



# JavaScript

---

# String Methods

---

JavaScript String methods with examples



# JavaScript String Methods

## at() Method

The `at()` method takes an index and returns the character at that index, accommodating negative indices to count back from the end of the string.



```
1  const str = 'hello';
2  // Using positive index
3  console.log(str.at(0)); // Output: "h"
4  // Using negative index
5  console.log(str.at(-1)); // Output: "o"
```

# JavaScript String Methods

## charAt() Method

The charAt() method is used to get the character at a specified index.



```
1  const str = 'hello';  
2  console.log(str.charAt(1));  // Output: "e"
```



# JavaScript String Methods

## charCodeAt() Method

The `charCodeAt()` method returns the Unicode of the character at the specified index.



```
1  const str = "hello";  
2  console.log(str.charCodeAt(0)); // 104
```



# JavaScript String Methods

## codePointAt() Method

The `codePointAt()` method returns a non-negative integer that is the Unicode code point value.



```
1  const str = "hello";  
2  console.log(str.codePointAt(0)); // 104
```



# JavaScript String Methods

## concat() Method

The `concat()` method is used to concatenate one or more strings to the existing one and returns the new concatenated string.



```
1  const str1 = 'Hello';  
2  const str2 = ' world!';  
3  console.log(str1.concat(str2)); // Output: "Hello world!"
```



# JavaScript String Methods

## endsWith() Method

The `endsWith()` method checks if the string ends with the specified string or characters.



```
1  const str = 'hello';  
2  console.log(str.endsWith('o'));    // Output: true  
3  console.log(str.endsWith('l'));    // Output: false
```



# JavaScript String Methods

## includes() Method

The includes() method checks if the string contains a specified string or characters.



```
1  const str = 'hello';  
2  console.log(str.includes('ll'));    // Output: true  
3  console.log(str.includes('z'));    // Output: false
```



# JavaScript String Methods

## indexOf() Method

The `indexOf()` method returns the index of the first occurrence of a specified text in a string.



```
1 const str = "hello world";  
2 console.log(str.indexOf("world")); // 6
```



# JavaScript String Methods

## lastIndexOf() Method

The `lastIndexOf()` method returns the index of the last occurrence of a specified text in a string.



```
1  const str = "hello world";  
2  console.log(str.lastIndexOf("world")); // 6
```



# JavaScript String Methods

## localeCompare() Method

The `localeCompare()` method returns a number indicating whether the reference string comes before, matches, or comes after the given string in sort order.



```
1  const str1 = 'a';  
2  const str2 = 'b';  
3  console.log(str1.localeCompare(str2)); // Output: -1
```



# JavaScript String Methods

## match() Method

The `match()` method retrieves the matches of a string against a search pattern specified by a regular expression object.



```
1  const str = 'hello world';  
2  const result = str.match(/hello/);  
3  console.log(result[0]); // Output: "hello"
```



# JavaScript String Methods

## matchAll() Method

The `matchAll()` method returns an iterator of all results matching a string against a regular expression, including capturing groups.



```
1  const str = 'hello world';  
2  const regex = /h(e)(l)(l)o/g;  
3  const array = [...str.matchAll(regex)];  
4  console.log(array[0]); // Output: ["hello", "e", "l", "l", "o"]
```



# JavaScript String Methods

## normalize() Method

The `normalize()` method returns the Unicode Normalization Form of the string, which is useful for comparing strings in an accent-insensitive manner.



```
1  const str = '\u1E9B\u0323';  
2  console.log(str.normalize('NFC')); // Output: "\u1E9B\u0323"
```



# JavaScript String Methods

## padEnd() Method

The `padEnd()` method pads the current string from the end with a given string (repeated, if needed) until the resulting string reaches a given length.



```
1  const str = 'hello';  
2  console.log(str.padEnd(8, '!')); // Output: "hello!!!"
```



# JavaScript String Methods

## padStart() Method

The `padStart()` method Pads the current string from the start with a given string until the resulting string reaches the given length.



```
1  const str = 'world';  
2  console.log(str.padStart(8, '!')); // Output: "!!!world"
```



# JavaScript String Methods

## repeat() Method

The repeat() method constructs and returns a new string that contains the specified number of copies of the string on which it was called, concatenated together.

```
1  const str = "ha";  
2  console.log(str.repeat(3)); // "hahaha"
```



# JavaScript String Methods

## replace() Method

The `replace()` method returns a new string with some or all matches of a pattern replaced by a replacement.



```
1  const str = 'hello world';  
2  console.log(str.replace('world', 'everyone')); // Output: "hello everyone"
```



# JavaScript String Methods

## replaceAll() Method

The `replaceAll()` method returns a new string with all matches of a pattern replaced by a replacement.



```
1  const str = 'hello world, hello';  
2  console.log(str.replaceAll('hello', 'hi')); // Output: "hi world, hi"
```



# JavaScript String Methods

## search() Method

The `search()` method executes a search for a match between a regular expression and this String object.

```
1  const str = 'hello world';  
2  console.log(str.search('world')); // Output: 6
```



# JavaScript String Methods

## slice() Method

The `slice()` method extracts a part of a string and returns it as a new string, without modifying the original string.



```
1  const str = 'hello world';  
2  console.log(str.slice(0, 5));  // Output: "hello"
```



# JavaScript String Methods

## split() Method

The `split()` method divides a String into an ordered list of substrings, puts these substrings into an array, and returns the array.



```
1  const str = 'hello world';  
2  console.log(str.split(' ')); // Output: ["hello", "world"]
```



# JavaScript String Methods

## startsWith() Method

The startsWith() method checks if a string starts with certain characters.



```
1  const str = 'hello';  
2  console.log(str.startsWith('he')); // Output: true
```



# JavaScript String Methods

## substr() Method

The `substr()` method extracts parts of a string, beginning at the character at the specified position, and returns the specified number of characters.



```
1  const str = 'hello world';  
2  console.log(str.substr(0, 5)); // Output: "hello"
```



# JavaScript String Methods

## substring() Method

The `substring()` method extracts characters from a string between two specified indices.



```
1  const str = 'hello world';  
2  console.log(str.substring(0, 5)); // Output: "hello"
```



# JavaScript String Methods

## toLocaleLowerCase() Method

The `toLocaleLowerCase()` method converts all the alphabetic characters in a string to lowercase, taking into account the host environment's current locale.



```
1  const str = 'HELLO';  
2  console.log(str.toLocaleLowerCase()); // Output: "hello"
```



# JavaScript String Methods

## toLocaleUpperCase() Method

The `toLocaleUpperCase()` method converts all the alphabetic characters in a string to uppercase, taking into account the host environment's current locale.



```
1  const str = 'hello';  
2  console.log(str.toLocaleUpperCase()); // Output: "HELLO"
```



# JavaScript String Methods

## toLowerCase() Method

The toLowerCase() method returns the calling string value converted to lowercase.



```
1  const str = 'HELLO';  
2  console.log(str.toLowerCase());  // Output: "hello"
```



# JavaScript String Methods

## toString() Method

The `toString()` method returns a string representing the specified object.



```
1  const strObj = new String('hello');  
2  console.log(strObj.toString()); // Output: "hello"
```



# JavaScript String Methods

## toUpperCase() Method

The toUpperCase() method converts a string to uppercase letters.



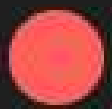
```
1  const str = 'hello';  
2  console.log(str.toUpperCase()); // Output: "HELLO"
```



# JavaScript String Methods

## trim() Method

The trim() method removes whitespace from both ends of a string.



```
1  const str = '  hello  ';  
2  console.log(str.trim()); // Output: "hello"
```



# JavaScript String Methods

## trimStart() Method

The trimStart() method removes whitespace from the beginning of a string.



```
1  const str = '  hello';  
2  console.log(str.trimStart()); // Output: "hello"
```



# JavaScript String Methods

## trimEnd() Method

The trimEnd() method removes whitespace from the end of a string.



```
1  const str = 'hello  ';  
2  console.log(str.trimEnd()); // Output: "hello"
```



# JavaScript String Methods

## valueOf() Method

The `valueOf()` method returns the primitive value of a `String` object.



```
1  const strObj = new String('hello');  
2  console.log(strObj.valueOf()); // Output: "hello"
```





# JavaScript String Methods

---

**Which of these JavaScript String methods do you use in your projects?**

---

Let's connect and expand our professional network together