



**Anchors**

**Sample Patterns**

^ Start of line +

\A Start of string +

$ End of line +

\Z End of string +

\b Word boundary +

\B Not word boundary +

\< Start of word

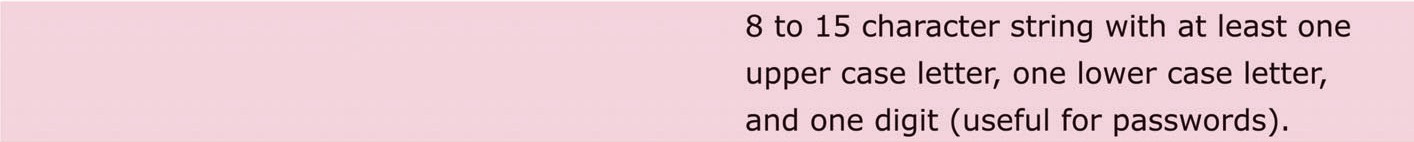
\> End of word

([A-Za-z0-9-]+)

(\d{1,2}\/\d{1,2}\/\d{4})

([^\s]+(?=\.(jpg|gif|png))\.\2)

(^[1-9]{1}$|^[1-4]{1}[0-9]{1}$|^50$)

(#?([A-Fa-f0-9]){3}(([A-Fa-f0-9]){3})?)

((?=.\*\d)(?=.\*[a-z])(?=.\*[A-Z]).{8,15})

Letters, numbers and hyphens Date (e.g. 21/3/2006)

jpg, gif or png image

Any number from 1 to 50 inclusive

Valid hexadecimal colour code

8 to 15 character string with at least one

upper case letter, one lower case letter, and one digit (useful for passwords).

**Character Classes**



(\w+@[a-zA-Z\_]+?\.[a-zA-Z]{2,6})

(\<(/?[^\>]+)\>)

Email addresses

\c

HTML Tags

\s

\S

\d

\D

\w

\W

\xhh

\Oxxx

**POSIX Character Classes**

[:upper:]

[:lower:]

[:alpha:]

[:alnum:]

[:digit:]

[:xdigit:]

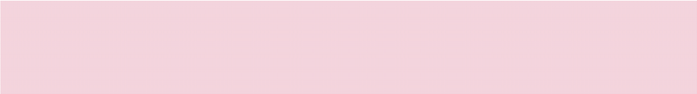
[:punct:]

[:blank:]

[:space:]

[:cntrl:]

[:graph:]

[:print:]

[:word:]

**Assertions**

?<=



?=

?!

Lookahead assertion +

Negative lookahead +

Control character White space

Not white space Digit

Not digit Word Not word

Hexadecimal character hh Octal character xxx

Upper case letters Lower case letters All letters

Digits and letters Digits Hexadecimal digits Punctuation

Space and tab Blank characters Control characters Printed characters

Printed characters and spaces

Digits, letters and underscore

Lookbehind assertion +

\*

*These patterns are intended for reference purposes and have not been extensively tested. Please use with caution and test thoroughly before use.*

**Note**

**Quantifiers**

\*?

+

+?

?

??

{3}

{3,}

{3,5}

{3,5}? 3, 4 or 5, ungreedy +

\

**Special Characters**

\n

\r

\t

\v

\f

\a

[\b] Backspace

\e

\N{name} Named Character

$n

**String Replacement (Backreferences)**

$2



?!= or ?<!

Negative lookbehind +

$1

0 or more +

1. or more, ungreedy +
2. or more +

1 or more, ungreedy +

0 or 1 +

0 or 1, ungreedy +

Exactly 3 + 3 or more + 3, 4 or 5 +

Escape Character + New line + Carriage return + Tab +

Vertical tab + Form feed + Alarm

Escape

nth non-passive group "xyz" in /^(abc(xyz))$/ "xyz" in /^(?:abc)(xyz)$/

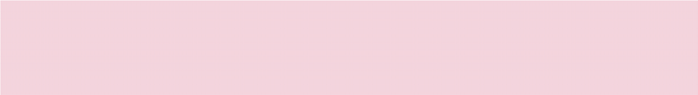
.

**Ranges**

(a|b) (...)

(?:...)

[abc]

[^abc] [a-q]

[A-Q]

[0-7]

*\n*

*Ranges are inclusive.*

**Note**

g i m s x

**Pattern Modifiers**

e U

Any character except new line (\n) +

a or b +

Group +

Passive Group + Range (a or b or c) + Not a or b or c +

Letter between a and q + Upper case letter + between A and Q +

Digit between 0 and 7 +

nth group/subpattern +

Global match Case-insensitive Multiple lines

Treat string as single line Allow comments and white space in pattern Evaluate replacement Ungreedy pattern

?>

**Metacharacters (must be escaped)**

?()

?()|

?#

Once-only Subexpression $`

Condition [if then] $'

Condition [if then else] $+

Comment $&

$\_ Entire input string

$$

**Note**

Items marked + should work in most regular expression implementations.

Before matched string After matched string Last matched string Entire matched string

Literal "$"

^ [ .

$ { \*

( \ +

) | ?

< >

*Available free from AddedBytes.com*