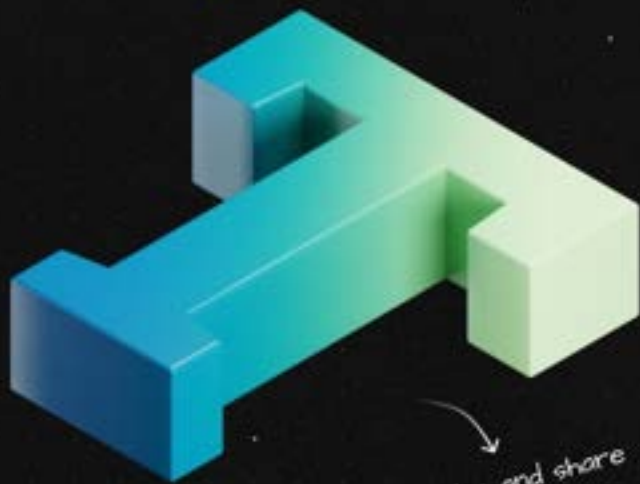


JS

String Methods

in **JavaScript**



like and share



Why do we use String Methods?

In JavaScript, strings are used to represent and work with a sequence of characters. A string can represent **an object** as well as the **primitive data type**. JavaScript **automatically converts** primitive strings to String objects so that it's possible to use String methods and access properties even for primitive strings.



Basics - 1

1. toUpperCase()


Converts all string characters to uppercase.

2. toLowerCase()

Converts all string characters to lowercase.

3. concat()

Used for concatenation of two strings.



```
let Str = "Code with Sloba";  
console.log(Str.toUpperCase());  
// CODE WITH SLOBA  
console.log(Str.toLowerCase());  
// code with sloba  
console.log(Str.concat(", subscribe  
for more"));  
// Code with Sloba, subscribe for more
```



Basics - 2

4. `replace()`

It only replaces the first match, and return new replaced string.

5. `trim()`

Used to remove whitespace from either sides of a string.

6. `split()`

Used to convert a string to an array.

```
let Str = "Code with Slobodan";  
console.log(Str.replace('Slobodan', 'Sloba'));  
// Code with Sloba  
console.log(Str.trim('   Code   with Sloba   '));  
// Code   with Sloba  
console.log(Str.split(' '));  
// 'Code', 'with', 'Slobodan'
```



String Length

1. Length

Returns the length of a string.



style.css

```
let myName = 'CodewithSlobo'  
const length = myName.length;  
console.log(length);  
//13
```



Finding Substring

1. indexOf()

A high-order function is a function that receives another **function as a parameter** or returns a **function as a return value**.

2. lastIndexOf()

this method traverses from **end to start**.

3. search()

you cannot pass a second parameter.

```
style.css

let myBio = "I am a React Developer";

console.log(myBio.indexOf('React'));
// 7

console.log(myBio.lastIndexOf('ac',14));
// 9

console.log(myBio.search('name'));
// -1
```



Extracting Substring

1. slice()

return the substring as a new string.

two parameter --> start position, End position.

2. substring()

start and end values less than 0 are treated as 0

3. substr()

In this case, second parameter specifies the length of extracted string.



```
let text = "Red, Green, Blue";  
console.log(text.slice(5, 10));  
// Green  
console.log(text.substring(5, 16));  
// Green, Blue  
console.log(text.substr(5,5));  
// Green
```



Extracting Character

1. `charAt()`


return the character at a specified index in the string, if it's not available then return nothing.

2. `charCodeAt()`

return the Unicode value of character. If its unavailable then return NAN.

3. `propertyAccess[]`

ECMAScript 5 allow `propertyAccess[]` on string



```
let text = "Code with Sloba";  
console.log(text.charAt(8));  
// h  
console.log(text.charCodeAt(10));  
// 104  
console.log(text[0]);  
// C
```



Do you find it helpful?


let me know down in the
comments !



Slobodan Gajić

Content Creator



Code with  **Sloba**

FOLLOW FOR MORE