Modifying a line matching pattern and all following lines starting with character in Vim

0

I have to prepend (comment) line matching a pattern (e.g. the string "foo") with character '*' and all following lines starting with '+'.

In the files I'm working on the syntax uses '+' as the first character of a line to mean that line is part of the preceding line, like:

```
this is a command line continuing through next lines and containing foo
+ this line continues the preciding one with other commands
+ this line continues the preciding one with other commands
this is a new command line
this is another command line
```

After the modification the text should become:

```
*this is a command line continuing through next lines and containing foo
*+ this line continues the preciding one with other commands
*+ this line continues the preciding one with other commands
this is a new command line
this is another command line
```

More minimal example:

```
gas
foo
+ abc
+ def
+ ghi
bar
+ qwe
+ rty
baz
foo
bor
```

Becomes:

```
baz
*foo
bor
```

Share Follow

2 Answers

Active | Oldest | Votes

0

You can select line range based on regex pattern. Sort of

```
:/foo/;/^[^+]/-1 s/^/\*
```

Here we select the lines starting from one containing "foo", until (and excluding) one not starting with "plus". Then simply substitute "the beginning of a line" with a literal "star".

Of course, that's a simplified example, and so the regex is a bit fragile.

To re-execute the last executed command you can press `@` `:` (and after that `@` `@`).
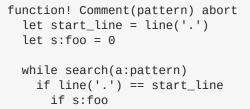
Share Follow

0

I've created a function which comments out lines containing the pattern, but leaves already commented ones. It also check if line continuations contains pattern.

```
function! Comment(pattern) abort
  let start_line = line('.')
  let s:foo = 0

  while search(a:pattern)
    if line('.') == start_line
      if s:foo
```

```
    let f = getline('.') =~ '^+'
    s/.*/\*&
    let l = line('.')
    let i = 0


    " " This commented block is for matching lines above pattern
    "
    " while getline(l-i) =~ '^+'
    "    normal! k
    "    s/.*/\*&
    "    let i += 1
    " endwhile
    "
    " if f
    "    normal! k
    "    s/.*/\*&
    " endif
    "
    " execute ':'.l

    let i = 1
```
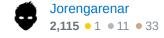
Effect of `:call Comment("foo")`:

```
BEFORE | AFTER
-------|---------
foo    |  *foo
gas    |  gas
foo    |  *foo
+ abc  |  *+ abc
+ def  |  *+ def
+ foo  |  *+ foo
+ ghi  |  *+ ghi
bar    |  bar
+ qwe  |  + qwe
+ rty  |  + rty
baz    |  baz
fab    |  fab
foo    |  *foo
bor    |  bor
hello  |  hello
+ foo  |  *+ foo
abc    |  abc
*foo   |  *foo
*+ g   |  *+ g
```

Share Follow

edited Jul 20 '20 at 21:00

answered Jul 15 '20 at 18:11

Jorengarenar
2,115 ● 1 ● 11 ● 33

yes, it helped me very much. there are those two points to fix. I'm working on it... – comaante Jul 21 '20 at 7:15