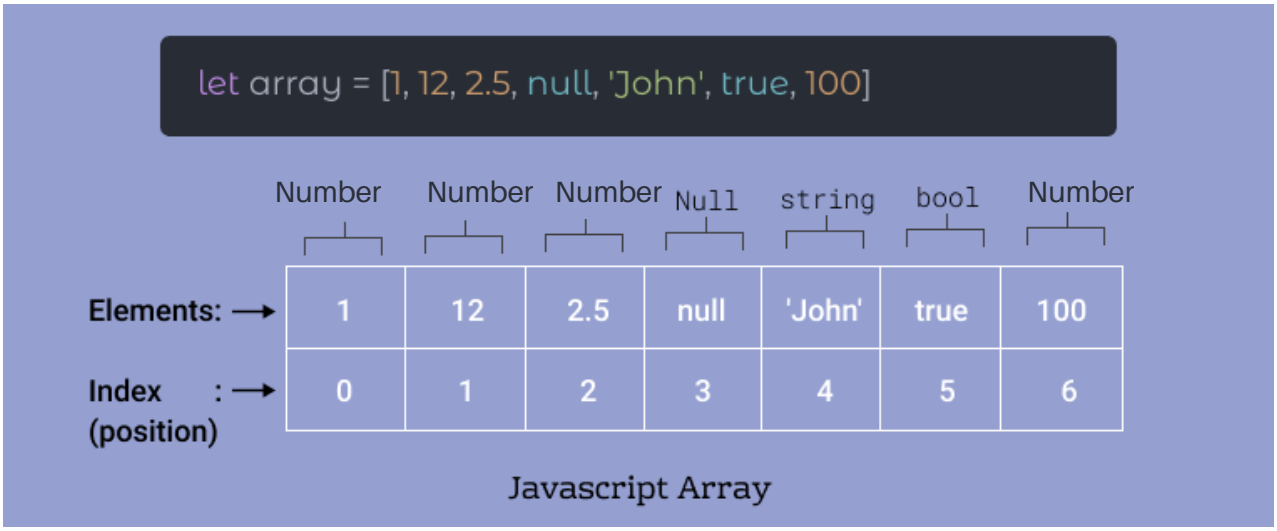
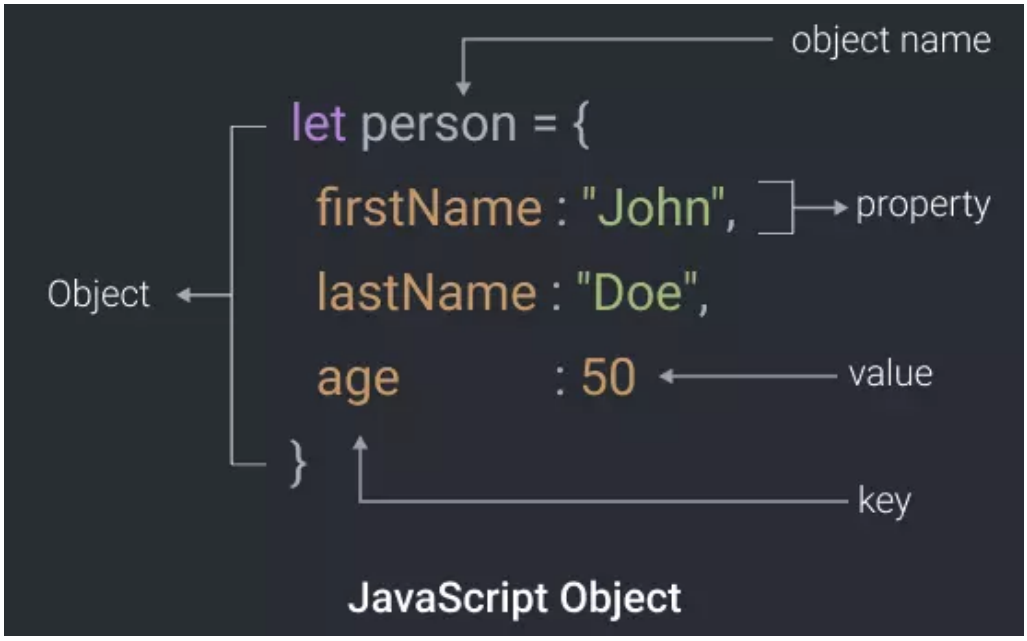
ARRAY IN JAVASCRIPT

In JavaScript, array is a single variable that is used to store different elements. It is often used when we want to store list of elements and access them by a single variable. Unlike most languages where array is a reference to the multiple variable, in JavaScript array is a single variable that stores multiple elements.



OBJECT IN JAVASCRIPT

Objects, in JavaScript, is the most important data-type and forms the building blocks for modern JavaScript. An object can be created with figure brackets {…} with an optional list of properties. A property is a “key: value” pair, where a key is a string (also called a “property name”), and value can be anything.



JAVASCRIPT - FOR LOOP

The 'for' loop is the most compact form of looping. It includes the following three important parts –

The loop initialization where we initialize our counter to a starting value. The initialization statement is executed before the loop begins.

The test statement which will test if a given condition is true or not. If the condition is true, then the code given inside the loop will be executed, otherwise the control will come out of the loop.

The iteration statement where you can increase or decrease your counter.

```
<SCRIPT TYPE = "TEXT/JAVASCRIPT">
VAR COUNT;
        DOCUMENT.WRITE("STARTING LOOP" + "<BR />");
FOR(COUNT = 0; COUNT < 10; COUNT++) {
          DOCUMENT.WRITE("CURRENT COUNT : " + COUNT );
          DOCUMENT.WRITE("<BR />");}
        DOCUMENT.WRITE("LOOP STOPPED!");
</SCRIPT>
```

OUTPUT :

```
STARTING LOOP
CURRENT COUNT : 0
CURRENT COUNT : 1
CURRENT COUNT : 2
CURRENT COUNT : 3
CURRENT COUNT : 4
CURRENT COUNT : 5
CURRENT COUNT : 6
CURRENT COUNT : 7
CURRENT COUNT : 8
CURRENT COUNT : 9
LOOP STOPPED!
```

FOR IN

The for...in loop iterates through properties in the prototype chain. This means that we need to check if the property belongs to the object using hasOwnProperty whenever we loop through an object with the for...in loop:

```
VAR POPULATION = {
  MALE: 4,
  FEMALE: 93,
  OTHERS: 10
};
FOR (CONST KEY IN POPULATION) {
    CONSOLE.LOG(KEY ,POPULATION[KEY]);
  }
}
```

Html DOM Property:

- **innerHTML:**

The innerHTML property sets or returns the HTML content

of an element.

- **value:**

The value property sets or return the value of Form inputs.

- **getAttribute:**

returns the value of particular attribute.

- **setAttribute:**

sets the value of particular attribute.

- **style.cssProperty:**

returns or set the css property.

- **src:**

sets or return src of image.

- **className:**

to replace the className or push the new className

Events:

HTML events are "things" that happen to HTML elements.

When JavaScript is used in HTML pages, JavaScript can

"react" on these events

- **Mouse Events:**

onclick, ondblclick, onmousedown, onmousemove, onmouseup

- **Keyboard Events:**

onkeydown, onkeyup ,keyup, keydown, keypressed, e.key, e.code, e.altKey, e.ctrlKey, e.shiftKey

- **Frame/Object Events:**

onload, onscroll

- **Form Events:**

onblur, onchange, onfocus, oninput, onselect, onsubmit

- **Clipboard Events:**

oncopy, oncut, onpaste

QUERYSELECTOR

It will return the first element that matches the specified CSS selector in the document. The querySelector() method only returns the first element that matches the specified selectors.

```
document.querySelector('#myclass');
document.querySelector('.myclass');
document.querySelector('li');
```

MULTIPLE ELEMENT SELECTOR:-

It is used to select multiple HTML elements within a document. There are three ways in which we can select elements in a DOM using multiple element selectors.

- QUERYSELECTORALL()
- GETELEMENTSBYTAGNAME()
- GETELEMENTSBYCLASSNAME()

QUERYSELECTORALL:-

It will return a list of the document's elements that match the specified group of selectors. The querySelectorAll() method returns all elements in the document as a static NodeList object. The NodeList object represents a collection of nodes. The nodes can be accessed by index numbers. The index starts at 0.

- document.querySelectorAll('.heading') // method 1

```
// method 2
<div>
<article> My article 1</article>
<article> My article 2</article>
</div>
var mytag = document.querySelectorAll('div article');
for (var  i = 0; i < mytag.length; i++) {
mytag[i].style.border = '1px solid blue';}
```

KEEP LEARNING AND HAPPY CODING