

SEMINARIO DE LENGUAJES. JAVASCRIPT

**ACCEDEMOS A KAHOOT.IT CON ESTE
CÓDIGO 4028675**

Breve Quiz

- El objeto document representa la página Web y todos los objetos están contenidos en él.
- Para poder manipular los elementos primero tenemos que localizarlos.
- Primero accedemos al document
- Es posible localizarlos por id, por clase, por tag, por propiedad del CSS y por colecciones de objetos, por ejemplo adentro de un form.
- Si no encuentra el objeto retorna null.

El objeto document representa la página Web y todos los objetos están contenidos en él. Para acceder a un elemento, primero accedemos al document

Para encontrar un elemento	
<u>document.getElementById(id)</u>	Encuentra un elemento por ID.
<u>document.getElementsByTagName(name)</u>	Encuentra un elemento por <u>name</u> .
<u>document.getElementsByClassName(name)</u>	Encuentra un elemento por la clase

Para <u>egregar</u> o eliminar un elemento	
<u>document.createElement(element)</u>	Crea un elemento HTML
<u>document.appendChild(element)</u>	Elimina un elemento HTML
<u>document.removeChild(element)</u>	Agrega un elemento HTML

getElementById y querySelector
getElementsByTagName

EVENTOS

- Los eventos son notificaciones que algo acaba de ocurrir.
- Estos eventos son generados por el navegador.
- HTML DOM permite identificar cuando el evento ocurre a través de la interfaz Event

NAVEGADOR

- Se carga la página Web
- Se hace click sobre un elemento
- Se modifica un input field.
- Se carga una imagen.
- Se envía un formulario.
- El mouse se mueve sobre un elemento.
- El usuario presiona una tecla.
- Se pone el foco sobre un elemento

Javascript reacciona a los eventos. Cuando ocurre un evento se ejecuta una determinada acción a través de funciones **event handlers**.

this es una variable especial, que se usa para referirse al elemento HTML que generó el evento.

Por ejemplo, se dispara el evento click sobre ese elemento HTML, y al tener un atributo onclick (cuando hagas click), se ejecutará el código que tenemos en el valor de dicho atributo El evento va en minúscula!!!

```
<elemento onclick="funcionAEje
```

Pueden agregarse a los elementos HTML para definir acciones.

Para <u>egregar</u> un manejador de eventos	
<u>document.getElementById(id).onclick = function(){code}</u>	Asocia una función a un evento
Para <u>cambiar</u> un elemento	
<u>element.innerHTML</u> = nuevo contenido	Cambia el contenido de un elemento.
<u>element.attribute</u> = nuevo valor	Cambia el valor de un atributo.
<u>element.style.property</u> = new <u>style</u>	Cambia el estilo de un elemento.
<u>element.setAttribute(attribute, value)</u>	Cambia el valor de un atributo.

Evento onclick Evento onmouseover

Los eventos se pueden gestionar de distinta manera. Primero es necesario elegir el Elemento donde ocurrirá el evento y luego configurarle un manejador. Los eventos son monothread. Se encolan uno después del otro.

Atributo HTML precedido de <u>on</u> que llama a una función.	<code><<u>button</u> <u>onClick</u>="..."></<u>button</u>></code>
Propiedades precedidas de <u>on</u> q incluye a la función	<code>.<u>onclick</u></code>
Escucha de eventos (<u>Listeners</u>)	<code>.<u>addEventListener</u>("click", ...)</code>

- Es la forma más sencilla.
- La desventaja es que se mezcla código HTML con JavaScript.
- Atenta contra la legibilidad y mantenimiento si es código extenso. No tiene reuso.

```
<button onclick="funcionAEjecutar(
```

- Gestiona eventos mediante Propiedades
- Es código Javascript.
- Se hace uso de una propiedad especial precedida con on para asignarle la función.

```
const x = document.querySelector("  
x.onclick = function () {  
    alert("Seminario JS");  
}
```

El método `.addEventListener()` pone en escucha un elemento para cuando se dispare un evento que se recibe como parámetro.

Es el más versátil y es el mecanismo que se debe usar en las páginas web modernas.

```
elemento.addEventListener("tipo-de
```

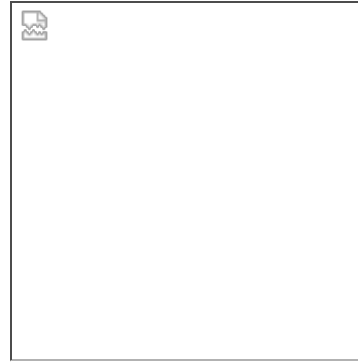
```
<div>
  <p class="nombre">Seminario JS
  <button>Cambia color</button>
</div>
<p> Acá va más código .. </p>
<script>
  const nombre = document.queryS
  const btn = documento.querySel
  btn.addEventListener("click",
    nombre.style.color = "gree
  });
</script>
```

DE LA CLASE ANTERIOR!

Adivina adivinador

TIEMPO

¿Vemos una animación?



> **Clickeame**

- Las funciones son bloques de código reutilizable.
- Existen varias maneras de definir funciones, por ahora vamos a ver un tipo simple.
- Se define con la palabra clave `function`, seguida de un nombre, con paréntesis después de él. Después de eso ponemos dos llaves `{ }`.
- Dentro de las llaves se encuentra todo el código que queremos ejecutar cuando llamamos a la función.
- Cuando queramos ejecutar el código, escribimos el nombre de la función seguido de los paréntesis.

Definición

Funciones en JS

Invocación

Funciones en JS