

JavaScript Arrays

Introduction to Arrays

- Arrays are used to store multiple values in a single variable.
- Each value in an array is called an element.
- Arrays are zero-indexed, meaning the first element is at index 0.

```
let fruits = ["Apple", "Banana", "Cherry"];  
console.log(fruits[0]); // Output: Apple
```

Creating Arrays

- Arrays can be created using the array literal syntax `[]`.

```
let numbers = [1, 2, 3, 4, 5];
```

- Arrays can also be created using the `Array` constructor.

```
let colors = new Array("Red", "Green", "Blue");
```

Accessing Array Elements

- Access elements using their index.

```
let animals = ["Dog", "Cat", "Elephant"];  
console.log(animals[1]); // Output: Cat
```

- Modify elements by assigning a new value to an index.

```
animals[2] = "Lion";  
console.log(animals); // Output: ["Dog", "Cat", "Lion"]
```

Array Methods

- `push()` : Adds one or more elements to the end of an array.

```
let fruits = ["Apple", "Banana"];  
fruits.push("Cherry");  
console.log(fruits); // Output: ["Apple", "Banana", "Cherry"]
```

- `pop()` : Removes the last element from an array.

```
fruits.pop();  
console.log(fruits); // Output: ["Apple", "Banana"]
```

Looping Through Arrays

- Use `for` loop to iterate over array elements.

```
let numbers = [1, 2, 3, 4, 5];  
for (let i = 0; i < numbers.length; i++) {  
    console.log(numbers[i]);  
}
```

- Use `for ... of` loop for a more concise syntax.

```
for (let number of numbers) {  
    console.log(number);  
}
```

Practice: Sum of Array Elements

- Write a function to calculate the sum of all elements in an array.

```
function sumArray(arr) {  
  let sum = 0;  
  for (let num of arr) {  
    sum += num;  
  }  
  return sum;  
}
```

```
let numbers = [1, 2, 3, 4, 5];  
console.log(sumArray(numbers)); // Output: 15
```

Array `map()` Method

- `map()` creates a new array with the results of calling a function on every element.

```
let numbers = [1, 2, 3, 4, 5];  
let squared = numbers.map(num ⇒ num * num);  
console.log(squared); // Output: [1, 4, 9, 16, 25]
```


Array `filter()` Method

- `filter()` creates a new array with all elements that pass the test implemented by the provided function.

```
let numbers = [1, 2, 3, 4, 5];  
let evenNumbers = numbers.filter(num => num % 2 === 0);  
console.log(evenNumbers); // Output: [2, 4]
```

Array `reduce()` Method

- `reduce()` executes a reducer function on each element, resulting in a single output value.

```
let numbers = [1, 2, 3, 4, 5];  
let sum = numbers.reduce((accumulator, currentValue) => accumulator + currentValue, 0);  
console.log(sum); // Output: 15
```

Practice: Find Maximum Element

- Write a function to find the maximum element in an array.

```
function findMax(arr) {  
    return arr.reduce((max, num) => (num > max ? num : max), arr[0]);  
}
```

```
let numbers = [1, 2, 3, 4, 5];  
console.log(findMax(numbers)); // Output: 5
```

Conclusion

- Arrays are a fundamental data structure in JavaScript.
- They provide various methods to manipulate and iterate over elements.
- Practice using arrays and their methods to become proficient.

Further Reading

- MDN Web Docs: Arrays
- JavaScript Info: Arrays