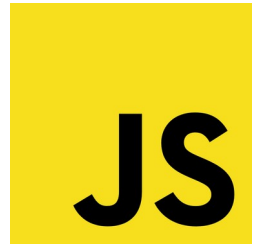


## JavaScript : Events

---

JavaScript és el llenguatge de programació front-end més popular actualment. Els seus scripts s'executen del costat del client, en el navegador web. Aquesta característica permet a JavaScript afegir major interactivitat i efectes dinàmics a les pàgines web mitjançant la manipulació del contingut retornat des d'un servidor web.



### Com afegir JavaScript a la teva pàgina web

Crearem un fitxer .js on hi escriurem totes les instruccions que ens permetran interactuar amb la pàgina web. Llavors el cridem des del fitxer .html escrivint aquesta instrucció al final de l'etiqueta `<body>`:

```
<script src="nom_fitxer.js"></script>
```

### Events

Degut a la interacció de l'usuari amb l'HTML o a l'acció del navegador es produeixen esdeveniments, amb JavaScript podem capturar aquests esdeveniments i generar accions a través dels Event Handlers.

Aquí pots veure varis Events Handlers que permet JS:

<https://www.tutorialrepublic.com/javascript-tutorial/javascript-events.php>

#### HTML

```
<button id="myBtn" type="button">Click Me!</button>
```

#### JavaScript

```
function sayHello() {  
    alert("Hello World!");  
}  
let buttonA = document.getElementById("myBtn");  
buttonA.onclick = sayHello;
```

En aquest cas, en prémer el botó amb id *myBtn* es mostra un missatge.

Per accedir a les dades de l'HTML DOM podem seleccionar per les propietats dels elements HTML utilitzant les següents funcions :

- a) **getElementById()**
- b) **getElementsByClassName()**
- c) **getElementsByName()**
- d) **getElementsByTagName()**

## Event Listeners

És una altra manera de capturar i tractar els esdeveniments. Els Event Listeners permeten més d'una acció per un esdeveniment aplicat a un element concret.

Utilitzant el codi de l'apartat anterior podem realitzar la mateixa acció de la següent manera:

```
buttonA.addEventListener("click", sayHello);
```

O reescrivint la funció:

```
buttonA.addEventListener("click", function() {  
    alert("Hello World!");  
});
```

Si passem un paràmetre a la funció podem interactuar amb l'esdeveniment:

```
buttonA.addEventListener("click", handleChange(event) {  
    alert(event.target.id);  
});
```

En aquest cas, obtenim l'id de l'element que ha provocat l'esdeveniment.

També podem controlar els window event, és a dir esdeveniments relacionats amb la mida de la pantalla. En aquests casos utilitzarem l'objecte *window*:

```
window.addEventListener("resize", displayWindowSize);
```

De la mateixa manera que podem afegir Event Listeners els podem eliminar:

```
buttonA.removeEventListener("click", sayHello);
```

## ACTIVITATS

1. Pren de base el fitxer .html que trobaràs aquí:

<https://github.com/xbaubes/DAM/blob/main/MP4-Llenguatges-marques-SGI/JavaScript/exercici-events-1/index.html>

Avui crearem la pàgina web més absurda de la història.

I en què consisteix?

Al prémer «Button 1» el tercer input ha de prendre de color de fons el color escrit a l'input 1.

Al prémer «Button 2» el tercer input ha de mostrar el resultat de la suma dels valors escrits als input 1 i 2.

Al prémer «Button 3» el tercer input ha de mostrar quin és el valor més alt entre els valors escrits als input 1 i 2.

Al passar per sobre del «Button 4» aquest ha de quedar deshabilitat.

Al prémer «Button 5» has de comparar els valors escrits als input 1 i 2, i escriure al tercer input si són «IGUALS» o «DIFERENTS». Al perdre el focus el «Button 5» el tercer input restaura el seu valor per defecte.

Per culminar amb èxit el desenvolupament de la pàgina web més absurda de la història necessitaràs accedir a la propietat *value* dels inputs.

## BIBLIOGRAFIA I WEBGRAFIA

«Mark Volkmann». <https://mvolkmann.github.io/MidwestJS/ES2015-talk.pdf>

«Tutorial Republic». <https://www.tutorialrepublic.com/javascript-tutorial/>



Autor: Xavier Baubés Parramon

Aquest document es llicència sota Creative Commons versió 4.0.  
Es permet compartir i adaptar el material però reconeixent-ne l'autor original.