Explicación paso a paso del uso de Array.map() en JavaScript

# Código original

// Definir un array de números  
let numeros = [1, 2, 3, 4, 5];  
  
// Usar Array.map() para crear un nuevo array donde cada número se multiplica por 2  
let numerosMultiplicados = numeros.map(function(numero) {  
 return numero \* 2;  
});  
  
// Mostrar el array original  
console.log("Array original:", numeros);  
  
// Mostrar el nuevo array con los números multiplicados por 2  
console.log("Array multiplicado:", numerosMultiplicados);  
  
// Otro ejemplo: convertir un array de cadenas a mayúsculas  
let palabras = ["hola", "mundo", "java"];  
let palabrasEnMayusculas = palabras.map(function(palabra) {  
 return palabra.toUpperCase();  
});  
  
console.log("Palabras originales:", palabras);  
console.log("Palabras en mayúsculas:", palabrasEnMayusculas);

# Explicación paso a paso

## 1. Definición del array de números

let numeros = [1, 2, 3, 4, 5];  
• ¿Qué hace?  
 Crea un array llamado "numeros" con los valores [1, 2, 3, 4, 5].  
• ¿Para qué sirve?  
 Es el array base sobre el que se aplicarán transformaciones usando map().

## 2. Usar map() para transformar cada elemento

let numerosMultiplicados = numeros.map(function(numero) {  
 return numero \* 2;  
});  
• ¿Qué hace?  
 Utiliza el método map() para crear un nuevo array donde cada elemento de "numeros" es multiplicado por 2.  
 La función anónima toma cada elemento (numero) del array original y devuelve ese elemento por 2.  
• ¿Para qué sirve?  
 Permite transformar todos los elementos de un array de manera sencilla y sin modificar el array original.  
• ¿Resultado?  
 El nuevo array es [2, 4, 6, 8, 10].

## 3. Mostrar el array original

console.log("Array original:", numeros);  
• ¿Qué hace?  
 Imprime el array original.  
• ¿Para qué sirve?  
 Demuestra que el método map() no cambia el array original, solo genera uno nuevo.  
• ¿Qué imprime?  
 Array original: [1, 2, 3, 4, 5]

## 4. Mostrar el nuevo array transformado

console.log("Array multiplicado:", numerosMultiplicados);  
• ¿Qué hace?  
 Imprime el nuevo array con los valores multiplicados por 2.  
• ¿Qué imprime?  
 Array multiplicado: [2, 4, 6, 8, 10]

## 5. Ejemplo con cadenas: convertir a mayúsculas

let palabras = ["hola", "mundo", "java"];  
let palabrasEnMayusculas = palabras.map(function(palabra) {  
 return palabra.toUpperCase();  
});  
• ¿Qué hace?  
 Crea un array de palabras en minúsculas. Luego usa map() para convertir cada palabra a mayúsculas con el método .toUpperCase().  
• ¿Para qué sirve?  
 Puedes transformar rápidamente un array de strings aplicando una función a cada elemento.  
• ¿Resultado?  
 El nuevo array es ["HOLA", "MUNDO", "JAVA"].

## 6. Mostrar resultados de las cadenas

console.log("Palabras originales:", palabras);  
console.log("Palabras en mayúsculas:", palabrasEnMayusculas);  
• ¿Qué hacen?  
 Imprimen el array original y el array transformado.  
• ¿Qué imprimen?  
 Palabras originales: ["hola", "mundo", "java"]  
 Palabras en mayúsculas: ["HOLA", "MUNDO", "JAVA"]

# Resumen general sobre map()

• map(): Recorre cada elemento de un array, aplica una función, y devuelve un nuevo array con los resultados.  
• No modifica el array original.  
• Es útil para transformar datos fácilmente (matemáticas, textos, objetos, etc.).