# qModule – 4 (JAVASCRIPT BASIC & DOM)

1. What is JavaScript?

* JavaScript is a scripting language used to create and control dynamic website content,

Ex: anything that moves, refreshes, or otherwise changes on your screen without requiring you to manually reload a web page.

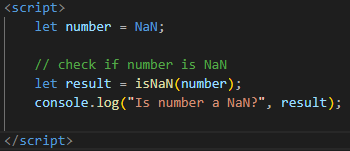
* JavaScript was initially created to enhance the user experience on web pages by adding interactivity, validating input forms, and manipulating the Document Object Model (DOM) to modify the structure and content of web pages in real time.
* It has evolved into a powerful and widely-used language for building web, mobile, server-side applications, and even desktop applications.
* JavaScript contains a standard library of objects like array, date, and math and a core set of language elements like operator control structure and statements.

1. What is the use of isNaN function?

* The isNaN() function checks if a value is NaN (Not a Number) or not.
* isNaN() is not associated with any object.
* This is a global function.
* The isNaN() method returns true if a value is NaN.
* The isNaN() method converts the value to a number before testing it.
* Syntax:

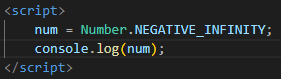
isNaN(value)

* Example:



1. What is negative Infinity?

* Number.NEGATIVE\_INFINITY returns negative infinity.
* The Number.NEGATIVE\_INFINITY property represents the negative infinity value.
* JavaScript shows the NEGATIVE\_INFINITY value as -Infinity.
* Syntax:
* Numner.NEGATIVE\_INFINITY
* Example:



1. Which company developed JavaScript?

* JavaScript was developed by Netscape Communications Corporation. The initial development of JavaScript was led by Brendan Eich in 1995. At that time, it was known as "Mocha" and later renamed to "LiveScript" before finally being named "JavaScript."

1. What are undeclared and undefined variables?

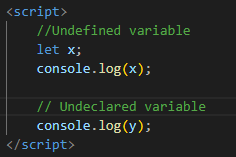
* This is an undefined and undeclared variable in JavaScript is:

1. Undefined

* Undefined variable means a variable has been declared but does not have a value.
* Undefined is not a keyword.
* Undefined is a property of the global object.
* Undefined is a variable in the global scope.

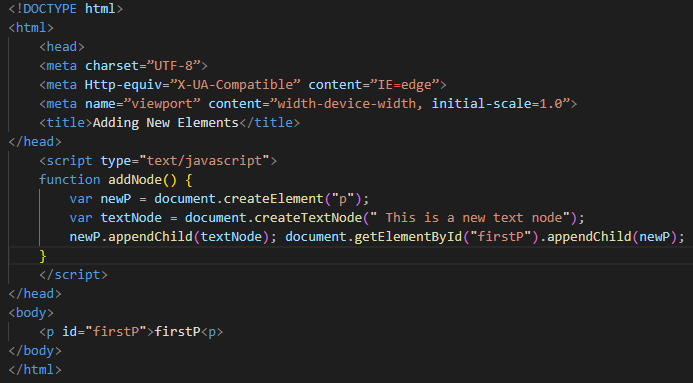
1. Undeclared

* Undeclared variable means that the variable does not exist in the program at all.
* When you use an undeclared variable, it means that the variable name is being used without any prior declaration using the var, let, or const keywords.
* Undeclared variable means that the variable does not exist in the program at all.
* Example



1. Write the code for adding new elements dynamically?

* The createElement() method in JavaScript can be used to create new items dynamically.
* We will create a Simple HTML page with some components defined.
* Once the page is created, we will be writing some outer HTML and adding a button to it so that we could add this HTML when a user clicks on the button.
* This HTML is dynamically added one by one in a list format.
* We are going to write a JavaScript function attached to the onclick event. This event will listen to the trigger and add the desired HTML to the page.
* We will also create a callback that will tell us if the HTML element was successfully added or not.
* Example:

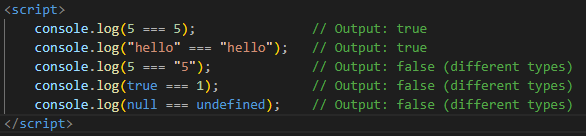


1. What is the difference between ViewState and SessionState?

|  |  |
| --- | --- |
| **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 post-back operation occurring. | In the session state, user data remains on the server. Data is available to the user until the browser is closed or there is a 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 a new page is loaded. | SessionState can be cleared by programmer or user or in case of timeouts. |

1. What is === operator?

* In JavaScript, the === operator is called the strict equality operator. It is used to compare two values for equality without performing type coercion.
* When you use the === operator to compare two values, it checks both the value and the data type of the operands.
* The comparison will return true if the values are equal and of the same type, and false if they are not.
* Example



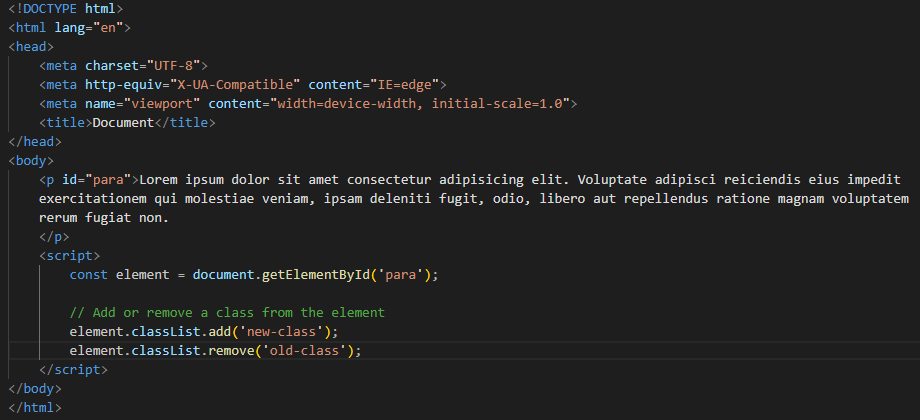
1. How can the style/class of an element be changed?

* In JavaScript, you can change the style or class of an element by accessing its properties and manipulating them.
* There are two ways.

1. Changing the style of an element:

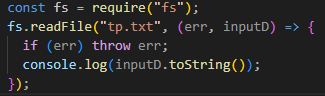


1. Changing the class of an element:



1. How to read and write a file using JavaScript?
2. Reading from the file

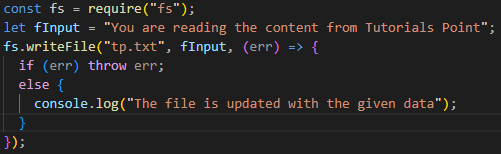
* After the File System module is imported, the reading of the file in JavaScript can be done by using the readFile() function.
* Syntax
* The syntax to read from a file is as follows –
* readFile (path, format, callBackFunc)
* The readFile() function accepts three parameters including one optional parameter.
* Path - The first parameter is the path of the test file from which the contents are to read. If the current location or directory is the same directory where the file which is to be opened and read is located then, only the file name has to be given.
* Format - The second parameter is the optional parameter which is the format of the text file. The format can be ASCII, utf-8 etc.
* CallBackFunc - The third parameter is the call back function which takes the error as the parameter and displays the fault is any raised due to the error.
* Example:
* Following example tries to read the contents of the file populate in the previous example and print it –



* Output: Following is the output of the above example –
* You are reading the content from Tutorials Point
* The text which is displayed in the console is the text which is in the given file.

1. Write operation on a file

* After the File System file is imported then, the writeFile() operation is called. The writeFile() method is used to write into the file in JavaScript. The syntax of this method is as follows –
* writeFile(path,inputData,callBackFunction)
* The writeFile() function accepts three parameters –
* Path - The first parameter is the path of the file or the name of the file into which the input data is to be written. If there is a file already, then the contents in the file are deleted and the input which is given by the user will get updated or if the file is not present, then the file with that will be created in the given path and the input information is written into it.
* inputData - The second parameter is the input data which contains the data to be written in the file that is opened.
* callBackFuntion - The third parameter is the function which is the call back function which takes the error as the parameter and shows the fault if the write operation fails.
* Example: Following is an example of the write operation in files in JavaScript.



1. What are all the looping structures in JavaScript?

* JavaScript provides several looping structures to iterate over a set of values or execute a block of code repeatedly. The looping structures in JavaScript are:

1. for loop:

* The for loop is the most commonly used looping structure in JavaScript. It allows you to define an initialization, condition, and iteration expression within the loop declaration.
* Syntax:

for (initialization; condition; iteration)

{

// Code to be executed

}

1. while loop:

* The while loop executes a block of code as long as a specified condition is true.
* Syntax

while (condition) {

// Code to be executed

}

1. do...while loop:

* The do...while loop is similar to the while loop, but it executes the code block at least once before checking the condition.
* Syntax:

do {

// Code to be executed

} while (condition);

1. for...in loop:

* The for...in loop iterates over the properties of an object. It allows you to loop through the enumerable properties of an object.
* Syntax

for (variable in object) {

// Code to be executed

}

1. for...of loop:

* The for...of loop was introduced in ECMAScript 6 (ES6) and is used to iterate over iterable objects like arrays, strings, maps, sets, etc.
* Syntax:

for (variable of iterable) {

// Code to be executed

}

1. forEach() method:

* The forEach() method is available for arrays and allows you to iterate over each element of an array and execute a provided function.
* Syntax:

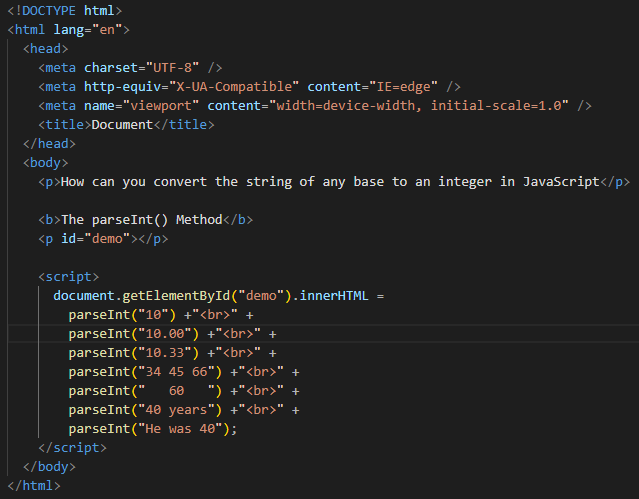
array.forEach((element, index) => {

// Code to be executed for each element

});

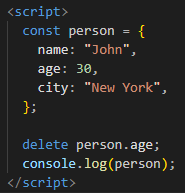
1. How can you convert the string of any base to an integer in JavaScript?

* JavaScript, you can convert a string of any base to an integer using the parseInt() function.
* The parseInt() function takes two arguments: the string to be converted and the base of the string.
* parseInt() function returns Nan( not a number) when the string doesn't contain number. If a string with a number is sent, then only that number will be returned as the output. This function won't accept spaces.
* If the first character cannot be converted, NaN is returned.
* Leading and trailing spaces are ignored.
* Only the first integer found is returned.
* Example:



1. What is the function of the delete operator?

* JavaScript, the delete operator is used to remove a property from an object or delete an element from an array. Its function depends on how it is used:
* Deleting object properties:
* Example



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

* List of types of Pop-up box

1. Alert Box:

* The alert() function displays an alert dialog box with a message and an OK button. It is used to show a message to the user.
* Syntax:

alert("This is an alert box!");

1. Confirm Box

* The confirm() function displays a dialog box with a message and OK and Cancel buttons. It is used to ask the user for confirmation and returns a boolean value indicating the user's choice.
* Syntax:

const result = confirm("Are you sure?");

if (result) {

// User clicked OK

} else {

// User clicked Cancel

}

1. Prompt Box

* The prompt() function displays a dialog box with a message, an input field, and OK and Cancel buttons. It is used to prompt the user for input and returns the entered value as a string.
* Syntax:

const name = prompt("Please enter your name:");

console.log("Hello, " + name);

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

* 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.
* The word void (0) means completely empty space.
* Example:



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

* JavaScript, you can force a page to load another page by modifying the window.location object. There are a few different methods you can use depending on the desired behavior
* Directly set the window.location.href property:
* Syntax:

window.location.href = “https://www.google.com”;

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

* Disadvantages of using innerHTML property in JavaScript:
* 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 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.
* Can break the document: There is no proper validation provided by innerHTML, so any valid HTML code can be used. This may break the document of JavaScript. Even broken HTML can be used, which may lead to unexpected problems.
* Can also be used for Cross-site Scripting (XSS): The fact that innerHTML can add text and elements to the webpage, can easily be used by malicious users to manipulate and display undesirable or harmful elements within other HTML element tags. Cross-site Scripting may also lead to loss, leak and change of sensitive information.