

GUÍA TEÓRICO-PRÁCTICA:

UNIDAD I

SEMANA 6: EVENTOS AVANZADOS

TEMA 6.1: EVENTOS AVANZADOS - onchange, onkeyup

Teoría

- **onchange:** Se activa cuando cambia el valor de un elemento y pierde el foco.
- **onkeyup:** Se activa cuando se libera una tecla.
- **onkeydown:** Se activa cuando se presiona una tecla.
- **oninput:** Se activa cuando cambia el valor de un input (en tiempo real).

Pregunta de reflexión 21: ¿Cuál es la diferencia entre onchange y oninput?

Ejemplo

```
html
```

```
1 <input type="text" id="buscador" onkeyup="buscar()"
2 |   |   placeholder="Escribe para buscar..."
3 |
4 <input type="select" id="pais" onchange="cambiarPais()">
5
6 <script>
7 function buscar() {
8 |   let texto = document.getElementById("buscador").value;
9 |   console.log("Buscando:", texto);
10 |   // Filtrar resultados en tiempo real
11 | }
12
13 function cambiarPais() {
14 |   let pais = document.getElementById("pais").value;
15 |   console.log("País seleccionado:", pais);
16 | }
17 </script>
```

Ejercicio 6.1

Crea:

1. Un buscador que filtre una lista mientras escribes (onkeyup)

2. Un formulario que valide campos en tiempo real (oninput)
3. Un select que muestre información diferente según la opción (onchange)

```
javascript
```

```
1 // Tu código aquí:
2
3
4
5
```

TEMA 6.2: USO DE addEventListener

Teoría

addEventListener: Forma moderna y recomendada de agregar eventos .

```
javascript
```

```
1 elemento.addEventListener("click", funcion);
2 elemento.addEventListener("click", () => {
3 |   console.log("Click");
4 | });
5
6 // Remover evento
7 elemento.removeEventListener("click", funcion);
8
9 // Múltiples eventos al mismo elemento
10 elemento.addEventListener("click", funcion1);
11 elemento.addEventListener("click", funcion2);
12
13 // Event object
14 elemento.addEventListener("click", (e) => {
15 |   console.log(e.target); // Elemento clickeado
16 |   console.log(e.type); // Tipo de evento
17 |   e.preventDefault(); // Prevenir comportamiento default
18 |   e.stopPropagation(); // Detener propagación
19 | });
```

Pregunta de reflexión 22: ¿Por qué es mejor addEventListener que onclick?

Ejemplo

javascript

```

1 let boton = document.getElementById("miBoton");
2
3 // Agregar evento
4 boton.addEventListener("click", function(e) {
5 |   console.log("Click en:", e.target);
6 | });
7
8 // Evento en múltiples elementos
9 document.querySelectorAll(".item").forEach(item => {
10 |   item.addEventListener("click", manejarClick);
11 | });
12
13 // Delegación de eventos
14 document.getElementById("lista").addEventListener("click", (e) => {
15 |   if (e.target.matches(".eliminar")) {
16 |     e.target.parentElement.remove();
17 |   }
18 | });
19
20 // Eventos de teclado
21 document.addEventListener("keydown", (e) => {
22 |   if (e.key === "Enter") {
23 |     console.log("Presionó Enter");
24 |   }
25 | });
26
27 // Evento de scroll
28 window.addEventListener("scroll", () => {
29 |   if (window.scrollY > 100) {
30 |     console.log("Scroll > 100px");
31 |   }
32 | });

```

Ejercicio 6.2**Crea:**

1. Un botón que cambie de color con `addEventListener`
2. Delegación de eventos para una lista dinámica
3. Atajo de teclado (presionar "S" para guardar)
4. Detectar si el usuario está haciendo scroll hacia arriba o abajo

```
javascript
```

```
1 // Tu código aquí:
2
3
4
5
```

TEMA 6.3: DOCUMENTACIÓN TÉCNICA BREVE

Teoría

Comentarios en JavaScript:

```
javascript
```

```
1 // Comentario de una línea
2
3 /* Comentario
4 | de múltiples
5 | líneas */
6
7 /**
8 | * Documentación JSDoc
9 | * @param {string} nombre - El nombre del usuario
10 | * @returns {string} Saludo personalizado
11 | */
12 function saludar(nombre) {
13 |     return "Hola " + nombre;
14 | }
```

Pregunta de reflexión 23: ¿Por qué es importante documentar el código?

Ejercicio 6.3

Documenta todas las funciones que creaste en ejercicios anteriores usando JSDoc.

```
javascript
```

```
1 /**
2 | * Descripción de La función
3 | * @param {tipo} parametro - Descripción
4 | * @returns {tipo} Descripción del retorno
5 | */
```

TEMA 6.4: CIERRE DE UNIDAD

Repaso general de todos los temas vistos.