Arrays in Detail

JavaScript array is an object that represents a collection of similar type of elements.

```
let months = ["January", "February", "March", "April"];

for (let i = 0; i < months.length; i++) {
    console.log(months[i]);
}
// January
// February
// March
// April
```

Array-Methods

```
const names = ["John", "Bob", "David", "Mark"];

// Array Push - Adds a new element containing
// the entered value to the end of the array
names.push("Shubham");
console.log(names); // [ 'John', 'Bob', 'David', 'Mark', 'Shubham' ]

// Array Pop - Deletes the last element of an array
names.pop();
console.log(names); // [ 'John', 'Bob', 'David', 'Mark' ]

// Array Shift - deletes the first element of the array
names.shift();
console.log(names); // [ 'Bob', 'David', 'Mark' ]

// Array Unshift - adds the new value to the start of the array
names.unshift("Samarth");
console.log(names); // [ 'Samarth', 'Bob', 'David', 'Mark' ]

// Array Splice - It adds or removes values in any position of an array
names.splice(2, 0, "Divyanshi", "Ayushi");
console.log(names); // [ 'Samarth', 'Bob', 'Divyanshi', 'Ayushi', 'David', 'Mark' ]

names.splice(2, 2);
console.log(names); // [ 'Samarth', 'Bob', 'David', 'Mark' ]

// Array Slice - Copies certain part of an array into a newly created array
const noOneLikesSam = names.slice(1);
console.log(noOneLikesSam); // [ 'Bob', 'David', 'Mark' ]
```

forEach method

```
const numbers = [2, 4, 6, 8];

numbers.forEach((value, i) => console.log(value, i)); // prints all the values and indexes

// when to use forEach :
// You want to do something with each element in the array

// Don't use when :
// You want to stop or break the loop when some condition is true.
// When you are working with async code.

let sum = 0;
numbers.forEach((value) => (sum += value));
console.log(sum); // 20
```

Map method

```
const inventory = [
    { price: 7, name: "egg" },
    { price: 10, name: "lays" },
    { price: 12, name: "maggie" },
];

// Array Map
const prices = inventory.map((item) => console.log(item.price));// displays only the prices
const names = inventory.map((item) => console.log(item.name)); // diplays only the names
```

Array Filter

```
const numbers = [2, 5, -2, 0, -5, 1];

const positiveNumbers = numbers.filter((number) => number >= 0);
console.log(positiveNumbers);

const negativeNumbers = numbers.filter((number) => number <= 0);
console.log(negativeNumbers);

// Another real life example
const employeesData = [
    { name: "Shubham", overtime: 5 },
    { name: "Samarth", overtime: 7 },
    { name: "Seema", overtime: 8 },
];

const employeesToReward = employeesData.filter(
    (employee) => employee.overtime >= 7
);

const employeeNames = employeesToReward.map((employee) => employee.name);

employeeNames.forEach((user)=>{
      console.log(`Congratulations, ${user}`);  // Congratulations, Samarth
                                                // Congratulations, Seema
})
```

Array Find

```
// Array Find

// The Find method for arrays returns the
// first value that satisfies the condition

const numbers = [1, 2, 3, 4, 5, 6, 7];

const value = numbers.find((number) => number > 5);
console.log(value); // 6

const cities = ["Bangalore", "Mumbai", "New Delhi", "Noida", "Hyderabad"];

const city = cities.find((city) => city.startsWith("N"));

console.log(city); // New Delhi
```

Array Includes

```
// Array Includes
const movies = ["Avengers", "Superman", "Batman"];

if (movies.includes("Avengers")) {
    console.log("The movie is available on prime");// The movie is available on prime
} else {
    console.log("The movie is not available on prime.");
}

// Note Includes method is case sensitive
```

Array Sort

```
// Array sort => Alphabetically,
// doesn't sort numbers
// This sort method mutates the original array
const names = ["Shubham", "Aditya", "Divyanshi", "Samarth"];
names.sort();
console.log(names); // [ 'Aditya', 'Divyanshi', 'Samarth', 'Shubham' ]

const numbers = [4, 12, 8, 5, 1];

// Ascending order
numbers.sort((a, b) => a - b);
console.log(numbers); // [ 1, 4, 5, 8, 12 ]

// Descending order
numbers.sort((a, b) => b - a);
console.log(numbers); // [ 12, 8, 5, 4, 1 ]
```

Some and Every

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

// Array Some => returns true if atleast one element passes the test
console.log(array.some((number) => number > 5)); // false

// Array Every => return true if all elements pass the test
console.log(array.every((number) => number > 0)); // true
```

Array Reduce

```
// Array Reduce

const groceryList = [29, 12, 45, 35, 87, 110];

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

console.log(total); // 318
```

All the code blocks above demonstrates how to use arrays and various array methods in JavaScript. An array is an object that represents a collection of similar type of elements. The code first defines an array `months` and then uses a `for` loop to iterate over the array, printing each element to the console. The code then demonstrates several array methods including `push`, `pop`, `shift`, `unshift`, `splice`, `slice`, `forEach`, `map`, `filter`, and `find`. The `push` method adds a new element to the end of the array, the `pop` method removes the last element of an array, the `shift` method removes the first element of an array, the `unshift` method adds a new value to the start of the array, the `splice` method adds or removes elements in any position of the array, the `slice` method copies a certain part of an array into a new array, the `forEach` method executes a provided function once for each array element, the `map` method creates a new array with the results of calling a provided function on every element in the array, the `filter` method creates a new array with all elements that pass the test implemented by the provided function, and the `find` method returns the first element in the array that satisfies a provided testing function.