ARRAY CHEATSHEET

JAVASCRIPT

Testleaf
Always Ahead

Creation and Initialization

- *Array()*

Creates a new array instance.

Example: let browsers = new Array("Chrome", "Firefox", "Safari");

- *[]*

Creates an array using literal notation.

Example: let browsers = ["Chrome", "Firefox", "Safari"];

Adding Elements

- *push()*

Adds elements to the end of an array.

Example: browsers.push("Edge");

- *unshift()*

Adds elements to the beginning of an array.

Example: browsers.unshift("Opera");

Removing Elements

- *pop()*

Removes the last element from an array.

Example: let lastBrowser = browsers.pop();

- *shift()*

Removes the first element from an array.

Example: let firstBrowser = browsers.shift();

Finding Elements

- *indexOf()*

Finds the first index of the specified element.

Example: let index = browsers.indexOf("Firefox");

- *lastIndexOf()*

Finds the last index of the specified element.

Example: let lastIndex = browsers.lastIndexOf("Chrome");

- *find()*

Finds the first element satisfying the provided function.

Example: let found = browsers.find(browser => browser.startsWith("F"));

- *findIndex()*

Finds the index of the first element satisfying the provided function.

Example: let foundIndex = browsers.findIndex(browser => browser.startsWith("F"));

Iterating Over Arrays

- *forEach()*

Executes a function for each array element.

Example: browsers.forEach(browser => console.log(browser));

- *map()*

Creates a new array populated with the results of the function on every element.

Example: let lengths = browsers.map(browser => browser.length);

- *every()*

Checks if every element passes the test.

Example: let allLong = browsers.every(browser => browser.length > 3);

- *some()*

Checks if at least one element passes the test.

Example: let anyLong = browsers.some(browser => browser.length > 6);

Transforming Arrays

- *filter()*

Creates a new array with elements that pass the test.

Example: let longNames = browsers.filter(browser => browser.length > 5);

- *reduce()*

Reduces the array to a single value.

Example: let totalLength = browsers.reduce((total, browser) => total + browser.length, 0);

- *reduceRight()*

Applies reduce() but from right to left.

Example: let totalLengthRight = browsers.reduceRight((total, browser) => total + browser.length, 0);

- *flat()*

Flattens nested arrays.

Example: let nested = [1, [2, 3], [4, 5]].flat();

- *flatMap()*

First maps each element using a mapping function, then flattens the result.

Example: let nestedOperations = [1, 2, 3].flatMap(x => [x, x * 2]);

Ordering and Sorting

- *sort()*

Sorts the array.

Example: let numbers = [10, 2, 15, 1]; numbers.sort((a, b) => a - b);

- *reverse()*

Reverses the order of the array.

Example: browsers.reverse();

Other Utility Methods

- *slice()*

Extracts a section of an array.

Example: let newBrowsers = browsers.slice(1, 3);

- *join()*

Joins all elements into a string.

Example: let browserString = browsers.join(", ");

- *concat()*

Merges two or more arrays.

Example: let combined = browsers.concat(["Vivaldi", "Brave"]);

- *includes()*

Determines whether the array contains a certain value.

Example: let hasChrome = browsers.includes("Chrome");

- *fill()*

Fills all the elements with a static value.

Example: browsers.fill("NewBrowser");

- *copyWithin()*

Copies part of an array to another location in the same array.

Example: browsers.copyWithin(2, 0);