### Anchors
- `^` Start of string, or start of line in multi-line pattern
- `\A` Start of string
- `$` End of string, or end of line in multi-line pattern
- `\Z` End of string
- `\b` Word boundary
- `\B` Not word boundary
- `<` Start of word
- `>` End of word

### 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

### Assertions
<table>
<thead>
<tr>
<th>Pattern</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>?=</code></td>
<td>Lookahead assertion</td>
</tr>
<tr>
<td><code>?!</code></td>
<td>Negative lookahead</td>
</tr>
<tr>
<td><code>?&lt;=</code></td>
<td>Lookbehind assertion</td>
</tr>
<tr>
<td><code>?!</code> or <code>?!</code></td>
<td>Negative lookbehind</td>
</tr>
<tr>
<td><code>?&gt;</code></td>
<td>Once-only Subexpression</td>
</tr>
<tr>
<td><code>()</code></td>
<td>Condition [if then]</td>
</tr>
<tr>
<td>`(()</td>
<td>)`</td>
</tr>
<tr>
<td><code>#</code></td>
<td>Comment</td>
</tr>
</tbody>
</table>

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

### Escape Sequences
- `\` Escape following character
- `\Q` Begin literal sequence
- `\E` End literal sequence

"Escaping" is a way of treating characters which have a special meaning in regular expressions literally, rather than as special characters.

### Common Metacharacters
- `^` Start of string
- `{` Start of pattern
- `*` Repeated
- `(` Group
- `)` End of group
- `+` One or more
- `?` Zero or one
- `>` Once-only
- `<` Start of pattern

The escape character is usually `\`.

### Groups and Ranges
- `.` Any character except new line (`\n`)
- `(a|b)` a or b
- `(\d)` Digit
- `([abc])` Range (a or b or c)
- `([^abc])` Not (a or b or c)
- `(a-q)` Lower case letter from a to q
- `(A-Q)` Upper case letter from A to Q
- `(0-7)` Digit from 0 to 7
- `\` Group/subpattern number "x"

### Pattern Modifiers
- `g` Global match
- `i` Case-insensitive
- `m` Multiple lines
- `s` Treat string as single line
- `x` Allow comments and whitespace in pattern
- `e` Evaluate replacement
- `u` Ungreedy pattern
- `*` PCRE modifier

### String Replacement
- `$n` nth non-passive group
- `$2` "xyz" in `/\(\d(xy)z)$/`
- `$1` "xyz" in `/\(?:\d(\xyz)$/`
- `$` Before matched string
- `$` After matched string
- `$+` Last matched string
- `$&` Entire matched string

Some regex implementations use `\` instead of `$`.