Agenda

→ Polyfills for HOF (map, filter, reduce)

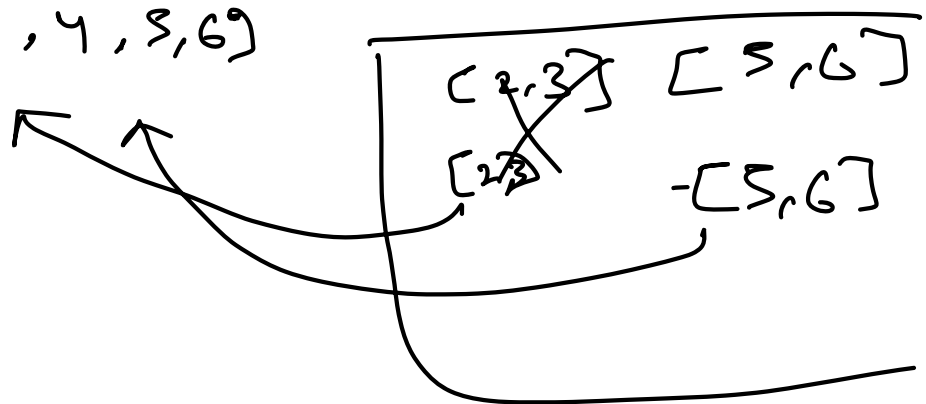→ Interview questions

```
const nestedArr = [1, [2, 3], [4, [5, 6]]];

function flattenArr(arr) {
  return arr.reduce((resultArr, item) => {
    if (Array.isArray(item)) {
      const subFlattenedArray = flattenArr(item); // Recursively flattening the nested array.
      for (let i = 0; i < subFlattenedArray.length; i++) {
        resultArr.push(subFlattenedArray[i]);
      }
    } else {
      resultArr.push(item); // Directly push it to the result array.
    }

    return resultArr;
  }, []);
}

const result = flattenArr(nestedArr); // [1, 2, 3, 4, 5, 6]
console.log(result);
```

[1 . 2, 3 , 4 , 5, 6]

[2, 3] [5, 6]

[2, 3]

- [5, 6]

🔑 Key Differences:

Aspect	Rest Operator	Spread Operator
Purpose	Gathers elements into an array	Spreads elements out individually
Usage Context	Function parameters, Destructuring	Array/Object literals, Function calls
Example Syntax	`function(a, ...args)`	`const arr = [...array]`
Data Flow	Collapses elements	Expands elements

✅ Summary:

- **Rest (`...`)**: Gathers elements into a collection.

- **Spread (`...`)**: Spreads elements out of a collection.