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

**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.**
* It represents any HTML document or web page that is loaded in the browser.
* It is loaded inside the window.
* It is the object of window property.
* All the tags, elements with attributes in HTML are part of the document.
* We can access the document from a window using the window. Document
* The document is part of BOM (browser object model) and dom (Document object model)
* Properties of document objects such as title, body, cookies, etc can also be accessed by a window like this window. document.title

**Syntax:**

document.property\_name;

document.method\_name;

**Properties of document:**

* **activeElement:** It returns the currently active elements in the document.
* **body:**  It returns the contents of the body element.
* **baseURI:** It returns a string value that represents the base URI of the document.
* **cookie:**  It returns the cookie of the current document.
* **charSet:**  It returns a string, representing the document’s character encoding.
* **defaultView:**  It returns the current Window Object.
* **designMode:**  It is used to set documents as editable or read-only.

**Methods of Document:**

* **addEventListener():**It is used to attach an event handler to the specified element.
* **adoptNode():**It is used to adopt a node from another document and it returns a node object, representing the adopted node.
* **close():**It is used to close the output stream.
* **createAttribute():** It is used to create an attribute node with the specified name and returns the attribute object.
* **createComment():** It is used to create a comment node with some text.
* **createDocumentFragment():** It is used to create the document fragment to change the content of the document.

**Key Characteristics of the Document Object:**

* DOM Structure: The document object provides a hierarchical representation of the HTML elements on a webpage, allowing developers to navigate, modify, and manipulate the content dynamically.
* Methods: The document object offers methods like getElementById(), getElementsByClassName(), and querySelector(), enabling developers to select specific elements based on their IDs, classes, or other attributes.
* Properties: Properties such as document.title, document.URL, and document.body provide information about the document's title, URL, and body, respectively.
* Content Manipulation: Through the document object, you can dynamically create, modify, or delete elements, attributes, and text content on a webpage.

**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.

## It represents a browser window or frame that displays the contents of the webpage.

## It is the very first object that is loaded in the browser.

## It is the object of the browser.

## Global objects, functions, and variables of JavaScript are members of the window object.

## We can access the window from the window only. i.e. window.window

## The window is part of BOM, not DOM.

## Properties of the window object cannot be accessed by the document object.

## Syntax:

window.property\_name;

window.method\_name;

**Properties of the window:**

* **Closed:** It holds a Boolean value that represents whether the window is closed or not.
* **Console:** It returns a reference to the console object which provides access to the browser’s debugging console.
* **defaultStatus:** It is used to define the default message that will be displayed in the status bar when no activity is carried on by the browser.
* **controllers:** It returns the XUL controller objects for the current Chrome window.
* **customElements:** It returns a reference to the CustomElementRegistry object, which can be used to register new custom elements and also get information about already registered custom elements.
* **crypto:**  It returns the browser crypto object.
* **devicePixelRatio:**  It returns the ratio between physical pixels and device-independent pixels in the current display.

**Methods of Window:**

* **alert():**It is used to display an alert box. It displays a specified message along with an OK button and is generally used to make sure that the information comes through the user.
* **atob():**It is used for decoding a base-64 encoded string. It is used to decode a string of data that has been encoded using the btoa() method.
* **blur():** It is used to remove focus from the current window.
* **btoa():** It is used for encoding a string in base-64 format.
* **clearInterval():** It clears the interval which has been set by the setInterval() function before that.
* **clearTimeout** It clears the timeout which has been set by the setTimeout()function before that.

# **Key Characteristics of the Window Object:**

* Global Scope: The window object serves as the global scope in JavaScript, meaning variables and functions declared without the var, let, or const keyword become properties of the window object.
* Browser Information: The window object offers properties like window.innerWidth, window.innerHeight, and window.navigator to provide information about the browser's dimensions, navigator object, and other related details.
* Navigation: Methods such as window.open(), window.close(), and window.location allow developers to control browser navigation, open new windows or tabs, and manipulate the current URL.
* Timers: The window object provides functions like setTimeout(), setInterval(), and clearTimeout() to manage time-based operations and execute code asynchronously.