Array Methods in JavaScript

Some of the useful methods are described below,

Method	Description
push()	Adds one or more elements to the **end** of the array.
pop()	Removes the **last** element from the array and returns it.
unshift()	Adds one or more elements to the **beginning** of the array.
shift()	Removes the **first** element from the array and returns it.
concat()	Combines two or more arrays into a **new array**.
slice()	Returns a **shallow copy** of a portion of the array (selected elements).
splice()	Adds, removes, or replaces elements in an array at a specified index.
indexOf()	Returns the **first index** of a specified element. Returns -1 if the element is not found.
lastIndexOf()	Returns the **last index** of a specified element. Returns -1 if the element is not found.
includes()	Checks if the array contains a specified element and returns true or false.
forEach()	Executes a function for each element in the array (does not return a new array).
sort()	Sorts the array **in place** (can handle strings and numbers, but needs a comparator for numbers).
reverse()	Reverses the order of the elements in the array **in place**.
fill()	Fills all or part of the array with a static value.
join()	Joins all elements of the array into a string, separated by a specified delimiter.
filter()	Creates a **new array** with elements that match a specified condition.

Examples:

1. push()

Adds elements to the end of an array.

```
let arr = [1, 2, 3];
arr.push(4);
console.log(arr); // Output: [1, 2, 3, 4]
```

---

2. pop()

Removes the last element and returns it.

```
let arr = [1, 2, 3];
let removed = arr.pop();
console.log(arr); // Output: [1, 2]
console.log(removed); // Output: 3
```

---

3. unshift()

Adds elements to the beginning of an array.

```
let arr = [1, 2, 3];
arr.unshift(0);
console.log(arr); // Output: [0, 1, 2, 3]
```

---

4. shift()

Removes the first element and returns it.

```
let arr = [1, 2, 3];
let removed = arr.shift();
console.log(arr); // Output: [2, 3]
console.log(removed); // Output: 1
```

---

5. concat()

Combines two or more arrays into a new array.

```
let arr1 = [1, 2];
let arr2 = [3, 4];
let combined = arr1.concat(arr2);
console.log(combined); // Output: [1, 2, 3, 4]
```

---

6. slice()

Returns a portion of an array as a new array.

```
let arr = [1, 2, 3, 4, 5];
let sliced = arr.slice(1, 4);
console.log(sliced); // Output: [2, 3, 4]
```

---

7. splice()

Adds, removes, or replaces elements at a specified index.

```
let arr = [1, 2, 3, 4];
arr.splice(1, 2, 'a', 'b'); // Removes 2 elements starting from index 1 and adds 'a', 'b'
console.log(arr); // Output: [1, 'a', 'b', 4]
```

---

8. indexOf()

Finds the first index of a specified element.

```
let arr = [1, 2, 3, 2];
console.log(arr.indexOf(2)); // Output: 1
```

9. lastIndexOf()

Finds the last index of a specified element.

```
let arr = [1, 2, 3, 2];
console.log(arr.lastIndexOf(2)); // Output: 3
```

10. includes()

Checks if the array contains a specified element.

```
let arr = [1, 2, 3];
console.log(arr.includes(2)); // Output: true
console.log(arr.includes(4)); // Output: false
```

11. forEach()

Executes a function for each element in the array.

```
let arr = [1, 2, 3];
arr.forEach(num => console.log(num * 2));
          (or)
arr.forEach(function(num) {
 console.log(num * 2);
});
// Output: 2, 4, 6
```

12. sort()

Sorts the array in place.

```
let arr = [3, 1, 4, 2];
arr.sort();
console.log(arr); // Output: [1, 2, 3, 4]
```

13. reverse()

Reverses the array in place.

```
let arr = [1, 2, 3];
arr.reverse();
console.log(arr); // Output: [3, 2, 1]
```

14. fill()

Fills an array with a static value.

```
let arr = [1, 2, 3];
arr.fill(0);
console.log(arr); // Output: [0, 0, 0]
```

15. join()

Joins all elements into a string.

```
let arr = ['a', 'b', 'c'];
let joined = arr.join('-');
console.log(joined); // Output: "a-b-c"
```

---

16. filter()

Creates a new array with elements that satisfy a condition.

```
let arr = [1, 2, 3, 4];
let even = arr.filter(num => num % 2 === 0);
console.log(even); // Output: [2, 4]
```

***