Regular Expressions Cheat Sheet

Metacharacters (must be escaped)

```
Symbol      Meaning
------      ----------------------
^           Start of string
$           End of string
( )         Grouping
< >         Word boundaries
.           Any character
* + ?       Quantifiers
[ ]         Character class
{ }         Repetition
\           Escape
|           Alternation
```

Quantifiers

```
Symbol      Meaning
------      ----------------------
*           0 or more
+           1 or more
?           0 or 1
{3}         Exactly 3
{3,}        3 or more
{3,5}       3 to 5
```

Anchors

```
Symbol      Meaning
------      ----------------------
^, \A       Start of string
$, \Z       End of string
\b          Word boundary
\B          Not word boundary
\<, \>      Word boundaries
```

POSIX Classes

```
[:upper:]     Uppercase letters
[:lower:]     Lowercase letters
[:alpha:]     All letters
[:alnum:]     Letters and digits
[:digit:]     Digits
[:xdigit:]    Hex digits
[:punct:]     Punctuation
[:space:]     Space characters
[:word:]      Letters, digits, underscore
```

Character Classes

```
Symbol      Meaning
------      ----------------------
\c          Control character
\s, \S      Whitespace / not whitespace
\d, \D      Digit / not digit
\w, \W      Word / not word
\x, \0      Hex / Octal digit
```

Special Characters

```
Symbol      Meaning
------      ----------------------
\n          Newline
\r          Carriage return
\t          Tab
\v          Vertical tab
\f          Form feed
\xxx        Octal character
\xhh        Hex character
```

Pattern Modifiers

```
g     Global match
i     Case-insensitive
m     Multi-line
s     Dot matches newline
x     Allow whitespace/comments
e     Evaluate replacement
U     Ungreedy
```

Assertions

```
(?=...)     Lookahead
(?!...)     Negative lookahead
(?<=...)    Lookbehind
(?<!...)    Negative lookbehind
(?>...)     Atomic group
(?()|)      Conditional
(?#...)     Comment
```

Replacement Tokens

```
$n        nth non-passive group
$1, $2    Match from group 1, 2
$`        Before match
$'        After match
$+        Last match
$&        Entire match
```

Sample Patterns

```
([A-Za-z0-9-]+)                     Alphanumeric +
    hyphens
(\d{1,2}/\d{1,2}/\d{4})             Date (e.g.
    21/3/2006)
([^\s]+(?=\.(jpg|gif|png))\.\2)     Image extensions
(^[1-9]$|^[1-4][0-9]$|^50$)         Number -150
(#?([A-Fa-f0-9]){3,6})             Hex color code
(?=.*\d)(?=.*[a-z])(?=.*[A-Z])     Strong password
(\w+@[a-zA-Z_]+?\.[a-zA-Z]{2,6})   Email
(\<(/?[^\>]+)\>)                    HTML tags
```