

Tema de manejo de DOM DWC

Método `querySelector()`

Este método se encarga de devolver el primer elemento que encuentre y que concuerde con el selector que le entre como parámetro.

Evento DOM

Todos estos eventos pueden ser escuchados por un `addEventListener()`.

Evento	Utilidad
keydown	Pulsar tecla
submit	Enviar formulario
mouseover.	Pasar ratón por encima de un elemento. Para terminar Este evento hay que incluir removeEventListener('mouseleave', 'metodo')
mouseenter	Cuando el ratón pasa sobre un elemento
load.	Cuando se carga una imagen
click	Cuando se hace click sobre un elemento
DOMContentLoaded	Cuando se carga una web

Nodos del DOM

Attr : Este nodo representa los atributos de las etiquetas que tienen formato atributo

Document: Es el nodo del que derivan los demás nodos

Element : Cada etiqueta es un nodo Element. Estos nodos tienen atributos y otros nodos derivan de estos

Text : Estos nodos almacenan texto

Tema de manejo de DOM DWC

Como acceder a los nodos

getElementById Se accede al elemento html por la id que tiene este elemento

getElementsByTagName Se accede a todos los nodos con una etiqueta concreta

getElementByName Se accede mediante el name(ya no se utiliza el name en HTML5)

getElementByClassName Se accede mediante el nombre de la clase

querySelector Accede al primer elemento DOM que concuerde con el selector que entra por parámetros

Como crear un nodo

1. Creamos un nodo tipo element. **document.createElement**("elemento")
2. Crear nodo nuevo tipo text. **document.createTextNode**("texto")
3. Vincular el nodo Text al Element. **nodoElement.appendChild**(nodoText)
4. Vincular nodo Element a la página. **document.body.appendChild**(nodoElement)

Eliminar un nodo

nodopadre.removeChild(hijoAEliminar);

Acceso a los nodos

Conocer el número de enlaces que tiene una web

document.getElementsByTagName("elemento")

Conocer la direccion a la que apunta el último enlace

varConElTotalDeEnlaces[varConElTotalDeEnlaces.length-1].href

Conocer el número de enlaces que están en X párrafo

varConTotalDeParrafos[X].getElementsByTagName('a')

Tema de manejo de DOM DWC

BOM

El browser object model fue implementado por los navegadores para hacer uso de sus métodos y navegadores de forma uniforme,

Diferencias entre DOM y BOM.

- Con el DOM se puede acceder a los elementos de una web mediante su estructura interna
- Con el DOM no se puede acceder a ciertos elementos del navegador.
- En el BOM el objeto raíz es el window

Objeto Window

El objeto window en JavaScript representa una ventana en el navegador. Es el elemento raíz de la estructura de objetos de cada página HTML.

Objeto location

El objeto location en JavaScript es una parte fundamental del Modelo de Objetos del Navegador (BOM). Permite interactuar con la URL actual de la página web que se está visualizando en un navegador.

objeto history

El objeto history en JavaScript es parte del Modelo de Objetos del Navegador (BOM) y nos permite interactuar con el historial de navegación del usuario en el navegador

objeto navigator

El objeto navigator en JavaScript representa el estado y la identidad del navegador o agente de usuario que está ejecutando el script. Es completamente consultable y proporciona información relevante sobre el entorno de navegación