



# Métodos funcionales en Arrays (Pt. 1)

Introducción a JavaScript  
Mentor: Joshua Eduardo González Ruíz

# OBJETIVO





Aprender cómo y cuándo utilizar los métodos funcionales de los Arrays en **JS**.

Utilizarlos para realizar procesos de forma adecuada en el código.

# ÍNDICE

- **Métodos funcionales en Arrays (Pt. 1)**
  - Ordenar los elementos del arreglo en orden ascendente o descendente .... pág. 6
  - Fusionar dos o más arreglos y devolver un nuevo arreglo, sin cambiar los arreglos existentes ..... pág. 7
  - Comprobar que todos los elementos del arreglo pasen una condición, devolviendo verdadero o falso según corresponda ..... pág. 8
  - Comprobar si al menos un elemento del arreglo pasa una condición, devolviendo verdadero o falso según corresponda ..... pág. 9
  - Comprobar si un arreglo incluye el elemento que pasa la condición, devolviendo verdadero o falso según corresponda ..... pág. 10

- Devolver el valor del primer elemento en un arreglo que pasa la prueba en una función de prueba ..... pág. 11
- Devolver el índice del primer elemento en un arreglo que pasa la prueba en una función de prueba ..... pág. 12
- Llenar los elementos de un arreglo con un valor estático y devolver el arreglo modificado ..... pág. 13
- Devolver un nuevo arreglo con elementos especificados desde el inicio hasta el final ..... pág. 14
- Invertir un arreglo en su lugar. El elemento en el último índice será el primero y el elemento en el índice 0 será el último ..... pág. 15
- **Práctica** ..... pág. 16
- **Referencias** ..... pág. 17

# sort()

```
1 let kodersNames = ["Francisco", "Jess", "Héctor", "Mónica", "Gustavo"];
2
3 //-----
4
5 console.log("Nombres en orden ascendente: ");
6
7 kodersNames.sort((a,b) => a > b ? 1 : -1);
8
9 console.log(kodersNames);
10
11 //-----
12
13 console.log("Nombres en orden descendente: ");
14
15 kodersNames.sort((a,b) => a > b ? -1 : 1);
16
17 console.log(kodersNames);
18
19 //-----
```

Nombres en orden ascendente:

```
[ 'Francisco', 'Gustavo', 'Héctor', 'Jess', 'Mónica' ]
```

Nombres en orden descendente:

```
[ 'Mónica', 'Jess', 'Héctor', 'Gustavo', 'Francisco' ]
```

Hint: hit control+c anytime to enter REPL.

```
>
```

# concat()

```
1 let kodersNames = ["Francisco", "Jess", "Héctor", "Mónica", "Gustavo"];
2
3 let otherKodersNames = ["Roberto", "Cristian", "Heidel", "Erick",
4 "Aldahir"];
5 //-----
6
7 console.log("Se unen los dos arreglos: ");
8
9 let allKodersNames = kodersNames.concat(otherKodersNames);
10
11 console.log(allKodersNames);
12
13 //-----
14
15 console.log("Corroboramos el primer arreglo: ");
16
17 console.log(kodersNames);
18
19 //-----
20
21 console.log("Corroboramos el segundo arreglo: ");
22
23 console.log(otherKodersNames);
```

Se unen los dos arreglos:

```
[
  'Francisco', 'Jess',
  'Héctor',    'Mónica',
  'Gustavo',   'Roberto',
  'Cristian',  'Heidel',
  'Erick',     'Aldahir'
]
Corroboramos el primer arreglo:
[ 'Francisco', 'Jess', 'Héctor', 'Mónica', 'Gustavo' ]
Corroboramos el segundo arreglo:
[ 'Roberto', 'Cristian', 'Heidel', 'Erick', 'Aldahir' ]
```

Hint: hit control+c anytime to enter REPL.

>

# every()

```
1 let kodersNames = ["Francisco", "Jess", "Héctor", "Mónica", "Gustavo",  
  "Roberto", "Cristian", "Heidel", "Erick", "Aldahir"];  
2  
3 //-----  
4  
5 console.log('¿Todos los elementos del arreglo contienen la letra "a"?');  
6  
7 let letterVerification = kodersNames.every(letter => letter.includes("a"))  
  ;  
8  
9 console.log(letterVerification);  
10  
11 //-----  
12  
13 console.log("¿Todos los elementos del arreglo tienen al menos 4  
  caracteres de longitud?");  
14  
15 let lengthVerification = kodersNames.every(letter => letter.length >= 4);  
16  
17 console.log(lengthVerification);
```

¿Todos los elementos del arreglo contienen la letra "a"?

false

¿Todos los elementos del arreglo tienen al menos 4 caracteres de longitud?

true

Hint: hit control+c anytime to enter REPL.

✂



# some()

```
1 let kodersNames = ["Francisco", "Jess", "Héctor", "Mónica", "Gustavo",  
  "Roberto", "Cristian", "Heidel", "Erick", "Aldahir"];  
2  
3 //-----  
4  
5 console.log('¿Alguno de los elementos del arreglo contienen la letra "a"?  
  ');  
6  
7 let letterVerification = kodersNames.some(letter => letter.includes("a"));  
8  
9 console.log(letterVerification);  
10  
11 //-----  
12  
13 console.log("¿Alguno de los elementos del arreglo tienen al menos 10  
  caracteres de longitud?");  
14  
15 let lengthVerification = kodersNames.some(letter => letter.length >= 10);  
16  
17 console.log(lengthVerification);
```

¿Alguno de los elementos del arreglo contienen la letra "a"?

true

Alguno de los elementos del arreglo tienen al menos 10 caracteres de longitud?

false

Hint: hit control+c anytime to enter REPL.

➤

# includes()

```
1 let kodersNames = ["Francisco", "Jess", "Héctor", "Mónica", "Gustavo",  
  "Roberto", "Cristian", "Heidel", "Erick", "Aldahir"];  
2  
3 //-----  
4  
5 console.log('¿Alguno de los elementos del arreglo es "Jess"?');  
6  
7 let firstVerification = kodersNames.includes("Jess");  
8  
9 console.log(firstVerification);  
10  
11 //-----  
12  
13 console.log('¿Alguno de los elementos del arreglo es "Vryahn"?');  
14  
15 let secondVerification = kodersNames.includes("Vryahn");  
16  
17 console.log(secondVerification);
```

```
¿Alguno de los elementos del arreglo es "Jess"?  
true  
¿Alguno de los elementos del arreglo es "Vryahn"?  
false  
Hint: hit control+c anytime to enter REPL.  
> []
```

# find()

```
1 let kodersNames = ["Francisco", "Jess", "Héctor", "Mónica", "Gustavo",  
  "Roberto", "Cristian", "Heidel", "Erick", "Aldahir"];  
2  
3 //-----  
4  
5 console.log('Encontrar el primer elemento que contiene la letra "o"');  
6  
7 let verification = kodersNames.find(element => element.includes("o"));  
8  
9 console.log(verification);
```

Encontrar el primer elemento que contiene la letra "o"  
Francisco

Hint: hit control+c anytime to enter REPL.

>

# findIndex()

```
1 let kodersNames = ["Francisco", "Jess", "Héctor", "Mónica", "Gustavo",  
  "Roberto", "Cristian", "Heidel", "Erick", "Aldahir"];  
2  
3 //-----  
4  
5 console.log('Encontrar el índice del primer elemento que contiene la  
  letra "o"');  
6  
7 let verification = kodersNames.findIndex(element => element.includes("o"))  
  ;  
8  
9 console.log(verification);
```

Encontrar el índice del primer elemento que contiene la letra "o"

0

Hint: hit control+c anytime to enter REPL.

>

# fill()

```
1 let kodersNames = new Array(3);
2
3 //-----
4
5 console.log('Llenar todos los espacios vacíos del arreglo con
"Francisco": ');
6
7 kodersNames.fill("Francisco");
8
9 console.log(kodersNames);
10
11 console.log("\n");
12
13 //-----
14
15 console.log('Llenar el espacio del índice 1 en el arreglo con "Jess": ');
16
17 kodersNames.fill("Jess", 1, 2);
18
19 console.log(kodersNames);
20
21 console.log("\n");
22
23 //-----
24
25 console.log('Llenar el espacio del índice 2 en el arreglo con "Héctor": ')
;
26
27 kodersNames.fill("Héctor", 2, 3);
28
29 console.log(kodersNames);
30
31 //-----
```

Llenar todos los espacios vacíos del arreglo con "Francisco":  
[ 'Francisco', 'Francisco', 'Francisco' ]

Llenar el espacio del índice 1 en el arreglo con "Jess":  
[ 'Francisco', 'Jess', 'Francisco' ]

Llenar el espacio del índice 2 en el arreglo con "Héctor":  
[ 'Francisco', 'Jess', 'Héctor' ]

Hint: hit control+c anytime to enter REPL.

>



# slice()

```
1 let kodersNames = ["Francisco", "Jess", "Héctor", "Mónica", "Gustavo",  
  "Roberto", "Cristian", "Heidel", "Erick", "Aldahir"];  
2  
3 //-----  
4 console.log('Se crea un nuevo arreglo a partir de los índices 0, 1, 2 del  
  arreglo "kodersNames"');  
5 let firstTeam = kodersNames.slice(0,3);  
6 console.log(firstTeam);  
7  
8 //-----  
9 console.log('Se crea un nuevo arreglo a partir de los índices 3, 4, 5 del  
  arreglo "kodersNames"');  
10  
11 let secondTeam = kodersNames.slice(3,6);  
12 console.log(secondTeam);  
13  
14 //-----  
15 console.log('Se crea un nuevo arreglo a partir de los índices 6, 7, 8 del  
  arreglo "kodersNames"');  
16  
17 let thirdTeam = kodersNames.slice(6,9);  
18 console.log(thirdTeam);  
19  
20 //-----  
21 console.log('Se crea un nuevo arreglo a partir del índice 10 del arreglo  
  "kodersNames"');  
22  
23 let fourthTeam = kodersNames.slice(9,10);  
24 console.log(fourthTeam);
```

Se crea un nuevo arreglo a partir de los índices 0, 1, 2 del arreglo "kodersNames"  
[ 'Francisco', 'Jess', 'Héctor' ]  
Se crea un nuevo arreglo a partir de los índices 3, 4, 5 del arreglo "kodersNames"  
[ 'Mónica', 'Gustavo', 'Roberto' ]  
Se crea un nuevo arreglo a partir de los índices 6, 7, 8 del arreglo "kodersNames"  
[ 'Cristian', 'Heidel', 'Erick' ]  
Se crea un nuevo arreglo a partir del índice 10 del arreglo "kodersNames"  
[ 'Aldahir' ]  
Hint: hit control+c anytime to enter REPL.

>

# reverse()

```
1 let kodersNames = ["Francisco", "Jess", "Héctor", "Mónica", "Gustavo",  
2 "Roberto", "Cristian", "Heidel", "Erick", "Aldahir"];  
3 //-----  
4 console.log('Muestra los nombres de los koders empezando desde el  
5 elemento final hasta el del principio: ');  
6 kodersNames.reverse();  
7 console.log(kodersNames);
```

Muestra los nombres de los koders empezando desde el elemento final hasta el del principio: 🔍 ✕

```
[  
  'Aldahir', 'Erick',  
  'Heidel', 'Cristian',  
  'Roberto', 'Gustavo',  
  'Mónica', 'Héctor',  
  'Jess', 'Francisco'  
]  
Hint: hit control+c anytime to enter REPL.  
> 
```

Tiempo de  
practicar





# Referencias

- **Arrays – Learn web development | MDN. (2021, November 19). MDN Web Docs. [https://developer.mozilla.org/en-US/docs/Learn/JavaScript/First\\_steps/Arrays](https://developer.mozilla.org/en-US/docs/Learn/JavaScript/First_steps/Arrays)**