

GM2

Arrays

GM2

Métodos

.includes()

Determina si una matriz incluye un determinado elemento, devuelve true o false según corresponda.

- **Recibe** un elemento a buscar en el array.
- **Retorna** *true* si encontró lo que buscábamos, *false* en caso contrario.

```
{}  
  
let frutas = ['Manzana', 'Pera', 'Frutilla'];  
  
frutas.includes('Frutilla');  
// Encontró lo que buscaba. Devuelve true  
  
frutas.includes('Banana');  
// No encontró lo que buscaba. Devuelve false
```

GM2

find()

GM2

MÉTODO FIND

El método **find()** devuelve el valor del primer elemento del Array que satisface la función de comprobación enviada por parámetro. Si ningún valor satisface la función de comprobación, se devuelve undefined.

```
let numeros    = [1, 2, 3, 4, 5];  
//La función parámetro generalmente es una función flecha sin cuerpo.  
let encontrado = numeros.find(function(numeros){  
    return numeros > 3  
}); //Encuentra 4
```

```
let nombres= ["Ana", "Ema", "Juan"];  
let nombreEma = nombres.find(function(nombre){  
    return nombre === "Ema"  
}); //Encuentra "Ema"  
let NombreEma = nombres.find(function(nombre){  
    return nombre === "Luis"  
}); //undefined
```

GM2

every()

GM2

MÉTODO EVERY

Determina si todos los elementos en el array cumplen una condición.
¡Llamar este método en un array vacío devuelve true para cualquier condición!

```
let numbers = [1, 2, 3, 4, 5];

// Chequea si todos los ítems en el carrito están en stock
let allPositive = numbers.every(function(num) {
  return num > 0;
});

console.log(allPositive) // true
```

GM2

`some()`

GM2

MÉTODO SOME

El método `some()` comprueba si al menos un elemento del array cumple con la condición implementada por la función proporcionada.

```
let numbers = [15, 20, 25, 30, 8];

// Usar el método every para verificar si todos los números son mayores que 10
let allGreaterThanTen = numbers.every(function(num) {
    return num > 10;
});

// Imprimir el resultado
console.log(allGreaterThanTen); // false
```

GM2

Deconstructing

“

El "**destructuring**" en JavaScript es una característica que permite descomponer un array o un objeto en variables más pequeñas y accesibles de manera más conveniente. Esto facilita la extracción de datos y asignación de variables de manera concisa.

”



GM2

DESTRUCTURING ARRAY

En este ejemplo, color1, color2 y color3 son variables que se asignan utilizando destructuring para extraer los valores del array colors. Cada variable se asigna a un elemento correspondiente del array en el mismo orden.

```
let colors = ['rojo', 'verde', 'azul'];

// Usando destructuring para asignar variables
let [color1, color2, color3] = colors;

console.log(color1 ) // rojo
console.log(color2 ) // verde
console.log(color3 ) // azul
```

GM2

¡Eso es todo amigos!



The background features a gradient of warm colors, transitioning from a light orange at the top to a dark brown at the bottom. There are several overlapping, semi-transparent geometric shapes, including circles and polygons, in various shades of orange and brown. The text 'GM2' is centered in a bold, white, sans-serif font.

GM2