**Arrays & Objects**

1. **For the given JSON iterate over all for loops (for, for in, for of forEach)**

Assuming JSON object:

const jsonObject = {

"name": "John",

"age": 30,

"city": "New York"

};

**For...in Loop (for JSON objects):**

for (const key in jsonObject) {

if (jsonObject.hasOwnProperty(key)) {

const value = jsonObject[key];

console.log(`${key}: ${value}`);

}

**}**

**For...of Loop (for JSON arrays):**

JSON arrays are just regular JavaScript arrays, so you can use the for...of loop as you would with any other array.

const jsonArray = [1, 2, 3, 4, 5];

for (const element of jsonArray) {

console.log(element);

}

**For Loop (for JSON arrays):**

You can also use a traditional for loop to iterate over JSON arrays.

const jsonArray = [1, 2, 3, 4, 5];

for (let i = 0; i < jsonArray.length; i++) {

console.log(jsonArray[i]);

}

**ForEach Loop (for JSON arrays):**

If you have an array, you can use the forEach method to iterate over its elements.

const jsonArray = [1, 2, 3, 4, 5];

jsonArray.forEach((element) => {

console.log(element);

});

Remember to choose the appropriate loop based on whether you are working with a JSON object or a JSON array, as shown in the examples above.

1. **Create your own resume data in JSON format**

{

"name": "Pavithra Velmurugan",

"contact": {

"email": "Pavithravelmurugan1497@gmail.com",

"phone": "+91 7395964293",

"address": "No:13A, 1st Main road, Thendral Nagar, Chennai-72"

},

"summary": "Process Executive with a positive attitude. Strong problem-solving skills and a proven track record of delivering high-quality code.",

"education": [

{

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

"school": "University of Madras",

"graduation\_year": 2019

},

],

"experience": [

{

"position": "Process Executive",

"company": "Cognizant Technology Solutions ",

"start\_date": "2020-03-01",

"end\_date": "2023-09-29",

"responsibilities": [

"Played a major role as a critical resource to save the project",

"Trained many cross training work flows",

"Handling Clients and solve their problems"

]

},

"skills": [

"JavaScript",

"React",

"Node.js",

"Python",

"SQL",

"RESTful APIs",

"Git",

"Agile Development",

"Problem Solving",

"Team Leadership"

],

"languages": ["English (Fluent)", "Spanish (Basic)"],

"interests": ["Singing", "vegetable carving", "Painting"]

}

1. **Difference between window, Screen, and document in javascript**

In JavaScript, window, screen, and document are three important objects that are part of the web browser environment. They serve different purposes and provide access to various properties and methods related to the web page and the browser. Here's a breakdown of the differences between these objects:

**window Object:**

The window object represents the browser window or the global context in the web browser environment.

It is the top-level object and serves as the global object for JavaScript in a web page.

It contains various properties and methods related to the browser window, such as window.location (to get or set the URL of the current page), window.alert() (to display a popup alert), and window.setTimeout() (to schedule a function to run after a specified time).

You can use window to access global variables and functions defined on the page.

Example:

window.location.href = 'https://example.com';

**screen Object:**

The screen object provides information about the user's screen or monitor.

It contains properties like screen.width (to get the width of the screen in pixels), screen.height (to get the height of the screen in pixels), and screen.availWidth (to get the available width for the browser window, excluding system toolbars).

It is useful for building responsive web applications or for determining the screen size for layout purposes.

Example:

const screenWidth = screen.width;

const screenHeight = screen.height;

**document Object:**

The document object represents the web page or document that is currently loaded in the

browser window.

It provides access to the content of the HTML document, including elements like forms, images, links, and text.

You can use it to manipulate the content and structure of the web page using methods like document.getElementById() (to select elements by their ID), document.querySelector() (to select elements using CSS selectors), and document.createElement() (to create new HTML elements).

The document object is a key tool for DOM (Document Object Model) manipulation.

Example:

const headingElement = document.getElementById('myHeading');

headingElement.textContent = 'New Heading';

In summary, window deals with the browser window and global environment, screen provides information about the user's screen, and document allows you to interact with and manipulate the content of the web page itself. These objects are essential for building interactive and dynamic web applications.