**Write a blog on the Differences between document and windows object**

**Introduction**

* In the world of web development, understanding the Document and Window objects is crucial as they play a fundamental role in manipulating and interacting with web pages.
* Both of these objects are part of the Document Object Model (DOM), which represents the structure and content of a web page. However, they serve different purposes and have distinct properties and methods.
* In this blog, we will delve into the differences between the Document and Window objects to help you grasp their individual roles and functionalities.

| * **Document** | * **Window Object** |
| --- | --- |
| * Main Scope of the document is limited to the content of a single web page. | * Window Object encompasses the entire browser window or tab, including multiple frames if present. |
| * Focused on manipulating and accessing the content and structure of a web page. | * Concerned with controlling the browser's behavior, managing multiple windows or frames, and interacting with browser-related features. |
| * Represents the root node of the DOM tree. | * Serves as a global object within which the Document object resides. |
| * Provides methods and properties related to document content manipulation (e.g., getElementById(), innerHTML). | * Offers methods and properties for browser control (e.g., open(), setTimeout()) and accessing browser-related information (e.g., location, navigator). |
| * Provides access to the content and structure of a single web page. | * Allows interaction with content from multiple windows or frames within the same browser instance. |
| * Used for handling events related to elements on the web page. | * Handles broader events such as window resizing, closing, and navigation. |
| * Used to create, modify, or remove HTML elements within the current web page. | * Used for tasks like opening new browser windows or tabs. |
| * Typically used for navigating within the document's history (e.g., back and forward buttons). | * Manages the browser's entire navigation history. |
| * Utilized for handling timing events specific to elements within the document. | * Used for broader timing events and delays, such as setTimeout() and setInterval(). |