Regular expressions Pattern matching syntax

CONCEPTS

A regex is matched left to right: at each position the engine tries the whole pattern and moves on if it fails.

Anchor ^ \b \A
↓ then
Atom a . \d [abc] (...)
↓ repeated by
Quantifier ? * + {n,m}
↓ joined by
Alternation a|b

Flavours differ. This sheet follows PCRE2 (Perl-style), also used by PHP and grep -P; JavaScript, Python and POSIX tools lack some features.

Prefer a specific class over .* : greedy wildcards cause surprising matches and slow backtracking.

CHARACTERS

`.`
Any char except newline

`\d` `\w` `\s`
Digit, word char, space

`[abc]`
Any one of a, b, c

`[^abc]`
Any char except a, b, c

`[a-z0-9]`
Ranges inside a set

`\.`
Escape a metacharacter

`\n` `\t` `\r`
Newline, tab, carriage return

`\x41` `\x{263A}`
Character by hex code

`[[:alpha:]]`
POSIX class inside a set

`\h` `\v` `\R`
Horizontal space, vertical space, any line break

`\Q...\E`
Treat text literally

CHARACTERS (cont.)

`\p{L}` `\p{Lu}`
Unicode letter, uppercase letter

QUANTIFIERS

`*`
Zero or more

`+`
One or more

`?`
Zero or one (optional)

`{n}`
Exactly n times

`{n,m}`
Between n and m times

`{n,}`
At least n times

`*?` `+?` `??`
Lazy: as few as possible

`*+` `++` `?+`
Possessive: never give back

ANCHORS

`^` `$`
Start, end of line

`\b`
Word boundary

`\B`
Not a word boundary

`\A` `\z`
Start, end of text

GROUPS

`(...)`
Capture group

`a|b`
Alternation

`(?:...)`
Group without capturing

`(?<name>...)`
Named capture group

`\1` `\2`
Backreference to group n

`\k<name>`
Backreference by name

LOOKAROUND

`(?=...)`
Followed by (lookahead)

`(?!...)`
Not followed by

`(?<=...)`
Preceded by (lookbehind)

`(?<!...)`
Not preceded by

`\K`
Reset start of reported match

Zero-width checks

Syntax	Asserts
`(?=x)`	x comes next
`(?!x)`	x does not come next
`(?<=x)`	x comes before
`(?<!x)`	x does not come before

FLAGS

`(?i)`
Ignore case

`(?m)`
^ and $ match at each line

`(?s)`
Dot matches newline

`(?x)`
Free-spacing: ignore blanks, allow #comments

ADVANCED

`(?R)`
Recurse the whole pattern

RECIPES

Integer or decimal

```
^-?\d+(?:\.\d+)?$
```

ISO date with named parts

```
(?<y>\d{4})-(?<m>\d{2})-(?
<d>\d{2})
```

Trim trailing whitespace

```
# match, replace with nothing
[ \t]+$   (with (?m))
```

Value after a key

```
# match only the number
price=\K\d+
# or with lookbehind
(?<=price=)\d+
```