

pythex

[Link to this regex](#)

Your regular expression:

[0-9\,\.]+(?:= a)

IGNORECASE

MULTILINE

DOTALL

VERBOSE

Your test string:

4.000,00 a 5.000,00

Match result:

4.000,00 a 5.000,00

Match captures:

No groups.

Regular expression cheatsheet

Special characters

\	escape special characters
.	matches any character
^	matches beginning of string
\$	matches end of string
[5b-d]	matches any chars '5', 'b', 'c' or 'd'
[^a-c6]	matches any char except 'a', 'b', 'c' or '6'
R S	matches either regex R or regex S
()	creates a capture group and indicates precedence

Quantifiers

*	0 or more (append ? for non-greedy)
+	1 or more (append ? for non-greedy)
?	0 or 1 (append ? for non-greedy)
{m}	exactly m occurrences
{m, n}	from m to n . m defaults to 0, n to infinity
{m, n}?	from m to n , as few as possible

Special sequences

\A	start of string
\b	matches empty string at word boundary (between \w and \w)
\B	matches empty string not at word boundary
\d	digit
\D	non-digit
\s	whitespace: [\t\n\r\f\v]
\S	non-whitespace
\w	alphanumeric: [0-9a-zA-Z_]
\W	non-alphanumeric
\Z	end of string
\g<id>	matches a previously defined group

Special sequences

(?iLmsux)	matches empty string, sets re.X flags
(?:...)	non-capturing version of regular parentheses
(?P...)	matches whatever matched previously named group
(?P=)	digit
(?#...)	a comment; ignored
(?=...)	lookahead assertion: matches without consuming
(?!...)	negative lookahead assertion
(?<=...)	lookbehind assertion: matches if preceded
(?<!=...)	negative lookbehind assertion

```
(?          match 'yes' if group 'id' matched, else  
(id)yes|no) 'no'
```

Based on tartley's [python-regex-cheatsheet](#).

Inspired by [Rubular](#). For a complete reference, see the official [re module documentation](#).

Made by [Gabriel Rodríguez](#). Powered by [Flask](#) and [jQuery](#).