



further enhance your understanding:

## 1. Adding/Removing Elements

- `pop()`: Removes the last element from an array and returns that element. Example:

```
let f
```

`unshift()` : Adds one or more elements to the beginning of an array and returns the new length of the array. Example:

- ```
fruits.unshift('apple'); // fruits is now ['apple', 'banana', 'orange']
```

javascrip

**Accessing/Searchng Elements**

- ```
let fruits = ['apple', 'banana', 'orange'];
let index = fruits.indexOf('banana'); // index is 1
```

```
let f
```

`includes()` : Checks if an array contains a specific element. Example:

- ### 3. Iterating/Transforming Arrays

javascript

```
fruits.forEach(fruit => console.log(fruit)); // logs
'orange'

map() : Creates a new array with the results of calling a provided function on every element.

Example:

javascript
let numbers = [1, 2, 3];
let squares = numbers.map(num => num * num); // squares = [1, 4, 9]

filter() : Creates a new array with all elements that pass the provided function. Example:

javascript
let numbers = [1, 2, 3, 4, 5];
let evens = numbers.filter(num => num % 2 === 0);

reduce() : Applies a function against an accumulator and each element
```

```
let numbers = [1, 2, 3, 4];
let sum = numbers.reduce((to
```

- ## 4. Manipulating Array Structure

## javascript

`splice()` : Changes the contents of an array by removing or replacing existing elements and/or adding new elements. Example:

- ```
let fruits = ['apple', 'banana', 'orange'];
fruits.splice(1, 1, 'kiwi'); // removes 'banana', adds 'kiwi'; fruits is now
['apple', 'kiwi', 'orange']
```

javascript

```
let allFruits = fruits1.concat(fruits2); // allFruits is ['apple', 'banana', 'orange', 'kiwi']
```

- ```
javascript

let numbers = [3, 1, 4, 1, 5];
numbers.sort(); // numbers is [1, 1, 3, 4, 5]
```

reverse

```
let numbers = [1, 2, 3];
numbers.reverse(); // numbers is [3, 2, 1]
```

```
javascript
```

## javascript

`find()` : Returns the first element that satisfies the provided testing function. Example:

- ```
let firstLarge = numbers.find(num => num > 10); // firstLarge is 16
```

javascript

```
let index = numbers.findIndex(num => num > 10); // index is
```