🗓️ **Internship Report – Day 3, Week 5**
Company: ProSensia
Intern: Hamza Rafique Awan
Topic: Array Methods in JavaScript – map(), filter(), reduce()
date: 23-7-2025

🔷 **Objective:**

To understand and implement the most commonly used array methods in JavaScript: map(), filter(), and reduce() for effective data manipulation.

🔶 **1. map() Method:**

✅ **Purpose:**

Creates a new array by applying a function to every element of the original array.

🧠 **Syntax:**

javascript

CopyEdit

array.map(callback(currentValue, index, array))

🧪 **Example:**

javascript

CopyEdit

const numbers = [1, 2, 3, 4];

const squared = numbers.map(num => num * num);

console.log(squared); // Output: [1, 4, 9, 16]

🔶 **2. filter() Method:**

✅ **Purpose:**

Returns a new array containing only elements that pass a specific test (return true).

🧠 **Syntax:**

javascript

CopyEdit

array.filter(callback(currentValue, index, array))

🧪 **Example:**

javascript

CopyEdit

const ages = [12, 17, 19, 21, 16];

const adults = ages.filter(age => age >= 18);

console.log(adults); // Output: [19, 21]

🔶 **3. reduce() Method:**

✅ **Purpose:**

Reduces the array to a single value by executing a reducer function on each element.

🧠 **Syntax:**

javascript

CopyEdit

array.reduce(callback(accumulator, currentValue, index, array), initialValue)

🧪 **Example:**

javascript

CopyEdit

const prices = [100, 200, 300];

const total = prices.reduce((sum, price) => sum + price, 0);

console.log(total); // Output: 600

📝 **Key Learnings:**

- map() is used to **transform** each element.
- filter() is used to **select** elements based on a condition.
- reduce() is used to **accumulate** or calculate a single value from multiple values.

🔚 **Conclusion:**

These array methods help write cleaner, more readable, and functional-style JavaScript. Mastering them is crucial for working with dynamic data structures in frontend or backend development.