Javascript - Day -3 : JS array & objects

To iterate over a JSON object in JavaScript, there are multiple types of for loops you can use, depending on your specific needs. Here are four common types of for loops you can use to iterate over a JSON object:

**For loop:**

A for loop is a basic loop construct in JavaScript. You can use a for loop to iterate over an array of keys or indices in a JSON object. Here's an example:

var myObj = {

"name": "John",

"age": 30,

"city": "New York"

};

for (var key in myObj) {

console.log(key + ": " + myObj[key]);

}

In this example, we define a JSON object called "myObj" with three key-value pairs. We then use a for loop with the "in" keyword to loop through each key in the object and log the key and value to the console.

**For...in loop:**

A for...in loop is similar to a basic for loop, but it is specifically designed for looping through the keys in an object. Here's an example:

var myObj = {

"name": "John",

"age": 30,

"city": "New York"

};

for (var key in myObj) {

console.log(key + ": " + myObj[key]);

}

This code will produce the same output as the previous example.

ForEach loop:

A forEach loop is a higher-order function in JavaScript that can be used to iterate over an array or object. However, it is only available for arrays and not objects. Here's an example:

var myArr = [1, 2, 3, 4];

myArr.forEach(function(value) {

console.log(value);

});

In this example, we define an array called "myArr" and use the forEach method to loop through each value in the array and log it to the console.

For...of loop:

A for...of loop is a newer loop construct in JavaScript that can be used to loop through iterable objects, including arrays, maps, and sets. However, it is not available for plain objects. Here's an example:

var myArr = [1, 2, 3, 4];

for (var value of myArr) {

console.log(value);

}

In this example, we use a for...of loop to loop through each value in the array and log it to the console.

In conclusion, these are four common types of for loops we can use to iterate over a JSON object in JavaScript. The best one to use depends on the specific requirements of the code.

1. Create your own resume data in JASON format

{

"name": "Sulfiya Beegum",

"email": ""Sulfiya @example.com",

"phone": "123-456-7890",

"address": "123 Main Street, Anytown, USA",

"summary": "Experienced software engineer with expertise in DB2 SQL, java and Devops",

"skills": [

"DB2 SQL",

java

"Devops"

],

"experience": [

{

"position": "ASE",

"company": "Acme Corporation",

"startDate": "November 2018",

"endDate": "Present",

"description": "Develop and maintain banking application."

},

{

"position": "SE",

"company": "XYZ Inc.",

"startDate": "May 2016",

"endDate": "December 2017",

"description": "Worked on front-end and back-end development for e-commerce applications."

}

],

"education": [

{

"degree": "Bachelor of Technology in Computer Science",

"school": "University of Anytown",

"graduationDate": "May 2016"

}

]

}

1. Write a blog on Difference between document and window object.

In JavaScript, the Document, Screen and Window objects are some most commonly used objects. These objects provide access to various properties and methods that allow you to manipulate and control the web page displayed in a browser. These three objects are essential to building dynamic and interactive web pages, but they have different roles and behaviors.

**The Window Object**

The Window object is the top-level object in the browser's object hierarchy. It represents the browser window or tab and provides access to the properties and methods of the current window. The Window object is created automatically by the browser when a new window or tab is opened and is accessible through the global "window" variable.

There are different methods and properties which are associated with the window object. We can use the location property to get the URL details, hostname, port number of the current window.

We can use alert () method to create an alert on the current window. We can use open () method to open a new window etc.

**The Document Object**

The Document object is a child of the Window object. It represents the HTML document that is currently loaded in the browser window. It provides access to the HTML elements of the page and allows you to manipulate the content and structure of the document.

The Document object contains various properties and methods that allow you to access and modify the content of the HTML document. For example, you can use the "getElementById()" method to retrieve a specific HTML element from the page or use the "write()" method to write text directly to the document.

**The Screen Object**

The screen object represents the user's screen or display, and it provides information about the screen's dimensions and characteristics. The screen object has several properties that can be useful in web development, such as:

screen.width: the width of the screen in pixels

screen.height: the height of the screen in pixels

screen.availWidth: the available width of the screen in pixels (excluding the taskbar and other system UI)

**Differences between the Window and Document Objects**

**Scope**:

The window object has a global scope and is accessible from anywhere within the JavaScript code. This means that any variable or function declared without the "var" keyword is automatically attached to the window object.

The screen object and document object are properties of the window object and have a more limited scope. They can only be accessed within the context of the current browser window or tab.

**Properties**:

The window object has several properties that are not available on the screen or document objects, such as the "innerWidth" and "innerHeight" properties, which represent the size of the browser window's viewport.

The screen object has properties that provide information about the physical screen dimensions, such as the "width" and "height" properties, which represent the total size of the screen.

The document object has properties that provide access to the content of the web page, such as the "title" and "URL" properties, which represent the title and URL of the current page.

**Methods**:

The window object has several methods that are not available on the screen or document objects, such as the "open" method, which opens a new browser window or tab.

The screen object and document object have methods that provide information or modify their respective areas. For example, the document object has methods like "getElementById" and "querySelector" that allow you to access and manipulate elements on the web page, while the screen object has methods like "availWidth" and "availHeight" that provide information about the available screen space.

In summary, while the window, screen, and document objects are all essential to building dynamic and interactive web pages, they have different scopes of properties and methods that are tailored to their specific roles and responsibilities.