

## Trabajo practico 2 Callback

1. **Multiplicar números con un callback:**
  - Crea una función multiplicar que acepte dos números y un callback.
  - Usa el callback para mostrar el resultado de multiplicar los dos números.
2. **Operación con números usando callbacks:**
  - Define una función operarNumeros que acepte dos números y un callback.
  - Crea dos callbacks, uno para sumar y otro para restar los números.
3. **Uso de setTimeout con callbacks:**
  - Usa setTimeout para imprimir "Esto es un callback" después de 2 segundos.
4. **Filtrar números pares con un callback:**
  - Define una función filtrarPares que acepte un array de números y un callback.
  - Usa el callback para filtrar solo los números pares.

### Ejercicios sobre Set

6. **Crear y manipular un Set:**
  - Crea un Set con los números 1, 2, 3, 4, y 5.
  - Añade el número 6 y elimina el número 3.
7. **Convertir un Set a un Array:**
  - Convierte el Set del ejercicio anterior en un array.
8. **Eliminar duplicados de un array usando Set:**
  - Dado un array [1, 2, 2, 3, 4, 4, 5], elimina los duplicados utilizando un Set.
9. **Comprobar existencia en un Set:**
  - Comprueba si el número 4 existe en el Set creado anteriormente y muestra el resultado.
10. **Iterar sobre un Set:**
  - Itera sobre el Set creado en el ejercicio 6 e imprime cada número.

**Ejercicios sobre Map****11. Crear y manipular un Map:**

- Crea un Map con tres entradas: "nombre" -> "Juan", "edad" -> 30, "ciudad" -> "Madrid".
- Añade una nueva entrada para "país" -> "España" y elimina la entrada "ciudad".

**12. Acceder a valores en un Map:**

- Accede al valor de "nombre" en el Map creado y muéstralo.

**13. Iterar sobre un Map:**

- Itera sobre las entradas del Map e imprime cada par clave-valor.

**14. Comprobar existencia en un Map:**

- Comprueba si el Map tiene la clave "edad" y muestra el resultado.

**15. Convertir un Map a un Array:**

- Convierte el Map creado en el ejercicio 11 a un array de arrays.

**Ejercicios sobre Arrays y Métodos de Array****16. Filtrar números mayores que 10:**

- Dado un array [5, 10, 15, 20, 25], usa filter para obtener los números mayores que 10.

**17. Doblar los números en un array:**

- Usa map para crear un nuevo array donde cada número en el array [1, 2, 3, 4, 5] se ha doblado.

**18. Convertir un array de nombres a mayúsculas:**

- Dado un array ["juan", "maria", "pedro"], usa map para convertir cada nombre a mayúsculas.

**19. Filtrar palabras cortas:**

- Dado un array ["árbol", "casa", "elefante", "sol"], usa filter para obtener las palabras con menos de 5 letras.

**20. Sumar números pares en un array:**

- Dado un array [1, 2, 3, 4, 5, 6, 7, 8, 9, 10], usa filter para obtener los números pares y luego utiliza reduce para sumarlos.

# Ejercicios avanzados

## 1. Transformar objetos usando map:

- Dado un array de objetos [{ nombre: "Juan", edad: 20 }, { nombre: "María", edad: 25 }, { nombre: "Pedro", edad: 30 }], usa map para crear un nuevo array de strings que contenga solo los nombres.

## 2. Filtrar objetos en un array:

- Dado un array de objetos [{ nombre: "Carlos", edad: 17 }, { nombre: "Ana", edad: 23 }, { nombre: "Luis", edad: 16 }], usa filter para obtener solo los objetos donde la persona sea mayor de edad (edad >= 18).

## 3. Contar elementos únicos en un array usando Map:

- Dado un array de números [2, 3, 4, 2, 3, 5, 4, 5, 6], utiliza un Map para contar cuántas veces aparece cada número en el array. El resultado debe ser un Map donde las claves sean los números del array y los valores sean la cantidad de veces que cada número aparece.

## 4. Encadenamiento de métodos de array:

- Dado un array de números [1, 2, 3, 4, 5, 6, 7, 8, 9, 10], usa una combinación de filter, map, y reduce para obtener la suma de los cuadrados de los números pares.

## 5. Transformar un objeto en un Map:

- Dado un objeto { a: 1, b: 2, c: 3 }, transforma el objeto en un Map donde las claves son las propiedades del objeto y los valores son los valores correspondientes.

## 6. Crear un objeto a partir de un Map:

- Dado un Map que contiene [{"nombre", "Pedro"}, {"edad", 30}, {"ciudad", "Sevilla"}], convierte este Map en un objeto.

## 7. Combinar arrays de objetos usando filter y map:

- Dado dos arrays de objetos: estudiantes con propiedades id y nombre, y notas con propiedades idAlumno y nota, usa filter y map para combinar ambos arrays en uno nuevo que contenga el nombre del alumno y su nota correspondiente.

## 8. Eliminar elementos duplicados y mantener el orden usando Set y filter:

- Dado un array de números [1, 2, 2, 3, 4, 4, 5, 1, 6, 7], usa Set junto con filter para eliminar los elementos duplicados, pero manteniendo el primer elemento encontrado en su posición original.