



Mini aventura conversacional

Requisitos técnicos obligatorios (ES6)

El programa **DEBE** cumplir los siguientes puntos:

- Uso de let y const (no se permite var)
- Uso de **funciones** (normales o funciones flecha)
- Uso de **arrays**
- Uso de **template strings** (`\$`{})
- Código estructurado y legible

Descripción general

Debes crear un programa en **ECMAScript 6** que simule una **mini aventura conversacional en texto**, donde el jugador pueda:

- Moverse entre varias habitaciones
- Recoger objetos
- Consultar su inventario
- Ganar la partida al cumplir una condición

El juego se controla mediante un **menú repetitivo** usando prompt.

Estructura de datos obligatoria

Se deben usar **arrays** para almacenar la información del juego. Como mínimo, deben existir los siguientes arrays:

- **habitaciones** → nombres de las habitaciones (mínimo 3)
- **descripciones** → descripción de cada habitación
- **objetos** → objeto presente en cada habitación
 - Uno de ellos debe ser "tesoro"
- **inventario** → objetos recogidos por el jugador

Ejemplo orientativo:

```
const habitaciones = ["Entrada", "Pasillo", "Sala final"];
const descripciones = [
    "La entrada de una cueva misteriosa.",
    "Un pasillo largo y oscuro.",
    "Una sala iluminada con un cofre antiguo."
];
let objetos = ["linterna", "llave", "tesoro"];
let inventario = [];
```



Inicio del juego

- El jugador comienza en la **habitación 0**
- El inventario empieza vacío
- Al iniciar el programa se muestra la información de la habitación actual

Mostrar información de la habitación

Pueden crearse tantas funciones como se consideren necesarias, pero **obligatoriamente** debe crearse una **función** que:

- Reciba como parámetro la posición actual del jugador
- Muestre por consola:
 - Nombre de la habitación
 - Descripción
 - Objeto presente (si lo hay)

Ejemplo:

Estás en: Pasillo

Un pasillo largo y oscuro.

Hay una llave en el suelo.

Menú de acciones

El programa mostrará un **menú en bucle** con las siguientes opciones:

1. Avanzar a la siguiente habitación
2. Retroceder a la habitación anterior
3. Recoger objeto
4. Ver inventario
0. Salir del juego
 - El menú se repetirá usando un **bucle**
 - La opción se gestionará con switch
 - La entrada se pedirá con prompt

Movimiento entre habitaciones

- Solo se puede **avanzar** si no está en la última habitación
- Solo se puede **retroceder** si no está en la primera habitación
- Si el movimiento no es válido, se mostrará un mensaje usando console.log

Recoger objetos

- Si en la habitación hay un objeto distinto de "ninguno":
 - Se añade al inventario
 - El objeto desaparece de la habitación
- Si no hay objeto, se muestra un mensaje informativo



Inventario

Debe crearse una **función** que muestre el inventario. En el caso de que el inventario esté vacío, muestra un mensaje que lo indica, y si contiene objetos, se muestra **uno por línea**.

Condición de victoria

Si el jugador llega a la **última habitación** Y tiene "tesoro" en el inventario se mostrará un mensaje usando **template strings** y el programa finalizará:

¡Has encontrado el tesoro y ganado la aventura!