operate on other functions, either by taking them as arguments or by returning them, are called higher-order functions.

Example:

function higherOrder(fn) {

fn();

}

higherOrder(function() { console.log("Hello world") });

**14) In JavaScript what is an argument object?**

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

**15)** **What is the difference between undeclared & undefined variables?** The difference between undeclared and undefined variables are: An undeclared variable has not been declared anywhere in the code, so said variable does not exist. If you try to read an undeclared variable, JavaScript throws an error. An undefined variable has been declared in the program, but no value has been assigned. This means the variable exists, but its value is yet to be defined.

**16) How to write HTML code dynamically using JavaScript?** The innerHTML property is used to write the HTML code using JavaScript dynamically.

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

**17) What are Constructors in JavaScript?**

Constructor in JS are the most powerful members of a class which will be used for object initialization. The special ability of a aconstructor is, it will be invoked only once during the creation of an object.

Eg: function User(name) {

this.name = name;

}

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

**18) How do you create an object in JavaScript?**

Since JavaScript is essentially an object-oriented scripting language, it supports and encourages the usage of objects while developing web applications.

Eg: const student = {

name: 'John',

age: 17

}

**19) What is DOM ?**

The Document Object Model (DOM) is a platform and language-neutral interface that allows programs and scripts to dynamically access and update the content, structure, and style of a document.

**20) 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.

Eg: let userName = "John";

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

**21) What are the arrow functions in JavaScript?**

Arrow functions are a short and concise way of writing functions in JavaScript. The general syntax of an arrow function is as below:

const helloWorld = () => {

console.log("hello world!");

};

**22) What are the conventions of naming a variable in JavaScript?**

* Following are the naming conventions for a variable in JavaScript:
* Variable names cannot be similar to that of reserved keywords. For example, var, let, const, etc.
* Variable names cannot begin with a numeric value. They must only begin with a letter or an underscore character.
* Variable names are case-sensitive.

**23) What is Callback in JavaScript?**

In JavaScript, functions are objects and therefore, functions can take other functions as arguments and can also be returned by other functions.

Fig: Callback function

A callback is a JavaScript function that is passed to another function as an argument or a parameter. This function is to be executed whenever the function that it is passed to gets executed.

**24) How do you debug a JavaScript code?**

All modern web browsers like Chrome, Firefox, etc. have an inbuilt debugger that can be accessed anytime by pressing the relevant key, usually the F12 key. There are several features available to users in the debugging tools.

We can also debug a JavaScript code inside a code editor that we use to develop a JavaScript application—for example, Visual Studio Code, Atom, Sublime Text, etc.

**25) What are the ways of adding JavaScript code in an HTML file?**

There are primarily two ways of embedding JavaScript code:

We can write JavaScript code within the script tag in the same HTML file; this is suitable when we need just a few lines of scripting within a web page.

We can import a JavaScript source file into an HTML document; this adds all scripting capabilities to a web page without cluttering the code.

**26) What do you understand about cookies?**

A cookie is generally a small data that is sent from a website and stored on the user’s machine by a web browser that was used to access the website. Cookies are used to remember information for later use and also to record the browsing activity on a website.

**27) How would you create a cookie?**

The simplest way of creating a cookie using JavaScript is as below:

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

**28) How would you read a cookie?**

Reading a cookie using JavaScript is also very simple. We can use the document.cookie string that contains the cookies that we just created using that string.

The document.cookie string keeps a list of name-value pairs separated by semicolons, where ‘name’ is the name of the cookie, and ‘value’ is its value. We can also use the split() method to break the cookie value into keys and values.

**29) What is the difference between Session storage and Local storage?**

**Session storage**

The data stored in session storage gets expired or deleted when a page session ends.

**Local storage**

Websites store some data in local machine to reduce loading time; this data does not get deleted at the end of a browsing session.

**30) What are Closures in JavaScript?**

Closures provide a better, and concise way of writing JavaScript code for the developers and programmers. Closures are created whenever a variable that is defined outside the current scope is accessed within the current scope.

function hello(name) {

var message = "hello " + name;

return function hello() {

console.log(message);

};

}

//generate closure

var helloWorld = hello("World");

//use closure

helloWorld();

**31) What are the different ways an HTML element can be accessed in a JavaScript code?**

Here are the ways an HTML element can be accessed in a JavaScript code:

* getElementByClass(‘classname’): Gets all the HTML elements that have the specified classname.
* getElementById(‘idname’): Gets an HTML element by its ID name.
* getElementbyTagName(‘tagname’): Gets all the HTML elements that have the specified tagname.
* querySelector(): Takes CSS style selector and returns the first selected HTML element.

**32) Difference between exec() and test() methods**

**exec()**

* It is an expression method in JavaScript that is used to search a string with a specific pattern.
* Once it has been found, the pattern will be returned directly, otherwise, it returns an “empty” result.

**test ()**

* It is an expression method in JavaScript that is also used to search a string with a specific pattern or text.
* Once it has been found, the pattern will return the Boolean value 'true', else it returns ‘false’.

**33) 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.

* Date objects are used to inherit properties from the Date prototype
* Math objects are used to inherit properties from the Math prototype
* Array objects are used to inherit properties from the Array prototype.

**34) Types of errors in javascript?**

Javascript has two types of errors, Syntax error, and Logical error.

**35) 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.

**36) 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.

**37) What is Spread Operator(...)?**

In a function call, we use the spread operator.

It's also to spread one or more arguments that are expected in a function call.The spread operator is used to take an array or an object and spread them.

**38) What are generator functions?**

Generator functions are declared with a special class of functions and keywords using function\*. It does not execute the code, however, it returns a generator object and handles the execution.

**39) What is Object Destructuring?**

Object destructuring is a method to extract elements from an array or an object.

Example 1: Array Destructuring

const arr = [1, 2, 3];

const first = arr[0];

const second = arr[1];

const third = arr[2];

Example 2: Object Destructuring

const arr = [1, 2, 3];

const [first,second,third,fourth] = arr;

console.log(first); // Outputs 1

console.log(second); // Outputs 2

console.log(third); // Outputs 3