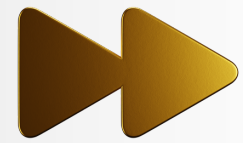
Saif Mujawar
@webbysaif



ES2023 introduces new array copying methods to JavaScript.

Saif Mujawar
@webbysaif

toSorted():

toSorted()

```
const numbers = [3, 4, 1, 5, 2];
const sortedNumbers = numbers.toSorted();

console.log(numbers); // Output: [3, 4, 1, 5, 2]
console.log(sortedNumbers); // Output: [1, 2, 3, 4, 5]
```

The toSorted() method of Array instances is the copying version of the sort() method. It returns a new array without modifying original array with the elements sorted in ascending order.

Saif Mujawar
@saifmujawar

The sort method, on the other hand, modifies the original array in place.

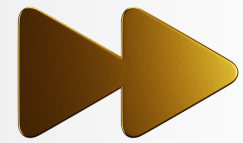
sort()

```
const numbers = [3, 4, 1, 5, 2];
numbers.sort();

console.log(numbers); // Output: [1, 2, 3, 4, 5]
```

Saif Mujawar
@saifmujawar

Saif Mujawar
@webbysaif



toReversed():

```
toReversed()

const numbers = [1, 2, 3, 4, 5];
const reversedNumbers = numbers.toReversed();

console.log(numbers); // Output: [1, 2, 3, 4, 5]
console.log(reversedNumbers); // Output: [5, 4, 3, 2, 1]
```

The toReversed() method is used to create a new array that is the reverse of the original array. The original array remains unchanged.

Saif Mujawar
@saifmujawar

On the other hand, the reverse method modifies the original array in place.

```
reversed()

const numbers = [1, 2, 3, 4, 5];
numbers.reverse();

console.log(numbers); // Output: [5, 4, 3, 2, 1]
```

Saif Mujawar
@saifmujawar

Saif Mujawar
@webbysaif

toSpliced():

The toSpliced() method of Array instances is the copying version of the splice() method. It returns a new array with some elements removed and/or replaced at a given index.

toSpliced()

```
const months = ["Jan", "Mar", "Apr", "May"];

// Inserting an element at index 1
const months2 = months.toSpliced(1, 0, "Feb");
console.log(months2); // ["Jan", "Feb", "Mar", "Apr", "May"]

// Replacing one element at index 1 with two new elements
const months4 = months3.toSpliced(1, 1, "Feb", "Mar");
console.log(months4); // ["Jan", "Feb", "Mar", "May"]

// Original array is not modified
console.log(months); // ["Jan", "Mar", "Apr", "May"]
```

Saif Mujawar
@saifmujawar

splice()

```
const months = ['Jan', 'March', 'April', 'June'];
months.splice(1, 0, 'Feb');
// Inserts at index 1
console.log(months);
// Expected output: ["Jan", "Feb", "March", "April", "June"]

months.splice(4, 1, 'May');
// Replaces 1 element at index 4
console.log(months);
// Expected output: ["Jan", "Feb", "March", "April", "May"]
```

The splice() method changes the contents of an array by removing or replacing existing elements and/or adding new elements in place.

Saif Mujawar
@saifmujawar

Saif Mujawar
@webbysaif

with():

The with() method is the copying version of using bracket notation to set the value of an element at a given index in an array. Unlike bracket notation, it doesn't mutate the array it's called upon.

It returns a new array, allowing you to chain array methods while performing manipulations without worrying about mutating the original array

with()

```
const numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9];

const replaceWith = numbers.with(1, 100);
console.log("with", replaceWith); // "with", [1, 100, 3, 4, 5, 6, 7, 8, 9]
console.log("original", numbers); // "original", [1, 2, 3, 4, 5, 6, 7, 8, 9]
```

Saif Mujawar
@saifmujawar

Without the .with() method, you would need to copy the array first, mutate the copy, and then sort the copy

Bonus

Find array from the last

This function will allow us to find element from the last to first of the array based on a condition.

getLastElement

```
const array = [{x: 1, y: 1}, {x: 2, y: 2}, {x: 3, y: 3}, {x: 4, y: 4}]

console.log(array.findLast(m => m)); //result -> {x: 4,y: 4 }

console.log(array.findLast(m => m.x * 5 === 20));
// result -> {x:4,y:4} as the condition is true so it returns the last element.

console.log(array.findLast(m => m.x * 5 === 21));
//result -> undefined as the condition is false so return undefined instead of {x:4,y:4}.

console.log(array.findLastIndex(m => m.x * 5 === 21));
// result -> -1 as the condition is not justified for returning the last element.

console.log(array.findLastIndex(m => m.x * 5 === 20));
// result -> 3 which is the index of the last element as the condition is true.
```

Saif Mujawar
@saifmujawar

Turn on notifications 🔔

And That's it!!!

Did you find it useful?

Saif Mujawar

@webbysaif

Follow for more!!