

01

10 JAVASCRIPT ARRAY METHODS IN JAVASCRIPT!



@souravcode

① FOREACH()

② SOME()

③ INCLUDES()

④ EVERY()

FILTER()

⑥ SORT()

⑦ MAP()

⑧ ARRAY.FROM()

⑨ REDUCE()

⑩ ARRAY.OF()

① FOREACH()

➡ THIS METHOD CAN HELP YOU TO LOOP OVER ARRAY'S ITEMS. THE FOREACH() METHOD CAN ALSO BE USED ON MAPS AND SETS.




```
const arr=[1,2,3,4,5,6];  
arr.forEach(items=>{  
    console.log(items);  
})
```

2 SOME()

⇒ THE SOME() METHOD EXECUTES THE FUNCTION ONCE FOR EACH ARRAY ELEMENT:

◆ IF THE FUNCTION RETURNS TRUE, SOME() RETURNS TRUE AND STOPS.


◆ IF THE FUNCTION RETURNS FALSE, SOME() RETURNS FALSE AND STOPS.



```
const arr = [1,2,3,4,5,6];  
  
const num = arr.some(num => num > 4);  
const newNum = arr.some(num => num <= 0)  
console.log(num); //true  
console.log(newNum); // false
```

③ INCLUDES()

⇒ THE INCLUDES() METHOD DETERMINES WHETHER AN ARRAY INCLUDES A CERTAIN VALUE AMONG ITS ENTRIES, RETURNING TRUE OR FALSE AS APPROPRIATE.



```
const arr =[1,2,3,4,5,6];  
  
let yes=arr.includes(4);// true return  
console.log(yes);
```

4 EVERY()

→ THE EVERY() METHOD TESTS WHETHER ALL ELEMENTS IN THE ARRAY PASS THE TEST IMPLEMENTED BY THE PROVIDED FUNCTION. IT RETURNS A BOOLEAN VALUE.

```
const arr = [1,2,3,4,5,6];

const num = arr.every(num => num > 4);
const newNum = arr.every(num => num <= 10)
console.log(num); // false
console.log(newNum); // true
```

5 FILTER()

==> THE FILTER() METHOD CREATES A NEW ARRAY FILLED WITH ELEMENTS THAT PASS A TEST PROVIDED BY A FUNCTION.

```
const arr=[1,2,3,4,5,6];  
const arrfilter=arr.filter(num=>num>3);  
console.log(arrfilter); // // (3) [4, 5, 6]  
console.log(arr); // (6) [1, 2, 3, 4, 5, 6]
```

6 SORT()

==> THIS METHOD IS USED TO ARRANGE/SORT AN ARRAY'S ITEM EITHER ASCENDING OR DESCENDING!

7 MAP()

⇒ MAP() CREATES A NEW ARRAY FROM CALLING A FUNCTION FOR EVERY ARRAY ELEMENT.

```
const arr = [1,2,3,4,5,6];  
const add = arr.map(num => num + 1);  
console.log(add); // (6) [2, 3, 4, 5, 6, 7]  
console.log(arr); // (6) [1, 2, 3, 4, 5, 6]
```

8 ARRAY.FROM()

⇒ THE ARRAY.FROM() METHOD RETURNS AN ARRAY FROM ANY OBJECT WITH A LENGTH PROPERTY.

```
const myName = 'Sourav';  
const nameArray = Array.from(myName);  
console.log(nameArray); // ['S', 'o', 'u', 'r', 'a', 'v']
```


9 REDUCE()

⇒ THE REDUCE() METHOD APPLIES A FUNCTION AGAINST AN ACCUMULATOR AND EACH ELEMENT IN THE ARRAY.

10. ARRAY.OF()

⇒ THE ARRAY.OF() METHOD CREATES A NEW ARRAY INSTANCE FROM A VARIABLE NUMBER OF ARGUMENTS, REGARDLESS OF THE NUMBER OR TYPE OF THE ARGUMENTS.

10

Thanks for reading

DONT FORGET YO SAVE AND SHARE



@souravcode



@souravcode