

MEMORIA DEL BINGO

2ºDAW

Autores: Sergio Belvís Barba, Damiem Rave Grizales,
Brian Camba Hipólito

ÍNDICE

Estructura principal.....	1
Creación de nodos.....	2
Navegación por el DOM.....	3
Modificación de nodos.....	4
Eliminación y reinicio.....	5
Explicación de las principales acciones.....	6

Estructura del DOM del Bingo

Nuestra aplicación del Bingo se basa en la **creación y manipulación dinámica de nodos del DOM**, sin recargar la página.

Estructura principal

- El documento HTML contiene un contenedor principal (`<main id="container">`) vacío.

```
    </div>
  </header>

  <main id="container"></main>
```

- Dentro de este contenedor se generan dinámicamente los **cartones de bingo** de cada alumno.
- Cada cartón es un nodo padre que contiene:
 - Un nodo **<h2>** con el nombre del alumno.

- Un nodo **<table>** que representa el cartón de bingo.

```
// Funcion que crea un carton completo en el DOM
function crearCarton(nombre){
  // Contenedor principal del carton
  let contCarton = document.createElement('div');
  contCarton.classList.add('carton');

  // Nodo que muestra el nombre del alumno
  let titulo = document.createElement('h2');
  titulo.textContent = nombre;

  // Tabla que representa el carton
  let carton = document.createElement('table');
```

Creación de nodos

Los nodos se crean utilizando:

- **document.createElement()**
- **appendChild()**
- **setAttribute()**

Ejemplo:

- Se crea un **<div>** con clase cartón.
- Dentro se insertan filas **<tr>** y celdas **<td>**.
- Cada celda tiene un atributo personalizado, un número que facilita su identificación.

```

for (let i = 0; i < 3; i++) {
  let tr = document.createElement('tr');
  for (let j = 0; j < 5; j++) {
    let td = document.createElement('td');

    // Span que contiene el numero visible
    let spanNumero = document.createElement('span');
    spanNumero.textContent = numeros[indice];
    spanNumero.classList.add('numero');

    // Se añade el número y un atributo personalizado
    td.appendChild(spanNumero);
    td.setAttribute("numero", numeros[indice]);

    tr.appendChild(td);
    indice++;
  }
  carton.appendChild(tr);
}

```

Navegación por el DOM

Para recorrer el árbol DOM se utilizan:

- **querySelectorAll()** para obtener colecciones de nodos.
- **foreach** para recorrer tablas y filas.

```

// Marca automaticamente las casillas que coincidan con el numero
function marcarNumero(numero){
  let casillas = document.querySelectorAll('td');
  casillas.forEach(td => {
    if(parseInt(td.getAttribute("numero")) === numero){
      td.classList.add('marcado');
    }
  });
}

```

- Comparación de atributos para localizar números.

Esto permite detectar automáticamente si un número está presente en un cartón.

Modificación de nodos

Cuando se extrae un número:

- Se modifica el contenido con **textContent**.

```
// Funcion para sacar un numero del bombo
function sacarNumero(){
    if(bombo.length === 0){
        alert("El bombo está vacío");
        return;
    }

    // Seleccion aleatoria de un numero disponible
    let indice = Math.floor(Math.random() * bombo.length);
    let numero = bombo.splice(indice, 1)[0];

    // Mostrar ultimo numero extraido
    ultNumero.textContent = numero;

    // Añadir numero al historial creando un nuevo nodo
    let registroHist = document.createElement('span');
    registroHist.textContent = numero + " ";
    historial.appendChild(registroHist);

    marcarNumero(numero);
    comprobarLogro();
}
```

- Se añaden clases con **classList.add()** para marcar visualmente las casillas.

- Se cambia el estilo del cartón ganador para destacar el bingo.

```
// Comprobacion de bingo completo
if(!carton.classList.contains('bingo')){

  if (!primerBingo && totalMarcadas === 15) {
    carton.classList.add('bingo');
    alert("Bingo completo de " + nombre + "!");

    carton.style.border = "3px solid #FFD700";

    let celdasMarcadas = carton.querySelectorAll('td.marcado');
    celdasMarcadas.forEach(td => td.classList.remove('linea'));
    primerBingo = true;

    // Como ha habido bingo, desaparece el botón de 'Sacar Número' para que se acabe el juego.
    btnSacarNumero.style.display = 'none';
    return;
  }
}
```

Eliminación y reinicio

Para reiniciar la partida:

- Se eliminan los nodos hijos del contenedor con **removeChild()**.
- Se limpian textos y estados sin recargar la página.
- Se generan nuevos cartones dinámicamente.

```

// Reinicia la partida eliminando nodos y estados
function reiniciarJuego(){
    bombo = [];
    for(let i = 1; i <= 90; i++) bombo.push(i);

    historial.innerHTML = "<p>Historial de números: </p>";
    ultNumero.textContent = "";

    // Eliminacion de todos los cartones del DOM
    while(container.firstChild){
        container.removeChild(container.firstChild);
    }

    primeraLinea = false;
    primerBingo = false;

    // Creacion de nuevos cartones
    participantes.forEach(nombre => crearCarton(nombre.trim()));

    btnSacarNumero.style.display = 'inline-block';
}

```

Explicación de las principales acciones

Creación de los cartones

La creación de los cartones se produce gracias a la función `crearCarton()`. Esta función requiere como parámetro el nombre del jugador al que es asignado, cuando lo obtiene lo primero que hace es crear un contenedor principal (div) que engloba todo el contenido y lo añade a la clase "carton" para los estilos. Después, crea un título en el que se almacenará y mostrará posteriormente el nombre del jugador, la tabla que almacena cada uno de los número del cartón, el conjunto de números aleatorios con los que se rellenan las celdas de la tabla (array), las filas y las celdas de la tabla (3 filas x 5 celdas). Acto seguido se rellenan las celdas de la tabla y se añaden a la

clase “numero” (también para los estilos). Por último se añaden todos los nodos del DOM.

```
// Funcion que crea un carton completo en el DOM
function crearCarton(nombre){
  // Contenedor principal del carton
  let contCarton = document.createElement('div');
  contCarton.classList.add('carton');

  // Nodo que muestra el nombre del alumno
  let titulo = document.createElement('h2');
  titulo.textContent = nombre;

  // Tabla que representa el carton
  let carton = document.createElement('table');

  // Array para guardar numeros aleatorios no repetidos
  let numeros = [];
  while (numeros.length < 15) {
    let numero = Math.floor(Math.random() * 90) + 1;
    if (!numeros.includes(numero)) {
      numeros.push(numero);
    }
  }

  // Creacion de filas y celdas del carton
  let indice = 0;
  for (let i = 0; i < 3; i++) {
    let tr = document.createElement('tr');
    for (let j = 0; j < 5; j++) {
      let td = document.createElement('td');

      // Span que contiene el numero visible
      let spanNumero = document.createElement('span');
      spanNumero.textContent = numeros[indice];
      spanNumero.classList.add('numero');

      // Se añade el número y un atributo personalizado
      td.appendChild(spanNumero);
      td.setAttribute("numero", numeros[indice]);

      tr.appendChild(td);
      indice++;
    }
    carton.appendChild(tr);
  }

  // Se insertan todos los nodos en el DOM
  contCarton.appendChild(titulo);
  contCarton.appendChild(carton);
  container.appendChild(contCarton);
}
```

Marcado de números en los cartones

Para que se marquen los número en los cartones del bingo de forma automática existe la función `marcarNumero()` la cuál recibe como parámetro

el último número que ha salido del bombo. Después se recoge en una variable las casillas del cartón (o celdas de la tabla) y se compara con un bucle cada celda con el número de bombo. Si la condición se cumple la celda se añade a la clase “marcado” para darle el estilo y que por ende salga marcado.

```
// Marca automaticamente las casillas que coincidan con el numero  
function marcarNumero(numero){  
    let casillas = document.querySelectorAll('td');  
    casillas.forEach(td => {  
        if(parseInt(td.getAttribute("numero")) === numero){  
            td.classList.add('marcado');  
        }  
    });  
}
```

Detección y exposición de ganadores

Bingo

La detección de los ganadores y el mostrado de los mismos se realiza mediante la función `comprobarLogro()`. Primero se recogen todos los cartones, se recogen las filas, las celdas marcadas y el nombre del dueño de cada cartón. Después de guardar esos 3 parámetros se cuentan todas las casillas marcadas para saber si hay bingo con la siguiente comprobación sencilla, un condicional que comprueba si el número de casillas marcadas es 15. Si la condición se cumple, el cartón se añade a la clase “bingo” (para el estilo), salta una alerta con el mensaje de ganador del bingo con el nombre del dueño del cartón y por último si hay alguna línea de antes se elimina de la lista para que no haya conflicto de estilos.

Línea

Dentro de la misma función se encuentra el bloque de código que comprueba el logro de línea. Con los parámetros recogidos antes se hace una comprobación mediante un bucle de cada una de las filas, se recoge el número de celdas y el número de celdas marcadas para luego con un condicional comparar si el número de celdas marcadas es igual al número total de celdas de la fila. Si la condición se cumple se añade la fila completa a la clase “línea” (para añadirle el estilo), con un bucle se añade cada una de las celdas también a la clase “línea” y por último se muestra un mensaje de alerta con el nombre del ganador de la línea y dueño del cartón.

```

// Comprueba si hay línea o bingo en los cartones
function comprobarLogro(){
    let cartones = document.querySelectorAll('.carton');

    cartones.forEach(carton => {
        let filas = carton.querySelectorAll('tr');
        let totalMarcadas = 0;
        let nombre = carton.querySelector('h2').textContent;

        // Contar todas las casillas marcadas del carton
        filas.forEach(tr => totalMarcadas += tr.querySelectorAll('td.marcado').length);

        // Comprobación de bingo completo
        if(!carton.classList.contains('bingo')){

            if (!primerBingo && totalMarcadas === 15) {
                carton.classList.add('bingo');
                alert("Bingo completo de " + nombre + "!");

                carton.style.border = "3px solid ■ #FFD700";

                let celdasMarcadas = carton.querySelectorAll('td.marcado');
                celdasMarcadas.forEach(td => td.classList.remove('linea'));
                primerBingo = true;

                // Como ha habido bingo, desaparece el botón de 'Sacar Número' para que se acabe el juego.
                btnSacarNumero.style.display = 'none';
                return;
            }

            // Comprobación de línea por filas
            filas.forEach(tr => {
                let celdas = tr.querySelectorAll('td');
                let marcadas = tr.querySelectorAll('td.marcado');

                if (celdas.length === marcadas.length && !tr.classList.contains('linea') && !tr.classList.contains('primeralineas')) {
                    // Si es la primera línea aparece un alert y la línea se rellena de amarillo. Si no, se rellena de verde.
                    if(!primeralineas){
                        tr.classList.add('primeralineas');
                        marcadas.forEach(td => td.classList.add('primeralineas'));
                        alert("Línea completa de " + nombre + "!");
                        primeralineas = true;
                    } else {
                        tr.classList.add('linea');
                        marcadas.forEach(td => td.classList.add('linea'));
                    }
                }
            });
        }
    });
}

```