

# Regex search

## Interactive regex tutorial

If you want to dedicate a bit of time to learn regex, we recommend using this interactive regex tutorial: <https://regexone.com/>

Alternatively, check to the right for a regex cheat-sheet and below for some quick regex examples.

## Regex search examples

Find any vlan with IDs 1020 or 1030:

```
| vlan (1020|1030)
```

Find any vlan in the 10xx range:

```
| vlan 10\d{2}
```

Find any router with OSPF router-id in 10.0.0.0/24

```
| router-id=10.0.0.\d+
```

Make search case insensitive (will find 'VLAN 1002' or 'vlan 1002')

```
| (?i)vlan 1002
```

Find all lines that start with 'hello' or 'helo' (multiple examples)

```
| (?m)^he{1,2}o  
| (?m)^hell?o
```

Find all lines that end with 'set', ignoring any trailing spaces

```
| (?m)set\h*$
```

Case-insensitive search for all lines starting with 'no' or 'deny', ignoring leading spaces

```
| (?im)^\h*(no|deny)
```

## Regex cheat-sheet

Characters:

Token	Description
.	any single character
\d	any number
\h	any horizontal white-space (space, tab, etc.)

Quantifiers:

Token	Description
?	one or zero times
*	any number of times (even zero - zero or more times)
+	at least once (one or more times)
{2}	repeats 2 times
{2,4}	repeats 2 to 4 times
\	escape character for example '\+' will look for a literal '+' sign

Groups:

Token	Description
[abc]	any of the characters inside ('a' or 'b' or 'c')
[a-z]	any of the characters inside the range (a through z)
	or
()	group - for example '(foo bar)' - 'foo' or 'bar'
\	escape character for example '\(' will look for a literal '(' character

Anchors:

Token	Description
^	start of text if used with 'm' modifier - start of line
\$	end of text if used with 'm' modifier - end of line

Behavior modifiers (flags):

Modifier	Description
----------	-------------

(?i)	Case insensitive
(?m)	'^' and '\$' anchors work per-line instead of on the whole text normally these modifiers would work on the whole text, using the 'm' modifier makes them per-line