1.What is JavaScript?

Ans->

JavaScript is the world's most popular programming language.

JavaScript is the programming language of the Web.

JavaScript is easy to learn.

2.What is the use of isNaN function?

Ans->

In JavaScript NaN is short for "Not-a-Number".

The isNaN() method returns true if a value is NaN.

The isNaN() method converts the value to a number before testing it.

3.What is negative Infinity?

Ans->

The negative infinity in JavaScript is a constant value which is used to represent a value which 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->

The first ever JavaScript was created by Brendan Eich at Netscape.

5.What are undeclared and undefined variables?

Ans->

Undeclared — It occurs when a variable which hasn't been declared using var, let or const is being tried to access.

Undefined — It occurs when a variable has been declared using var, let or const but isn't given a value.

6.Write the code for adding new elements dynamically?

Ans->

```
<!DOCTYPE html>

<html>

<head>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></script>

<script>

function appendText() {

    var txt1 = "<p>Text.</p>";              // Create text with HTML

    var txt2 = $("<p></p>").text("Text.");    // Create text with jQuery

    var txt3 = document.createElement("p");

    txt3.innerHTML = "Text.";               // Create text with DOM

    $("body").append(txt1, txt2, txt3);      // Append new elements
```

}

</script>

</head>

<body>

<p>This is a paragraph.</p>

<button onclick="appendText()">Append text</button>

</body>

</html>

7.What is the difference between ViewState and SessionState?

Ans->

ViewState
SessionState

1.Maintained at page level only.	1.Maintained at session level.
2.View state can only be visible from a single available in a user session. page and not multiple pages.	2.Session state value availability is across all pages 3.In session state, user data remains in the server. Data is available to user until the browser is closed or there is session expiration.
3.It will retain values in the event of a	

postback operation occurring.

4.Information is stored on the client's end only.   4.Information is stored on the server.

5.used to allow the persistence of   5.used for the persistence
of user-specific data on the server's end.

page-instance-specific data.

8.What is === operator?

Ans->

the '===' operator tests for strict equality.

i.e it will not do the type conversion hence if the two values are not of the same type, when compared, it will return false.

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

Ans->

Approach 1: Changing CSS with the help of the style property.

Approach 2: Changing the class itself.

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

Ans->

1.filename: It contains the filename to be read, or the whole path if the file is saved elsewhere.

encoding: It stores the file's encoding. 'utf8' is the default setting.

2.callback function: This is a function that is invoked after the file has been read. It requires two inputs:

A.err: If there was an error.

B.data: The file's content.

3.Return Value: It returns the contents contained in the file, as well as any errors that may have occurred.

11.What are all the looping structures in JavaScript?

Ans->

JavaScript supports different kinds of loops:

1.for - loops through a block of code a number of times

2.for/in - loops through the properties of an object

3.for/of - loops through the values of an iterable object

4.while - loops through a block of code while a specified condition is true

5.do/while - also loops through a block of code while a specified condition is true.

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

Ans->

In JavaScript parseInt() function (or a method) is used to convert the passed in string parameter or value to an integer value itself.

This function returns an integer of base which is specified in second argument of parseInt() function.

parseInt() function returns Nan(not a number) when the string doesn't contain number.

13.What is the function of the delete operator?

Ans->

Delete is comparatively a lesser-known operator in JavaScript. This operator is more specifically used to delete JavaScript object properties.

The JavaScript pop(), shift() or splice() methods are available to delete an element from an array. But because of the key-value pair in an object,

the deleting is more complicated.

Note that, the delete operator only works on objects and not on variables or functions.

14.What are all the types of Pop up boxes available in JavaScript?

Ans->

JavaScript has three kind of popup boxes: Alert box, Confirm box, and Prompt box.

1.Alert box:-

An alert box is often used if you want to make sure information comes through to the user.

When an alert box pops up, the user will have to click "OK" to proceed.

2.Confirm box:-

A confirm box is often used if you want the user to verify or accept something.

When a confirm box pops up, the user will have to click either "OK" or "Cancel" to proceed.

If the user clicks "OK", the box returns true. If the user clicks "Cancel", the box returns false.

3.Prompt box:-

A prompt box is often used if you want the user to input a value before entering a page.

When a prompt box pops up, the user will have to click either "OK" or "Cancel" to proceed after entering an input value.

If the user clicks "OK" the box returns the input value. If the user clicks "Cancel" the box returns null.

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

Ans->

It is often used when inserting an expression in a web page might produce some unwanted effect. To remove this effect, "javascript:void(0)" is used.

This expression returns undefined primitive value.

This is often used with hyperlinks.

Sometimes, you will decide to call some JavaScript from inside a link.

Normally, when you click a link, the browser loads a brand new page or refreshes the same page (depending on the URL specified).

But you most likely don't desire this to happen if you have hooked up some JavaScript thereto link.

----

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

Ans->

Step 1: Create a file named index.html. Add a heading and two buttons to it.

One button forcefully loads a page with a live URL and the other button loads a local HTML page.

In the <script> tag we have two functions, one loads gfg home page, and the second loads a local HTML page using window.location property.

Step 2: Create a file named newPage.html. This is the local HTML page that would be loaded by Javascript.

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

Ans->

1.The use of innerHTML very slow.

2.Preserves event handlers attached to any DOM elements.

3.Content is replaced everywhere.

4.Appending to innerHTML is not supported.

5.Old content replaced issue.

6.Can break the document.

7.Can also be used for Cross-site Scripting(XSS).