**Difference between Document &Window Object**

horizontal line

# Placeholder image

# Introduction

When working with JavaScript, two essential objects come into play: the document object and the window object. These objects are crucial for manipulating and interacting with web pages. However, understanding the differences between them is essential for effective web development. In this blog post, we will delve into the dissimilarities between the document and window objects and explore their unique roles in the JavaScript environment.

The Document Object:

The document object represents the web page itself. It serves as an interface to access and manipulate the content within an HTML document. The document object provides various properties, methods, and events that allow developers to interact with elements present on the page. Through the document object, you can access and modify HTML elements, add or remove elements dynamically, change styles and attributes, and even handle user interactions.

For example, to access an HTML element with the id "myElement" using the document object, you would use the following code:

javascript

Copy code

var element = document.getElementById("myElement");

The Window Object:

On the other hand, the window object represents the browser window or the frame in which the web page is displayed. It acts as the global object in the browser's JavaScript environment, providing access to various properties, methods, and events related to the window and its content.

The window object encompasses the entire browser window, including the document object. It provides functionalities beyond the document, such as managing browser history, controlling navigation, opening new windows or tabs, and interacting with the browser's location and history. The window object also allows access to the global scope, which means that global variables and functions are accessible through the window object.

For instance, to open a new browser window using the window object, you can use the following code:

javascript

Copy code

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

Key Differences:

Scope: The document object refers to the content within an HTML document, while the window object refers to the entire browser window or frame.

Hierarchy: The document object is a property of the window object. It can be accessed using the window.document or simply document notation.

Functionality: The document object is primarily focused on manipulating and interacting with the content of a single HTML document. On the other hand, the window object provides a broader range of functionalities, including managing windows, frames, and the browser environment.

Global Scope: Global variables and functions are accessible through the window object due to its global scope nature. However, the document object does not provide direct access to the global scope.

Conclusion:

Understanding the distinction between the document and window objects is vital for effective JavaScript programming in the web development realm. The document object enables interaction with the content of an HTML document, while the window object represents the browser window itself, providing additional functionalities beyond the document. By leveraging the capabilities of both objects, developers can create dynamic and interactive web pages with ease.

As you delve deeper into JavaScript development, remember to consider the document and window objects' unique roles, properties, methods, and events to make the most out of your web projects.

### *Window object and Document object often look alike and confusing ??*

### A clear understanding of Browser Object Model (BOM) and Document Object Model(DOM) resolves the problem.

## Browser Object Model (BOM)

### The Browser Object Model contains objects that represent the current browser window or tab. The topmost object in the BOM is the *window* object representing the window or tab or an iframe sometimes. Window object has properties like browser history, location and the device’s screen etc. In case of multi tab browser, *a window object represents a single tab*, but some of its properties like *innerHeight*, *innerWidth* and methods like *resizeTo()* will affect the whole browser window.

## Document Object Model

### When a web page is loaded, the browser creates a Document Object Model of the page. The *document object* represents the whole html document as a tree of Objects(HTML, HEAD, BODY, and other HTML tags). It is the root element that represents the html document.

### 

### Figure 1 : Hierarchy of Window and Document object using BOM and DOM

### Now lets visualize the difference between window and document.

## Window Vs Document

### Window object : It is the top most object and outermost element of the object hierarchy as shown in *Figure 1.*

### Document object : Each HTML document that gets loaded into a window becomes a document object. The document contains the contents of the page. Using document object, JavaScript can modify, add and delete the HTML elements, attributes CSS styles in the page

### The *window object* represents a window/tab containing a DOM document where as *document object* is property of *window object* that points to the DOM document loaded in that window.

### You can access a document object either using *window.document* property or using document object directly as window is global object. In the below example, title is the property of document object.

### 

### The other major difference is that both window object and document object have properties and methods. Few method names are same in both objects but with different behavior. In the below example *window.open()* opens a new tab or window and *document.open()* creates a blank document within the window.

### 

## Screen

### Screen is the window property that holds information of browser screen. It refers to screen object associated with that window object. Used to display screen width, height, colorDepth, pixelDepth etc

### Similar to document screen can be accessed either by *window.screen* or *screen* object directly. Screen object doesn't have any methods as in window and document objects.

### 

### 