Methods on arrays

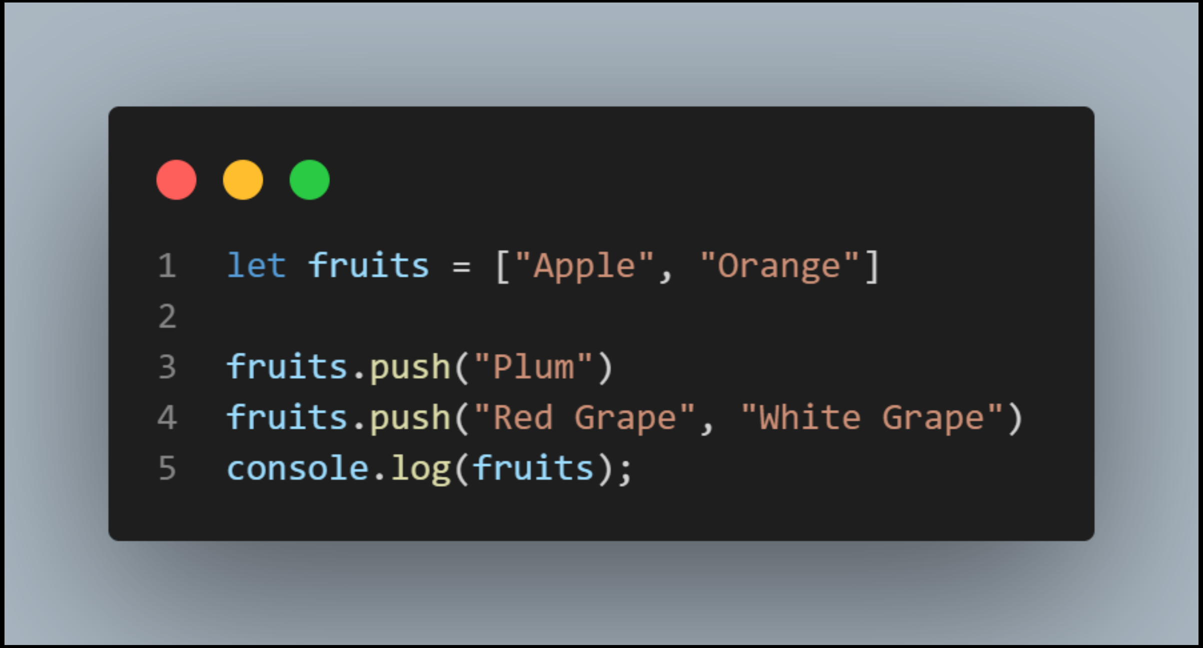
Add/Remove elements

- push(…elems) – add elements to the end
- pop() – extracts an element from the end
- shift() – extracts an element from the beginning
- unshift(…elems) – add elements to the beginning
- splice(pos, delCount, …elems) – in the pos index, deletes delCount elements and inserts elems
- slice(init,end) – creates new array and copies elements from position init to end
- concat(…elems) – returns new array: copies current members and adds elems to it

Methods on arrays

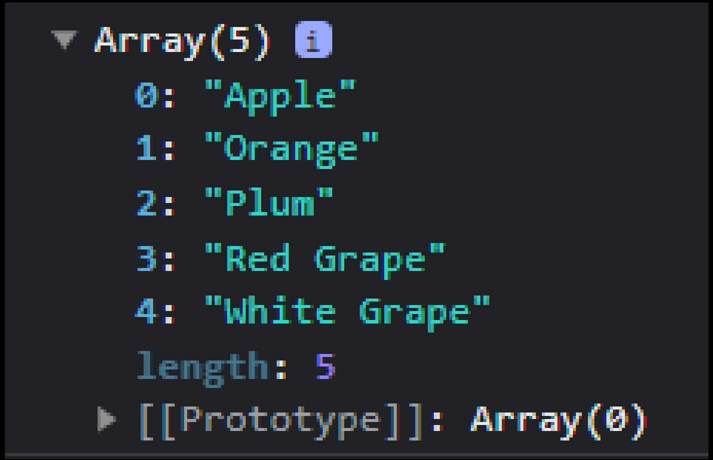
Add/Remove elements

- push(...elems)
  - add elements to the end
  - returns the current length (length) of the array



```javascript
let fruits = ["Apple", "Orange"]

fruits.push("Plum")
fruits.push("Red Grape", "White Grape")
console.log(fruits);
```

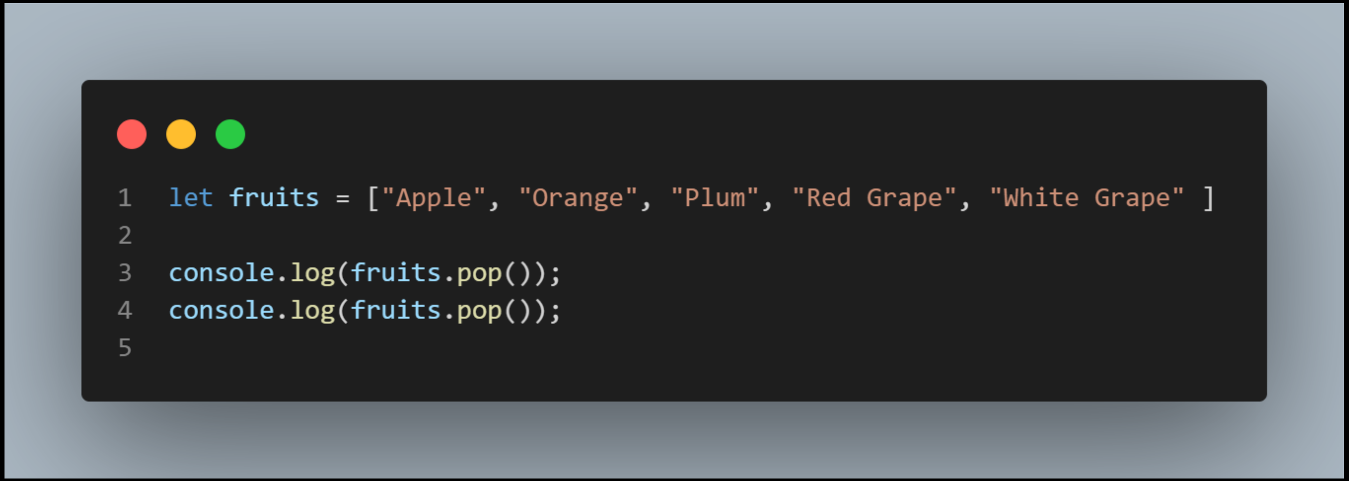


```
▼ Array(5) i
    0: "Apple"
    1: "Orange"
    2: "Plum"
    3: "Red Grape"
    4: "White Grape"
    length: 5
  ▶ [[Prototype]]: Array(0)
```

Methods on arrays

Add/Remove elements

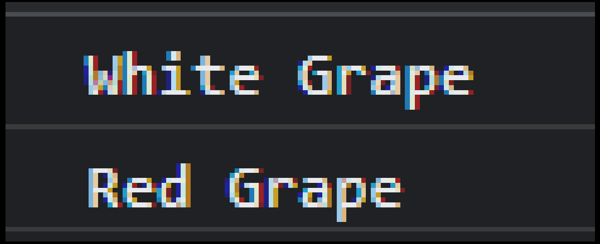
- pop( )
  - extracts an element from the end of the array
  - returns the removed element



```
let fruits = ["Apple", "Orange", "Plum", "Red Grape", "White Grape" ]

console.log(fruits.pop());
console.log(fruits.pop());

```



```
White Grape
Red Grape
```

Methods on arrays

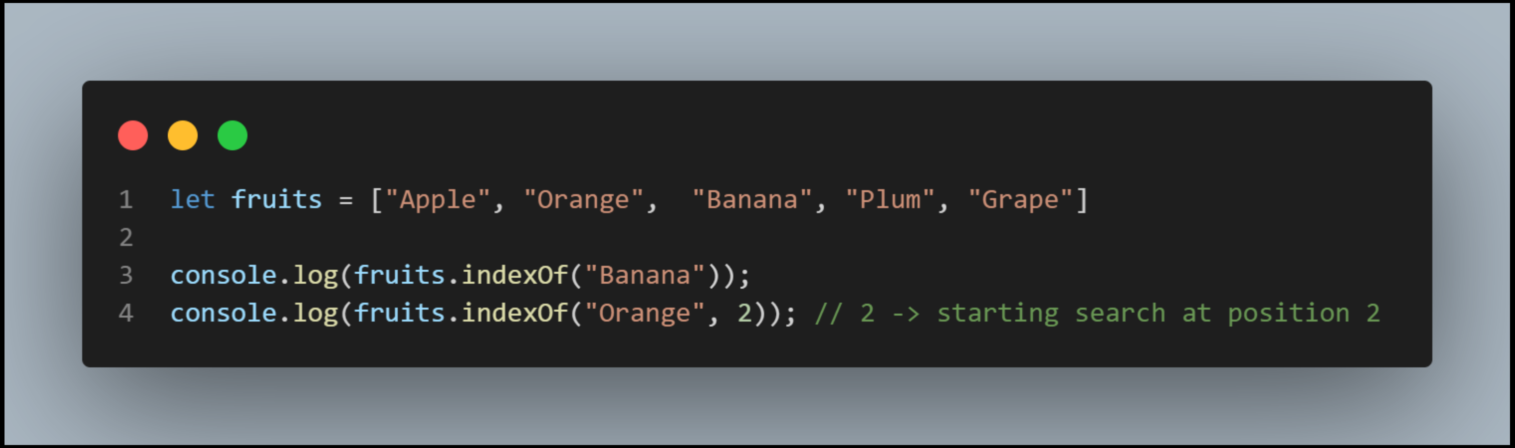
Search for elements in an array

- indexOf/lastIndexOf(elem,pos) – searches for elem starting at the pos position, and returns the index or -1 if not found
- includes(value) – returns true if the array has a value, otherwise false
- some(fn) – tests if at least one element of the array passes the test implemented by the function provided
- every(fn) – tests whether all elements of the array pass the test implemented by the function provided
- find/filter(func)– filters the elements through the function, returns the first/all values that make it return true
- findIndex(func)– it’s like find, but returns the index instead of the value

Methods on arrays

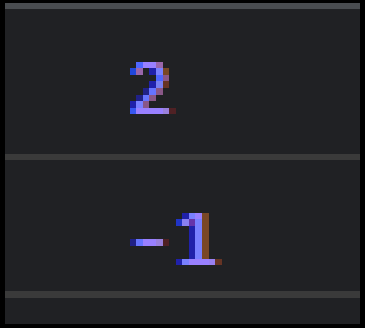
Search for elements in an array

- indexOf(elem,pos) – searches for elem starting at the pos position, and returns the index or -1 if not found
- lastIdexOf(elem,pos) – the same, but looking from the right to left



```
let fruits = ["Apple", "Orange",  "Banana", "Plum", "Grape"]

console.log(fruits.indexOf("Banana"));
console.log(fruits.indexOf("Orange", 2)); // 2 -> starting search at position 2
```

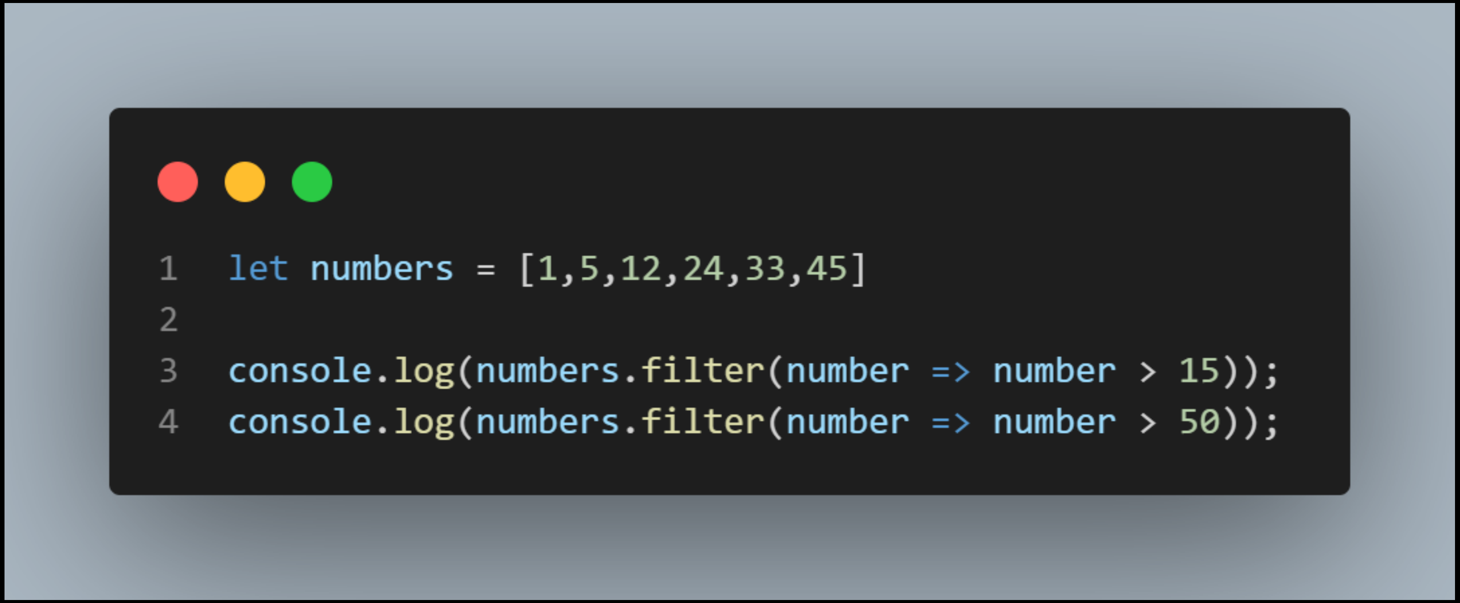


```
2
-1
```

Methods on arrays

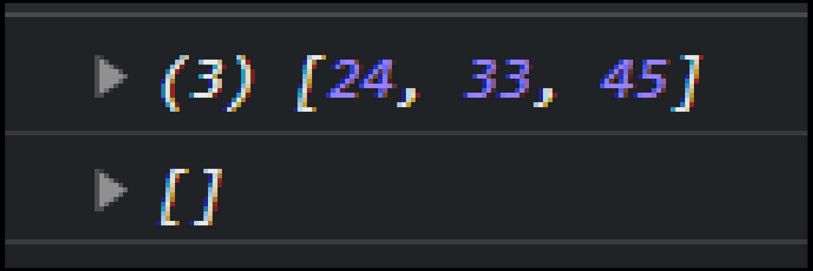
Search for elements in an array

- filter(func)
  - Filters the elements with a function
  - Returns an array with all elements that make the function return true
  - If not found, returns []



```
let numbers = [1,5,12,24,33,45]

console.log(numbers.filter(number => number > 15));
console.log(numbers.filter(number => number > 50));
```



```
▶ (3) [24, 33, 45]
▶ []
```

Methods on arrays

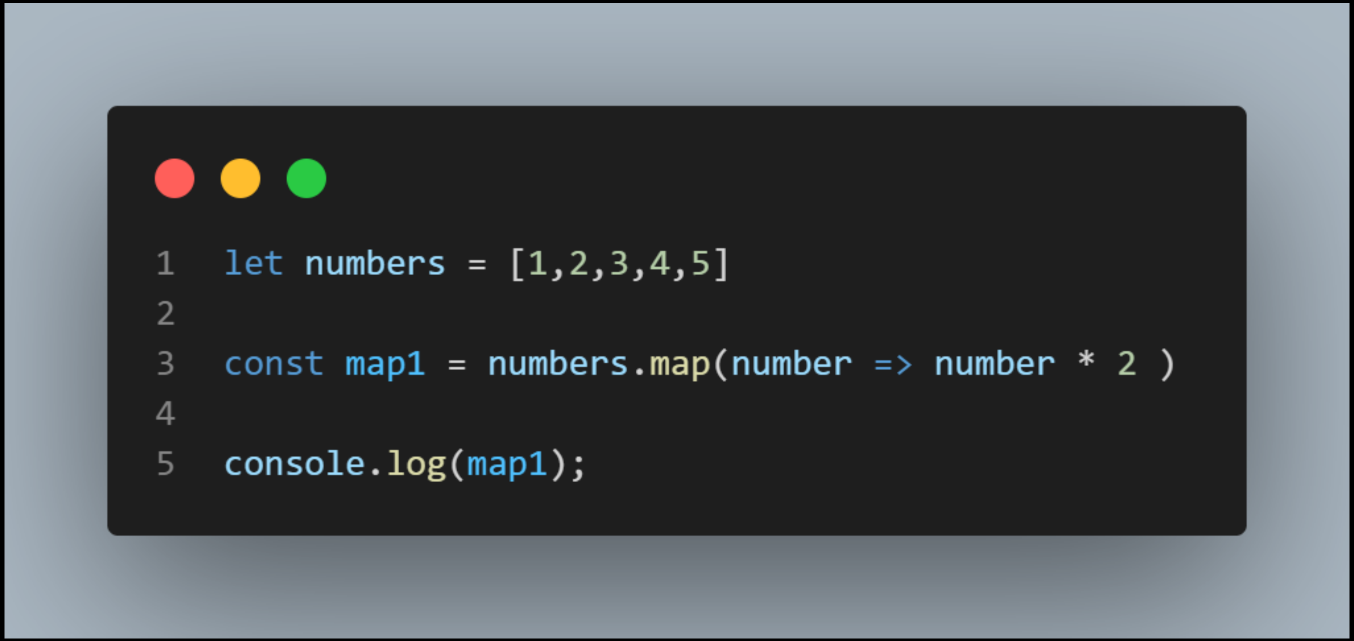
Transform an array

- map(func) – creates a new array from the results of the func call for each element of the array
- sort(func) – orders an array *(in place)*. Uses func to control ordering
- reverse() – inverts the array *in place*
- Split(sep)/join(sep) – converts a string to an array and vice verse based on sep
- reduce(func,init) – calculates a single value on the matrix calling func for each element and passing an intermediate result between calls
- fill(value,start,end) – fills the array with repeated values from the beginning to the end of the index

Methods on arrays

Transform an array

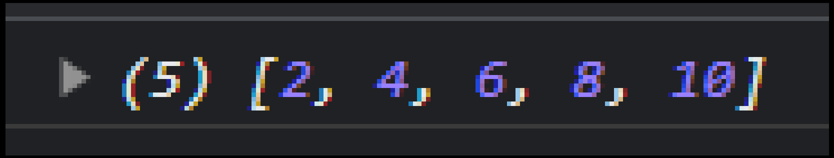
- map(func)
  - Creates a new array from the results of the func call for each element of the array



```
let numbers = [1,2,3,4,5]

const map1 = numbers.map(number => number * 2 )

console.log(map1);
```

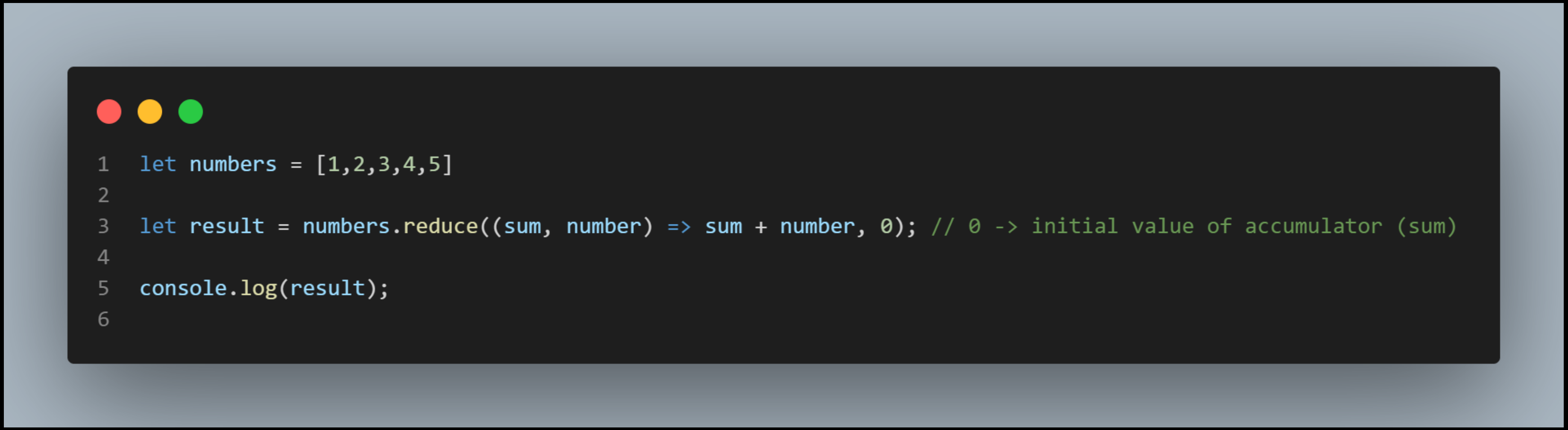


```
▶ (5) [2, 4, 6, 8, 10]
```

Methods on arrays

Transform an array

- reduce(func)
  - Calculates a single value on the matrix calling func for each element and passing an intermediate result between calls

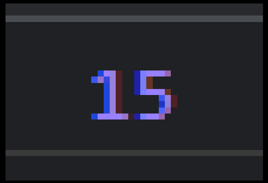


```
let numbers = [1,2,3,4,5]

let result = numbers.reduce((sum, number) => sum + number, 0); // 0 -> initial value of accumulator (sum)

console.log(result);

```



```
15
```