

DOM y Eventos

Explicación del Ejercicio

¿Qué vamos a hacer?

Vamos a crear eventos y manipular el DOM (Document Object Model) usando JavaScript para realizar diferentes interacciones con elementos HTML.

¿Dónde lo vamos a hacer? Crearemos el archivo `index.html` y los archivos de javascript que necesites, donde implementaremos los ejercicios en un entorno de desarrollo local utiizando **Live Server** para ver los resultados.

¿Cómo lo hacemos?

Mediante el uso de JavaScript, manipulación del DOM y eventos del navegador, escribiremos código que permitirá interactuar con los elementos de la página, como botones, campos de texto, listas, y más.

¿Con qué lo hacemos?

Utilizaremos HTML, JavaScript y un editor de texto como Visual Studio Code.

¿Cómo se evalúa lo que hacemos?

El ejercicio será evaluado en función de la correcta implementación de los eventos y la manipulación del DOM en las diferentes tareas propuestas.

Objetivo del Ejercicio

Objetivos Específicos

- **Manipulación del DOM:** Modificar los elementos del DOM a través de JavaScript, como cambiar el contenido, ocultar o mostrar elementos, y alterar atributos o estilos.
 - **Implementación de eventos:** Asignar y gestionar eventos del navegador para interactuar con los usuarios.
 - **Interacciones dinámicas:** Crear interacciones dinámicas, como cambiar el contenido de un párrafo, contar clics, o generar listas de elementos en respuesta a las acciones del usuario.
-

Criterios de Evaluación

- **Implementación de eventos:** Demostrar comprensión en la asignación y uso de eventos, así como en la gestión de respuestas a la interacción del usuario.
 - **Manipulación efectiva del DOM:** Modificar elementos del DOM de manera clara y eficaz, como cambiar estilos o contenido de texto en respuesta a eventos.
 - **Claridad y estructura del código:** El código debe estar bien organizado, con nombres descriptivos para variables y funciones, lo que facilita su lectura y mantenimiento.
 - **Cumplimiento de los objetivos del ejercicio:** Se evaluará la correcta implementación de las instrucciones especificadas para cada tarea.
-

Lista de Ejercicios

Ejercicios DOM y eventos

1. **Cambiar el texto de un elemento:**
 - Crea un párrafo con un texto inicial. Utiliza JavaScript para seleccionar el párrafo y cambiar su contenido por un nuevo mensaje.
2. **Cambiar el estilo de un elemento:**
 - Crea un párrafo. Utilizando JavaScript, selecciona el párrafo y cambia su color de texto a rojo.
3. **Agregar y eliminar elementos:**
 - Crea una lista vacía en HTML. Con JavaScript, crea un nuevo elemento de lista, añádele texto y agrégalo como hijo de la lista.
4. **Mostrar una lista de nombres desde un array:**
 - Crea un array de 10 nombres y una función que imprima esos nombres en pantalla como una lista HTML usando DOM.
5. **Contar elementos de un array:**
 - Crea un array de números y una función que imprima en pantalla cuántos elementos tiene el array.
6. **Crear un formulario con nodos:**
 - Mediante JavaScript, genera los elementos necesarios para crear un formulario básico (input, label, botón) y agrégalo al DOM.
7. **Crear una tabla a partir de un array de nombres:**
 - Crea una tabla HTML usando JavaScript para mostrar los nombres de un array en filas y columnas.
8. **Mostrar un objeto en una tabla:**

- Usando el siguiente array de objetos, imprime cada elemento en la tabla creada en el ejercicio anterior.

None

javascript

Copiar código

```
let array = [
  {
    "id": 1,
    "name": "Rick Sanchez",
    "status": "Alive",
    "species": "Human",
    "type": "",
    "gender": "Male",
  },
  {
    "id": 2,
    "name": "Morty Smith",
    "status": "Alive",
    "species": "Human",
    "type": "",
    "gender": "Male",
  },
  {
    "id": 3,
    "name": "Summer Smith",
    "status": "Alive",
    "species": "Human",
    "type": "",
    "gender": "Female",
  },
  {
    "id": 4,
    "name": "Beth Smith",
    "status": "Alive",
    "species": "Human",
    "type": "",
    "gender": "Female",
  },
  {
```

```
"id": 5,
"name": "Jerry Smith",
"status": "Alive",
"species": "Human",
"type": "",
"gender": "Male",
},
{
  "id": 6,
  "name": "Abadango Cluster Princess",
  "status": "Alive",
  "species": "Alien",
  "type": "",
  "gender": "Female",
},
{
  "id": 7,
  "name": "Abradolf Lincler",
  "status": "unknown",
  "species": "Human",
  "type": "Genetic experiment",
  "gender": "Male",
},
{
  "id": 8,
  "name": "Adjudicator Rick",
  "status": "Dead",
  "species": "Human",
  "type": "",
  "gender": "Male",
},
{
  "id": 9,
  "name": "Agency Director",
  "status": "Dead",
  "species": "Human",
  "type": "",
  "gender": "Male",
},
}
```

```
{
  "id": 10,
  "name": "Alan Rails",
  "status": "Dead",
  "species": "Human",
  "type": "Superhuman (Ghost trains summoner)",
  "gender": "Male",
}

];
```

9.

Eliminar un elemento de la lista al hacer clic:

- Crea una lista con algunos elementos. Al hacer clic en cualquier elemento de la lista, elimínalo del DOM.

10. Cambiar el contenido de un párrafo con un evento de teclado:

- Crea un campo de texto y un párrafo. Cada vez que el usuario presione una tecla, actualiza el contenido del párrafo con el texto del input.

11. Ocultar y mostrar un párrafo:

- Crea un botón y un párrafo. Al hacer clic en el botón, alterna entre ocultar y mostrar el párrafo.

12. Cambiar el color de fondo de la página:

- Crea un menú desplegable con varios colores. Al seleccionar un color, cambia el fondo de la página al color seleccionado.

13. Contar el número de clics en un botón:

- Crea un botón que cuente cuántas veces se ha hecho clic en él y muestre el resultado en un párrafo.

14. Mostrar la longitud del texto ingresado en tiempo real:

- Crea un input de texto. Muestra en un párrafo la cantidad de caracteres que se ingresan en tiempo real.

15. Cambiar la imagen al pasar el ratón:

- Crea una imagen que cambie cuando el ratón pasa por encima de ella y vuelva a la original cuando se retira el ratón.

16. Añadir una tarea a una lista de tareas:

- Crea un campo de texto y un botón "Agregar Tarea". Al hacer clic en el botón, añade el contenido del campo de texto a una lista de tareas.
17. **Contar palabras de un párrafo en tiempo real:**
- Crea un textarea y un párrafo. A medida que el usuario escribe en el textarea, muestra cuántas palabras ha escrito.
18. **Mostrar una alerta cuando se completa un campo:**
- Crea un input de texto. Cuando el usuario deje de escribir y salga del input, muestra una alerta con el contenido ingresado.
19. **Cambiar el estilo de los elementos de la lista al hacer doble clic:**
- Crea una lista de elementos. Al hacer doble clic en cualquier elemento, cambia su estilo (color, tamaño de fuente, etc.).
20. **Crear una lista desplegable con contenido dinámico:**
- Crea una lista desplegable (select) que se rellene dinámicamente con elementos de un array al cargar la página.

Paso a Paso

Paso 1: Crear la Carpeta del Proyecto

- Crea una carpeta llamada `dom-eventos` en tu computadora donde vas a guardar todos los archivos del proyecto.

Paso 2: Crear el Archivo HTML y Javascript

- Dentro de la carpeta `dom-eventos`, crea un archivo llamado `index.html`.
- Dentro de la carpeta `dom-eventos`, crea los archivos de javascript que creas necesario y vincúlalos al archivo HTML.

Paso 3: Implementar los Ejercicios de 1 a 20 especificados en la Lista de Ejercicios

Paso 4: Guardar y Probar el Código

- Guarda todos los cambios en los archivos `index.html` y `script.js`.
- Revisa en el navegador si los ejercicios funcionan correctamente utilizando **Live Server**.

Recursos y Herramientas

- **Editor de Texto:** [Visual Studio Code](#)
- **Documentación DOM:**
 - [JavaScript DOM Manipulation](#)
 - [Event Listeners in JavaScript](#)

Entrega del Ejercicio

- **Formato de entrega:**
 - Crea un repositorio en gitHub con el nombre **dom-eventos**.
 - Sube la solución de los ejercicios realizados en el repositorio creado.
 - Entrega el link del repositorio a las formadoras, a través del **Discord**.

Este ejercicio te ayudará a profundizar en el manejo del DOM y la creación de interacciones dinámicas en las páginas web mediante el uso de JavaScript.