### Anchors

<table>
<thead>
<tr>
<th>Character</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>^</td>
<td>Start of line +</td>
</tr>
<tr>
<td>\A</td>
<td>Start of string +</td>
</tr>
<tr>
<td>$</td>
<td>End of line +</td>
</tr>
<tr>
<td>\Z</td>
<td>End of string +</td>
</tr>
<tr>
<td>\b</td>
<td>Word boundary +</td>
</tr>
<tr>
<td>\B</td>
<td>Not word boundary +</td>
</tr>
<tr>
<td>&lt;</td>
<td>Start of word</td>
</tr>
<tr>
<td>&gt;</td>
<td>End of word</td>
</tr>
</tbody>
</table>

### Sample Patterns

- `(\[A-Za-z0-9-\]+)`
- `(\d{1,2}(/\d{1,2}(/\d{4}))` (Date e.g. 21/3/2006)
- `([^\s]+(?!.*\.(jpg|gif|png))\.(?))` (jpg, gif or png image)
- `(^[\1-\9]{1}\$|^\[\4-\9]{1}\$|^\$\$\$\$\$\$\$\$)$` (Any number from 1 to 50 inclusive)
- `([^A-Za-f\0-9])\(\((\[A-Za-f\0-9])\(\))\)` (Valid hexadecimal colour code)
- `(\[a-zA-Z]+\@[a-zA-Z_]+\.[a-zA-Z]{2,6})` (Email addresses)
- `^(\w+@[a-zA-Z_]+\?[a-zA-Z]+?\{2,6})` (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.

### Quantifiers

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

### Ranges

- `.` Any character except new line
- `(a|b)` a or b +
- `\{2,\}` Group +
- `\{a-q\}` Letter between a and q +
- `\^[a-q\]` Letter between A and Q +
- `[0-7]` Digit between 0 and 7 +

### POSIX Character Classes

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

### Sample Patterns

- `[^a-zA-Z_\]+(?=.*\.(jpg|gif|png))\.(?))` (jpg, gif or png image)
- `([^A-Za-f\0-9])\(\((\[A-Za-f\0-9])\(\))\)` (Valid hexadecimal colour code)
- `(\[a-zA-Z]+\@[a-zA-Z_]+\.[a-zA-Z]{2,6})` (Email addresses)
- `^(\w+@[a-zA-Z_]+\?[a-zA-Z]+?\{2,6})` (HTML Tags)

### Note

Ranges are inclusive.

### Pattern Modifiers

- `g` Global match
- `i` Case-insensitive
- `m` Multiple lines
- `s` Treat string as single line
- `x` Allow comments and white space in pattern
- `e` Evaluate replacement
- `U` Ungreedy pattern

### String Replacement (Backreferences)

- `$n` nth non-passive group
- `\s$` "xyz" in /\s(\s(xy)\s)/
- `$1` "xyz" in /\$1(\s(xy)\s)/
- `$s` Before matched string
- `\$` After matched string
- `$+` Last matched string
- `\&` Entire matched string
- `\$` Entire input string
- `\$` Literal "$"

### Sample Patterns

- `^[\1-\9]{1}\$|^\[\4-\9]{1}\$|^\$\$\$\$\$\$\$\$)$` (Any number from 1 to 50 inclusive)
- `([^A-Za-f\0-9])\(\((\[A-Za-f\0-9])\(\))\)` (Valid hexadecimal colour code)
- `(\[a-zA-Z]+\@[a-zA-Z_]+\.[a-zA-Z]{2,6})` (Email addresses)
- `^(\w+@[a-zA-Z_]+\?[a-zA-Z]+?\{2,6})` (HTML Tags)

### Note

Items marked + should work in most regular expression implementations.

### Available free from

AddedBytes.com