



Data Types

By

Mulia Arifandi Nasution

 mul14



Data Types in JavaScript

- Primary Data Types

- boolean / Boolean
- string / String
- number / Number

- Special Data Types

- null
- undefined
- function

- Composite Data Types (Reference)

- object / Object
- array / Array
- Symbol

- Primitive Data Types

- boolean
- null
- undefined
- number
- string
- Symbol (new in ECMAScript 2015)

Primitive Data Types

A primitive is data that is not an object and has no methods.

Variables

var, let, const.

Primary Data Types

Primary Data Types

- String
- Number
- Boolean

String

Strings Declaration

```
let a = `Hello World`
```

```
let b = 'Hello Universe'
```

```
let c = "Hello JavaScript"
```

```
let d = new String('This is a string')
```

Concatenation

```
const first_name = 'Mulia'  
  
const last_name = 'Nasution'  
  
const full_name = first_name + ' ' + last_name  
  
console.log(full_name)
```

Split strings into array

```
const text = 'Lorem ipsum dolor sit amet'  
  
splitted_text = text.split(' ')  
  
console.log(splitted_text)
```

Convert array into a string

```
const list = ['Rasa', 'sate', 'ini', 'enak']  
  
const text = list.join(' ')  
  
console.log(text)
```

Uppercase, Lowercase, and Capitalize.

```
const text = 'kelapa sawit'  
  
console.log(text.toUpperCase())  
  
console.log(text.toLowerCase())  
  
const caps = text.toLowerCase()  
    .replace(/\b\w/g, (letter) => letter.toUpperCase())  
  
console.log(caps)
```

Find a string in a string - includes

```
const text = 'Saya ingin makan apel merah'  
const find = 'apel'  
  
console.log(text.includes(find))
```

Find a string in a string - startsWith

```
const text = 'Saya ingin makan apel merah'  
const find = 'Saya'  
  
console.log(text.startsWith(find))
```


Find a string in a string - endsWith

```
const text = 'Saya ingin makan apel merah'  
const find = 'merah'  
  
console.log(text.endsWith(find))
```

Number

Numbers Declaration

```
const first = 1
```

```
const last = new Number(88)
```

Number

- 10
- 19.84
- 0733
- 0o644
- 0x12
- 0b1101
- NaN (Not a Number)
- Infinity/-Infinity

Combine Number and String

```
const price = 103494  
  
const currency = 'Rp'  
  
const amount = currency + price  
  
console.log(amount)
```

Boolean

Boolean - Truthy and Falsy

- Truthy

- 1
- -1
- '1'
- 'abc'
- ' '
- true
- {}
- []
- Infinity / -Infinity
- function(){} / () => {}

- Falsy

- false
- ''
- 0
- null
- undefined
- NaN

Special Data Types

Special Data Types

- null
- undefined
- function

Collective Data Types

Collective Data Types (Reference)

- Object
- Array

Object

Object Declaration

```
const a = {}
```

```
const b = new Object()
```

```
const c = { a: 1 }
```

```
const d = { a: 1, b: 4 }
```

Access value of an Object

```
const shirt = { color: 'red', size: 'L' }  
  
shirt.color      // red  
  
shirt['size']    // L
```

Assign value to an Object

```
const blazer = {}  
  
const jacket = blazer  
  
jacket.color = 'black'  
  
jacket['size'] = 'L'  
  
console.log(blazer)
```

Trick to Clone an Object

```
const blazer = { color: 'red', size: 'L' }  
  
const jacket = JSON.parse(JSON.stringify(blazer))  
  
jacket.color = 'blue'  
  
console.log(blazer)  
  
console.log(jacket)
```


Object Iteration

```
const person = { name: 'Mulia', age: 14, blood: 'B' }  
  
for (let key in person) {  
    console.log(key)    // name, age, blood  
}
```

Object Iteration

```
const person = { name: 'Mulia', age: 14, blood: 'B' }  
  
for (let key in person) {  
    console.log(person[key])    // Mulia, 14, B  
}
```

Array

Arrays Declaration

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

```
const cities = ['Jakarta', 'Bandung', 'Medan', 'Jogja']
```

```
const foods = new Array('Bakso', 'Sate', 'Martabak')
```

Access an Array

```
const cities = ['Jakarta', 'Bandung', 'Medan', 'Jogja']  
  
cities[0]      // Jakarta  
  
cities[1]      // Bandung  
  
cities[2]      // Medan
```

Array Manipulation

```
const cities = ['Jakarta', 'Bandung', 'Medan', 'Jogja']  
  
cities[0] = 'Balikpapan'  
  
console.log(cities)
```

Array Manipulation: pop

```
const cities = ['Jakarta', 'Bandung', 'Medan']  
  
const p = cities.pop()  
  
console.log(cities)  
  
console.log(p)           // Medan
```

Array Manipulation: shift

```
const cities = ['Jakarta', 'Bandung', 'Medan']  
  
const p = cities.shift()  
  
console.log(cities)  
  
console.log(p)           // Jakarta
```


Array Manipulation: concat

```
const fruits = ['Rambutan', 'Jeruk', 'Anggur']  
const vegetables = ['Sawi', 'Kubis', 'Bayam']  
const foods = fruits.concat(vegetables)  
console.log(foods)
```

Array Manipulation: push

```
const fruits = ['Rambutan', 'Jeruk', 'Anggur']  
  
fruits.push('Ubi')  
  
console.log(fruits)
```

Array Manipulation: unshift

```
const fruits = ['Rambutan', 'Jeruk', 'Anggur']  
  
fruits.unshift('Ubi')  
  
console.log(fruits)
```

Array Manipulation: sort

```
const fruits = ['Rambutan', 'Jeruk', 'Anggur']  
  
fruits.sort()  
  
console.log(fruits)
```

Array Manipulation: reverse

```
const fruits = ['Rambutan', 'Jeruk', 'Anggur']  
  
fruits.sort().reverse()  
  
console.log(fruits)
```

Array Manipulation: indexOf

```
const fruits = ['Rambutan', 'Jeruk', 'Anggur']  
  
const index = fruits.indexOf('Jeruk')  
  
console.log(index)
```

Array Manipulation: splice

```
const fruits = ['Rambutan', 'Jeruk', 'Anggur']  
  
const index = fruits.indexOf('Jeruk')  
  
fruits.splice(index, 1)  
  
console.log(fruits)
```

Loop in an Array: for

```
const cities = ['Jakarta', 'Bandung', 'Medan']  
  
for (let i = 0; i < cities.length; i++) {  
    console.log(cities[i])  
}
```


Loop in an Array: for-of

```
const cities = ['Jakarta', 'Bandung', 'Medan']  
  
for (let city of cities) {  
    console.log(city)    // Jakarta, Bandung, Medan  
}
```

Loop in an Array: for-in

```
const cities = ['Jakarta', 'Bandung', 'Medan']  
  
for (let index in cities) {  
    console.log(index)    // 0, 1, 2  
}
```

Complex Array

```
const purchases = [  
  { items: ['Pisang', 'Apel'], total: 1000, onSale: false },  
  { items: ['Cabe'], price: 450, onSale: true },  
  { items: ['Bawang'], price: 200, onSale: false },  
]
```

Questions ... please.