

CAPTURANDO EVENTOS

ÍNDICE

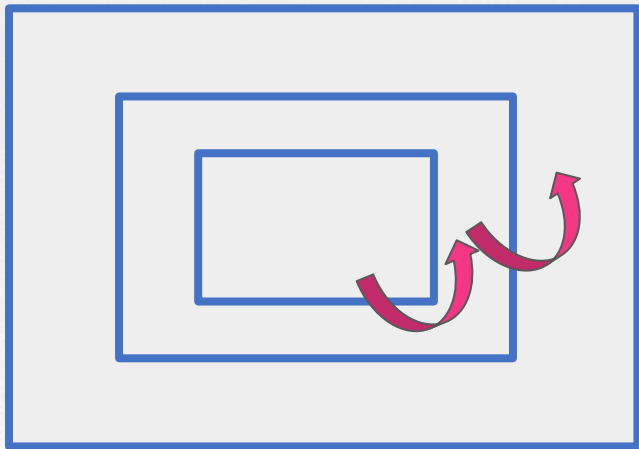
- ¿CAPTURAR Eventos?
- FORMAS de capturar eventos: ¿Capturing vs Bubbling?
- FORMAS para capturar eventos con JavaScript.
- El objeto EVENT.
- Referencias.
- Ejemplo.

¿CAPTURAR EVENTOS?

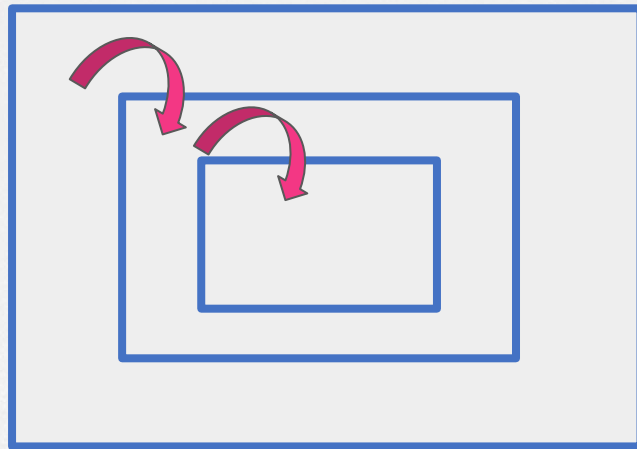
- TRATAR un EVENTO que se ha producido en un ELEMENTO teniendo en cuenta que este evento se PROPAGA.

FORMAS DE CAPTURAR: CAPTURING VS BUBBLING

Bubbling



Capturing



FORMAS DE CAPTURAR: CAPTURING VS BUBBLING

- Por DEFECTO el comportamiento en BUBBLING.
- CAPTURING se usa raramente (no recuerdo haberlo usado nunca).
- NO TODOS los eventos se PROPAGAN (focus).

FORMAS DE CAPTURAR EVENTOS

FORMAS DE CAPTURAR EVENTOS

- Desde el **HTML** (**inline** - no recomendado)
- Desde el Script con **Método Propio**.
- Desde el Script con **addEventListener(...)** / **removeEventListener()**.

INLINE - DESDE HTML

```
<tag eventAttribute = "someJsFunction(event)" ></tag>
```

Example:

```
<div onclick="doSomething(event)" ></div>  
<script>  
    function doSomething(e) {  
        console.log("Has hecho click");  
    }  
</script>
```


CON PROPIEDADES PROPIAS

someJsElement.eventProperty = someJsFunction;

Example:

<div id="example"></div>

<script>

*function doSomething(e) {
 console.log("Has hecho click");
}*

*let element = doc.getElementById("example");
element.onclick = doSomething*

</script>

CON LA FUNCIÓN **addEventListener**

```
someJsElement.addEventListener(event,someFunction);
```

Example:

```
<div id="listener"></div>  
<script>  
    function doSomething(e) {  
        console.log("Has hecho click");  
    }  
    let element = doc.getElementById("listener");  
    element.addEventListener("click",doSomething);  
</script>
```

CON LA FUNCIÓN `removeEventListener()`

```
someJsElement.removeEventListener(event,someFunction);
```

Example:

```
<div id="listener"></div>
```

```
<script>
```

```
//En caso de que quiera deshabilitar un manejador de eventos  
element.removeEventListener("click",doSomething);
```

```
</script>
```

EL OBJETO EVENT

EL OBJETO EVENT

- Desde los manejadores tenemos acceso al **OBJETO** que define el **EVENTO** y que tiene muchas **PROPIEDADES** y **MÉTODOS** interesantes.

EL OBJETO EVENT

- No todos los eventos tienen las mismas propiedades

EL OBJETO EVENT

Propiedad del Objeto	Descripción de la propiedad
<i>.bubbles</i>	Si es un “bubbling” event.
<i>.preventDefault()</i>	Evita el comportamiento por defecto.
<i>.stopPropagation()</i> / <i>.stopImmediatePropagation()</i>	Para la propagación / Para también otros listeners que puedan estar capturando dicho evento
<i>.target</i> / <i>.currentTarget</i>	Elemento que lanzó el evento / cuyos manejadores lanzaron el evento
<i>.pageX</i> / <i>.pageY</i>	Posición X/Y de donde se produjo el evento
...	

REFERENCIAS

[https://developer.mozilla.org/en-US/docs/Learn/JavaScript/Building blocks/Events](https://developer.mozilla.org/en-US/docs/Learn/JavaScript/Building_blocks/Events)

[https://developer.mozilla.org/en-US/docs/Web/Events/Event handlers](https://developer.mozilla.org/en-US/docs/Web/Events/Event_handlers)

<https://developer.mozilla.org/en-US/docs/Web/Events>

<https://developer.mozilla.org/en-US/docs/Web/API/Event>

EJEMPLOS



GRACIAS!
¿PREGUNTAS?

 @pekechis

 /pekechis