1. What is javascript ?

Ans: JavaScript is a scripting language. It’s a client-side scripting language that browser can read.

2. What is the use of isNaN function?

Ans: In Javascript, The isNaN() method returns true if a value is NaN (Not-a-Number). The isNaN () method converts the value to a number before testing it..

3. What is the negative infinity?

Ans: The negative infinity in JS is a constant value that is used to represent a value that is the lowest available. This means that no other number is lesser than this value. It can be generated using a self-made function or by an arithmetic operation.

4. Which company developed javascript?

Ans: JavaScript was developed by **Netscape** Software Company in 1995.

5. What are undeclared and undefined variables?

Ans: Undefined: It occurs when a variable has been declared but has not been assigned with any value.Undefined is not a keyword.

Undeclared: It occers when we try to access any variable that is not initialized or declared earlier using var or const keyword.

6. Write the code for adding new elements dynamically?

Ans: // Create a new paragraph element

let newP = document.createElement("p");

// Add some text content to the paragraph

newP.textContent = "This is a new paragraph";

// Find the element where you want to insert the new paragraph

let container = document.getElementById("container");

// Append the new paragraph as the last child of the container

container.appendChild(newP);

7. What is the difference between ViewState and SessionState?

Ans: The basic difference between these two is that the Viewstate is to manage state at the client’s end, making state managementeasy for end-user.

While SessionState manages state at the server’s end, making easy to manage content from this end too.

8. What is === operator?

Ans: The strict equality (===) **operator** checks whether its two operands are equal, returning a Boolean result. Unlike the equality **operator**, the strict equality **operator** always considers operands of different types to be different.

9. How can the style/class of an element be changed?

Ans: [You can use the **element.style** property to set the style attribute of an element](https://www.bing.com/search?q=How+can+the+style%2Fclass+of+an+element+be+changed%3F&qs=n&form=QBRE&sp=-1&lq=0&pq=how+can+the+style%2Fclass+of+an+element+be+changed%3F&sc=0-49&sk=&cvid=C2417A49DD174AD88DDEBF5B88A3B6B9&ghsh=0&ghacc=0&ghpl=).

[You can use the **element.classList** property to add, remove or toggle classes of an element](https://www.geeksforgeeks.org/how-to-change-style-class-of-an-element-using-javascript/).

[You can use the **document.querySelectorAll** method to get a collection of elements with a specific class and change their styles using a loop](https://bobbyhadz.com/blog/javascript-change-style-of-all-elements-with-class).

10. How to read and write a file using JavaScript?

Ans: There are different ways to read and write a file using JavaScript, depending on the environment and the purpose. For example:

* [If you are using **Node.js**, you can use the **fs module** to access the file system and perform read and write operations](https://www.geeksforgeeks.org/explain-about-read-and-write-of-a-file-using-javascript/).
* [If you are using **JavaScript in a browser**, you can use the **FileReader API** to read files from the user’s machine](https://www.digitalocean.com/community/tutorials/js-file-reader).
* [If you are using **JavaScript in a web app**, you can use the **Quota Management API** to request storage space and access files](https://stackoverflow.com/questions/585234/how-to-read-and-write-into-file-using-javascript).

11. What are all the looping structures in JavaScript?

Ans: The statements for loops provided in JavaScript are:

* **for** statement: repeats until a specified condition evaluates to false.
* **do…while** statement: repeats until a specified condition evaluates to false, but executes at least once.
* **while** statement: repeats until a specified condition evaluates to false.
* **for…in** statement: iterates over the enumerable properties of an object.
* **for…of** statement: iterates over the values of an iterable object, such as an array or a string.
* **labeled** statement: provides a way to identify a loop and break out of it or continue it.
* **break** statement: terminates the current loop and transfers control to the next statement.
* **continue** statement: skips the rest of the current loop iteration and continues with the next one.

12. How can you convert the string of any base to an integer in JavaScript?

Ans: One way to convert a string of any base to an integer in JavaScript is to use the **parseInt()** function. This function takes two arguments: the string value and the base (or radix) of the string.

var num = parseInt("1002", 2);

console.log(num);

13. What is the function of the delete operator?

Ans:  The **delete operator** removes a given property from an object. On successful deletion, it will return true, else false will be returned.

delete object.property

delete object[property]

14. What is the use of Void (0)?

Ans: void(0) is just a short and simple script that evaluates to undefined.

The reason you’d want to do this with the href of a link is that normally, a javascript: URL will redirect the browser to a plain text version of the result of evaluating that JavaScript. But if the result is undefined, then the browser stays on the same page.

15. How can a page be forced to load another page in JavaScript?

Ans:

<script>

window.location = <Path / URL>

</script>

16. What are the disadvantages of using innerHTML in JavaScript?

Ans: Using innerHTML in JavaScript has some disadvantages, such as:

* [It is slow because it re-parses the content even if it is already parsed](https://www.tutorialspoint.com/disadvantages-of-using-innerhtml-in-javascript).
* [It can cause security issues if the content is not sanitized properly](https://www.geeksforgeeks.org/what-is-the-disadvantage-of-using-innerhtml-in-javascript/).
* [It can break event listeners attached to the existing elements](https://www.geeksforgeeks.org/what-is-the-disadvantage-of-using-innerhtml-in-javascript/).
* [It can invalidate references to the existing elements](https://www.geeksforgeeks.org/what-is-the-disadvantage-of-using-innerhtml-in-javascript/).