Blog on difference between **Document** and **Window** Object

* Scope and Hierarchy:
* The Document Object represents the HTML document loaded in the browser window. It is part of the DOM (Document Object Model) and provides access to the structure and content of the document, including elements like `<html>`, `<head>`, `<body>`, and all other HTML elements.
* The Window Object represents the browser window or tab. It is the top-level object in the browser's JavaScript object hierarchy and serves as the global object for all JavaScript code running in that window.
* Access to DOM Elements:
* The Document Object provides methods and properties to interact with the DOM elements within the document. Methods like `getElementById`, `querySelector`, `querySelectorAll`, can be used to access and manipulate specific elements, modify their attributes, content, or structure.
* While the Window Object does not directly provide methods for accessing individual DOM elements like the Document Object does, it does provide access to the global `document` object. Document object can be accessed through `window.document` or `document`.
* Lifecycle and Events:
* The Document Object is closely tied to the lifecycle of the HTML document. It provides events such as `DOMContentLoaded` (fires when the initial HTML document has been completely loaded and parsed) and `load` (fires when all resources of the page including images and scripts have finished loading).
* The Window Object manages the lifecycle of the browser window or tab. It provides events such as `load` (similar to the document's `load` event but at the window level), `resize` (fires when the window is resized), `scroll` (fires when the window is scrolled), and many others related to user interactions and window state changes.
* Global Context:
  + The Document Object is specific to the content and structure of the HTML document. It is primarily concerned with representing and manipulating the document's elements, attributes, and content.
  + The Window Object provides a global context for JavaScript code running within that window. It includes not only the document-related functionality but also browser-related features such as timing events (`setTimeout`, `setInterval`), local storage (`localStorage`, `sessionStorage`), history navigation (`history` object), and more.
* Parent-Child Relationship:
  + Document Object: In terms of hierarchy, the Document Object is a child of the Window Object. Each window/tab in a browser has its own Document Object representing the loaded HTML document.
  + The Window Object serves as the parent object for the Document Object within a browser window or tab. This relationship reflects the structure of the DOM and the browser environment.