

# Cleanor **Regex Cheat Sheet**

Regular expression tokens, quantifiers and flags

Regular expressions match patterns in text. This is the common syntax shared by JavaScript, Python, PCRE and most engines.

## REGEX TOKENS

TOKEN	MATCHES
<code>.</code>	Any single character except newline
<code>\d</code>	A digit (0-9)
<code>\D</code>	A non-digit
<code>\w</code>	A word character (letter, digit, underscore)
<code>\W</code>	A non-word character
<code>\s</code>	Any whitespace (space, tab, newline)
<code>\S</code>	Any non-whitespace
<code>^</code>	Start of the string / line
<code>\$</code>	End of the string / line
<code>\b</code>	A word boundary
<code>\B</code>	Not a word boundary
<code>*</code>	0 or more of the previous
<code>+</code>	1 or more of the previous
<code>?</code>	0 or 1 of the previous (optional)
<code>{n}</code>	Exactly n of the previous
<code>{n,}</code>	n or more of the previous
<code>{n,m}</code>	Between n and m of the previous
<code>[abc]</code>	Any one of a, b or c
<code>[^abc]</code>	Any character except a, b or c
<code>[a-z]</code>	Any character in the range a to z
<code>(...)</code>	A capturing group
<code>(?:...)</code>	A non-capturing group
<code>a b</code>	a or b (alternation)
<code>\1</code>	Backreference to group 1
<code>(?=...)</code>	Lookahead: followed by ...
<code>(?!...)</code>	Negative lookahead: not followed by ...
<code>(?&lt;=...)</code>	Lookbehind: preceded by ...
<code>\.</code>	A literal dot (escape a special char)
<code>flag g</code>	Global: find all matches
<code>flag i</code>	Case-insensitive
<code>flag m</code>	Multiline: ^ and \$ match each line