DIFFERENCES BETWEEN DOCUMENT AND WINDOW OBJECTS

INTRODUCTION:

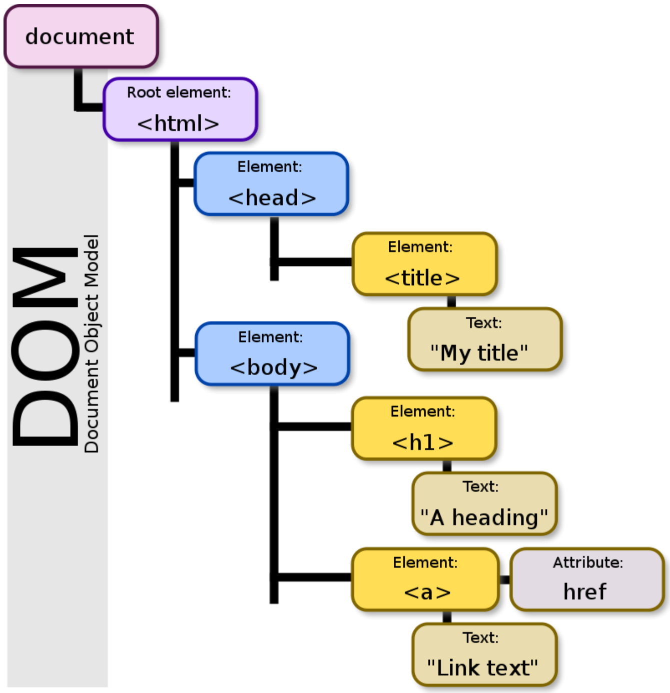
In the realm of web development, two fundamental objects play pivotal roles in shaping the user's experience: the document and window objects. These objects are central to understanding how web page function and interact with the user's browser. While both objects are part of the JavaScript environment, they serve distinct purposes and occupy different positions in the web development landscape.

DOCUMENT OBJECT:

The document object represents a web page that is loaded in the browser. By accessing the document object, we can access the element in the HTML page. With the help of document objects, we can add dynamic content to our web page. The document object can be accessed with a **window. Document** or just **document.**

SYNTAX:

document.property_name;

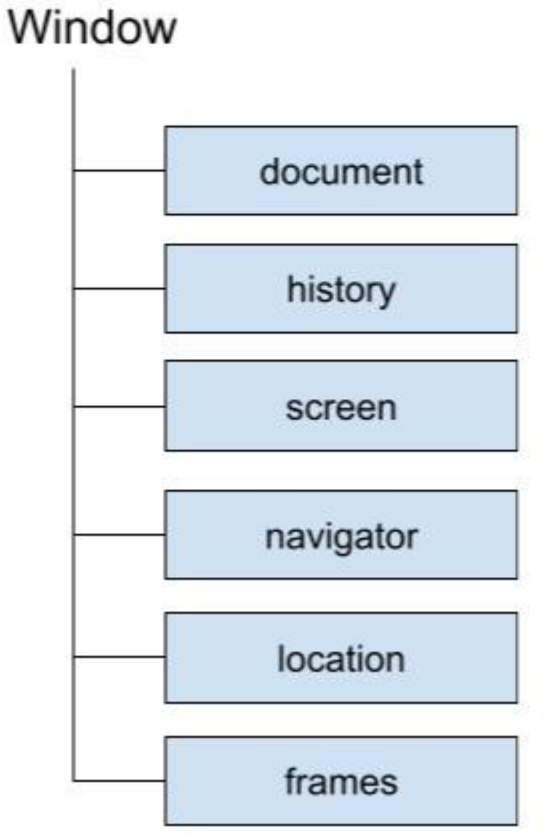


WINDOW OBJECT:

The window object is the topmost object of the DOM hierarchy. It represents a browser window or frame that displays the contents of the webpage. Whenever a window appears on the screen to display the contents of the document, the window object is created.

SYNTAX:

window.property_name;

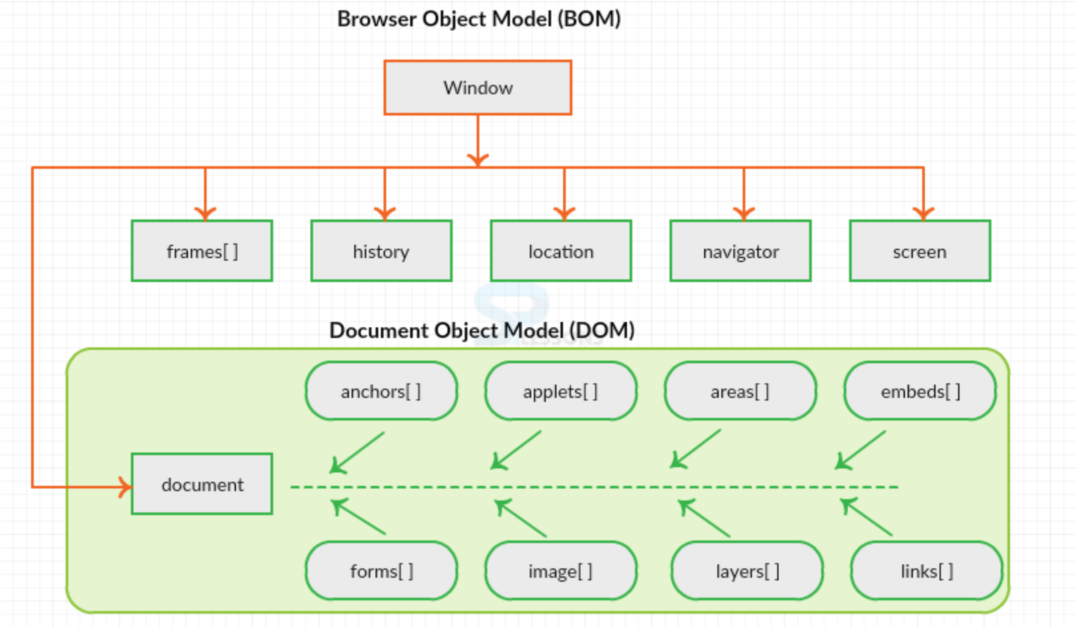


DIFFERENCE BETWEEN DOCUMENT AND WINDOW:

S.NO	DOCUMENT	WINDOW
1	It represents any HTML document or web page that is loaded in the browser.	It represents a browser window or frame that displays the contents of the webpage.
2	It is loaded inside the window.	It is the very first object that is loaded in the browser.

3	It is the object of window property.	It is the object of the browser.
4	All the tags, elements with attributes in HTML are part of the document.	Global objects, functions, and variables of JavaScript are members of the window object.
5	We can access the document from a window using the window. document	We can access the window from the window only. i.e. window.window
6	The document is part of BOM (browser object model) and dom (Document object model)	The window is part of BOM, not DOM.
7	Properties of document objects such as title, body, cookies, etc can also be accessed by a window like this window. document.title	Properties of the window object cannot be accessed by the document object.
8	syntax: document.propertyname;	syntax: window.propertyname;
9	example:	example:

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



CONCLUSION:

In conclusion, the document and window objects are foundational concepts in web development, each with a unique set of responsibilities and capabilities. The document object serves as the gateway to a web page's content, allowing developers to manipulate its structure, access elements, and handle events. On the other hand, the window object manages the broader browser environment, controlling windows and tabs, handling navigation, and providing global functionalities.

By recognizing the distinctions between these objects, developers can harness their full potential to create seamless and interactive web experiences. Whether you're crafting a user interface, managing browser windows, or implementing complex functionalities, knowing when to use the document and window objects is essential for building robust and user-friendly web applications.