JavaScript Arrays - Complete Study Guide

A comprehensive guide to JavaScript arrays, including all concepts from your `index.js`, organized explanations, and real-world project problems with solutions.

Table of Contents

---

1. Array Basics

```
let arr = [1, 2, 3, 4, 5, 5];
```

- Arrays are ordered collections of values (numbers, strings, objects, etc).
- Indexing starts at 0.

---

2. Array Modification

Change Value by Index

```
arr[2] = 0; // arr is now [1,2,0,4,5,5]
```

Add Elements

```
arr.push(11); // Adds 11 to the end
```

Sort Array

```
let sr = arr.sort(function(a, b) {
    return a - b; // Ascending order
    // return b - a; // Descending order
});
```

---

3. Array Iteration Methods

forEach

```
arr.forEach(function(val) {
    console.log(val + 5); // Adds 5 to each value and prints
});
```

- forEach does not return a new array.

---

4. Array Transformation

map

```
let newArr = arr.map(function(val) {
    return (val > 10) ? val : 0;
});
console.log(newArr); // [0,0,0,0,0,0,11]
```

- map creates a new array by transforming each element.
- Must use `return` inside map's callback.

---

5. Array Searching & Filtering

filter

```
let filter = arr.filter(function(val) {
    return val > 4;
});
console.log(filter); // [5,5,11]
```

- filter returns a new array with elements that pass the test.

reduce

```
let reduce = arr.reduce(function(accumulator, val) {
    return accumulator + val;
}, 0); // Sum of all elements
```

find

```
let find = arr.find(function(val) {
    return val === 5;
});
console.log(find); // 5 (first match)
```

some

```
let some = arr.some(function(val) {
    return val > 10;
});
console.log(some); // true if any element > 10
```

every

```
let every = arr.every(function(val) {
    return val > 5;
});
console.log(every); // true if all elements > 5
```

---

6. Array Copying & Destructuring

Destructuring

```
let [a, b, , d] = arr; // a=1, b=2, d=4
```

Spread Operator

```
let arr2 = arr; // Reference, not a copy!
let arr_2 = [...arr]; // Proper shallow copy
```

---

7. Real-World Array Problems & Solutions

Problem 1: Remove Duplicates from Array

```
let numbers = [1,2,2,3,4,4,5];
let unique = [...new Set(numbers)];
console.log(unique); // [1,2,3,4,5]
```

Problem 2: Find the Most Frequent Element

```
let arr = [1,2,2,3,3,3,4];
let freq = {};
arr.forEach(val => freq[val] = (freq[val] || 0) + 1);
let max = Object.entries(freq).reduce((a,b) => a[1]>b[1]?a:b);
console.log(`Most frequent: ${max[0]} (${max[1]} times)`);
```

Problem 3: Chunk Array into Smaller Arrays

```
function chunkArray(array, size) {
    let result = [];
    for (let i = 0; i < array.length; i += size) {
        result.push(array.slice(i, i + size));
    }
    return result;
}
console.log(chunkArray([1,2,3,4,5,6,7], 3)); // [[1,2,3],[4,5,6],[7]]
```

Problem 4: Flatten a Nested Array

```
let nested = [1, [2, [3, 4], 5], 6];
let flat = nested.flat(Infinity);
console.log(flat); // [1,2,3,4,5,6]
```

Problem 5: Sum of All Even Numbers

```
let arr = [1,2,3,4,5,6];
let sumEven = arr.filter(x => x%2===0).reduce((a,b)=>a+b,0);
console.log(sumEven); // 12
```

---

8. Practice Exercises

1. **Reverse an Array**
   - Write a function to reverse an array without using `reverse()`.
2. **Find All Prime Numbers in an Array**
   - Use `filter` and a helper function.
3. **Group Objects by Property**
   - Given an array of objects, group them by a property value.
4. **Remove Falsy Values**
   - Remove all falsy values (`false`, `0`, `""`, `null`, `undefined`, `NaN`) from an array.
5. **Find Intersection of Two Arrays**
   - Return an array of elements present in both arrays.

Additional Resources

- MDN Array Reference
- JavaScript.info Arrays

Happy Coding!

Master arrays by practicing real-world problems and exploring all the methods JavaScript offers!