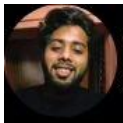

@codebysid
Siddhant Jain
16 JavaScript
Array Methods



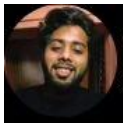
@codebysid
Siddhant Jain

```
const list = [1, 2, 3, 4, 5];
list.shift(); // 1
list; // [2, 3, 4, 5]
```

Removes the first element from an array and returns that element..

```
const list = [1, 2, 3, 4, 5];
list.unshift(0); // 6
list; // [0, 1, 2, 3, 4, 5]
```

Adds new elements to the beginning of an array, and returns the new length.



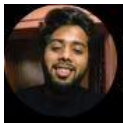
@codebysid
Siddhant Jain

```
const list = [1, 2, 3, 4];
list.map((el) => el * 2); // [2, 4, 6, 8]
```

creates a new array populated with the results of calling a provided function on every element in the calling array.

```
const list = [1, 2, 3, 4];
list.filter((el) => el % 2 === 0); // [2, 4]
```

Returns a new array with all elements that pass the test implemented by the provided function.



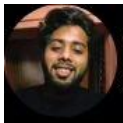
@codebysid
Siddhant Jain

```
const list = [1, 2, 3, 4, 5];
list.indexOf(3); // 2
list.indexOf(6); // -1
```

Returns the first index at which a given element can be found in the array, or -1 if it is not present.

```
const list = [1, 2, 3, 4, 5];
list.includes(3); // true
list.includes(6); // false
```

Returns true if the given element is present in the array



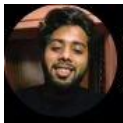
@codebysid
Siddhant Jain

```
const list = [1, 2, 3, 4, 5];
list.reduce((total, item) => total + item, 0); // 15
```

Reduce the array to a single value. The value returned by the function is stored in an accumulator (result/total).

```
const list = [1, 2, 3, 4, 5];
list.find((el) => el === 3); // 3
list.find((el) => el === 6); // undefined
```

Returns the value of the first element in the array that satisfies the provided testing function. Otherwise undefined is returned.



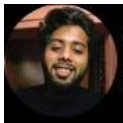
@codebysid
Siddhant Jain

```
const list = [1, 2, 3, 4, 5];
list.pop(); // 5
list; // [1, 2, 3, 4]
```

Removes the last element from an array and returns that element.

```
const list = [1, 2, 3, 4, 5];
list.push(6); // 6
list; // [1, 2, 3, 4, 5, 6]
```

Appends new elements to the end of an array, and returns the new length.



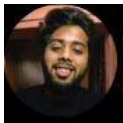
@codebysid
Siddhant Jain

```
const list = [1, 2, 3, 4, 5];
list.reverse(); // [5, 4, 3, 2, 1]
list; // [5, 4, 3, 2, 1]
```

Reverses the order of the elements in an array.

```
const list = [1, 2, 3, 4, 5];
list.some((el) => el === 3); // true
list.some((el) => el === 6); // false
```

Returns true if at least one element in the array passes the test implemented by the provided function



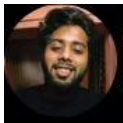
@codebysid
Siddhant Jain

```
const months = ['Jan', 'March', 'April',
'June'];
months.splice(1, 0, 'Feb');
// Inserts at index 1 , remove 0(no-
element), add 'Feb'
//Array ["Jan", "Feb", "March", "April",
"June"]
```

Changes the contents of an array by removing or replacing existing elements and/or adding new elements in place

```
const list = [1, 2, 3, 4, 5];
list.join(', '); // "1, 2, 3, 4, 5"
```

creates and returns a new string by concatenating all of the elements in this array, separated by commas or a specified separator string



@codebysid
Siddhant Jain

```
const list = [1, 2, 3, 4, 5];
list.every((el) => el === 3); // false

const list = [2, 4, 6, 8, 10];
list.every((el) => el%2 === 0); // true
```

Returns true if all elements in the array pass the test implemented by the provided function.

```
const list = [1, 2, 3, 4, 5];
list.at(1); // 2
list.at(-1); // 5
list.at(-2); // 4
```

Returns a value at the specified index.