• What is JavaScript?

(Ans) :

JavaScript is the **Programming Language** for the Web.

JavaScript can update and change both **HTML** and **CSS.**

JavaScript can **calculate**, **manipulate** and **validate** data.

• What is the use of isNaN function?

(Ans): The JavaScript isNaN() Function is used **to check whether a given value is an illegal number or not**. It returns true if the value is a NaN else returns false. It is different from the Number. isNaN() Method.

• What is negative Infinity?

(Ans): The **negative infinity** in JavaScript 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.

• Which company developed JavaScript?

(Ans): JavaScript was invented by **Brendan Eich** in 1995. It was developed for Netscape 2, and became the ECMA-262 standard in 1997. After Netscape handed JavaScript over to ECMA, the Mozilla foundation continued to develop JavaScript for the Firefox browser. Mozilla's latest version was 1.8.5.

• 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 occurs when we try to access any variable that is not initialized or declared earlier using *var* or *const keyword*. If we use *‘typeof’* operator to get the value of an undeclared variable, we will face the *runtime error* with return value as **“undefined”**. The scope of the undeclared variables is always global.

**For example:**

* **Undefined:**

var tops;

undefined

console.log(tops)

**Undeclared:**

//ReferenceError: myVariable is not defined

console.log(myVariable)

• Write the code for adding new elements dynamically?

(Ans): New elements can be dynamically created in JavaScript with the help of **createElement() method**. The attributes of the created element can be set using the **setAttribute() method**. The examples given below would demonstrate this approach.

<!DOCTYPE HTML>

<html>

<head>

<title>

How to dynamically create

new elements in JavaScript?

</title>

</head>

<body>

<h1 style="text-align:center; color:green;">

Tops

</h1>

<!-- Form to add item -->

<form action="#" style="text-align:center;">

<!-- Type of Element -->

<label for="type">

Add Element Type

</label>

<input type="text" id="type"

placeholder="Like: div, h1, li...."

value="li" />

<br /><br />

<!-- Text/Value for the element --->

<label for="value">

Add Element Value

</label>

<input type="text" id="value"

placeholder="Like: Hello Tops"

value="CHILD 2" />

<br /><br />

<!-- Submit the Form -->

<button type="button"

onClick="addItem()">

Add

</button>

</form>

<!-- Parent tag where we add

item as child -->

<ol id="parent">

<li>List Item 1</li>

</ol>

<script>

// Define the addItem() function

// to be called through onclick

function addItem() {

// Get type of element from form

let type = document.

getElementById("type").value;

// Get the text/value for the tag

// from the form

let value = document.

getElementById("value").value;

// createElement() is used for

// creating a new element

type

= document.createElement(type);

// Use value as textnode in this example

type.appendChild(

document.createTextNode(value));

// Append as child to the parent

// tag i.e. ol

document.getElementById(

"parent").appendChild(type);

}

</script>

</body>

</html>

• What is the difference between ViewState and SessionState?

(Ans):

| ViewState | SessionState |
| --- | --- |
| Maintained at page level only. | Maintained at session level. |
| View state can only be visible from a single page and not multiple pages. | Session state value availability is across all pages available in a user session. |
| It will retain values in the event of a postback operation occurring. | In session state, user data remains in the server. Data is available to user until the browser is closed or there is session expiration. |
| Information is stored on the client’s end only. | Information is stored on the server. |
| used to allow the persistence of page-instance-specific data. | used for the persistence of user-specific data on the server’s end. |
| ViewState values are lost/cleared when new page is loaded. | SessionState can be cleared by programmer or user or in case of timeouts. |

• What is === operator?

# (Ans): Strict equality (===)

The **strict equality (===)** operator checks whether its two operands are equal, returning a Boolean result. Unlike the [equality](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Equality) operator, the strict equality operator always considers operands of different types to be different.

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

(Ans):

1. The add() method: It adds one or more classes.
2. The remove() method: It removes one or more classes.
3. The toggle() method: If the class does not exist it adds it and returns true.

• How to read and write a file using JavaScript?

(Ans): The[fs.readFile()](https://www.geeksforgeeks.org/node-js-fs-readfile-method/) and [rs.writeFile()](https://www.geeksforgeeks.org/node-js-fs-writefile-method/) methods are used to read and write of a file using javascript. The file is read using the fs.readFile() function, which is an inbuilt method. This technique reads the full file into memory and stores it in a buffer.

**Syntax:**

fs.readFile( file\_name, encoding, callback\_function )

• What are all the looping structures in JavaScript?

(Ans):

JavaScript supports different kinds of loops:

* for - loops through a block of code a number of times
* for/in - loops through the properties of an object
* for/of - loops through the values of an iterable object
* while - loops through a block of code while a specified condition is true
* do/while - also loops through a block of code while a specified condition is true

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

(Ans): We will convert a string into an integer in Javascript. In JavaScript **[parseInt()](https://www.geeksforgeeks.org/javascript-parseint-function/)** 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 the base which is specified in the second argument of the **parseInt() function**. JavaScript parseInt() function returns Nan( not a number) when the string doesn’t contain a number.

• What is the function of the delete operator?

(Ans): The delete operator **removes a property from an object**. If the property's value is an object and there are no more references to the object, the object held by that property is eventually released automatically.

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

## (Ans): 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.

### Syntax

window.alert("*sometext*");

The window.alert() method can be written without the window prefix.

### Example

alert("I am an alert box!");

## 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**.

### Syntax

window.confirm("*sometext*");

The window.confirm() method can be written without the window prefix.

### Example

if (confirm("Press a button!")) {  
  txt = "You pressed OK!";  
} else {  
  txt = "You pressed Cancel!";  
}

## 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.

### Syntax

window.prompt("*sometext*","*defaultText*");

The window.prompt() method can be written without the window prefix.

### Example

let person = prompt("Please enter your name", "Harry Potter");  
let text;  
if (person == null || person == "") {  
  text = "User cancelled the prompt.";  
} else {  
  text = "Hello " + person + "! How are you today?";  
}

## Line Breaks

To display line breaks inside a popup box, use a back-slash followed by the character n.

### Example

alert("Hello\nHow are you?");

• 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.

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

(Ans): **Approach:**We can use ***[window.location](https://www.geeksforgeeks.org/javascript-window-location-and-document-location-objects/)*** property inside the *script* tag to forcefully load another page in Javascript. It is a reference to a Location object that is it represents the current location of the document. We can change the URL of a window by accessing it.

**Syntax:**

<script>window.location = <Path / URL></script>

• What are the disadvantages of using innerHTML in JavaScript?

(Ans): **Disadvantages of innerHTML**

* Event handlers attached to any DOM element are preserved.
* Replacement is done everywhere.
* It is not possible to append innerHTML.
* Breaks the document.
* Used for Cross-site Scripting.