

Method	Syntax	Description	Example
push	array.push(element)	Adds one or more elements to the end of an array and returns the new length.	let arr = [1, 2]; arr.push(3); // [1, 2, 3]
pop	array.pop()	Removes the last element from an array and returns it.	let arr = [1, 2, 3]; arr.pop(); // [1, 2]
shift	array.shift()	Removes the first element from an array and returns it.	let arr = [1, 2, 3]; arr.shift(); // [2, 3]
unshift	array.unshift(element)	Adds one or more elements to the start of an array and returns the new length.	let arr = [2, 3]; arr.unshift(1); // [1, 2, 3]
concat	array.concat(array2)	Merges two or more arrays and returns a new array.	let arr1 = [1, 2]; let arr2 = [3, 4]; arr1.concat(arr2); // [1, 2, 3, 4]
slice	array.slice(start, end)	Returns a shallow copy of a portion of an array into a new array.	let arr = [1, 2, 3, 4]; arr.slice(1, 3); // [2, 3]
splice	array.splice(start, deleteCount, ...items)	Changes the content of an array by removing or replacing elements.	let arr = [1, 2, 3]; arr.splice(1, 1, 4); // [1, 4, 3]
indexOf	array.indexOf(element)	Returns the first index of the element in the array or -1 if not found.	let arr = [1, 2, 3]; arr.indexOf(2); // 1
lastIndexOf	array.lastIndexOf(el)	Returns the last index of the element in the array or -1 if not found.	let arr = [1, 2, 3, 2]; arr.lastIndexOf(2); // 3
includes	array.includes(el)	Determines if an array contains a certain element. Returns true or false.	let arr = [1, 2, 3]; arr.includes(2); // true
reverse	array.reverse()	Reverses the order of the elements in the array in place.	let arr = [1, 2, 3]; arr.reverse(); // [3, 2, 1]
join	array.join(separator)	Joins all elements of an array into a string.	let arr = [1, 2, 3]; arr.join('-'); // '1-2-3'
sort	array.sort(compareFn)	Sorts the elements of an array in place, optionally using a custom comparator function.	let arr = [3, 1, 2]; arr.sort(); // [1, 2, 3]
flat	array.flat(depth)	Creates a new array with sub-array elements concatenated to a specified depth.	let arr = [1, [2, [3]]]; arr.flat(2); // [1, 2, 3]
fill	array.fill(value, start, end)	Fills all elements from start to end with a static value.	let arr = [1, 2, 3]; arr.fill(0, 1, 3); // [1, 0, 0]
copyWithin	array.copyWithin(target, start, end)	Copies part of the array to another location in the same array.	let arr = [1, 2, 3, 4]; arr.copyWithin(0, 2); // [3, 4, 3, 4]