



# Eventos

```
add.EventListener("evento",function)
```



# Eventos

Captan lo que sucede en el navegador o alguna acción realizada por el usuario. En respuesta, se ejecute una función determinada.

Algunos ejemplos de eventos HTML:

- Una página web HTML finalizó su carga.
- Se hizo clic en un botón HTML.

# Eventos - sintaxis



```
elemento.addEventListener("evento",function) //evento de escucha|
```

Hay 2 maneras en que vamos a ver los eventos creados....



```
//obtenemos el elemento que queremos manipular
var btn = document.querySelector("button");

//le asignamos el evento y la funcion a ejecutar
// nota que es una funcion anonima
btn.addEventListener("click", function(){
    document.querySelector('#hora').textContent= Date();
})
```



```
//obtenemos el elemento que queremos manipular
var btn = document.querySelector("button");

//declaramos la funcion a ejecutar
function Hora(){
    document.querySelector('#hora').textContent= Date();
}

//le asignamos el evento y el nombre de la funcion
btn.addEventListener("click", Hora);
```

# Eventos - Ejemplo para practicar

Pongamos a prueba este ejemplo y miremos lo que pasa:

```
<body>
<button>¿Qué hora es?</button>
<p id="hora"></p>
</body>
```

HTML

```
//obtengo mi elemento
var btn = document.querySelector('button');
//creo mi función
function(){
  document.querySelector('#hora').textContent = Date();
}
//asigno al elemento, el evento y la funcion a ejecutar
btn.addEventListener('click',)
```

JS



# Eventos - tipos de eventos

A continuación conoceremos los más utilizados:

- **click:** Se activa cuando hacemos click en un elemento HTML.
- **dblclick:** Se activa cuando hacemos doble click en un elemento HTML.
- **load:** Se activa cuando la página está cargada completamente.
- **mouseover:** Se activa cuando el ratón "entra" en el elemento.
- **mouseleave:** Se activa cuando el ratón "sale" del elemento.

# Evento a varios elementos - this



En manipulación de DOM, aprendimos que si queremos trabajar varios elementos del DOM, debemos obtenerlos mediante el método `document.querySelectorAll()`. Este método, regresa un array, el cual debo recorrer, para poder manipular cada uno de los elementos contenidos en el mismo.

Ahora, si quisiéramos poner eventos a cada uno de esos elementos. Vamos a necesitar un nuevo concepto que es **this**.

**this** dentro de un evento hace referencia al elemento específico que está disparando ese evento en ese momento.

# Evento a varios elementos - Ejemplo de this

HTML

```
<h3>Asistencia de alumnos</h3>
  <p>Pilar</p>
  <p>Harry</p>
  <p>Bruce</p>
  <p>Nati</p>
```

JS

```
//obtenemos los elementos
var alumnos = document.querySelectorAll("p");
//recorremos el array que devuelve
for(var i = 0; i < alumnos.length; i++){
  //le asignamos el evento y la funcion a ejecutar, a cada
  //uno de los elementos que itera el for
  alumnos[i].addEventListener("click", function(){
    //utilizamos la palabra this para referirnos al
    //elemento que esta recibiendo el evento.
    this.style.color="red"
  })
}
```

Prueba este ejemplo y mira lo que sucede....



# **¡MANOS A LA OBRA!**

Busca el archivo que se encuentra en la carpeta Tareas> Ejercicios clase 7