JavaScript Array Methods

Method	Description	Example
length	Returns the number of elements in an array	[1, 2, 3].length // 3
push()	Adds one or more elements to the end	let a = [1]; a.push(2); // [1, 2]
pop()	Removes the last element	let a = [1, 2]; a.pop(); // [1]
shift()	Removes the first element	let a = [1, 2]; a.shift(); // [2]
unshift()	Adds one or more elements to the beginning	let a = [2]; a.unshift(1); // [1, 2]
concat()	Merges arrays	[1].concat([2, 3]) // [1, 2, 3]
join()	Joins all elements into a string	[1, 2].join('-') // '1-2'
slice()	Returns a shallow copy of a portion	[1, 2, 3].slice(1) // [2, 3]
splice()	Adds/removes elements	let a = [1, 2]; a.splice(1, 0, 3); // [1, 3, ...
indexOf()	Returns first index of element	[1, 2, 3].indexOf(2) // 1
lastIndexOf()	Returns last index of element	[1, 2, 1].lastIndexOf(1) // 2
includes()	Checks if array includes element	[1, 2, 3].includes(2) // true
forEach()	Executes function for each element	[1, 2].forEach(x => console.log(x))
map()	Creates new array by applying function	[1, 2].map(x => x * 2) // [2, 4]
filter()	Filters elements by condition	[1, 2, 3].filter(x => x > 1) // [2, 3]
reduce()	Reduces array to a single value	[1, 2, 3].reduce((a, b) => a + b) // 6
find()	Returns first matching element	[1, 2, 3].find(x => x > 1) // 2
findIndex()	Returns index of first match	[1, 2, 3].findIndex(x => x > 1) // 1
every()	Checks if all elements meet condition	[2, 4].every(x => x % 2 === 0) // true
some()	Checks if at least one meets condition	[1, 2].some(x => x > 1) // true
sort()	Sorts array in place	[3, 1, 2].sort() // [1, 2, 3]
reverse()	Reverses the array in place	[1, 2].reverse() // [2, 1]
flat()	Flattens nested arrays	[1, [2, 3]].flat() // [1, 2, 3]
flatMap()	Maps and flattens	[1, 2].flatMap(x => [x, x * 2]) // [1, 2, 2, ...

fill()	Fills array with static value	[1, 2, 3].fill(0) // [0, 0, 0]
copyWithin()	Copies part of array to another part	[1, 2, 3, 4].copyWithin(1, 2) // [1, 3, 4, 4]
toString()	Converts array to string	[1, 2].toString() // '1,2'
entries()	Returns iterator of key/value pairs	Array.from([1, 2].entries()) // [[0,1],[1,2]]
keys()	Returns iterator of keys	Array.from([1, 2].keys()) // [0, 1]
values()	Returns iterator of values	Array.from([1, 2].values()) // [1, 2]