1. Differences between Document and Window Objects in JavaScript

Document Objects:

* 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.

Syntax:

*document.property\_name;*

Properties of document:

* charSet: It returns a string, representing the document’s character encoding.
* defaultView: It returns the current Window Object.
* domain: It returns the domain name of the document server.
* doctype: It returns the document’s doctype.
* embeds: It returns the collection of all embedded elements.
* URL: It returns the complete URL of the document.
* forms: It returns all the elements of the form.
* fullScreenElement: It returns the element that is currently present in full-screen mode.
* title: It returns the title element of the document.
* head: It returns the head element of the document.
* links: It returns all <area> and <a> elements that have a href attribute.

Methods of Document:

Syntax:

*document.method\_name;*

The lists of most commonly used methods are listed below:

* 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.
* createElement(): It is used to create HTML element .
* createEvent(): It is used to create a new events object.
* getElementById(): It returns the object of the given ID. If no object with that id exists then it returns null.
* getElementsByClassName(): It returns an object containing all the elements with the specified class names in the document as objects.
* getElementsByName(): It returns an object containing all the elements with the specified name in the document as objects.

**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.

Syntax:

*window.property\_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.
* Document: It returns a reference to the document object of that window.
* DOMMatrix: It returns a reference to a DOMMatrix object, which represents 4×4 matrices, suitable for 2D and 3D operations.
* frames[]: It represents an array that contains all the frames of a given window.
* DOMPoint: It returns a reference to a DOMPoint object, which represents a 2D or 3D point in a coordinate system.
* History: It provides information on the URLs visited in the current window.
* Length: It represents the number of frames in the current window.
* DOMRect: It returns a reference to a DOMRect object, which represents a rectangle.
* fullScreen: This property indicates whether the window is displayed on full screen or not.
* Location: It contains the URL of the current window.
* innerHeight: It is used to get the height of the content area of the browser window.
* innerWidth: It is used to get the width of the content area of the browser window.
* Name: It contains the name of the referenced window.
* Window: It returns the current window or frame.
* Navigator: It returns a reference to the navigator object.
* outerHeight: It will get the height of the outside of the browser window.

**Methods of Window:**

**Syntax:**

**window.method\_name;**

**The methods of Window objects that are commonly used are listed in the below table:**

* 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.
* 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.
* close(): It is used for closing a certain window or tab of the browser which was previously opened.
* confirm(): It is used to display a modal dialog with an optional message and two buttons i.e. OK and Cancel. It returns true if the user clicks “OK”, and false otherwise.
* focus(): It is used to give focus to an element in the current window.
* getComputedStyle(): It is used to get all the computed CSS properties and values of the specified element.
* getSelection(): It returns a Selection object representing the range of text selected by the user
* matchMedia(): It is used to return a MediaQueryList object which represents the result of the specified CSS media query string.
* open(): It is used to open a new tab or window with the specified URL and name.

| **document** | **window** |
| --- | --- |
| It represents any HTML document or web page that is loaded in the browser. | It represents a browser window or frame that displays the contents of the webpage. |
| It is loaded inside the window. | It is the very first object that is loaded in the browser. |
| It is the object of window property. | It is the object of the browser. |
| All the tags, elements with attributes in HTML are part of the document. | Global objects, functions, and variables of JavaScript are members of the window object. |
| We can access the document from a window using the window. document | We can access the window from the window only. i.e. window.window |
| The document is part of BOM (browser object model) and dom (Document object model) | The window is part of BOM, not DOM. |
| Properties of document objects such as title, body, cookies, etc can also be accessed by a window like this window. document.title | Properties of the window object cannot be accessed by the document object. |
| syntax:        document.propertyname; | syntax:  window.propertyname; |