**Session2:**

**Write a blog on the difference between document and window objects.**

In this blog, we'll dissect the differences between these two objects and explore their unique characteristics. Two crucial objects that play pivotal roles in manipulating and interacting with web pages are the document and window objects.

**The Window Object:**

The window object represents the browser window or tab in which the web page is loaded. It serves as the global object in client-side JavaScript and provides access to various properties and methods related to the browser environment.

Browser Interaction: The window object facilitates interactions with the browser environment, such as navigating to URLs, opening new windows or tabs, and managing the history stack.

Global Scope: The window object is the global scope in JavaScript. All global variables and functions are implicitly defined as properties and methods of the window object.

Properties and Methods: It exposes properties and methods for manipulating the browser window, accessing screen dimensions, setting timeouts, and handling events like onload and onresize.

**The Document Object:**

The document object represents the web page loaded in the browser window. It provides access to the structure and content of the HTML document, enabling dynamic manipulation of elements and their attributes.

DOM Manipulation: The document object serves as the entry point to the Document Object Model (DOM), allowing developers to access, modify, and manipulate elements within the web page.

Element Selection: It offers methods like getElementById, getElementsByClassName, and querySelector to select specific elements within the document for manipulation.

Content Access: The document object provides properties and methods to access the content of the document, such as textContent, innerHTML, and innerText.

Session 3:

1.How to compare two JSON have the same properties without order?

1. let obj1 = {name: “Person 1”, age:5};
2. let obj2 = {age:5, name: “Person 1”};

Ans=> To compare two JSON that have same properties without order, we have to follow 3 steps:

1. Convert the both JSON objects into strings.
2. Sort the properties within the each JSON.
3. Compare it.

Code:

Function compareIT(obj1, obj2){

var str1 = JSON.stringify(obj1, object.keys(obj1).sort());

var str2 = JSON.stringify(obj2, object.keys(obj2).sort());

return str1 === str2;

}

let obj1 = {name: “Person 1”, age:5};

let obj2 = {age:5, name: “Person 1”};

console.log(compareIt(obj1, obj2));

the output will be: Objects are equal(ignoring property order) because the properties of both objects are the same, regardless of their order.

2. Use the rest countries’ API URL -> <https://restcountries.com/v3.1/all> and display all the country flags in the console.

Ans=> Code:

fetch('https://restcountries.com/v3.1/all')

.then(response => response.json())

.then(data => {

// Iterate over each country object and display its flag

data.forEach(country => {

if (country.flags) {

console.log(country.flags.png); // Assuming the flags are PNGs

}

});

})

.catch(error => {

console.log('Error fetching data:', error);

});

This code fetches the data from the given API URL, parses the JSON response , and then iterates over each country object. If a country object has a flag (assuming its represented by a property named ‘flags’ and contains a URL to the flag image ), it logs the URL of the flag image to the console.

3. Use the same rest countries and print all the countries names, regions sub-regions and population.

Ans=> Code:

fetch('https://restcountries.com/v3.1/all')

.then(response => response.json())

.then(data => {

data.forEach(country => {

console.log('Name:', country.name.common);

console.log('Region:', country.region);

console.log('Sub-region:', country.subregion);

console.log('Population:', country.population);

console.log('-------------------------');

});

})

.catch(error => {

console.log('Error fetching data:', error);

});

This code uses the **fetch()** function to make a GET request to the REST Countries API and retrieve data about all countries. It then parses the response as JSON and logs the country name, region, sub-region, and population for each country in the console.