

Backslashes Are Complicated

Backslashes are a bit complicated in Python's regular expressions. The reason being that regular expressions use backslashes to indicate special forms or to allow a special character to be searched for instead of invoking it, such as when we want to search for an dollar sign: `$`. If we didn't backslash that, we'd just be creating an anchor. The issue comes in because Python uses the backslash character for the same thing in literal strings. Let's say you want to search for a string like this (minus the quotes): `"python"`.

To search for this in a regular expression, you will need to escape the backslash but because Python also uses the backslash, then that backslash also has to be escaped so you'll end up with the following search pattern: `"\\python"`. Fortunately, Python supports raw strings by pre-pending the string with the letter `'r'`. So we can make this more readable by doing the following: `r"python"`.

So if you need to search for something with a backslash, be sure to use raw strings or you may end up with some unexpected results!

Wrapping Up

This chapter barely scratches the surface of all you can do with regular expressions. In fact, there is much more to the module itself. There are entire books on the subject of regular expressions, but this should give you the basics to get started. You will need to search for examples and read the documentation, probably multiple times when you're using regular expressions, but they are a handy tool to have when you need it.