



# String.prototype.search()

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

## JavaScript Demo: String.search()

```
1 const paragraph = 'The quick brown fox jumps over the lazy dog. If the dog bark
2
3 // any character that is not a word character or whitespace
4 const regex = /^[^\w\s]/g;
5
6 console.log(paragraph.search(regex));
7 // expected output: 43
8
9 console.log(paragraph[paragraph.search(regex)]);
10 // expected output: "."
11
```

Run ›

Reset

## Syntax

`search(regex)`



## Parameters

### regex

A [regular expression](#) object.

If a non-RegExp object `regex` is passed, it is implicitly converted to a [RegExp](#) with `new RegExp(regex)`.

## Return value

The index of the first match between the regular expression and the given string, or `-1` if no match was found.

## Description

When you want to know whether a pattern is found, and *a*lso know its index within a string, use `search()`. (If you only want to know if it exists, use the similar `test()` method on the `RegExp` prototype, which returns a boolean.)

For more information (but slower execution) use `match()` (similar to the regular expression `exec()` method).

## Examples

### Using search()

The following example searches a string with two different regex objects to show a successful search (positive value) vs. an unsuccessful search (`-1`)

```
let str = "hey Jude"
let re = /[A-Z]/g
let reDot = /[.]/g
console.log(str.search(re))    // returns 4, which is the index of the first
console.log(str.search(reDot)) // returns -1 cannot find '.' dot punctuation
```

## Specifications

Specification
<a href="#">ECMAScript Language Specification (ECMAScript)</a> <a href="#"># sec-string.prototype.search</a>

## Browser compatibility

[Report problems with this compatibility data on GitHub](#)

search	
Chrome	1

Edge	12
Firefox	1
Internet Explorer	4
Opera	4
Safari	1
WebView Android	1
Chrome Android	18
Firefox for Android	4
Opera Android	10.1
Safari on iOS	1
Samsung Internet	1.0
Deno	1.0
Node.js	0.10.0
flags  	
Chrome	No
Edge	No
Firefox	1–49
Internet Explorer	No
Opera	No
Safari	No
WebView Android	No
Chrome Android	No
Firefox for Android	4–49
Opera Android	No
Safari on iOS	No
Samsung Internet	No
Deno	No
Node.js	No



Full support



No support



Non-standard. Check cross-browser support before using.



Deprecated. Not for use in new websites.

## See also

- [Using regular expressions in JavaScript](#)
- [String.prototype.match\(\)](#)
- [RegExp.prototype.exec\(\)](#)

**Last modified:** Jul 20, 2021, [by MDN contributors](#)