Start Copying From Here

🔥 JavaScript Array Cheat Sheet (Freshers Edition) 🔥

1. The Basics (बुनियादी बातें)

- **What is an Array?** Ek special variable jo ek se zyada values store kar sakta hai. (A special variable that can store more than one value).
- **How to Create?** let myArray = ["car", "bike", "bus"]; (Literal syntax is best!)
- **How to Access?** Use index. myArray[0] → "car"
- **Get Length:** myArray.length → 3
- **Last Element:** myArray[myArray.length - 1] → "bus"

2. Mutating vs. Non-Mutating Methods (सबसे ज़रूरी Concept)

This is a very common interview question.

- **Mutating (Original Array Badal Jayega):** These methods change the original array.
  - push(), pop(), shift(), unshift(), splice(), reverse(), sort()
-
- **Non-Mutating (Naya Array Milega):** These methods DO NOT change the original array. They return a new array.
  - slice(), concat(), map(), filter(), reduce(), join(), toString()
-

3. Common Methods - At a Glance

Method	Kaam Kya Hai? (What it does?)	Original Array Change Hoga?	Example
push(val)	Adds an element to the END.	✅ **Yes**	arr.push(4);
pop()	Removes an element from the END.	✅ **Yes**	arr.pop();

unshift(val)	Adds an element to the START.	✅ **Yes**	arr.unshift(1);
shift()	Removes an element from the START.	✅ **Yes**	arr.shift();
splice(start, count...)	Adds/Removes elements from the MIDDLE.	✅ **Yes**	arr.splice(1, 1, "B");
reverse()	Reverses the order of elements.	✅ **Yes**	arr.reverse();
slice(start, end)	Copies a piece of the array into a NEW array.	❌ **No**	arr.slice(1, 3);
includes(val)	Checks if a value exists (returns true/false).	❌ **No**	arr.includes(2);
indexOf(val)	Returns the first index of a value (-1 if not found).	❌ **No**	arr.indexOf(20);
join(separator)	Joins all elements into a STRING with a separator.	❌ **No**	arr.join("-");

---

4. Higher-Order Functions (Most Important for Interviews)

These functions take another function as an argument.

- **forEach(callback):** Runs a function for each element. Returns undefined.
  - let nums = [1, 2, 3];
  - nums.forEach(num => console.log(num * 2)); // Prints 2, 4, 6
-
- **map(callback):** Creates a **new array** by transforming every element.
  - let nums = [1, 2, 3];
  - let doubled = nums.map(num => num * 2); // returns [2, 4, 6]
-
- **filter(callback):** Creates a **new array** with only elements that pass a condition.
  - let nums = [1, 2, 3, 4];
  - let evens = nums.filter(num => num % 2 === 0); // returns [2, 4]
-
- **reduce(callback, initialValue):** Reduces the entire array to a **single value**.
  - let nums = [1, 2, 3, 4];
  - let sum = nums.reduce((acc, current) => acc + current, 0); // returns 10
-

🎙️ JavaScript Array Interview Questions (For Freshers)

Level 1: The Basics (Direct Questions)

1. **What is an array in JavaScript?**
   - **Answer:** An array is a single, special variable that can hold an ordered collection of multiple values. These values can be of any data type (number, string, object, etc.).
2.
3. **How do you access the first and last element of an array?**
   - **Answer:** First element: arr[0]. Last element: arr[arr.length - 1].
4.
5. **Are arrays in JavaScript objects?**
   - **Answer:** Yes. In JavaScript, arrays are technically a special type of object. typeof [] returns "object". This is why they can have properties like .length.
6.

Level 2: Method-Based Questions (What's the difference?)

1. **What is the main difference between slice() and splice()?**
   - **Answer:** splice() **changes (mutates)** the original array. slice() **does not change** the original array; it returns a new, shallow copy.
2.
3. **Difference between join() and toString()?**
   - **Answer:** toString() always uses a comma , as the separator. With join(), you can specify a custom separator, like arr.join("-").
4.
5. **What is the difference between forEach() and map()?**
   - **Answer:** forEach() just iterates and returns nothing (undefined). map() iterates, transforms elements, and **returns a new array** of the same length.
6.
7. **What does pop() return? What does push() return?**
   - **Answer:** pop() returns the **element that was removed**. push() returns the **new length of the array**.
8.

Level 3: "What will be the output?" (Code Snippets)

Generated javascript

```
    const fruits = ["Apple", "Banana"];
fruits.push("Mango");
console.log(fruits);
```

   - **Answer:** ["Apple", "Banana", "Mango"]. You can modify an array declared with const.

2.Generated javascript
```javascript
    let numbers = [10, 20, 30, 40];
let removed = numbers.splice(1, 2);
console.log(removed);      // Output 1?
console.log(numbers);      // Output 2?
```

      ○ **Answer:** Output 1: [20, 30]. Output 2: [10, 40].

3.Generated javascript
```javascript
    let arr1 = [1, 2];
let arr2 = [1, 2];
console.log(arr1 == arr2);   // Output 1?
console.log(arr1 === arr2);  // Output 2?
```

      ○ **Answer:** Both are false. Arrays are reference types; they are stored in different memory locations.

Level 4: Practical Coding Problems (Write the code)

1. **How would you reverse a string using array methods?**
   - **Answer:** str.split('').reverse().join('')
2.
3. **How do you find the largest number in an array?**
   - **Answer:** Math.max(...myArray)
4.
5. **Write a function to remove all duplicate numbers from an array.**
   - **Answer:**

Generated javascript
```javascript
    function removeDuplicates(arr) {
  return [...new Set(arr)];
}
```

📘 JavaScript Arrays Complete Fresher Cheat Sheet

🔷 What is an Array?

A special variable that can store **multiple values** in a single variable.

🔷 Creating Arrays

✅ 1. Literal

let arr = [10, 20, 30];

✅ 2. Constructor

let arr = new Array(10, 20, 30);

🔷 join() vs toString()

Feature	join()	toString()
Separator	Custom (e.g. "-")	Always comma ","
Modifies array?	No	No
Example	arr.join("-") → "1-2-3"	arr.toString() → "1,2,3

💼 Interview Q&A (Fresher Level)

✅ **Q1. What is an array?**

A collection of values in a single variable.

✅ **Q3. Get the length of an array?**

Arr.length;

✅ **Q4. Get last element?**

arr[arr.length - 1];

✅ **Q5. Add at end?**

arr.push(4);

✅ **Q6. Remove last?**

arr.pop();

✅ **Q7. Add at start?**

arr.unshift(0);

✅ **Q8. Remove first?**

arr.shift();

✅ **Q9. Check if contains value?**

arr.includes(2);

✅ **Q10. Get index of value?**

arr.indexOf(2);

✅ **Q12. What is slice()?**

Returns new sub-array without changing original.

✅ **Q14. What is reverse()?**

Reverse the original array.

✅ **Q15. What does lastIndexOf() do?**

Last index of value (searches right to left).

🧪 Practice Coding Problems

✅ **Q1. Add "grape" to end.**

let fruits = ["apple", "banana"];

fruits.push("grape");

✅ **Q2. Remove first element.**

let colors = ["red", "green", "blue"];

colors.shift();

✅ **Q3. Add "pink" at start.**

colors.unshift("pink");✅

Q4. Replace "banana" with "mango".

let fruits = ["apple", "banana", "cherry"];

fruits.splice(1, 1, "mango");

✅ **Q5. Get [20, 30] from [10,20,30,40,50].**

let numbers = [10, 20, 30, 40, 50];

let subArray = numbers.slice(1, 3);

✅ **Q6. Check if 25 is present.**

let nums = [10, 20, 30];

nums.includes(25);

✅ **Q7. Reverse array.**

nums.reverse();

✅ **Q8. Join with "-".**

let items = [1, 2, 3];

items.join("-");

🎯 Output-Based Questions

✅ **Q1.**

let arr = [1, 2, 3];

console.log(arr.length);

✅ **Q2.**

let arr = [10, 20, 30];

```
arr.push(40);

console.log(arr);
```

✅ Q3.

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

arr.pop();

console.log(arr);
```

✅ Q4.

```
let arr = ["a", "b", "c"];

console.log(arr.indexOf("b"));
```

✅ Q5.

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

console.log(arr.lastIndexOf(2));
```

✅ Q6.

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

console.log(arr.join("-"));
```

✅ Q7.

```
let arr = [10, 20, 30, 40];

console.log(arr.slice(1, 3));
```

✅ Q8.

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

arr.splice(1, 1, 99);

console.log(arr);
```

✅ Q9.

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

console.log(arr.toString());
```

✅ Q10.

```
let arr = [5, 6, 7];
```

```
arr.reverse();

console.log(arr);
```