



Ejercicio 1: Buscaminas en consola

Crea un programa en **JavaScript** que simule el juego del **Buscaminas** en modo texto, ejecutándose **en la consola del navegador** (no en una interfaz gráfica).

Requisitos básicos

El programa debe:

- Pedir al usuario, mediante `prompt`, el **tamaño del tablero** (por ejemplo, 5 para un tablero 5x5).
- Generar **aleatoriamente las minas** en el tablero.
 - Puedes decidir la cantidad de minas tú mismo, por ejemplo, un 20% de las casillas totales o un número fijo.
- Representar el tablero con un array bidimensional (**Array de Array**), donde cada casilla puede tener:
 - "*" si es mina.
 - Números indicando la cantidad de minas adyacentes.
 - "X" si está sin descubrir.
- Para mostrar el tablero por consola se puede usar `console.table()` o texto formateado.

Lógica del juego

El juego debe funcionar mediante rondas. En cada ronda:

1. El programa pedirá al usuario (con `prompt`) las **coordenadas de la casilla a descubrir** (fila y columna).
 - a. Las coordenadas se corresponderán con los **índices del array** (por ejemplo, fila 0 y columna 2).
2. Según el contenido de la casilla:
 - a. Si hay una **mina**, mostrar un `alert` que diga "¡BOOM! Has perdido" y terminar el programa.
 - b. Si no hay mina:
 - i. Si la casilla tiene un **número**, se desvela mostrando ese número en el tablero.
 - ii. Si se trata de una **casilla vacía** (sin minas adyacentes), se deben desvelar todas las casillas adyacentes a esta que estén vacías o sean numéricas, de forma recursiva.
3. El juego continúa hasta que **todas las casillas sin mina se descubran**, y en ese momento se muestra un mensaje que diga: "Enhorabuena, has ganado".



Estructura recomendada

Debes modularizar el código usando **funciones**, por ejemplo (puedes usar más si lo necesitas o lo ves necesario, o usar otros nombres):

- `generarTablero(tamaño)`
- `colocarMinas(tablero, cantidad)`
- `contarMinasAdyacentes(tablero, fila, columna)`
- `mostrarCasillasAdyacentesVaciasONumericas(fila, columna)`
- `mostrarTablero(tablero)`
- `jugar()`

Importante: cada función debe tener una responsabilidad clara. **EVITA ESCRIBIR TODA LA LÓGICA DENTRO DE `jugar()`** .

Desafío opcional

Como mejora adicional, crea un **objeto** para guardar el estado de la partida. Por ejemplo:

```
let partida = {  
  tablero: tablero,  
  minasRestantes: numMinas,  
  movimientos: intentos  
};
```

Convierte el objeto a formato JSON con `JSON.stringify(partida)` y muéstralo por consola para comprobar su estructura.