DIFFERENCE BETWEEN WINDOW AND DOCUMENT OBJECT

WINDOW

The window object represents the browser window.

The document object represents the HTML document loaded in that window

Well the window is the first thing that get loaded into the browser.

This window object has the majority of the properties like

Length

Inner width

Inner height

for example we are writing simple programme in window page

<!DOCTYPE html>

<html>

<body>

<h1>Window Document Object</h1>

<h2>The URL Property</h2>

<p id="demo"></p>

<script>

let url = window.document.URL;

document.getElementById("demo").innerHTML = url;

</script>

</body>

</html>

* \*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.
* 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.
* outerWidth: It will get the width of the outside of the browser window.
* Status: It overrides the default status and places a message in the status bar.
* Top: It returns a reference to the topmost window containing a frame if many windows are opened.
* Toolbar: It will result in the toolbar object, whose visibility can be toggled in the window.
* Opener: It contains a reference to the window that opened the current window.
* \*Parent: It refers to the frameset in which the current frame is contained.
* \*Screen: It refers to the screen object
* \*Self: It provides another way to refer to the current window.
* Methods of Window:

Syntax:

window.method\_name;

DOCUMENT

The window object represents a window/tab containing a DOM document

where as document object is property of window object that points to the

DOM document loaded in that window.

can access a document object either using <window.document>

<script>

<document.h1>

<!DOCTYPE html>

<html>

<body>

<h1>Window Document Object</h1>

<h2>The URL Property</h2>

<p id="demo"></p>

<script>

let url = document.URL;

document.getElementById("demo").innerHTML = url;

</script>

</body>

</html>

DOM TREE

* <window.document.getelementby<h1>
* <h1>details are acesssed
* in document there are some properites
* Properties of document:
* activeElement: It returns the currently active elements in the document.
* body: It returns the contents of the body element.
* anchors: It returns all <a> elements that have a name attribute.
* 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.
* 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.
* lastModified: It returns the date and time of the current document that was last modified.
* images: It returns the collection of <img> elements in the document.
* \*implementation: It returns the DOMImplementation object associated with the current document.
* \*readyState: It returns the loading status of the current document.
* \*referrer: It returns the URI of the page that is linked to the current page.
* \*scripts: It returns all script elements present in the document.
* \*strictErrorChecking: It sets or returns whether strict error checking can be enforced on a document or not.