

Regex to replace multiple spaces with a single space

[Ask Question](#)

Given a string like:

389

"The dog has a long tail, and it is RED!"



What kind of jQuery or JavaScript magic can be used to keep spaces to only one space max?



73

Goal:

"The dog has a long tail, and it is RED!"

javascript

jquery

regex

edited May 3 '16 at 16:14



[TylerH](#)

16.2k 10 55 70

asked Dec 30 '09 at 17:28



[AnApprentice](#)

39.4k 156 510 902

4 Do you also want to match whitespace tab characters? – [Chris Farmer](#) Dec 30 '09 at 17:34

@Chris, Yes please, great question.... With all these different answers, how is one supposed to know which is the most

Join **Stack Overflow** to learn, share knowledge, and build your career.

Email Sign Up

OR SIGN IN WITH



Google

Facebook

- 1 There's not going to be any noticeable difference in performance. You could always profile it, but I doubt it would be worth it. I would go for the clearest. – [Draemon](#) Dec 30 '09 at 18:15

@EvanCarroll: Not true -- at least on Firefox. That version runs significantly slower. See the profiling results in my answer (below). – [Edward Loper](#) Dec 30 '09 at 19:28

21 Answers



738



Given that you also want to cover tabs, newlines, etc, just replace `\s\s+` with `' '`:

```
string = string.replace(/\s\s+/g, ' ');
```

If you really want to cover only spaces (and thus not tabs, newlines, etc), do so:

```
string = string.replace(/ +/g, ' ');
```

edited May 18 '15 at 7:52

answered Dec 30 '09 at 17:31



[BalusC](#)

867k 308 3206

3255

- 2 Does this also remove, whitespace tab characters? –

Join **Stack Overflow** to learn, share knowledge, and build your career.


Email Sign Up

OR SIGN IN WITH




Google



tab, \r, \n, \v \f) with space-character. – [Rafael](#) Dec 30 '09 at 17:40 

4 This is not working when a blank instead of a tab or newline is needed. Right? \s+/ would be working. – [Fabian](#) Feb 20 '13 at 11:01

4 @Ethan: JS has a builtin function for that: `trim()` . It's faster than regex. You could just do
`string.trim().replace(/\s\s+/g, ' ');` or
`string.replace(/\s\s+/g, ' ').trim();` . – [BalusC](#) Apr 12 '18 at 6:12 



Since you seem to be interested in performance, I profiled these with firebug. Here are the results I got:

141



```
str.replace( / +/g, ' ' )      -> 380ms
str.replace( /\s\s+/g, ' ' )   -> 390ms
str.replace( / {2,}/g, ' ' )   -> 470ms
str.replace( / +/g, ' ' )      -> 790ms
str.replace( / +(?= )/g, ' ' ) -> 3250ms
```

This is on Firefox, running 100k string replacements.

I encourage you to do your own profiling tests with firebug, if you think performance is an issue. Humans are notoriously bad at predicting where the bottlenecks in their programs lie.

(Also, note that IE 8's developer toolbar also has a profiler built in -- it might be worth checking what the performance is like in IE.)

Join Stack Overflow to learn, share knowledge, and build your career.

Email Sign Up

OR SIGN IN WITH





Edward Loper

10.3k 3 31 45

- 4 jsperf.com/removing-multiple-spaces Go forth and JSPerf! The last method; `(/ +(?=)/g, ' ');` fails in IE9, it leaves double spaces: `"Foo Bar Baz".replace(/ +(?=)/g, ' ');` --> `"Foo Bar Baz"` – [Nenotlep](#) Apr 26 '13 at 10:22

how there is much diff bw 1 and 2nd line – [Vivek Panday](#) Feb 24 '16 at 9:16

@VivekPanday - I imagine that this is because the second line is only replacing occurrences of double spaces with a single space, while the first is replacing any space with a space as well. Whether this is time saved during the search or the actual replace, I don't know. – [Maloric](#) Aug 16 '16 at 11:27

This doesn't remove the initial and trailing whitespaces. For that see [this answer](#). – [Ethan](#) Apr 12 '18 at 5:04

Edited to order by decreasing speed. Vivek's and Maloric's comments refer to lines with 380 ms and 790 ms. – [Skippy le Grand Gourou](#) Feb 16 at 21:04



```
var str = "The      dog      has a long tail,      and i·
str = str.replace(/ {2,}/g, ' ');
```

38



EDIT: If you wish to replace all kind of whitespace characters the most efficient way would be like that:

```
str = str.replace(/\s{2,}/g, ' ');
```

Join **Stack Overflow** to learn, share knowledge, and build your career.

Email Sign Up

OR SIGN IN WITH



Google





watain

2,150 4 25 32

Funny your test string doesn't even have two spaces in it. – [Josh Stodola](#) Dec 30 '09 at 17:31

just realized you already had what i recently came up with, +1 :) – [meder omuraliev](#) Dec 30 '09 at 17:40

2 For some reason this isn't working... A lot of " " are showing up... Likely due to CKEDITOR... – [AnApprentice](#) Dec 30 '09 at 18:28

K turns out JQUERY's text() was messing things up. fixed - thanks all! – [AnApprentice](#) Dec 30 '09 at 18:30



This is one solution, though it will target *all* space characters:

16



```
"The    dog      has a long tail,    and it is RED!"
```

```
"The dog has a long tail, and it is RED!"
```

Edit: This is probably better since it targets a space followed by 1 or more spaces:

```
"The    dog      has a long tail,    and it is RED!"
```

```
"The dog has a long tail, and it is RED!"
```

Alternative method:

Join **Stack Overflow** to learn, share knowledge, and build your career.

Email Sign Up

OR SIGN IN WITH



Google

Facebook

I didn't use `/\s+/` by itself since that replaces spaces that span 1 character multiple times and might be less efficient since it targets more than necessary.

I didn't deeply test any of these so lmk if there are bugs.

Also, if you're going to do string replacement remember to re-assign the variable/property to its own replacement, eg:

```
var string = 'foo'  
string = string.replace('foo', '')
```

Using `jQuery.prototype.text`:

```
var el = $('span:eq(0)');  
el.text( el.text().replace(/\d+/, '') )
```

edited Dec 30 '09 at 17:45

answered Dec 30 '09 at 17:32



[meder omuraliev](#)

143k 56 342 416

-
- 1 The first one is totally pointless, `\s\s+` means, an `\s` followed by one or more `\s+`, which can be reduced to a single `\s+`, the second example is more accurate because we only want to replace double spaces, not newlines, the third is more optimized because it only applies to examples with 2+ spaces. But `str.replace(/ +(?=)/g, "");` only applies to examples with 2+ spaces but saves overwriting a space with a space step. – [Evan Carroll](#) Dec 30 '09 at 17:55
-

- 3 [EvanCarroll](#) you **fail** because `\s\s+` is definitely different to `\s+`.

Join **Stack Overflow** to learn, share knowledge, and build your career.

Email Sign Up

OR SIGN IN WITH



'16 at 13:03 

I have this method, I call it the Derp method for lack of a better name.

13



```
while (str.indexOf(" ") !== -1) {  
    str = str.replace(/ /g, " ");  
}
```

[Running it in JSPerf gives some surprising results.](#)

answered Apr 26 '13 at 10:36



[Nenotlep](#)

9,240 4 52 89

2 I'm going to be embarrassed as hell if it turns out I fudged up the test case instead of it actually being fast :D – [Nenotlep](#) Apr 26 '13 at 10:38

Providing a test case... Excellent answer! – [Oytun](#) Jun 18 '14 at 8:56

2 This made my day :-) Funny how "derping" often works better than being all "clever". The "Derp split" appears to have kicked its ass though. Still, deserves upvoting. – [Fred Gandt](#) Apr 7 '15 at 3:01

for me only replace(/ +/g, " ") worked – [Miguel](#) Jun 16 '17 at 10:03

Join **Stack Overflow** to learn, share knowledge, and build your career.

Email Sign Up

OR SIGN IN WITH



Google

Facebook 

12

```
// NOTE the possible initial and trailing spaces
var str = "  The dog      has a long  tail, and it    is

str = str.replace(/^\s+|\s+$|\s+(?=\s)/g, "");

// str -> "The dog has a long tail, and it is RED !"
```

Your example didn't have those spaces but they are a very common scenario too, and the accepted answer was only trimming those into single spaces, like: " The ... RED! ", which is not what you will typically need.

edited Apr 12 '18 at 5:00

answered Aug 31 '16 at 19:58



Ethan

3,642 1 19 30

3 I used this pattern on PHP and works. \$parts = preg_split("/^\s+|\s+\$|\s+(?=\s)/", "Avenida Tancredo Neves, 745 Piso Térreo Sala"); – Bruno Ribeiro Jan 16 '17 at 1:32



More robust:

10



```
function trim(word)
{
    word = word.replace(/[\x21-\x7E]+/g, ' '); // change !
    return word.replace(/^\s+|\s$/g, ''); // remove :
}
```

Join **Stack Overflow** to learn, share knowledge, and build your career.

Email Sign Up

OR SIGN IN WITH



Google



answered May 10 '11 at 15:37



Chris

101 1 2



I suggest

8

```
string = string.replace(/ +/g, " ");
```

for just spaces
OR

```
string = string.replace(/(\s+)/g, "$1");
```

for turning multiple returns into a single return also.

answered May 14 '13 at 0:59



Leonard Meagher

119 1 2

Here is an alternate solution if you do not want to use
replace (replace spaces in a string without using replace
javascript)

6



```
var str="The dog      has a long  tail, and it      is RED  
var rule=/\s{1,}/g;  
str = str.split(rule).join(" ");
```

Join Stack Overflow to learn, share knowledge, and build your career.

Email Sign Up

OR SIGN IN WITH



Google





8,895 5 69 87

answered Mar 15 '13 at 5:10



imos

87 1 7



I know that I am late to the party, but I discovered a nice solution.

5

Here it is:



```
var myStr = myStr.replace(/[ ]*/g, ' ');
```

answered Apr 22 '15 at 6:16



ToXic73

169 4 15

For IntelliJ regex this worked for me – [Seabass77](#) May 2 '18 at 1:56



Comprehensive unencrypted answer for newbies et al.

5

This is for all of the dummies like me who test the scripts written by some of you guys which do not work.



The following 3 examples are the steps I took to remove

Join Stack Overflow to learn, share knowledge, and build your career.

[Email Sign Up](#)

OR SIGN IN WITH

[Google](#)

We have chained these together with over 50 at a time and NO problems.

// This removed special characters + 0-9 and allows for just letters (upper and LOWER case)

```
function NoDoublesPls1()
{
  var str=document.getElementById("NoDoubles1");
  var regex=/[^a-z]/gi;
  str.value=str.value.replace(regex, "");
}
```

// This removed special characters and allows for just letters (upper and LOWER case) and 0-9 AND spaces

```
function NoDoublesPls2()
{
  var str=document.getElementById("NoDoubles2");
  var regex=/[^a-z 0-9]/gi;
  str.value=str.value.replace(regex, "");
}
```

// This removed special characters and allows for just letters (upper and LOWER case) and 0-9 AND spaces //
The .replace(/\\s\\s+/g, " ") at the end removes excessive spaces // when I used single quotes, it did not work.

```
function NoDoublesPls3()
{
  var str=document.getElementById("NoDoubles3");
  var regex=/[^a-z 0-9]/gi;
  str.value=str.value.replace(regex, "").replace(/\\s\\s+/g, " ");
}
```

::NEXT:: Save #3 as a .js // I called mine NoDoubles.js

Join **Stack Overflow** to learn, share knowledge, and build your career.

Email Sign Up

OR SIGN IN WITH



Include this in your form field:: such as

```
<INPUT type="text" name="Name"
      onKeyUp="NoDoublesPls3()" onKeyDown="NoDoublesPls3()"
```

So that it looks like this

```
<INPUT type="text" name="Name" onKeyUp="NoDoublesPls3()" or
id="NoDoubles3"/>
```

This will remove special characters, allow for single spaces and remove extra spaces.

edited Jan 24 '18 at 8:24



semuzaboi

2,130 2 14 28

answered Jan 9 '15 at 20:33



PatFoster

51 1 2

What is going on here? The formatting looks very, very broken.

– Nenotlep Oct 8 '15 at 11:19

Also a possibility:

4

```
str.replace( /\s+/g, ' ' )
```

answered Dec 30 '09 at 17:32

Join Stack Overflow to learn, share knowledge, and build your career.

Email Sign Up

OR SIGN IN WITH



Google

Facebook



Jquery has trim() function which basically turns something like this " FOo Bar " into "FOo Bar".

1



```
var string = "  My      String with Multiple lines  ";  
string.trim(); // output "My String with Multiple Lines"
```

It is much more usefull because it is automatically removes empty spaces at the beginning and at the end of string as well. No regex needed.

edited Aug 17 '17 at 7:30

answered Aug 16 '17 at 12:13



Eryk Wróbel

138 1 10

- 2 As you said, trim() removes empty spaces at the *beginning* and at the *end* of string, but not in the middle of the string, so, it doesn't work in this case, the output would just be "My String with Multiple lines". api.jquery.com/jQuery.trim – egvaldes Aug 25 '17 at 18:35



```
var myregexp = new RegExp(/ {2,}/g);
```

0

```
str = str.replace(myregexp, ' ');
```

Join **Stack Overflow** to learn, share knowledge, and build your career.

Email Sign Up

OR SIGN IN WITH



Google





```
var string = "The dog      has a long  tail, and it    i:  
var replaced = string.replace(/ +/g, " ");
```

Or if you also want to replace tabs:

```
var replaced = string.replace(/\s+/g, " ");
```

edited Dec 30 '09 at 17:38

answered Dec 30 '09 at 17:32



[Brian Campbell](#)

241k 51 320 321

using + seems cleaner but it will also replace single spaces with single spaces, a little bit redundant and I'm not sure but it may create performance problems with a much longer text. – [marvin](#) Dec 30 '09 at 17:39

I tend to use the shortest, simplest solution that will work, and only worry about that sort of optimization if I know that I need to be matching against a very large string, and at that point I will actually measure different solutions to see which will be faster. It can be hard to predict in advance what will be fastest without testing; for instance, in JavaScript interpreters, some complicated regular expressions will cause you to switch from a fast JIT compiled implementation to a slow interpreted one. – [Brian Campbell](#) Dec 30 '09 at 17:44

Join **Stack Overflow** to learn, share knowledge, and build your career.

Email Sign Up

OR SIGN IN WITH





Add the text into some file say test

```
manjeet-laptop:Desktop manjeet$ cat test
"The dog      has a long  tail, and it      is RED!"
```

We can use the following regex to replace all white spaces with single space

```
manjeet-laptop:Desktop manjeet$ sed 's/ \{1,\}/ /g' test
"The dog has a long tail, and it is RED!"
```

Hope this serves the purpose

answered Jan 19 '15 at 9:35



minhas23

6,561 2 45 36



Try this to replace multiple spaces with a single space.

0

```
<script type="text/javascript">
  var myStr = "The dog      has a long  tail, and it
  alert(myStr); // Output 'The dog      has a Long  ta

  var newStr = myStr.replace(/ +/g, ' ');
  alert(newStr); // Output 'The dog has a Long tail, an
</script>
```

Read more @ [Replacing Multiple Spaces with Single Space](#)

answered Apr 14 '16 at 14:50

Join **Stack Overflow** to learn, share knowledge, and build your career.

Email Sign Up

OR SIGN IN WITH



Google





```
var text = `xxx df dfvdfv df
            dfv`.split(/\s,\t,\r,\n]+)/.filter(x:
```

0



result:

```
"xxx df dfvdfv df dfv"
```

edited Jun 5 '18 at 15:22

answered Jun 5 '18 at 13:44



Toolkit

6,678 4 40 53



For more control you can use the replace callback to handle the value.

0



```
value = "tags:HUNT tags:HUNT tags:HUNT tags:HUNT"
value.replace(new RegExp(`(?:\\s+)(?:tags)`), 'g'), $1 => `
// "tags:HUNT tags:HUNT tags:HUNT tags:HUNT"
```

answered Oct 14 '18 at 19:17



jackotonye

1,684 12 23

Join **Stack Overflow** to learn, share knowledge, and build your career.

Email Sign Up

OR SIGN IN WITH



Google



0

```
// Trims & replaces any whitespace to single space between words
String.prototype.clearExtraSpace = function(){
    var _trimLeft = /^s+/,
        _trimRight = /\s+$/,
        _multiple = /\s+/g;

    return this.replace(_trimLeft, '').replace(_trimRight, '').replace(_multiple, ' ');
};
```

answered Oct 17 '18 at 23:21



Oriol

4,200 1 24 33



is replace is not used, string = string.split(/\W+/);

0



answered Nov 2 '18 at 8:41



Lin

1

Join **Stack Overflow** to learn, share knowledge, and build your career.

[Email Sign Up](#)

OR SIGN IN WITH

 Google[Facebook](#)

Join Stack Overflow to learn, share knowledge, and build your career.

Email Sign Up

OR SIGN IN WITH

 Google

Facebook 