

## INDICE

INTRODUCCIÓN .....	1
EJERCICIOS .....	2
EJERCICIO 1 .....	2
Código: .....	2
Explicación:.....	3
Vista:.....	3
Ejercicio 2.....	3
Código: .....	4
Explicación:.....	5
Vista:.....	5
BIBLIOGRAFÍA .....	6

## INTRODUCCIÓN

### Ejercicio 1:

Crea un formulario HTML con dos elementos: uno select y otro datalist con los mismos datos.

A continuación, mediante código javascript, visualiza el contenido de los datos elegidos de cada elemento, comentando las diferencias que encuentres a la hora de referenciarlos (obtener sus valores).

### Ejercicio 2:

Crea un formulario HTML con dos controles o elementos: un select con opciones y el otro un select con agrupaciones de opciones (optgroup) de manera que el primero permita deshabilitar un grupo o varios grupos de opciones del segundo.

## EJERCICIOS

### EJERCICIO 1

#### Código:

```
<!DOCTYPE html>
<html lang="es">

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link rel="stylesheet" href="styles.css">
  <script src="script.js" defer></script>
  <title>Ejercicio 1</title>
</head>

<body>
  <div>
    <h3>Select HTML</h3>
    <label for="selectFrutas">Elige una fruta:</label>
    <select id="selectFrutas" name="frutaSeleccionada">
      <option value="" disabled selected>-- Selecciona una opción --</option>
      <option value="manzana">Manzana</option>
      <option value="platano">Plátano</option>
      <option value="fresa">Fresa</option>
      <option value="uva">Uva</option>
      <option value="kiwi">Kiwi</option>
    </select>
  </div>
  <div>
    <h3>Datalist HTML</h3>
    <label for="inputFrutas">Escribe una fruta (o elige de la sugerencia):</label>
    <input list="listaFrutas" id="inputFrutas" name="frutaSugerida">

    <datalist id="listaFrutas">
      <option value="Manzana">
      <option value="Plátano">
      <option value="Fresa">
      <option value="Uva">
      <option value="Kiwi">
    </datalist>
  </div>
  <div id="ultimoContainer">
    <p id="texto"></p>
  </div>
</body>

</html>
```

```
document.getElementById("selectFrutas").addEventListener("change", verDatos);
document.getElementById("inputFrutas").addEventListener("change", verDatos);
const p = document.getElementById("texto");
const div = document.getElementById("ultimoContainer");
const input = document.getElementById("inputFrutas");

function verDatos(e) {
  p.textContent = e.target.value;
  p.textContent = e.target.value;

  comprobarContainer();
}

function comprobarContainer() {
  if (p.textContent === "") {
    div.style.display = "none";
  } else {
    div.style.display = "block";
  }
}
```

## Explicación:

Construyo una página HTML básica que presenta dos formas de selección de datos: un select tradicional y un input con una datalist.

El select (selectFrutas) permite elegir una fruta de una lista de opciones definidas con `<option>`. La datalist (listaFrutas) ofrece sugerencias a un campo de texto (inputFrutas) mientras el usuario escribe, combinando la libertad del input con la ayuda de la lista de opciones.

En JavaScript, asigno la función `verDatos()` al evento `change` de ambos selectores. La función `verDatos()` captura el valor seleccionado y lo muestra en un párrafo (`p.textContent`). Inmediatamente, llamo a `comprobarContainer()` que verifica si el párrafo tiene contenido; si el párrafo está vacío (por ejemplo, al seleccionar la opción deshabilitada), oculto el contenedor (`div.style.display = "none"`), y si hay valor, lo hago visible (`"block"`).

## Vista:

The image displays three screenshots of a web application interface, illustrating the functionality of a Select HTML form and a Datalist HTML form.

**Top Left Screenshot:** Shows the "Select HTML" form. The label "Elige una fruta:" is followed by a dropdown menu with the placeholder text "-- Selecciona una opción --". Below it, the "Datalist HTML" form is shown with the label "Escribe una fruta (o elige de la sugerencia):" and a text input field.

**Top Right Screenshot:** Shows the "Select HTML" form with the dropdown menu displaying "Plátano". Below it, the "Datalist HTML" form is shown with the label "Escribe una fruta (o elige de la sugerencia):" and a text input field. A suggestion box below the input field displays "platano".

**Bottom Screenshot:** Shows the "Select HTML" form with the dropdown menu displaying "Plátano". Below it, the "Datalist HTML" form is shown with the label "Escribe una fruta (o elige de la sugerencia):" and a dropdown menu displaying "Manzana". A suggestion box below the input field displays "Manzana".

## Ejercicio 2

Código:

```
<html lang="es">
<head>
  <title>Selectores Flexibles</title>
  <link rel="stylesheet" href="styles.css">
  <script src="script.js" defer></script>
</head>

<body>
  <div class="container">
    <label for="selectSimple">Elige una Mascota:</label>
    <select id="selectSimple" name="frutaSeleccionada" size="2">
      <option value="Frutas">Frutas</option>
      <option value="Verduras">Verduras</option>
    </select>

    <label for="selectAgrupado">Elige un Alimento:</label>
    <select id="selectAgrupado" name="frutaSugerida" size="4">
      <optgroup id="frutas" label="Frutas" disabled>
        <option value="manzana">🍏 Manzana</option>
        <option value="uva">🍇 Uva</option>
      </optgroup>
      <optgroup id="verduras" label="Verduras" disabled>
        <option value="brocoli">🥦 Brócoli</option>
        <option value="zanahoria">🥕 Zanahoria</option>
      </optgroup>
    </select>
  </div>
</body>
</html>
```

```
document.getElementById("selectSimple").addEventListener("change", desactivar);
const frutas = document.getElementById("frutas");
const verduras = document.getElementById("verduras");
function desactivar(e) {
  console.log(e.target.value);
  if (e.target.value === "Frutas") {
    frutas.removeAttribute("disabled");
    verduras.setAttribute("disabled", true);
  } else {
    verduras.removeAttribute("disabled");
    frutas.setAttribute("disabled", true);
  }
}
```

### Explicación:

Construyo dos estructuras de selección de datos en HTML. La primera es un select simple (selectSimple) con opciones agrupadas por optgroup (Frutas/Verduras), y asocio la función desactivar() a su evento change.

desactivar() obtiene el valor seleccionado del select simple; si es "Frutas", activo el grupo de frutas (removeAttribute("disabled")) y desactivo el de verduras (setAttribute("disabled", true)) en el select agrupado (selectAgrupado), y viceversa si se selecciona "Verduras".

De esta manera, logro una interdependencia entre selectores, activando o desactivando dinámicamente las opciones del segundo select en función de la elección realizada en el primero.

### Vista:

**Elige una Mascota:**

Frutas  
Verduras

**Elige un Alimento:**

Frutas  
Manzana  
Uva  
Verduras

**Elige una Mascota:**

Frutas  
Verduras

**Elige un Alimento:**

Frutas  
Manzana  
Uva  
Verduras

## BIBLIOGRAFÍA

MDN Web Docs (Mozilla Developer Network) Uso: Consulto la documentación esencial sobre manipulación del DOM, específicamente para verificar y aplicar los métodos para modificar atributos de los elementos HTML (removeAttribute y setAttribute), que me permiten activar o desactivar los grupos de opciones del selector (optgroup).

Enlace:

[https://developer.mozilla.org/es/docs/Web/API/Document\\_Object\\_Model/Introduccion](https://developer.mozilla.org/es/docs/Web/API/Document_Object_Model/Introduccion)

W3Schools Uso: Reviso los tutoriales sobre Eventos del DOM para reforzar la sintaxis del evento change que utilizo para detectar la selección del usuario, y la navegación básica para obtener el valor seleccionado (e.target.value) que uso en la lógica de interdependencia.

Enlace: [https://www.w3schools.com/js/js\\_htmlDOM\\_events.asp](https://www.w3schools.com/js/js_htmlDOM_events.asp)

Referencias de JavaScript y Selectores HTML Uso: Verifico el uso correcto de los selectores HTML anidados (select con optgroup) y la manera de obtener referencias a los elementos mediante su id (getElementById) para controlar su comportamiento dinámicamente con JavaScript.

Enlace: <https://developer.mozilla.org/es/docs/Web/HTML/Element/select>