JavaScript Array Cheat Sheet

By Web Dev Simplified   https://courses.webdevsimplified.com

Basic Methods

Name	Return Type	Description	Results
concat `[1,2,3].concat([4,5])`	Array	Concatenate the array passed to concat to the *end* of the array Add the elements 4 and 5 to the *end* of the array	1 2 3 **4 5**
slice `[1,2,3,4].slice(1, 3)`	Array	Copy part of an array starting at the index of the first parameter and stopping at the index of the second parameter, but not including the last index Copy the values in index 1 and 2 into a new array	1 **2 3** 4
reverse `[1,2,3].reverse()`	Array Modifies The Array	Reverse the order of the elements in an array	3 2 1
join `[1,2,3].join(", ")`	String	Combine each element of the array into a string with the value passed to join placed between each value Add a comma and a space between each element in the array	1, 2, 3

Array Manipulation

Name	Return Type	Description	Results
push `[1,2,3].push(4, 5)`	Number Modifies The Array *Return length of array*	Add one or more elements to the *end* of the array Add 4 and 5 to the *end* of the array	1 2 3 **4 5**
pop `[1,2,3].pop()`	Element Modifies The Array *Return removed element*	Remove the *last* element from the array Remove the value 3 from the array	1 2 3
shift `[1,2,3].shift()`	Element Modifies The Array *Return removed element*	Remove the *first* element from the array Remove the value 1 from the array	1 2 3
unshift `[1,2,3].unshift(4, 5)`	Number Modifies The Array *Return length of array*	Add one or more elements to the *start* of the array Add 4 and 5 to the *start* of the array	**4 5** 1 2 3
splice `[1,2,3].splice(1, 1, 4, 5)`	Number Modifies The Array *Return length of array*	Starting at the index specified by the first parameter, remove the number of elements specified by the second parameter and then add elements to the array for all remaining parameters at the index specified by the first parameter Starting at index 1, remove 1 element and then add the elements 4 and 5 at index 1.	1 2 **4 5** 3

Higher Order Functions

Name	Return Type	Description	Results
forEach `[1,2,3].forEach(n => console.log(n))`	Undefined	Execute the function passed to forEach for each element in the array similar to a for loop *Log out each element in the array*	None
map `[1,2,3].map(n => n * 2)`	Array	Return a new array which contains the return result of running each item through the function passed to map *Create a new array with all values doubled*	2 4 6
filter `[1,2,3].filter(n => n > 2)`	Array	Return a new array which only contains values that return true to the function passed to filter *Create a new array with only values greater than 2*	1 2 3
find `[1,2,3].find(n => n > 1)`	Element	Select the first element that returns true to the function passed to find *Get the first element greater than 1*	2
findIndex `[1,2,3].findIndex(n => n > 1)`	Number	Get the index of the first element that returns true to the function passed to find *Get the index of the first element greater than 1*	1
every `[1,2,3].every(n => n > 1)`	Boolean	Return true if every element in the array returns true from the function passed to every *Check if every element is greater than 1*	false
some `[1,2,3].some(n => n > 1)`	Boolean	Return true if at least one element in the array returns true from the function passed to some *Check if any element is greater than 1*	true
reduce `[1,2,3].reduce((sum, n) => sum + n, 0)`	Any Value	Reduce the array to one single value by starting the sum at the second value passed to reduce and updating the sum with the return value of each iteration *Sum all the numbers in the array starting with 0*	6

Basic Value Checks

Name	Return Type	Description	Results
indexOf `[1,2,1].indexOf(1)`	Number	Return the *first* index of the value passed to indexOf or return -1 if the value cannot be found *Get the first index of 1 in the array*	0
lastIndexOf `[1,2,1].lastIndexOf(1)`	Number	Return the *last* index of the value passed to indexOf or return -1 if the value cannot be found *Get the last index of 1 in the array*	2
includes `[1,2,3].includes(4)`	Boolean	Return true if the value passed to includes is in the array *Check if the value 4 is in the array*	false
length `[1,2,3].length`	Number	Return the length of the array	3