**How are the JavaScript window and JavaScript document different from one another?**

**What is a** **JavaScript window?**

The window is at a root/top level at the JavaScript object hierarchy. It is a global/root object in JavaScript and it is the root object of the Document object model(DOM);

**What is a JavaScript document?**

A document is an object inside the window object and we use a document object for manipulation inside the document.

The first thing that gets loaded into the browser is the window and the properties related to that window are stored in the window object. Properties related to window objects are length, innerWidth, innerHeight, caches, etc.

There was a document object too so what about it then?

So after the window gets loaded then there’s a document (HTML, PHP, or another document) loaded inside that window, and the properties related to that document is stored in the document object. Properties related to document objects are title, URL, cookie, etc.

**Syntax:**

* window object:

window.propertyname;

* document object:

document.propertyname

// OR

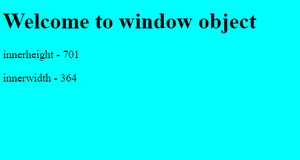
window.document.propertyname

**Example 1:**Focus on the window object.

* Javascript

|  |
| --- |
| window.document.bgColor = "aqua";          const height = document.getElementById('innerheight');          const width = document.getElementById('innerwidth');          window.document.querySelector('h1').textContent =              "Welcome to window object";          height.textContent = "innerheight - " +              window.innerHeight;          width.textContent = "innerwidth - " +              window.innerWidth; |

**Output:**



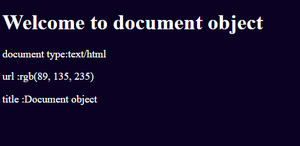
*focused on the window object*

**Example 2:** Focus on the document object.

* HTML

|  |
| --- |
| <!DOCTYPE html>  <html lang="en">    <head>      <meta charset="UTF-8">      <meta http-equiv="X-UA-Compatible" content="IE=edge">      <meta name="viewport"            content="width=device-width, initial-scale=1.0">      <title>Document object</title>  </head>    <body style="color:white">      <h1></h1>      <p class="details"></p>      <p class="details"></p>      <p class="details"></p>      <script>          document.bgColor = "rgb(89, 135, 235)";          const text = prompt('enter the text to appear on screen');          const h1 = document.querySelector('h1');          h1.textContent = text;          const p = document.getElementsByClassName('details');          p[0].textContent = "document type:" +                document.contentType;          p[1].textContent = "url :" + document.bgColor;          p[2].textContent = "title :" + document.title;      </script>  </body>    </html> |

**Output:**



*focused on the document object*

|  |  |
| --- | --- |
| **Document** | **Window** |
| It represents the document loaded inside the window or browser. | It represents the browser window in which you are seeing the content. |
| The properties related to it are stored in the document object. | The properties related to it are stored in the window object. |
| It is loaded after the loading window because the window contains a document. | It is loaded before the document because window container document. |
| It is the root element of the document object model. | The window is the global element for all objects, functions, etc. |
| It is an object of window. | It is an object of the browser. |
| We can not access windows objects properties inside the document. | We can access document object properties inside the window. |
| logically:  document:{ properties} | logically:      window:{         document:{properties}     } |
| Example: document.title will return the title of the document | Example: window.document.title will return the title of the document. |

# Screen

Screen is a small information object about physical **screen dimensions** . It can be used to display screen width, height, colorDepth, pixelDepth etc. It is not mandatory to write **window prefix** with screen object. It can be written without window prefix.

Properties:

screen.width  
screen.height  
screen.availWidth  
screen.availHeight  
screen.colorDepth  
screen.pixelDepth