

## Re: split regex

---

*Source:* <http://coding.derkeiler.com/Archive/Perl/perl.beginners/2005-12/msg00589.html>

---

- *From:* [umesh.perl@xxxxxxxxxx](mailto:umesh.perl@xxxxxxxxxx) (Umesh T G)
  - *Date:* Fri, 16 Dec 2005 11:53:29 +0530
- 

Sorry for that John. It works fine.

Many thanks for your help.

Umesh

On 12/16/05, Umesh T G <[umesh.perl@xxxxxxxxxx](mailto:umesh.perl@xxxxxxxxxx)> wrote:

```
>
> Thanks Very much John. :) it worked, but a small problem. if I have
> "\" after the directory name, it does not take as new instance.
> for ex:
> if my $line is *Options=-a"hello" -mt -ml3 -i"dir1\dir2"-k -p
> -i"dir3\dir4\" -m -n -i"dir5\dir6" -k*
> **
> then, I'm getting output like this,
>
> dir1\dir2
> dir3\dir4\dir5\dir6
>
> This is because an extra "\" is present at the end of dir4.
>
> can you help again pls.? thanks so much.
>
> I want the output as
>
> dir1\dir2
> dir3\dir4
> dir5\dir6
>
>
>
> Umesh
>
>
>
>
>
```

Re: split regex

```
>
>
> On 12/15/05, John Doe <security.department@xxxxxxxx> wrote:
>>
>> Umesh T G am Donnerstag, 15. Dezember 2005 15.31:
>>> Hi Joe,
>>>
>>> just to correct my prevoius one, my $line does not have * in the
>> begining,
>>> that was a type. the $line value is like this
>>>
>>> *Options=-a"hello" -mt -ml3 -i"dir1\dir2"-k -p -i"dir3\dir4" -m*
>>>
>>>
>>> my grep return the values correct to my array and the values in array
>> will
>>> be like this obviously:
>>>
>>> *Options=-a"hello" -mt -ml3 *
>>> *"dir1\dir2"-k -p *
>>> *"dir3\dir4" -m*
>>> **
>>> **
>>> But, what I'm interested in getting the output values are:
>>> *"dir1\dir2"*
>>> *"dir3\dir4"*
>>>
>>> I do not know how to get them.....can u suggest pls.?
>>
>>
>> use strict;
>> use warnings;
>>
>> my $str=q(Options=-a"hello" -mt -ml3 -i"dir1\dir2"-k -p -i"dir3\dir4"
>> -m);
>>
>> # this is only one possibility. I use \s* in the case spaces are
>> # allowed after -i.
>> # Instead of [^"]+ also .+? works.
>> # You could also modify the regex to allow " as well as ' around dirs.
>> #
>> my @dirs=$str=~m,-i\s*"([^\"]+)",g;
>>
>> print join "\n", @dirs;
>>
>> # output:
>> dir1\dir2
>> dir3\dir4
>>
>> ok?
```

Re: split regex

Re: split regex

```
>> :-)
>>
>> [top post history following]
>>
>>> On 12/15/05, John Doe < security.department@xxxxxxx> wrote:
>>>> Umesh T G am Donnerstag, 15. Dezember 2005 14.20:
>>>>> Hi John,
>>>>>
>>>>> Hi Umesh
>>>>>
>>>>> I tried like this, where $line=*Options=-a"hello" -mt -ml3
>>>>>
>>>>> -i"dir1\dir2"
>>>>>
>>>>> -k -p -i"dir3\dir4" -m'*
>>>>>
>>>>>
>>>>>
>>>>> if (grep/^Options/, $line)
>>>>> {
>>>>>
>>>>> @inc=split(/-i/, $line);
>>>>>
>>>>> foreach $word (@inc)
>>>>> {
>>>>> print "$word\n";
>>>>> }
>>>>>
>>>>>
>>>>>
>>>>> I do not know how to proceed after this....
>>>>> Thanks for your quick reply...
>>>>>
>>>>> You're welcome...
>>>>>
>>>>> try the following:
>>>>>
>>>>> a) put
>>>>> use strict;
>>>>> use warnings;
>>>>> at the beginning of the script
>>>>>
>>>>> b) look at the warnings and then declare the vars with my, see
>>>>> perldoc -f my
>>>>>
>>>>> c)
>>>>> read the documentation
>>>>> perldoc -f grep
>>>>> to see why grep is not appropriate in the if clause
>>>>>
>>>>> d) check the documentation
```

Re: split regex

>>>> perldoc -f perlre  
>>>> and the others mentioned at the bottom to see why  
>>>> /^Options/ won't ever match a string beginning with '\*' and why your  
>>  
>>>> script  
>>>> doesn't print anything  
>>>>  
>>>> e) hint:  
>>>> don't use if, only use a regex to extract the parts you like, and  
>> extract  
>>>> them  
>>>> directly into an array by catching the parts you like  
>>>>  
>>>> ... and post the new code then :-)  
>>>>  
>>>> greetings joe  
>>>>  
>>>>> Thanks  
>>>>> Umesh  
>>>>>  
>>>>> On 12/15/05, John Doe < security.department@xxxxxxx> wrote:  
>>>>>> Umesh T G am Donnerstag, 15. Dezember 2005 13.00:  
>>>>>>> Hi List,  
>>>>>>>  
>>>>>>> Hi Umesh  
>>>>>>>  
>>>>>>> I want to get that value of \*-i\* set in below line.  
>>>>>>>  
>>>>>>> \*Options=-a"hello" -mt -ml3 -i"dir1\dir2" -k -p -i"dir3\dir4"  
>> -m\*  
>>>>>>>  
>>>>>>> What is the best way to get the value of \*-i\* values into one  
>>>>>>>  
>>>>>>> array. For  
>>>>>>>  
>>>>>>> ex: I want the output as: \*dir1\dir2\* and  
>> \*dir3\dir4\* into  
>>>>>>> one  
>>>>>>>  
>>>>>>> array.  
>>>>>>>  
>>>>>>> What possibilities did you try (show us some code), from which  
>> the  
>>>>  
>>>> best  
>>>>  
>>>>>> could be choosed, or to which a better could be added?  
>>>>>>>  
>>>>>>> joe  
>>>>>>>  
>>>>>>> —

Re: split regex

## Re: split regex

>>>>> To unsubscribe, e-mail: [beginners-unsubscribe@xxxxxxxx](mailto:beginners-unsubscribe@xxxxxxxx)  
>>>>> For additional commands, e-mail: [beginners-help@xxxxxxxx](mailto:beginners-help@xxxxxxxx)  
>>>>> <<http://learn.perl.org/>> <<http://learn.perl.org/first-response>>  
>>>>  
>>>> --  
>>>> To unsubscribe, e-mail: [beginners-unsubscribe@xxxxxxxx](mailto:beginners-unsubscribe@xxxxxxxx)  
>>>> For additional commands, e-mail: [beginners-help@xxxxxxxx](mailto:beginners-help@xxxxxxxx)  
>>>> <<http://learn.perl.org/>> <<http://learn.perl.org/first-response>>  
>>  
>> --  
>> To unsubscribe, e-mail: [beginners-unsubscribe@xxxxxxxx](mailto:beginners-unsubscribe@xxxxxxxx)  
>> For additional commands, e-mail: [beginners-help@xxxxxxxx](mailto:beginners-help@xxxxxxxx)  
>> <<http://learn.perl.org/>> <<http://learn.perl.org/first-response>>  
>>  
>>  
>>  
>

---

### • *References:*

- ◆ *[split regex](#)*
    - ◇ *From:* Umesh T G
  - ◆ *[Re: split regex](#)*
    - ◇ *From:* John Doe
  - ◆ *[Re: split regex](#)*
    - ◇ *From:* Umesh T G
  - ◆ *[Re: split regex](#)*
    - ◇ *From:* John Doe
  - ◆ *[Re: split regex](#)*
    - ◇ *From:* Umesh T G
- 
- Prev by Date: *[Re: split regex](#)*
  - Next by Date: *[Re: maximum outgoing connections](#)*
  - Previous by thread: *[Re: split regex](#)*
  - Next by thread: *[Re: split regex](#)*
  - Index(es):
    - ◆ *[Date](#)*
    - ◆ *[Thread](#)*