***JavaScript - Day -2: Datatypes  
21/09/2023 - Thursday - 9:00 AM : 11:30 AM***

***DAY 2 ACTIVITY:***

***1. Write a blog on the difference between document and window object:***

|  |  |
| --- | --- |
| **DOCUMENT OBJECT** | **WINDOW OBJECT** |
| 1. It is a top-level object within the DOM hierarchy and provides access to the content and structure of the HTML document. | 1. It is the global object in the browser's JavaScript environment and serves as the entry point for interacting with the browser and controlling the window or tab. |
| 2. The Document object is primarily responsible for managing and providing access to the document's content, structure, and elements. You can use it to manipulate the HTML content, access and modify elements, and handle events within the document. | 2. The Window object is responsible for managing the browser window or tab. It provides methods and properties for controlling browser behavior, such as navigating to URLs, opening new windows or tabs, setting timeouts, handling user interactions, and managing the global JavaScript environment. |
| 3. It's a child of the Window object. In other words, the Window object contains the Document object as one of its properties | 3. It's the top-level object in the browser's JavaScript environment. |
| 4. Primarily used for interacting with the content and structure of the HTML document. You can access and manipulate elements, modify content, and handle events within the document. | 4. Used for controlling the browser itself. It manages browser behavior, navigation, opening new windows or tabs, timing events, and the global JavaScript environment. |
| 5. Provides methods like getElementById, querySelector, and more to access and manipulate HTML elements within the document. | 5. Provides methods like getElementById, querySelector, and more to access and manipulate HTML elements within the document. |
| 6. Handles events related to elements within the document, such as click events on buttons or form submissions. | 6. Handles broader events related to the browser window or tab, like resize events, focus/blur events, and page load events. |
| 7. Allows you to modify the content, structure, and styling of elements within the document. | 7. Doesn't directly deal with content manipulation but can be used to trigger actions that affect the document, such as opening a new window or tab. |
| 8. Provides information about the current document, such as its URL and referrer. | 8. Manages navigation, loading new pages, and controlling the history stack. |
| 9. Doesn't exist in the global scope. You typically access it via window.document or simply document within JavaScript. | 9. Is the global scope in a browser's JavaScript environment. |
| 10. Can be used to create timers, such as setTimeout and setInterval, which operate within the context of the document. | 10. Manages timers and timeouts at the window level. |
| 11. Doesn't directly manage the browsing history but can be used to navigate within the document using methods like history.back(). | 11. Manages the browsing history through methods like history.back(), history.forward(), and history.pushState(). |

**####### THANK YOU #######**