JavaScript is a dynamic computer programming language. It is lightweight and most commonly used as a part of web pages, whose implementations allow client-side script to interact with the user and make dynamic pages. It is an interpreted programming language with object-oriented capabilities.

JavaScript was first known as **LiveScript,** but Netscape changed its name to JavaScript, possibly because of the excitement being generated by Java. JavaScript made its first appearance in Netscape 2.0 in 1995 with the name **LiveScript**. The general-purpose core of the language has been embedded in Netscape, Internet Explorer, and other web browsers.

The [ECMA-262 Specification](http://www.ecma-international.org/publications/index.html" \t "https://www.tutorialspoint.com/javascript/_blank) defined a standard version of the core JavaScript language.

* JavaScript is a lightweight, interpreted programming language.
* Designed for creating network-centric applications.
* Complementary to and integrated with Java.
* Complementary to and integrated with HTML.
* Open and cross-platform

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.

NEGATIVE\_INFINITY is **the same as the negative value of the global object's Infinity property**. This value behaves slightly differently than mathematical infinity: Any positive value, including POSITIVE\_INFINITY , multiplied by NEGATIVE\_INFINITY is NEGATIVE\_INFINITY .

Brendan Eich, a **Netscape Communications Corporation** programmer, created JavaScript in September 1995. It took Eich only 10 days to develop the scripting language, then known as Mocha.

**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.

Syntax: **document.** **createElement("<tagName>");** // Where <tagName> can be any HTML // tagName like div, ul, button, etc. // newDiv element has been created For Eg: let newDiv = document. createElement("div");

The values of controls of a particular page of the client browser is persisted by ViewState at the time of post back operation is done. If the user requests another page, the data of previous page is no longer available.  
  
- The data of a particular server persists in the server by SessionState. The availability of the user data is up to the completion of a session or closure of the browser.

|  |
| --- |
| A ViewState is a state of a page within a browser wherein the values of controls persist when post back operation is done. When another page is loaded, the previous page data is no longer available.  - SessionState is the data of a user session and is maintained on the server side. This data available until user closes the browser or session time-outs. |

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.

Another way to alter the style of an element is by **changing its class attribute**. class is a reserved word in JavaScript, so in order to access the element's class, you use element. className .

**readFile() and rs.** **writeFile() 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.

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

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.

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

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

JavaScript void 0 means returning undefined (void) as a primitive value. You might come across the term “JavaScript:void(0)” while going through HTML documents. It is used **to prevent any side effects caused while inserting an expression in a web page**.

Approach: We can **use window.** **location 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.

· **The use of innerHTML very slow:** The process of using innerHTML is much slower as its contents as slowly built, also already parsed contents and elements are also re-parsed which h takes time.

· **Preserves event handlers attached to any DOM elements:** The event handlers do not get attached to the new elements created by setting innerHTML automatically. To do so one has to keep track of the event handlers and attach it to new elements manually. This may cause a memory leak on some browsers.

· **Content is replaced everywhere:** Either you add, append, delete or modify contents on a webpage using innerHTML, all contents is replaced, also all the DOM nodes inside that element are reparsed and recreated.

· **Appending to innerHTML is not supported:** Usually, += is used for appending in JavaScript. But on appending to an Html tag using innerHTML, the whole tag is re-parsed.