How to escape a square bracket for Pattern compilation

Asked 10 years, 5 months ago Active 1 year, 8 months ago Viewed 49k times

26

I have comma separated list of regular expressions:

```
.{8},[0-9],[^0-9A-Za-z ],[A-Z],[a-z]
```

I have done a split on the comma. Now I'm trying to match this regex against a generated password. The problem is that `Pattern.compile` does not like square brackets that is not escaped. Can some please give me a simple function that takes a string like so: `[0-9]` and returns the escaped string `\[0-9\]`.

4

java regex escaping special-characters

edited Mar 29 '16 at 17:49
Cullub
2,061 1 20 36

asked Jul 16 '09 at 20:58
Afamee
4,029 9 32 42

4 Answers

24

You can use `Pattern.quote(String)`.

From the docs:

> `public static String quote(String s)`
>
> Returns a literal pattern `String` for the specified `String`.
>
> This method produces a String that can be used to create a Pattern that would match the string s as if it were a literal pattern.
>
> Metacharacters or escape sequences in the input sequence will be given no special meaning.

edited Apr 10 '18 at 22:25

answered Jul 16 '09 at 21:03

Laurence Gonsalves
114k 28 203 248

What value do you put in for `String` ? `Pattern.quote("\[0-9\]")` ? – Danny Bullis Apr 10 '18 at 22:16

1 @DannyBullis From the question "a simple function that takes a string like so: `[0-9]` and returns the escaped string `\[0-9\]` ". So you'd give this `"[0-9]"` , and it will return something equivalent to `"\[0-9\]"` . (It actually uses `\Q` and `\E` , but the end result has the same effect when given to `Pattern.compile` .) – Laurence Gonsalves Apr 10 '18 at 22:28

awesome. Thanks for the quick help, even 9 years later :) – Danny Bullis Apr 11 '18 at 0:45

34

For some reason, the above answer didn't work for me. For those like me who come after, here is what I found.

I was expecting a single backslash to escape the bracket, however, you must use two if you have the pattern stored in a string. The first backslash escapes the second one into the string, so that what regex sees is `\]` . Since regex just sees one backslash, it uses it to escape the square bracket.

```
\\]
```

In regex, that will match a single closing square bracket.

If you're trying to match a newline, for example though, you'd only use a single backslash. You're using the string escape pattern to insert a newline character into the string. Regex doesn't see `\n` - it sees the newline character, and matches that. You need two backslashes because it's not a string escape sequence, it's a regex escape sequence.

edited Oct 25 '16 at 17:18

answered Jul 29 '15 at 4:50

Cullub
2,061 1 20 36

5 When thinking about it I came up why this is like that: The regex is a String and whatever processes this regex will look for a single backslah as an escape character. However as the regex is passed as a String you have to escape the backslah as well in order to get it into a String properly and that's the readon why you need two backslashes – Raven Mar 27 '16 at 18:51

You can use the \Q and \E special characters...anything between \Q and \E is automatically escaped.

13

```
\Q[0-9]\E
```

answered Jul 16 '09 at 21:06

Dan Breen
10.1k 4 31 47

Sounds a bit perlish if you ask me, have you tried it in java (I havn't, that's why I ask). – Fredrik Jul 18 '09 at 8:30

2 It's valid in Java too: java.sun.com/javase/6/docs/api/java/util/regex/Pattern.html (ctrl-F for "\Q") – MatrixFrog Jul 18 '09 at 8:48

6 In Java string literal format it would be "\\Q[0-9]\\E" or "\\Q" + regex + "\\E". But the quote() method does that for you, plus it deals correctly with strings that already have \E in them. – Alan Moore Jul 19 '09 at 4:55

3

`Pattern.compile()` likes square brackets just fine. If you take the string

```
".{8},[0-9],[^0-9A-Za-z ],[A-Z],[a-z]"
```

and split it on commas, you end up with five perfectly valid regexes: the first one matches eight non-line-separator characters, the second matches an ASCII digit, and so on. Unless you really want to match strings like `".{8}"` and `"[0-9]"`, I don't see why you would need to escape anything.

answered Jul 18 '09 at 8:27

Alan Moore
64.1k 10 84 141