Regex match entire words only

Asked 9 years, 9 months ago Active 2 months ago Viewed 187k times

76

17

I have a regex expression that I'm using to find all the words in a given block of content, case insensitive, that are contained in a glossary stored in a database. Here's my pattern:

```
/($word)/i
```

The problem is, if I use `/(Foo)/i` then words like `Food` get matched. There needs to be whitespace or a word boundary on both sides of the word.

How can I modify my expression to match only the word `Foo` when it is a word at the beginning, middle, or end of a sentence?

regex word-boundary

edited Jan 6 '14 at 17:54

Eric Leschinski
96.3k 43 341 293

asked Nov 17 '09 at 19:49

Aaron
487 3 6 7

5 Answers

99

Use word boundaries:

```
/\b($word)\b/i
```

Or if you're searching for "S.P.E.C.T.R.E." like in Sinan Ünür's example:

```
/(?:\W|^)(\Q$word\E)(?:\W|$)/i
```

edited Nov 17 '09 at 21:14

answered Nov 17 '09 at 19:51

Richard Simões
8,935 3 34 48

1 I was just typing up the long-hand version of this answer when you posted. :) – ZombieSheep Nov 17 '09 at 19:52

@RichardSimoes `\b(<|>=)\b` doesn't match `>=` – alhelal Jan 21 '18 at 1:40

@RichardSimoes and `\b[-|+][0-9]+\b` match `+10` in `43E+10` . Both I don't want. – alhelal Jan 21 '18 at 1:47

what if i want to search word which is not appended or does not contained in any other word. then this logic won't work – Prasanna Sasne Nov 14 '18 at 8:49

How would someone get the mathematical comparison operators >= and <=? – AntonSack Jun 21 at 7:30

8

Using `\b` can yield surprising results. You would be better off figuring out what separates a word from its definition and incorporating that information into your pattern.

```
#!/usr/bin/perl

use strict; use warnings;

use re 'debug';

my $str = 'S.P.E.C.T.R.E. (Special Executive for Counter-intelligence,
Terrorism, Revenge and Extortion) is a fictional global terrorist
organisation';

my $word = 'S.P.E.C.T.R.E.';

if ( $str =~ /\b(\Q$word\E)\b/ ) {
    print $1, "\n";
}
```

Output:

```
Compiling REx "\b(S\.P\.E\.C\.T\.R\.E\.)\b"
Final program:
   1: BOUND (2)
```

```
   2: OPEN1 (4)
   4:   EXACT  (9)
   9: CLOSE1 (11)
  11: BOUND (12)
  12: END (0)
anchored "S.P.E.C.T.R.E." at 0 (checking anchored) stclass BOUND minlen 14
Guessing start of match in sv for REx "\b(S\.P\.E\.C\.T\.R\.E\.)\b" against "S.P
.E.C.T.R.E. (Special Executive for Counter-intelligence,"...
Found anchored substr "S.P.E.C.T.R.E." at offset 0...
start_shift: 0 check_at: 0 s: 0 endpos: 1
Does not contradict STCLASS...
Guessed: match at offset 0
Matching REx "\b(S\.P\.E\.C\.T\.R\.E\.)\b" against "S.P.E.C.T.R.E. (Special Exec
utive for Counter-intelligence,"...
   0           |  1:BOUND(2)
   0           |  2:OPEN1(4)
   0           |  4:EXACT (9)
  14      |  9:CLOSE1(11)
  14      | 11:BOUND(12)
                                failed...
Match failed
Freeing REx: "\b(S\.P\.E\.C\.T\.R\.E\.)\b"
```

answered Nov 17 '09 at 20:03

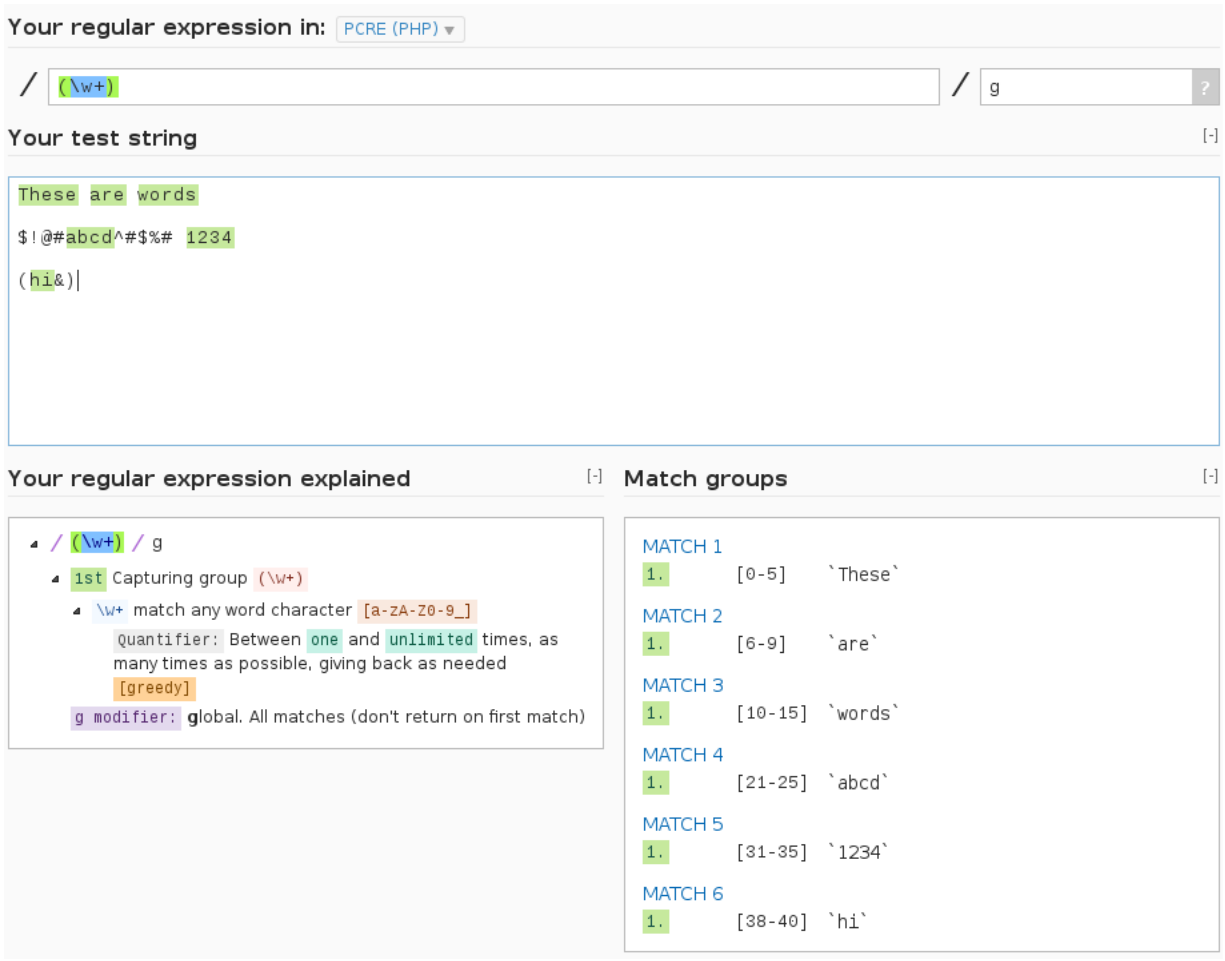
Sinan Ünür
109k 15 178 314

1 I think a word will typically be a \w word, but interesting point. – Richard Simões Nov 17 '09 at 20:09

36

To match any whole word you would use the pattern `(\w+)`

Assuming you are using PCRE or something similar:



Above screenshot taken from this live example: http://regex101.com/r/cU5IC2

Matching any whole word on the commandline with `(\w+)`

I'll be using the phpsh interactive shell on Ubuntu 12.10 to demonstrate the PCRE regex engine through the method known as preg_match

Start phpsh, put some content into a variable, match on word.

```
el@apollo:~/foo$ phpsh

php> $content1 = 'badger'
php> $content2 = '1234'
php> $content3 = '$%^&'

php> echo preg_match('(\w+)', $content1);
1

php> echo preg_match('(\w+)', $content2);
1

php> echo preg_match('(\w+)', $content3);
0
```

The preg_match method used the PCRE engine within the PHP language to analyze variables: `$content1` , `$content2` and `$content3` with the `(\w)+` pattern.

$content1 and $content2 contain at least one word, $content3 does not.

Match a number of literal words on the commandline with `(dart|fart)`

```
el@apollo:~/foo$ phpsh

php> $gun1 = 'dart gun';
php> $gun2 = 'fart gun';
php> $gun3 = 'farty gun';
php> $gun4 = 'unicorn gun';

php> echo preg_match('(dart|fart)', $gun1);
1

php> echo preg_match('(dart|fart)', $gun2);
1

php> echo preg_match('(dart|fart)', $gun3);
1

php> echo preg_match('(dart|fart)', $gun4);
0
```

variables gun1 and gun2 contain the string dart or fart. gun4 does not. However it may be a problem that looking for word `fart` matches `farty` . To fix this, enforce word boundaries in regex.

Match literal words on the commandline with word boundaries.

```
el@apollo:~/foo$ phpsh

php> $gun1 = 'dart gun';
php> $gun2 = 'fart gun';
php> $gun3 = 'farty gun';
php> $gun4 = 'unicorn gun';

php> echo preg_match('(\bdart\b|\bfart\b)', $gun1);
1

php> echo preg_match('(\bdart\b|\bfart\b)', $gun2);
1

php> echo preg_match('(\bdart\b|\bfart\b)', $gun3);
0

php> echo preg_match('(\bdart\b|\bfart\b)', $gun4);
0
```

So it's the same as the previous example except that the word `fart` with a `\b` word boundary does not exist in the content: `farty` .

edited Jan 6 '14 at 18:11

answered Jan 6 '14 at 17:51

Eric Leschinski

96.3k 43 341 293

a.m., p.m. ain't words? – minion Jun 27 '18 at 17:28

If you want to force a.m. and p.m. to be words, (they're not, they're acronyms) then add period as a word character for your regex engine. For you it appears you've set period as not a word character, so therefore regex words won't be one-to-one and onto for the standard definition of "word" that you were taught in your European Dictionary for your hybrid European language (or any other language for that matter). – Eric Leschinski Nov 18 '18 at 20:36

1

use word boundaries \b,

The following (using four escapes) works in my environment: Mac, safari Version 10.0.3 (12602.4.8)

```
var myReg = new RegExp('\\\\b'+ variable + '\\\\b', 'g')
```

answered Jun 7 '18 at 18:11

X. L

266 3 9

0

If you are doing it in Notepad++

```
[\w]+
```

Would give you the entire word, and you can add parenthesis to get it as a group. Example: `conv1 = Conv2D(64, (3, 3), activation=LeakyReLU(alpha=a), padding='valid', kernel_initializer='he_normal')(inputs)` . I would like to move `LeakyReLU` into its own line as a comment, and replace the current activation. In notepad++ this can be done using the follow find command:

```
([\w]+)( = .+)(LeakyReLU.alpha=a.)(.+)
```

and the replace command becomes:

```
\1\2'relu'\4 \n    # \1 = LeakyReLU\(alpha=a\)\(\1\)
```

The spaces is to keep the right formatting in my code. :)

answered Jun 11 at 10:55

JTIM

896 19 58