Java script

1) What is JavaScript?

**JavaScript** is *a scripting language*. It is different from Java language. It is object-based, lightweight, cross-platform translated language. It is widely used for client-side validation.

### What are the key differences between Java and JavaScript? / How is JavaScript different from Java?

|  |  |
| --- | --- |
| **Java** | **JavaScript** |
| Java is a complete and strongly typed programming language used for backend coding. In Java, variables must be declared first to use in the program, and the type of a variable is checked at compile-time. | JavaScript is a weakly typed, lightweight programming language (most commonly known as scripting language) and has more relaxed syntax and rules. |
| Java is an object-oriented programming (OOPS) language or structured programming languages such as C, C++, or .Net. | JavaScript is a client-side scripting language, and it doesn't fully support the OOPS concept. It resides inside the HTML documents and is used to make web pages interactive (not achievable with simple HTML). |
| Java creates applications that can run in any virtual machine (JVM) or browser. | JavaScript code can run only in the browser, but it can now run on the server via Node.js. |
| The Java code needs to be compiled. | The JavaScript code doesn't require to be complied. |
| Java Objects are class-based. You can't make any program in Java without creating a class. | JavaScript Objects are prototype-based. |
| Java is a Complete and Standalone language that can be used in backend coding. | JavaScript is assigned within a web page and integrates with its HTML content. |
| Java programs consume more memory. | JavaScript code is used in HTML web pages and requires less memory. |
| The file extension of the Java program is written as ".Java" and it translates source code into bytecodes which are then executed by JVM (Java Virtual Machine). | The JavaScript file extension is written as ".js" and it is interpreted but not compiled. Every browser has a JavaScript interpreter to execute the JS code. |
| Java supports multithreading. | JavaScript doesn't support multithreading. |
| Java uses a thread-based approach to concurrency. | JavaScript uses an event-based approach to concurrency. |

[**What are JavaScript Data Types?**](https://www.geeksforgeeks.org/variables-datatypes-javascript/)

There are three major Data types in JavaScript.

* Primitive
  + [Numbers](https://www.geeksforgeeks.org/javascript-numbers/)
  + [Strings](https://www.geeksforgeeks.org/javascript-string/)
  + [Boolean](https://www.geeksforgeeks.org/javascript-boolean/)
  + [Symbol](https://www.geeksforgeeks.org/javascript-symbol-method/)
* Trivial
  + [Undefined](https://www.geeksforgeeks.org/undefined-in-javascript/)
  + [Null](https://www.geeksforgeeks.org/null-in-javascript/)
* Composite
  + [Objects](https://www.geeksforgeeks.org/objects-in-javascript/)
  + [Functions](https://www.geeksforgeeks.org/functions-in-javascript/)
  + [Arrays](https://www.geeksforgeeks.org/arrays-in-javascript/)

Name the types of functions

The types of function are:

* Named - These type of functions contains name at the time of definition. For Example:
  1. function display()
  2. {
  3. document.writeln("Named Function");
  4. }
  5. display();
* Anonymous - These type of functions doesn't contain any name. They are declared dynamically at runtime.
  1. var display=function()
  2. {
  3. document.writeln("Anonymous Function");
  4. }
  5. display();

### 10) In JavaScript what is an argument object?

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

Define closure.

In JavaScript, we need closures when a variable which is defined outside the scope in reference is accessed from some inner scope.

1. var num = 10;
2. function sum()
3. {
4. document.writeln(num+num);
5. }
6. sum();

### 19) 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.

### 20) 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.

|  |  |
| --- | --- |
| **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. |

### 21) What is the use of history object?

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

### 25) What is the difference between == and ===?

The == operator checks equality only whereas === checks equality, and data type, i.e., a value must be of the same type.

### 26) 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:

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

28) How to create objects in JavaScript?

There are 3 ways to create an object in JavaScript.

1. By object literal
2. By creating an instance of Object
3. By Object Constructor
4. emp={id:102,name:"Rahul Kumar",salary:50000}

29) How to create an array in JavaScript?

There are 3 ways to create an array in JavaScript.

1. By array literal
2. By creating an instance of Array
3. By using an Array constructor

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

1. var emp=["Shyam","Vimal","Ratan"];

### 33) Difference between Client side JavaScript and Server side JavaScript?

**Client-side JavaScript** comprises the basic language and predefined objects which are relevant to running JavaScript in a browser. The client-side JavaScript is embedded directly by in the HTML pages. The browser interprets this script at runtime.

**Server-side JavaScript** also resembles client-side JavaScript. It has a relevant JavaScript which is to run in a server. The server-side JavaScript are deployed only after compilation.

### 34) In which location cookies are stored on the hard disk?

The storage of cookies on the hard disk depends on the OS and the browser.

The Internet Explorer stores the cookies on a file username@website.txt. The path is: c:\Windows\Cookies\username@Website.txt.

### 8) What is the difference between undefined value and null value?

**Undefined value:** A value that is not defined and has no keyword is known as undefined value. For example:

1. int number;//Here, a number has an undefined value.

**Null value:** A value that is explicitly specified by the keyword "null" is known as a null value. For example:

1. String str=null;//Here, str has a null value.

### 44) What are the pop-up boxes available in JavaScript?

* Alert Box
* Confirm Box
* Prompt Box

#### **Example of alert() in JavaScript**

1. **<script** type="text/javascript"**>**
2. function msg(){
3. alert("Hello Alert Box");
4. }
5. **</script>**
6. **<input** type="button" value="click" onclick="msg()"**/>**

49) How to handle exceptions in JavaScript?

By the help of try/catch block, we can handle exceptions in JavaScript. JavaScript supports try, catch, finally and throw keywords for exception handling.

50) How to validate a form in JavaScript?

1. **<script>**
2. function validateform(){
3. var name=document.myform.name.value;
4. var password=document.myform.password.value;
6. if (name==null || name==""){
7. alert("Name can't be blank");
8. return false;
9. }else if(password.length**<6**){
10. alert("Password must be at least 6 characters long.");
11. return false;
12. }
13. }
14. **</script>**
15. **<body>**
16. **<form** name="myform" method="post" action="abc.jsp" onsubmit="return validateform()" **>**
17. Name: **<input** type="text" name="name"**><br/>**
18. Password: **<input** type="password" name="password"**><br/>**
19. **<input** type="submit" value="register"**>**
20. **</form>**

51) How to validate email in JavaScript?

1. **<script>**
2. function validateemail()
3. {
4. var x=document.myform.email.value;
5. var atposition=x.indexOf("@");
6. var dotposition=x.lastIndexOf(".");
7. if (atposition**<1** || dotposition**<atposition**+2 || dotposition+2**>**=x.length){
8. alert("Please enter a valid e-mail address \n atpostion:"+atposition+"\n dotposition:"+dotposition);
9. return false;
10. }
11. }
12. **</script>**
13. **<body>**
14. **<form** name="myform"  method="post" action="#" onsubmit="return validateemail();"**>**
15. Email: **<input** type="text" name="email"**><br/>**
17. **<input** type="submit" value="register"**>**
18. **</form>**

### 52) What is this keyword in JavaScript?

The this keyword is a reference variable that refers to the current object. For example:

53) What is the requirement of debugging in JavaScript?

JavaScript didn't show any error message in a browser. However, these mistakes can affect the output. The best practice to find out the error is to debug the code. The code can be debugged easily by using web browsers like Google Chrome, Mozilla Firebox.

To perform debugging, we can use any of the following approaches:

* Using console.log() method
* Using debugger keyword

62) What is the use of a Set object in JavaScript?

The JavaScript Set object is used to store the elements with unique values. The values can be of any type i.e. whether primitive values or object references. For example:

1. function display()
2. {
3. var set = new Set();
4. set.add("jQuery");
5. set.add("AngularJS");
6. set.add("Bootstrap");
7. for (let elements of set) {
8. document.writeln(elements+"**<br>**");
9. }
10. }
11. display();

What is the use of a Map object in JavaScript?

The JavaScript Map object is used to map keys to values. It stores each element as key-value pair. It operates the elements such as search, update and delete on the basis of specified key. For example:

1. function display()
2. {
3. var map=new Map();
4. map.set(1,"jQuery");
5. map.set(2,"AngularJS");
6. map.set(3,"Bootstrap");
7. document.writeln(map.get(1)+"**<br>**");
8. document.writeln(map.get(2)+"**<br>**");
9. document.writeln(map.get(3));
10. }
11. display();

**22. What are object prototypes?**

All javascript objects inherit properties from a prototype. For example,

* Date objects inherit properties from the Date prototype
* Math objects inherit properties from the Math prototype
* Array objects inherit properties from the Array prototype.

### 26. What is recursion in a programming language?

Recursion is a technique to iterate over an operation by having a function call itself repeatedly until it arrives at a result.

function computeSum(arr){

if(arr.length === 1){

return arr[0];

}

else{

return arr.pop() + computeSum(arr);

}

}

computeSum([7, 8, 9, 99]); // Returns 123

### 23. What are callbacks?

A callback is a function that will be executed after another function gets executed.

**function** **divideByHalf**(sum){

console.log(Math.floor(sum / 2));

}

**function** **multiplyBy2**(sum){

console.log(sum \* 2);

}

**function** **operationOnSum**(num1,num2,operation){

**var** sum = num1 + num2;

operation(sum);

}

operationOnSum(3, 3, divideByHalf); // Outputs 3

operationOnSum(5, 5, multiplyBy2); // Outputs 20