### Anchors
- `^` Start of string
- `\A` Start of string
- `$` End of string
- `\Z` End of string
- `\b` Word boundary
- `\B` Not word boundary
- `<` Start of word
- `>` End of word

### Quantifiers
- `*` 0 or more
- `+` 1 or more
- `?` 0 or 1
- `{3}` Exactly 3
- `{3,}` 3 or more
- `{3,5}` 3, 4 or 5

### Quantifier Modifiers
- `x?` Ungreedy version of “x”

### Character Classes
- `\c` Control character
- `\s` White space
- `\S` Not white space
- `\d` Digit
- `\D` Not digit
- `\w` Word
- `\W` Not word
- `\x` Hexadecimal digit
- `\O` Octal digit

### POSIX
- `[::upper:]` Upper case letters
- `[::lower:]` Lower case letters
- `[::alpha:]` All letters
- `[::alnum:]` Digits and letters
- `[::digit:]` Digits
- `[::xdigit:]` Hexadecimal digits
- `[::punct:]` Punctuation
- `[::blank:]` Space and tab
- `[::space:]` Blank characters
- `[::cntrl:]` Control characters
- `[::graph:]` Printed characters
- `[::print:]` Printed characters and spaces
- `[::word:]` Digits, letters and underscore

### Groups and Ranges
- `.` Any character except new line (\n)
- `(a|b)` a or b
- `(\d)` Group
- `(?::...)` Passive Group
- `[\abcd]` Range (a or b or c)
- `[^abcd]` Not a or b or c
- `[a-q]` Letter between a and q
- `[A-Q]` Upper case letter between A and Q
- `[0-7]` Digit between 0 and 7
- `\n` nth group/subpattern

Note: Ranges are inclusive.

### Escape Character
- `\` Escape Character

### Metacharacters (must be escaped)
- `^` [ . `\` `{` `*` `( ` `)` `|` `?` `<` `>`

### Special Characters
- `\n` New line
- `\r` Carriage return
- `\t` Tab
- `\v` Vertical tab
- `\f` Form feed
- `\x` Octal character
- `\xhh` Hex character

### String Replacement (Backreferences)
- `$n` nth non-passive group
- `$2` “xyz” in /^\(abc\)xyz$/
- `$1` “xyz” in /^\(?abc\)xyz$/
- `$` Before matched string
- `$` After matched string
- `$+` Last matched string
- `$&` Entire matched string

### Sample Patterns
- `([A-Za-z0-9-]+)` Letters, numbers and hyphens
- `\d(1,2)\d(1,2)\d(4)` Date (e.g. 21/3/2006)
- `[^\s]+(\s+\(jpg\|gif\|png\))\s+` jpg, gif or png image
- `^[1-9]\{1\}[0-9]\{1\}[0-9]\{1\}[0-9]\{5\}$` Any number from 1 to 50 inclusive
- `[#?[(A-Fa-f0-9)]\{3\}((A-Fa-f0-9)]\{3\}?)` Valid hexadecimal colour code
- `((?:[A-Za-z])\{8,15\})` String with at least one upper case letter, one lower case letter, and one digit (useful for passwords).
- `\w+[@a-zA-Z]_+?\[a-zA-Z]\{2,6\}` Email addresses
- `\<\(?[^\^]+\)+\>` HTML Tags

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