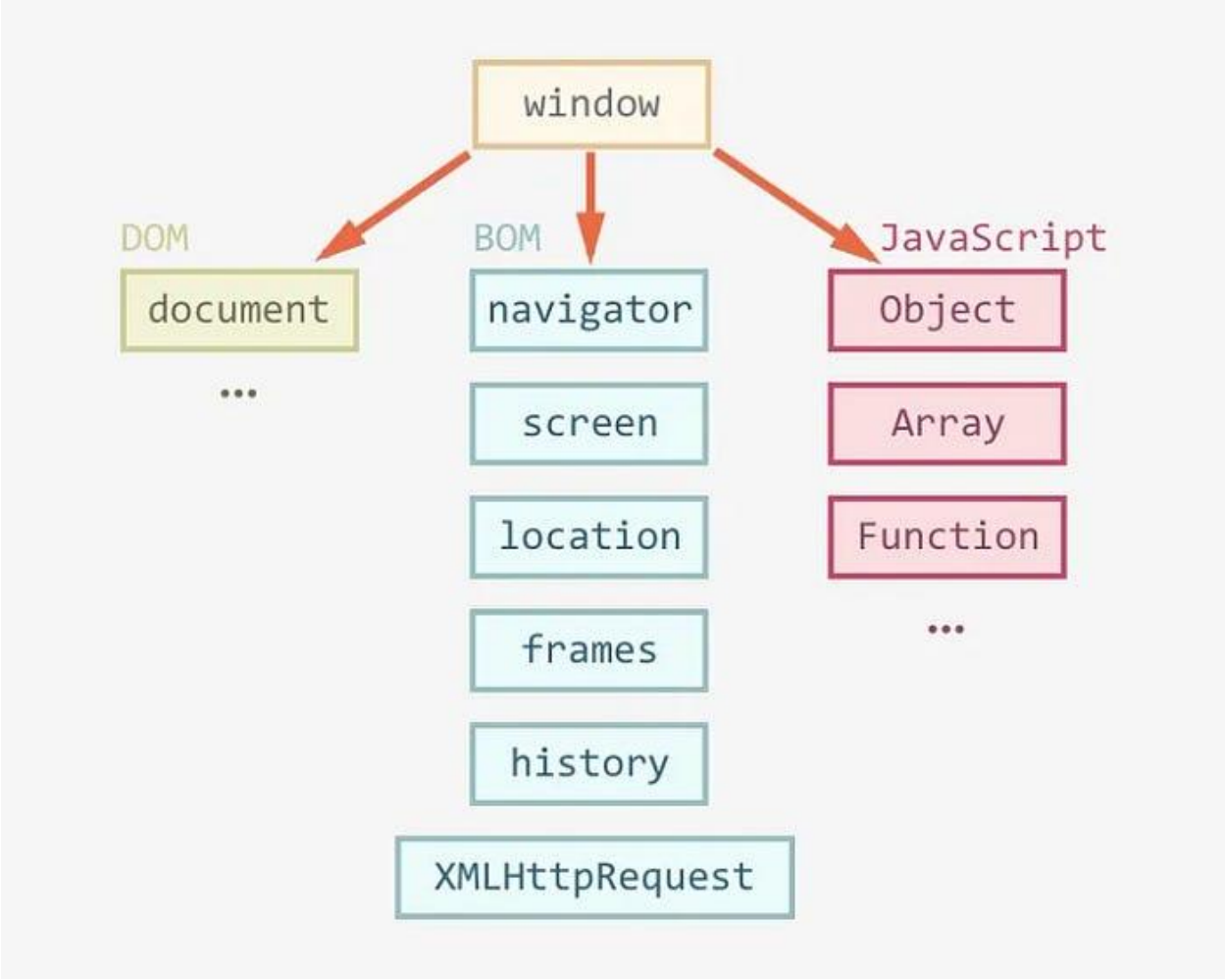
This blog explores the differences between Document and Window object in JavaScript, focusing on their roles, scopes, and functionalities, as they are essential components of the Document Object Model (DOM) for efficient web development.

Window Object

The Window object is a global object in client-side JavaScript, representing the browser window containing a DOM document and acting as the root of the document object model.

The window object, supported by all browsers, represents the browser's window and automatically includes global JavaScript objects, functions, and variables as members.

The Window object is responsible for managing global variables, functions, and objects, providing methods for browser interaction and managing properties related to frames, tabs, or windows, such as alert(), confirm(), setTimeout(), and setInterval().



Window Object

Few properties and methods of window objects are,

Properties

`window.innerHeight` - *the inner height of the browser window (in pixels)*

`window.innerWidth` - *the inner width of the browser window (in pixels)*

Methods

`window.open()` - *open a new window*

`window.close()` - *close the current window*

`window.moveTo()` - *move the current window*

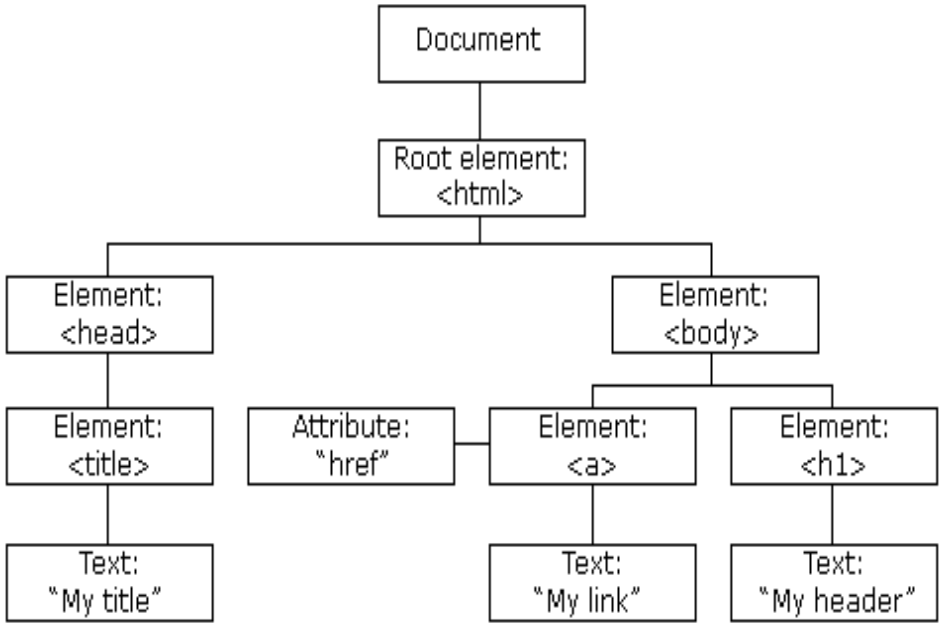
`window.resizeTo()` - *resize the current window*

Document Object

The Document object is the HTML document that appears in the browser window and serves as an interface for interacting with the web page's content. The browser generates a Document Object Model of a web page upon loading it.

The W3C Document Object Model (DOM) is a platform-neutral interface that enables dynamic access and updating of a document's content, structure, and style by programs and scripts.

The DOM is a logical tree in a document, with methods allowing programmatic access to change its structure, style, or content.



Document Object Model

The DOM manipulation tool offers methods to access and manipulate the structure and content of a document, such as getElementById(), querySelector(), createElement(), and innerHTML. It represents the entire HTML document as a node tree, with the Document object as the root node. It allows dynamic updates and interactions with web page content.

Few properties and methods of Document Object Model,

Properties

`element.innerHTML = new html content`*//Change the inner HTML of an element*

`element.attribute = new value`*//Change the attribute value of an HTML element*

Methods

`document.createElement(element`*//Create an HTML element*

`document.removeChild(element)`*//Remove an HTML element*

These are the few methods and properties of DOM, to know more you can refer to **MDN** docs here.

Conclusion

Understanding the differences between Window and Document objects is crucial in JavaScript and web development. Window manages browser interactions, while Document acts as an interface for content manipulation. Using these functionalities allows developers to create interactive web experiences.