JavaScript Array Methods

ARRAY METHOD	DESCRIPTION	SYNTAX	USE CASE
length	Returns the number of elements.	`array.length`	`[1,2,3].length // 3`
toString()	Converts array to string.	`array.toString()`	`[1,2,3].toString() // "1,2,3"`
at()	Returns element at index.	`array.at(index)`	`[10,20,30].at(-1) // 30`
join()	Joins elements with a separator.	`array.join(separator)`	`["a","b"].join("-") // "a-b"`
pop()	Removes last element.	`array.pop()`	`[1,2,3].pop() // 3`
push()	Adds elements to the end.	`array.push(item)`	`[1,2].push(3) // [1,2,3]`
shift()	Removes first element.	`array.shift()`	`[1,2,3].shift() // 1`
unshift()	Adds elements to the start.	`array.unshift(item)`	`[2,3].unshift(1) // [1,2,3]`

ARRAY METHOD	DESCRIPTION	SYNTAX	USE CASE
delete	Deletes item at index (leaves empty).	`delete array[index]`	`delete [1,2,3][1] // [1, empty, 3]`
concat()	Merges arrays.	`array1.concat(array2)`	`[1,2].concat([3,4]) // [1,2,3,4]`
copyWithin()	Copies part of array internally.	`array.copyWithin(target, start, end)`	`[1,2,3,4].copyWithin(1,2) // [1,3,4,4]`
flat()	Flattens nested arrays.	`array.flat(depth)`	`[1,[2,[3]]].flat(2) // [1,2,3]`
slice()	Returns shallow copy of portion.	`array.slice(start, end)`	`[1,2,3,4].slice(1,3) // [2,3]`
splice()	Modifies array by removing/replacing.	`array.splice(start, deleteCount, ...items)`	`[1,2,3].splice(1,1,4) // [1,4,3]`
toSpliced()	Non-destructive version of splice.	`array.toSpliced(start, deleteCount, ...items)`	`[1,2,3].toSpliced(1,1,4) // [1,4,3]`