**Difference between Document and window Object**

**Introduction:**

* When working with JavaScript, developers frequently encounter the Document and Window objects, both of which play crucial roles in manipulating the content and behavior of web pages.
* Although they might seem similar at first glance, these objects serve distinct purposes and operate within different contexts.
* In this blog post, we will delve into the key differences between the Document and Window objects in JavaScript.

**Document Object:**

* The Document object represents the entire HTML document and serves as an entry point to manipulate its content.
* It provides methods and properties to access, modify, and create elements within the document.
* When you interact with the Document object, you are essentially working with the structure and content of the web page.

**Key characteristics of the Document object include:**

Content Manipulation:

* The Document object allows you to access and modify the content of HTML elements.
* You can change text, attributes, and even add or remove elements dynamically.

1. Changing the text content of an element with ID 'exampleElement'

Example:

const element = document.getElementById('exampleElement');

element.textContent = 'New Text Content';

1. DOM Manipulation: The Document Object Model (DOM) is a programming interface provided by the Document object. It represents the document as a tree structure of elements, allowing developers to navigate and manipulate its hierarchy.

Example:

Accessing the parent element of an element with ID 'childElement'

const childElement = document.getElementById('childElement');

const parentElement = childElement.parentNode;

**Window Object:**

* On the other hand, the Window object represents the browser window or a tab.
* It is the global object in a client-side JavaScript environment and encompasses not only the document but also additional features related to the browser.

**Key characteristics of the Window object include:**

1. Global Scope: Variables and functions declared in the global scope are attached to the Window object. This means that you can access them globally within your JavaScript code.

// Example: Creating a global variable attached to the Window object

window.globalVariable = 'I am a global variable';

1. Browser-Related Functionality: The Window object provides methods to interact with the browser, such as opening and closing tabs, navigating to different URLs, and managing cookies.

// Example: Opening a new browser window

window.open('https://www.example.com', '\_blank');

1. Timers and Events: The Window object handles timers (e.g., 'setTimeout' and 'setInterval') and events (e.g., 'onload', 'onresize'). These features are related to the overall browser environment rather than the specific document.

// Example: Setting a timer to execute a function after 2000 milliseconds

window.setTimeout(() => {

console.log('Timer executed!');

}, 2000);

**Difference:**

**Document-**

* It represents any HTML document or web page that is loaded in the browser.
* It is loaded inside the window.
* It is the object of window property.
* All the tags, elements with attributes in HTML are part of the document.
* We can access the document from a window using the window. Document
* The document is part of BOM (browser object model) and dom (Document object model)
* Properties of document objects such as title, body, cookies, etc can also be accessed by a window like this window. document.title
* syntax:

document.propertyname;

* example:

document.title : will return the title of the document.

**Window-**

* It represents a browser window or frame that displays the contents of the webpage.
* It is the very first object that is loaded in the browser.
* It is the object of the browser.
* Global objects, functions, and variables of JavaScript are members of the window object.
* We can access the window from the window only. i.e. window.window
* The window is part of BOM, not DOM.
* Properties of the window object cannot be accessed by the document object.
* syntax:

window.propertyname;

* example:

window.innerHeight : will return the height of the content area of the browser

**Conclusion:**

In summary, while the Document and Window objects in JavaScript are interconnected, they serve distinct purposes. The Document object focuses on the content and structure of the HTML document, providing methods for document manipulation, while the Window object encompasses broader functionalities related to the browser environment. Understanding the differences between these two objects is essential for effective client-side web development.