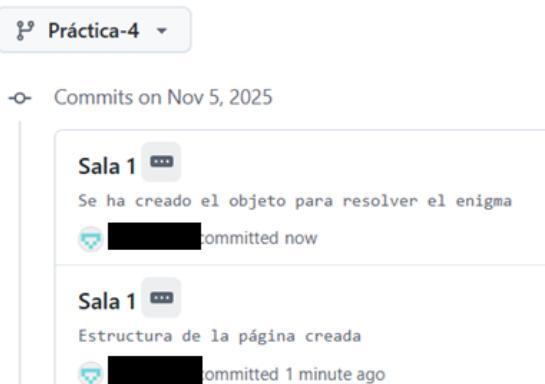


## INSTRUCCIONES

- La práctica es individual. Se dedicarán horas de clase a la resolución de dudas.
- Si se detecta alguna copia, la práctica se calificará con un 0 y no se podrá recuperar hasta junio.
- La entrega final será en un enlace habilitado en el aula virtual. La práctica se comprimirá en un archivo denominado “Práctica4\_Nombre\_Apellidos.zip”. Por ejemplo, “Práctica4\_José\_López\_Pérez.zip”. El código JavaScript siempre estará en archivos externos.

## ENTREGAS PARCIALES

- Se subirá el progreso actual a GitHub a modo de seguimiento. Puedes crear una rama específica para la práctica y así facilitar la organización del repositorio. No se corregirá la práctica si no se observan cambios en GitHub gradualmente. Cada sala tendrá al menos 5 commits con descripciones claras de la funcionalidad desarrollada. Ejemplos de formato para los commits:



- En relación a lo anterior, se realizarán varias entregas parciales del progreso actual en el aula virtual y se pedirá añadir código en clase en clase sin internet o justificar algún commit.
- Es obligatorio cumplir los requisitos de las entregas parciales para poder realizar la entrega final.

## ENUNCIADO

El objetivo de la práctica es crear un [juego de escape](#) con tres salas cerradas, que solo se podrán abrir si se resuelven una serie de retos manipulando las estructuras de datos empleadas en la unidad.

Para cambiar de pantalla se podrá emplear `window.location.href` o `window.location.reload()` en caso de tener que recargar la misma pantalla.

Cada una de las salas tendrá como requisito emplear las siguientes herramientas:

1. Primera sala: Tendrá algún objeto con [propiedades de tipo symbol](#), [funciones](#) y [otros objetos anidados](#). Se usarán [métodos de clonación de objetos](#) al menos una vez.

2. Segunda sala: Empleará las siguientes herramientas:

- Al menos tres métodos de Arrays
- Un Map y Set para almacenar pistas.
- Se aplicará el operador spread o la desestructuración.
- Se usará Array.from para convertir claves o valores a un Array.

3. Tercera sala: Empleará las siguientes herramientas:

- Creará alguna clase con herencia, propiedades o métodos privados y estáticos.
- Se modificará al menos un prototipo de objetos creados en una clase.
- Todas las clases se guardarán en módulos y se importarán posteriormente.

Además de los requisitos de cada sala, la pantalla principal o las diferentes salas cargarán los datos o instrucciones desde un JSON.

Por otro lado, se requiere controlar excepciones al menos una vez y añadir errores personalizados.