



Sayyed Siddique

[sayyed-siddique-834245255](#)

Mastering in **String** Data Structure in **Javascript**

1 String Creation

Strings can be created using single quotes ('), double quotes ("), or backticks (`) for template literals

```
let singleQuote = 'Hello';  
let doubleQuote = "World";  
let templateLiteral = `Hello, ${singleQuote}!`;
```



Sayyed Siddique

[sayyed-siddique-834245255](#)



2 Immutability

Once a string is created, its value cannot be changed. Any operations on a string will create a new string.

```
let str = "Hello";  
str[0] = "h"; // This will not change the string
```



Sayyed Siddique

[sayyed-siddique-834245255](#)



3 Concatenation

Strings can be concatenated using the + operator or the concat method.

```
let str1 = "Hello";  
let str2 = "World";  
let concatenated = str1 + " " + str2; // "Hello World"  
let concatenated2 = str1.concat(" ", str2); // "Hello World"
```



Sayyed Siddique

[sayyed-siddique-834245255](#)



4 Accessing Characters

You can access individual characters in a string using bracket notation or the `charAt` method.

```
let str = "Hello";  
console.log(str[0]); // "H"  
console.log(str.charAt(0)); // "H"
```



Sayyed Siddique

[sayyed-siddique-834245255](#)



5 Substring

Extract a part of the string using substring, substr, or slice.

```
let str = "Hello World";  
console.log(str.substring(0, 5)); // "Hello"  
console.log(str.substr(0, 5)); // "Hello"  
console.log(str.slice(0, 5)); // "Hello"
```



Sayyed Siddique

[sayyed-siddique-834245255](#)



6 Searching

Find the position of a substring using `indexOf` or `lastIndexOf`.

```
let str = "Hello World";  
console.log(str.indexOf("World")); // 6  
console.log(str.lastIndexOf("o")); // 7
```



Sayyed Siddique

[sayyed-siddique-834245255](#)



7 Replacing

Replace parts of the string using replace.

```
let str = "Hello World";  
let newStr = str.replace("World", "JavaScript"); // "Hello JavaScript"
```



Sayyed Siddique

[sayyed-siddique-834245255](#)



8 Case Conversion

Convert to uppercase or lowercase using `toUpperCase` and `toLowerCase`.

```
let str = "Hello World";  
console.log(str.toUpperCase()); // "HELLO WORLD"  
console.log(str.toLowerCase()); // "hello world"
```



Sayyed Siddique

[sayyed-siddique-834245255](#)





Splitting and Joining

Split a string into an array of substrings using `split` and join an array of strings into a single string using `join`.

```
let str = "Hello World";  
let arr = str.split(" "); // ["Hello", "World"]  
let joined = arr.join(", "); // "Hello, World"
```



Sayyed Siddique

[sayyed-siddique-834245255](#)



10 Trimming

Remove whitespace from both ends of a string using trim.

```
let str = "  Hello World  ";  
console.log(str.trim()); // "Hello World"
```



Sayyed Siddique

[sayyed-siddique-834245255](#)



10 Trimming

Remove whitespace from both ends of a string using trim.

```
let str = "  Hello World  ";  
console.log(str.trim()); // "Hello World"
```



Sayyed Siddique

[sayyed-siddique-834245255](#)



11 Template Literals

Template literals allow for multi-line strings and expression interpolation.

```
let name = "John";  
let greeting = `Hello, ${name}!`;  
console.log(greeting); // "Hello, John!"  
  
let multiLine = `This is  
a multi-line  
string.`;  
console.log(multiLine);
```



Sayyed Siddique

[sayyed-siddique-834245255](#)



12 Reversing a String

```
let str = "Hello World";  
let reversed = str.split('').reverse().join('');  
console.log(reversed); // "dlrow olleH"
```



Sayyed Siddique

[sayyed-siddique-834245255](#)



13 Checking for Palindrome

```
function isPalindrome(str) {  
  let cleaned = str.replace(/[^a-zA-Z0-9]/g, '').toLowerCase();  
  let reversed = cleaned.split('').reverse().join('');  
  return cleaned === reversed;  
}  
  
console.log(isPalindrome("A man, a plan, a canal: Panama")); // true  
console.log(isPalindrome("Hello")); // false
```



Sayyed Siddique

[sayyed-siddique-834245255](#)





Sayyed Siddique

[sayyed-siddique-834245255](#)

Thanks for reading

Engage with Us!

**Share your thoughts on
DSA in frontend
development.**