

Ace your JavaScript Interview. [Get my ebook](#). 100 solved Javascript, 20 solved React, & 2 frontend system design questions (**1160+ copies sold**). Get a [Free preview](#).

Advertisements

# How to find the substring in javascript

Posted on [February 22, 2019](#) | by [Prashant Yadav](#)

Posted in [Javascript](#), [String](#)

Learn how to find the substring from a given [string](#) in javascript. We will use different methods to find the substrings.

## Using [slice\(\)](#)

[slice\(\)](#) methods returns a part of the string as a substring without affecting the original string.

```
let str = 'I am Prashant Yadav';
let substring = str.slice(5);
console.log(substring);
// "Prashant Yadav"
```

Copy

It has extracted substring starting at index 5 till the end of the string.

The [slice\(\)](#) methods takes two parameters as input.

**String.slice(startIndex, endIndex)**

**startIndex**

- **Required.** Zero-based index at which substring should start from.
- If **negative** number is provided then it will be treated as **stringLength + startIndex** where **stringLength** is the length of the string. Like **stringLength + (-3)**.
- If the **startIndex** is greater than or equal to the string's length then an empty substring will be returned.

**endIndex**

- **Optional.** Zero-based index before which to end extraction of the substring.
- If **negative** number is provided then it will be treated as **stringLength + endIndex** where **stringLength** is the length of the string. Like **stringLength + (-3)**.

Practically  
prepare for  
your  
JavaScript  
interview

[JavaScript  
Revision](#)

[JavaScript-  
Concept Based  
Problems](#)

[Data Structures](#)

[Algorithms](#)

[Machine](#)

[Coding](#)

[Web](#)

[Fundamentals](#)

Advertisements

- If `endIndex` is not provided the substring till the end of the string is returned.

```
let str = 'I am Prashant Yadav';

//With endIndex
let substring = str.slice(5, 8);
console.log(substring);
//Pra

//With negative startIndex
let substring = str.slice(-5);
console.log(substring);
//"Yadav"

//With negative endIndex
let substring = str.slice(5, -4);
console.log(substring);
//"Prashant Y"

//With startIndex > string length
let substring = str.slice(15, -4);
console.log(substring);
// ""
```

Copy

## Using substring()

substring() methods removes and return the new substring from a given string between the two specified indexes or to the end of the string.

Advertisements



```
let str = 'I am Prashant Yadav';
let substring = str.substring(5);
console.log(substring);
//"Prashant Yadav"
```

Copy

Returns all the characters starting at index `5` till end of the given string. It takes two parameters as input.

`String.substring(startIndex, endIndex)`

`startIndex`

- **Required.** Zero-based index from where the substring extraction should start.

## endIndex

- **Optional.** Zero-based index before which substring extraction should end.
- The `endIndex` character is not included in the string.
- If omitted it will return all the characters till the end of the string.

```
let str = 'I am Prashant Yadav';
let substring = str.substring(5, 10);
console.log(substring);
// "Prash"
```

Copy

There are few things you should keep note of while using `substring()`.

- If `startIndex` and `endIndex` are equal then it will return empty substring.
- If `startIndex` is greater than `endIndex` then it will swap the indexes and return the substring accordingly. Like `str.substring(10, 5)` will be swapped to `str.substring(5, 10)`

```
let str = 'I am Prashant Yadav';

//When startIndex and endIndex are same
let substring = str.substring(5, 5);
console.log(substring);
// ""

//When startIndex > endIndex
let substring = str.substring(10, 5);
console.log(substring);
// "Prash"
```

Copy

## Difference between `substring()` and `slice()`

Both methods may seem to do the same work but there is a subtle difference in them.

- If the `startIndex` is greater than `endIndex`, `substring()` will swap the two indexes, while `slice()` will return an empty substring.
- If any one or both the indexes are given `negative` or `NaN`, `substring()` will treat them as `0`. While `slice()` treats `NaN` as `0` but for the `negative` values it starts the count from the end of the string.

You can also use `substr` method. But it is going to be deprecated in the future so its better to avoid it.

Prepare for your **JavaScript Interview** practically on each Interview rounds and grab that job.

BEGIN LEARNING

## Recommended Posts:

- [Filter nested object in Javascript](#)
- [Proxy design pattern in JavaScript](#)
- [Program to print all the permutation of string](#)
- [6 ways to convert string to a number in javascript](#)
- [How to encode a url with javascript](#)
- [How to open new tab in javascript](#)
- [Valid Palindrome II](#)
- [How to check undefined in javascript](#)
- [useHasFocus\(\) hook in React](#)
- [Implement custom instanceof method](#)

[Prev](#)

[Next](#)

## Leave a Reply

Your email address will not be published. Required fields are marked \*

Comment

Start typing...

Name\*

Name

Email\*

Email

POST COMMENT

Advertisements



[About Us](#)

[Contact Us](#)

[Privacy Policy](#)

[Advertise](#)



Handcrafted with ♥ somewhere in **Mumbai**

© 2023 [LearnersBucket](#) | [Prashant Yadav](#)

