



Advanced Data Types

Software Development Bootcamp

Understanding Arrays in JavaScript



Topic

Arrays



What Are Arrays?

- An array stores multiple values in one place
- Each value is called an element
- Elements are accessed by their position (index)
- Indexes start at 0



Array Syntax

- Arrays are created using square brackets `[]`
- In this example the `fruits` array holds three strings.

```
let fruits = ['apple', 'banana', 'cherry']
```



Array Indexes

- Arrays are indexed starting at 0.
- Use indexes to refer to individual values.
- `fruits[0]` gets the first element in the `fruits` array

```
let fruits = ['apple', 'banana',  
             'cherry']  
let numbers = [1, 2, 3, 4, 5]  
let mix = ['hello', 89, 'world',  
           true, 98, false]
```



Length Property

- The `length` property tells you how many elements are in an array.
- It can be used to add or remove elements from the end of an array
- It can also be used to get the last value in an array

```
let fruits = ['apple',  
             'banana', 'cherry']  
  
// output is 3  
  
fruits.length  
  
// output is 'cherry'  
  
fruits[fruits.length - 1]
```



Adding Values to an Array

- We can use square bracket notation to add, or “write” values to an array

```
let fruits = ['apple', 'banana',  
             'cherry']  
// fruits is now ['apple',  
                 'banana', 'cherry', 'orange']  
fruits[3] = 'orange'  
// fruits is now ['grape',  
                 'banana', 'cherry', 'orange']  
fruits[0] = 'grape'
```



Why We Use Arrays

- **Efficient storage:** Store multiple related items in a single variable
- **Easy access:** Quickly retrieve or modify any element using its index
- **Iteration:** Easily loop through all elements for processing
- **Organization:** Group related data together (e.g., list of users, scores in a game)



Topic

Array Methods



What Are Array Methods?

Array methods are special tools or actions you can use to with arrays. They help you:

- Add elements to an array
- Remove elements from an array
- Find specific element(s) in an array



Simple Array Methods

- `.push()`: Adds elements to the end of an array
- `.pop()`: Removes elements from the end of an array
- `.shift()`: Removes the first element from an array
- `.unshift()`: Adds a new element to the beginning of an array.

```
let fruits = ['apple', 'banana',  
             'cherry']  
// .push() method takes the string  
'pineapple' as an argument  
fruits.push('pineapple')  
// .pop() method doesn't take an  
argument, just removes the last element  
in the array  
fruits.pop()  
// .shift() method doesn't take an  
argument, just removes the first  
element in the array  
fruits.shift()  
// .unshift method takes the string  
'kiwi' as an argument  
fruits.unshift('kiwi')
```



Slice Method

- The `slice()` method copies a part of the array and creates a new array without changing the original array.

```
let fruits = ["apple", "banana",  
"cherry", "date", "elderberry"];  
let someFruits = fruits.slice(1, 4)  
// output: ["apple", "banana",  
"cherry", "date", "elderberry"]  
console.log(fruits)  
// output: ["banana", "cherry",  
"date"]  
console.log(someFruits)
```



Join Method

- The `join()` method combines array elements into a string, using a specified separator

```
let words = ['Hello', 'world',  
             'JavaScript', 'is', 'fun']  
  
// separator given to the join  
method is an empty space  
  
let sentence = words.join(' ')  
  
// output: "Hello world JavaScript  
is fun"  
  
console.log(sentence)
```

Splice Method

- The `splice()` method can add new elements to an array, remove elements, or both. `splice()` changes the original array.

```
let fruits = ['apple', 'banana',  
             'cherry', 'date']  
  
// fruits array is now ['apple', 'date']  
fruits.splice(1, 2);  
  
// fruits array is now ['apple',  
                        'blueberry', 'kiwi', 'date']  
fruits.splice(1, 0, 'blueberry', 'kiwi')  
  
// fruits array is now ['apple',  
                        'blueberry', 'mango', 'pear', 'date']  
fruits.splice(2, 1, 'mango', 'pear')
```



Split Method

- The `split()` method divides a string into smaller pieces and puts those pieces into an array.

```
let sentence = 'I love learning  
JavaScript'  
// call the split method and use "  
" (space) as the separator  
let words = sentence.split(" ")  
// value of words is ['I', 'love',  
'learning', 'JavaScript']
```



Why We Use Array Methods?

- **Simplicity:** Perform complex operations with a single line of code
- **Readability:** Make code more understandable and self-explanatory
- **Chaining:** Many methods can be chained together for powerful data transformations
- **Immutability:** Some methods create new arrays, leaving the original unchanged, which is useful for maintaining data integrity



Exercise

Fibonacci Stack