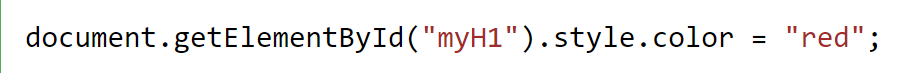
DOM Style

Descripcion: Sirve para poder interactuar con css desde javascript

Ejemplo:



Funciones mas importantes:

**Style.background**



Devuelve 8 propiedades diferentes del fondo: background-color

* background-image
* background-repeat
* background-attachment
* background-position
* background-size
* background-origin
* background-clip

**style.border**



Devuelve 3 propiedades diferentes del los bordes: border-width

* border-style
* border-color

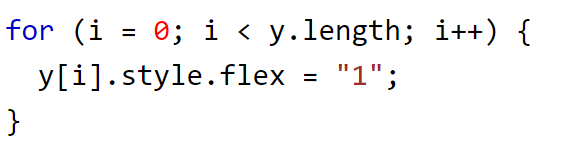
**style.font**



Devuelve 8 propiedades diferentes del tipo de letra: font-style

* font-variant
* font-weight
* font-size
* line-height
* font-family

**style.flex**



La propiedad flex establece o devuelve la longitud del elemento, en relación con el resto de los elementos flexibles dentro del mismo contenedor, tiene 3 proiedades diferentes:

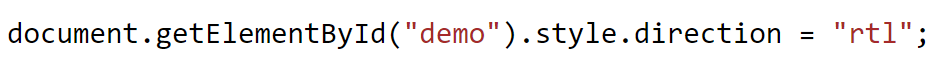
* flexGrow
* flexShrink
* flexBasis

**style.display**



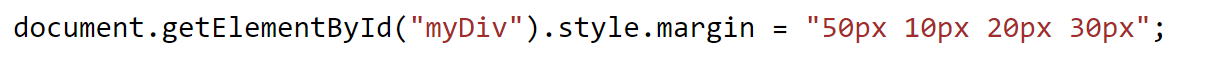
Devuelve el tipo de “display” (si aparece el elemento o no)

**style.direction**



Devuelve la dirección de la letra

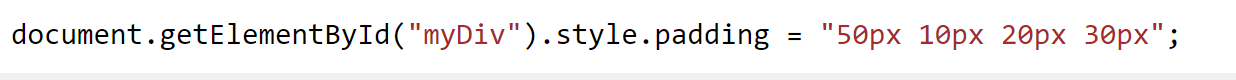
**style.margin**



Devuelve los márgenes de un elemento , tiene 4 propiedades:

* marginBottom
* marginLeft
* marginRight
* marginTop

**style.padding**



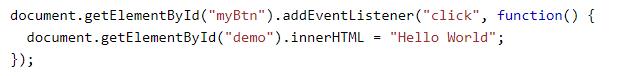
Devuelve el padding de un elemento, tiene 4 propiedades:

* paddingTop
* paddingBottom
* paddingRight
* paddingLeft

DOM ELEMENT

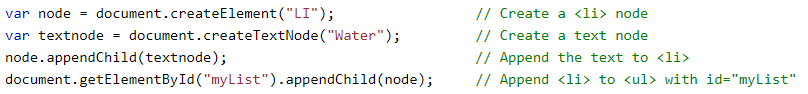
Descripcion: el DOM Element representa un elemento HTML, como P, DIV, A, TABLE o cualquier otro elemento HTML.

**.addEventListener()**



Añade un evento al elemento especificado

**.appendChild()**



Añade el nodo como el ultimo nodo hijo

**.lastChild()**



Devuelve la ultimo nodohijo del nodo especificado

**.id**



Define o devuelve la id del elemento indicado

**.innerHTML**



Define el contenido html de un elemento html (div, p, h1,etc.)

**.innerText**



Define el contenido(“texto en concreto”) del nodo escogido

**.remove()**



Elimina el elemento especifiado

DOM DOCUMENT

Descripcion: Representa su página web. Si quieres acceder a cualquier elemento de una página HTML, siempre comienzas accediendo al objeto del documento. Existen diferentes “funcionalidades” que hacen que algunos document sean específicamente para una cosa u otra.

**Encontrar elementos del HTML**

**document.getElementByid()**



Encuentra el elemento por el id

**document.getElementsByTagName()**



Encuentra elementos por el tag

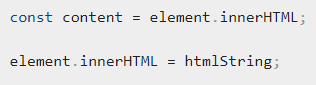
**document.getElementsByClassName()**



Encuentra los elementos por el nombre de la clase

**Cambiar elementos del HTML**

**element.innerHTML**



Cambia el el contenido del html de un elemento

**element.attribute**



Devuelve el valor del atributo de un elemento html

**Añadir y eliminar elementos**

**document.createElement()**



Crea un elemento de html

**document.removeChild()**



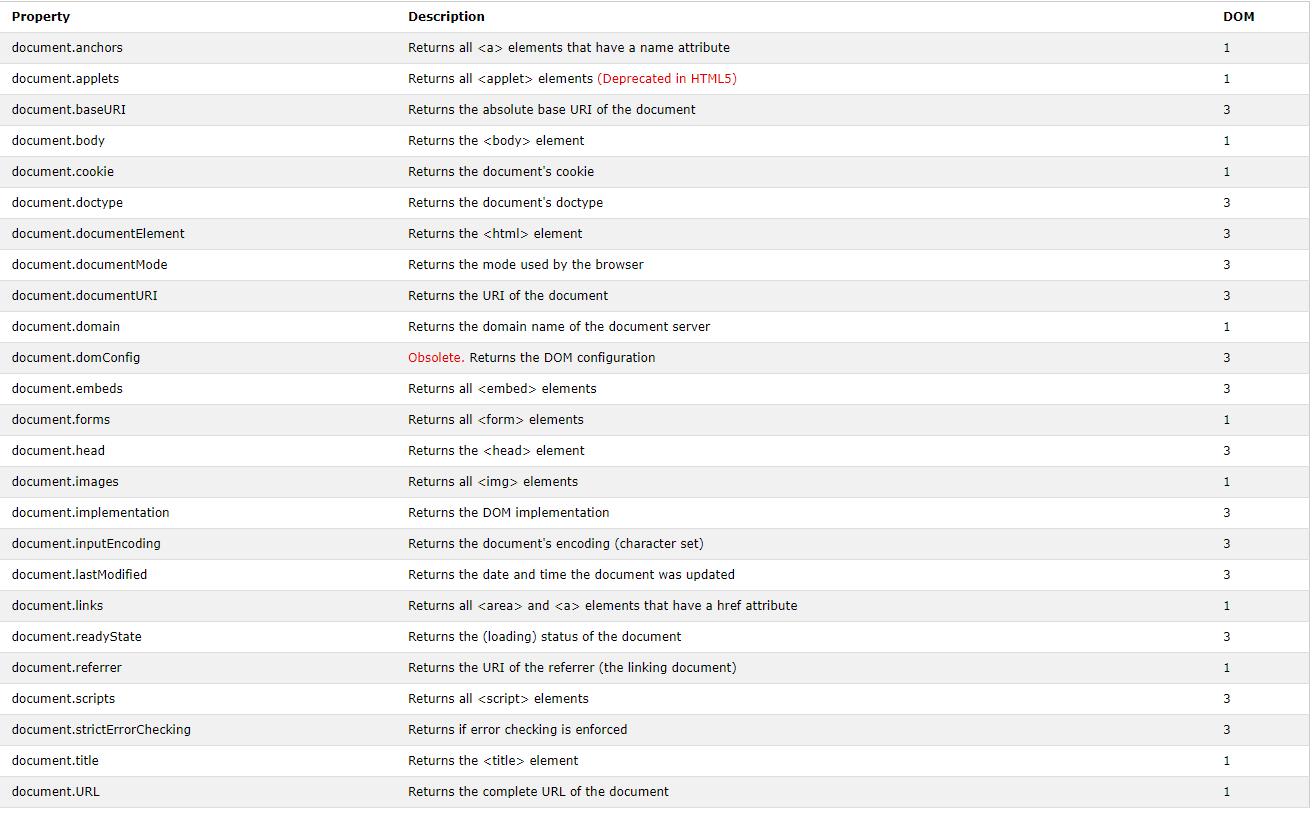
Elimina un elemento html

**document.replaceChild(new,old)**

Reemplaza un elemento html

**Encontrar objetos de html**

“Pondre una captura de imagen porque me parece que todos los igual de importantes ya que hacen la misma funcion”



DOM Events

Descripcion: Permite al JavaScript poder detectar “eventos” de los elementos html

“Lo mismo que antes, ya que no puedo escoger cuales son mas importantes porque todos lo son ya que hacen la misma funcionalidad, hare una captura de todos los eventos”

