# **Javascript String Methods: A Cheat Sheet for Developer**

## [**Javascript Cheat Sheet (4 Part Series)**](https://dev.to/devsmitra/series/18393)

[4Javascript String Methods: A Cheat Sheet for Developer](https://dev.to/devsmitra/javascript-string-methods-a-cheat-sheet-for-developer-4kbk)

Let's understand javascript String functions and how to use them.

#### **String.charAt()**

###### **Returns a string representing the character at the given index.**

const str = "Hello World";

str.charAt(0); // "H"

#### **String.charCodeAt()**

###### **Returns a number representing the UTF-16 code unit value of the character at the given index.**

const str = "Hello World";

str.charCodeAt(0); // 72

#### **String.concat()**

###### **Returns a new string containing the concatenation of the given strings.**

const str = "Hello";

const str2 = " World";

str.concat(str2); // "Hello World"

console.log(`${str}${str2}`); // "Hello World"

console.log(str + str2); // "Hello World"

#### **String.endsWith()**

###### **Returns true if the string ends with the given string, otherwise false.**

const str = "Hello World";

str.endsWith("World"); // true

#### **String.includes()**

###### **Returns true if the string contains the given string, otherwise false.**

const str = "Hello World";

str.includes("World"); // true

#### **String.indexOf()**

###### **Returns the index within the string of the first occurrence of the specified value, or -1 if not found.**

const str = "Hello World";

str.indexOf("World"); // 6

#### **String.lastIndexOf()**

###### **Returns the index within the string of the last occurrence of the specified value, or -1 if not found.**

const str = "Hello World";

str.lastIndexOf("World"); // 6

#### **String.match()**

###### **Returns a list of matches of a regular expression against a string.**

const str = "Hello World";

str.match(/[A-Z]/); // ["H"]

#### **String.matchAll()**

###### **Returns a list of matches of a regular expression against a string.**

const str = "Hello World";

str.matchAll(/[A-Z]/g); // ["H", "W"]

// OR

str.match(/[A-Z]/g); // ["H", "W"]

#### **String.padEnd()**

###### **Returns a new string with some content padded to the end of the string.**

const str = "Hello";

str.padEnd(15, "World"); // "HelloWorldWorld"

#### **String.padStart()**

###### **Returns a new string with some content padded to the start of the string.**

const str = "Hello";

str.padStart(15, "World"); // "WorldWorldWorldHello"

#### **String.repeat()**

###### **Returns a new string which contains the specified number of copies of the string.**

const str = "Hello";

str.repeat(3); // "HelloHelloHello"

#### **String.replace()**

###### **Returns a new string with some or all matches of a regular expression replaced by a replacement string.**

const str = "Hello World";

str.replace("l", "\*"); // "He\*lo World"

#### **String.replaceAll()**

###### **Returns a new string with some or all matches of a regular expression replaced by a replacement string.**

const str = "Hello World";

str.replaceAll("l", "\*"); // "He\*\*o Wor\*d"

OR;

str.replace(/l/g, "\*"); // "He\*\*o Wor\*d"

#### **String.search()**

###### **Returns the index within the string of the first occurrence of the specified value, or -1 if not found.**

const str = "Hello World 1";

const regex = /[^\D\s]/g; // Find digit

str.search(regex); // 12

#### **String.slice()**

###### **Returns a new string containing the characters of the string from the given index to the end of the string.**

const str = "Hello World";

str.slice(6); // "World"

#### **String.split()**

###### **Returns an array of strings split at the given index.**

const str = "Hello World";

str.split(" "); // ["Hello", "World"]

#### **String.startsWith()**

###### **Returns true if the string starts with the given string, otherwise false.**

const str = "Hello World";

str.startsWith("Hello"); // true

#### **String.substring()**

###### **Returns a new string containing the characters of the string from the given index to the end of the string.**

const str = "Hello World";

str.substring(1, 2); // "e"

## **NOTE: substring takes parameters as (from, to).**

#### **String.substr()**

###### **Returns a new string containing the characters of the string from the given index to the end of the string.**

const str = "Hello World";

str.substr(1, 2); // "el"

NOTE: substr takes parameters as (from, length).

#### **String.toLowerCase()**

###### **Returns a new string with all the uppercase characters converted to lowercase.**

const str = "Hello World";

str.toLowerCase(); // "hello world"

#### **String.toUpperCase()**

###### **Returns a new string with all the lowercase characters converted to uppercase.**

const str = "Hello World";

str.toUpperCase(); // "HELLO WORLD"

#### **String.toString()**

###### **Returns the string representation of the specified object.**

const str = new String("Hello World");

console.log(str); // Object of String

str.toString(); // "Hello World"

#### **String.trim()**

###### **Returns a new string with the leading and trailing whitespace removed.**

const str = " Hello World ";

str.trim(); // "Hello World"

#### **String.trimEnd()**

###### **Returns a new string with the trailing whitespace removed.**

const str = " Hello World ";

str.trimEnd(); // " Hello World"

#### **String.trimStart()**

###### **Returns a new string with the leading whitespace removed.**

const str = " Hello World ";

str.trimStart(); // "Hello World "