

# JavaScript Array Methods - Full Guide

## 1. Creation & Basic

**Array.isArray():** Checks if a value is an array.

**Example:** `Array.isArray([1,2,3]) // true`

**Array.of():** Creates an array from arguments.

**Example:** `Array.of(1,2,3) // [1,2,3]`

**Array.from():** Creates an array from iterable or array-like.

**Example:** `Array.from('abc') // ['a','b','c']`

## 2. Adding & Removing

**push():** Adds elements to the end.

**Example:** `let arr=[1]; arr.push(2); // [1,2]`

**pop():** Removes last element.

**Example:** `let arr=[1,2]; arr.pop(); // [1]`

**unshift():** Adds elements to the start.

**Example:** `let arr=[2]; arr.unshift(1); // [1,2]`

**shift():** Removes first element.

**Example:** `let arr=[1,2]; arr.shift(); // [2]`

**splice():** Adds/removes elements at specific index.

**Example:** `let arr=[1,2,3]; arr.splice(1,1,99); // [1,99,3]`

**slice():** Extracts part of array.

**Example:** `let arr=[1,2,3]; arr.slice(1,3); // [2,3]`

## 3. Searching & Checking

**includes():** Checks if array contains value.

**Example:** `[1,2,3].includes(2) // true`

**indexOf():** Returns first index of value.

**Example:** `[1,2,3].indexOf(2) // 1`

**lastIndexOf():** Returns last index of value.

**Example:** `[1,2,2].lastIndexOf(2) // 2`

**find():** Finds first element that satisfies condition.

**Example:** `[1,2,3].find(x=>x>1) // 2`

**findIndex():** Finds index of first match.

**Example:** `[1,2,3].findIndex(x=>x>1) // 1`

**some():** Checks if any element satisfies condition.

**Example:** `[1,2,3].some(x=>x>2) // true`

**every():** Checks if all elements satisfy condition.

**Example:** `[1,2,3].every(x=>x>0) // true`

## 4. Iteration

**forEach():** Executes function on each element.

**Example:** `[1,2,3].forEach(x=>console.log(x))`

**map():** Creates new array after applying function.

**Example:** `[1,2,3].map(x=>x*2) // [2,4,6]`

**filter():** Creates new array with elements passing condition.

**Example:** `[1,2,3].filter(x=>x>1) // [2,3]`

**reduce():** Reduces array to single value.

**Example:** `[1,2,3].reduce((a,b)=>a+b,0) // 6`

**reduceRight():** Same as reduce but right-to-left.

**Example:** `[1,2,3].reduceRight((a,b)=>a+b) // 6`

## 5. Sorting & Reversing

**sort():** Sorts array (by default as strings).

**Example:** `[40,1,5].sort((a,b)=>a-b) // [1,5,40]`

**reverse():** Reverses order.

**Example:** `[1,2,3].reverse() // [3,2,1]`

## 6. Flattening & Mapping

**flat():** Flattens nested arrays.

**Example:** `[1,[2,3]].flat() // [1,2,3]`

**flatMap():** Maps and flattens one level.

**Example:** `[1,2,3].flatMap(x=>[x,x*2]) // [1,2,2,4,3,6]`

## 7. Joining & Converting

**join():** Joins array into string.

**Example:** `[1,2,3].join('-') // '1-2-3'`

**toString():** Converts to string.

**Example:** `[1,2,3].toString() // '1,2,3'`

**toLocaleString():** Locale-specific string.

**Example:** `[1000].toLocaleString() // '1,000'`

## 8. Utility

**fill():** Fills array with value.

**Example:** `new Array(3).fill(0) // [0,0,0]`

**copyWithin():** Copies part of array within array.

**Example:** `[1,2,3,4].copyWithin(1,2) // [1,3,4,4]`

**keys():** Returns iterator of keys.

**Example:** `[1,2,3].keys()`

**values():** Returns iterator of values.

**Example:** `[1,2,3].values()`

**entries():** Returns iterator of [index, value].

**Example:** `[1,2].entries()`