Different Kinds of Loops

**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
* for/each - method calls a function (a callback function) once for each array element.

1) JavaScript For loop

The **JavaScript for loop** *iterates the elements for the fixed number of times*. It should be used if number of iteration is known. The syntax of for loop is given below.

**for (initialization; condition; increment)**

**{**

**code to be executed**

**}**

**As well as :**

**<script>**

**for (i=1; i<=2; i++)**

**{**

**document.write(i + "<br/>")**

**}**

**</script>**

**Output:**

**1**

**2**

2) JavaScript: For-In Loop

This JavaScript tutorial explains how to use the for-in loop with syntax and examples.

**Description**

In JavaScript, the for-in loop is a basic control statement that allows you to loop through the properties of an object. The statements of code found within the loop body will be executed once for each property of the object.

**Syntax**

The syntax for the for-in loop in JavaScript is:

for (variable in object) {

// statements

}

Parameters or Arguments

variable

The name of a variable, an element of an array, or the property of an object.

object

A valid object or name of an object whose properties will be iterated through.

statements

The statements of code to execute each pass through the loop.

**Note**

The body of the for-in loop will execute once for each property of the object.

The for-in loop does not loop through all of the properties of the object. It is can only loop through the enumerable properties such as user-defined properties or inherited user-defined properties. Any properties that are flagged as non-enumerable (such as built-in properties or methods) will not be iterated through using the for-in loop.

3)The For Of Loop

The JavaScript for of statement loops through the values of an iterable object.

It lets you loop over iterable data structures such as Arrays, Strings, Maps, NodeLists, and more:

**Syntax**

for (variable of iterable) {  
  // *code block to be executed*  
}

variable - For every iteration the value of the next property is assigned to the variable. Variable can be declared with const, let, or var.

iterable - An object that has iterable properties.

4)JavaScript forEach()

The forEach() method calls a function and iterates over the elements of an array. The forEach() method can also be used on Maps and Sets.

**JavaScript forEach**

The syntax of the forEach() method is:

array.forEach(function(currentValue, index, arr))

Here,

function(currentValue, index, arr) - a function to be run for each element of an array

currentValue - the value of an array

index (optional) - the index of the current element

arr (optional) - the array of the current elements

-----------------------End--------------------------

**RESUME in JSON format :**

{

"basics": {

"name": “vigneshwar”,

"email": “vigneshmenon2303@gmail.com”,

"phone": “8526857001”,

"summary": “To work hard with full determination and dedication to achieve organizational as well as personal goals”,

"location": {

"address": “3/5,KRMS Garden,Sulur”,

"postalCode": “641402”,

"district": “coimbatore”,

"state": “Tamil Nadu”,

"country": “INDIA”,

},

},

"work": [{

"name": “PEPS industries”,

"position": “Asst,supervisor”,

"url": “https//:PEPSindustries.pvt.ltd.com”,

"startDate": “14-04-2019”,

"endDate": “07-07-2021”,

"summary": “I was asst.supervisor in that company and I achieve my start from the company”,

"education": [{

"institution": “Ambal Professional Group Of Institutions”,

"area": “Engineering”,

"studyType": “Bachelor Of Engineering”,

"from": “2017”,

"to": “2021”,

"Grade": “6.5”,

}],

"skills": [{

"name": “MS- office”,

"level": “advanced”,

"name": “designing-CATIA”,

"level": “Basics”,

}],

"languages": [{

"language": “English”,

"fluency": “Read,Write and Speak”,

"language": “Tamil”,

"fluency": “Read,Write and Speak”,

"language": “Malayalam”,

"fluency": “Read,Write and Speak”,

"language": “Hindi”,

"fluency": “Speak”,

}],

“Aknowledgement”: [{

“I am here by declare the above given datails is true and best of my knowledge”,

}],

}

“date” : “\_\_\_\_\_\_\_\_\_\_\_”

“place” : “\_\_\_\_\_\_\_\_\_\_\_”

----------------------------------------- END-----------------------------------------------------

# JavaScript: window, document and screen

**window is the root of everything, screen just has screen dimensions, and document is top DOM object**.

## Window

The JavaScript **window object** sits at the top of the JavaScript Object hierarchy and represents the browser window. The window object is supported by all browsers. All global **JavaScript objects** , functions, and variables automatically become members of the window object. The window is the first thing that gets loaded into the **browser** . This window object has the majority of the properties like length, innerWidth, innerHeight, name, if it has been closed, its parents, and more.

The window object represents the current **browsing context** . It holds things like window.location, window.history, window.screen, window.status, or the **window.document** . Each browser tab has its own top-level window object. Each of these windows gets its own separate global object. window.window always refers to window, but **window.parent** and window.top might refer to enclosing windows, giving access to other execution contexts. Moreover, the window property of a window object points to the window object itself. So the following ststements all return the same window object:

EXAMPLE (properties) :

window.window

window.window.window

window.window.window.window

## Document

The **Document interface** represents any web page loaded in the browser and serves as an entry point into the web page's content, which is the DOM tree. When an HTML document is loaded into a **web browser** , it becomes a document object. It is the root node of the HTML document. The document actually gets loaded inside the window object and has properties available to it like title, URL, cookie, etc. HTML documents, served with the **"text/html"** content type, also implement the HTMLDocument interface, whereas XML and SVG documents implement the XMLDocument interface.

## Screen

Screen is a small information object about physical **screen dimensions** . It can be used to display screen width, height, colorDepth, pixelDepth etc. It is not mandatory to write **window prefix** with screen object. It can be written without window prefix.

**Properties :**

screen.width

screen.height

screen.availWidth

screen.availHeight

screen.colorDepth

screen.pixelDepth

---------------------------------------END--------------------------------------------------