

# addEventListener

---

El método addEventListener, es un escuchador que indica al navegador que esté atento a la interacción del usuario.

La ventaja es que se escribe totalmente en el JS, sin necesidad de tocar el HTML.

Su sintaxis es muy sencilla:

```
target.addEventListener('tipo de evento', función_a_lanzar , booleano){  
    código a ejecutar  
};
```

**Target**, es elemento que se escucha generalmente se identifica por su ID.

Los tipos de **evento** a escuchar, son los eventos ya conocidos de javascript

- **blur** Cuando el elemento pierde el foco.
- **click** El usuario hace clic sobre el elemento.
- **dblclick** El usuario hace doble clic sobre el elemento.  
**focus** El elemento gana el foco.
- **keydown** El usuario presiona una tecla.
- **keypress** El usuario presiona una tecla y la mantiene pulsada.
- **keyup** El usuario libera la tecla.
- **load** El documento termina su carga.
- **mousedown** El usuario presiona el botón del ratón en un elemento.
- **mousemove** El usuario mueve el puntero del ratón sobre un elemento.
- **mouseout** El usuario mueve el puntero fuera de un elemento.
- **mouseover** El usuario mantiene el puntero sobre un elemento.
- **mouseup** El usuario libera el botón pulsado del ratón sobre un elemento.
- **unload** El documento se descarga, bien porque se cierra la ventana, bien porque se navega a otra página.

**Función** que se ejecuta cuando se escucha el evento.

El **booleano**, es opcional, es un valor que define el orden del flujo de los eventos, cuando hay varios eventos posibles.

Si tenemos un botón dentro de un div y dentro del body.

**true**: El flujo de eventos es como el representado, y la fase de captura ocurre al lanzarse el evento. El orden de propagación para el ejemplo seguiría el orden del árbol del documento en el DOM y sería, por tanto, el indicado, **body-div-button**

**false**: Permite saltar la fase de captura, y la propagación seguiría sólo la burbuja. Así, el orden sería **button-div-body**.