

#### -----Atomic Characters-----

\s - match any whitespace character (including new line)  
\S - match any non-whitespace character  
\w - match any "word" character (alphanumeric + '\_' )  
\W - match any non-word character  
\d - match any digit character  
\D - match any non-digit character

#### -----Non-Atomic (meta)Characters-----

\b - true when at a word boundary (non-atomic)  
\B - true when not at a word boundary (non-atomic)  
\A - true at the beginning of a string  
\Z - true at the end of string only  
\z - true at the end of string or before optional new line

#### -----Regex Quantifiers-----

\* - match 0 or more  
+ - match 1 or more  
? - match 1 or 0 times (optional characters)  
{A} - match exactly A times  
{A,} - match at least A times  
{A,B} - match at least A but no more than B times

#### -----Other-----

() - string grouping (e.g. (123) must match "123")  
[] - character grouping (e.g. [123] can match 1, 2, or 3)  
. - match any one character (wild card) (except new line)  
^ - true at the beginning of a string (or at after new line, maybe)  
\$ - true at the end of a string (or after before new line, maybe)  
| - specifies or (e.g. (1|2) can match 1 or 2)

#### ----- Assertions -----

g - progressive matching (don't go back to the beginning when a match is made)  
i - ignore case  
m - make ^ match at the beginning of a new line and \$ match at the end of a line (just before \n). ^ and \$ will still match at the beginning and end of the text  
x - ignore (most) whitespace and permit comments in pattern

#### -----Useful Hints-----

=~ m/\Q\$notRegex\E/ - leave text as is and use as a regular expression (useful for reading in strings and then treating them as a regular expression)  
[<characters>] - matches anything but <characters>  
=~ m/(<pattern>)?/ - non-greedy match (match 1 or 0 times) but can be used with normal regular expressions (useful when parsing files with the same pattern repeated)

#### -----Useful Perl-----

split/<pattern>/<expression> - splits the <expression> on the <pattern>.  
If <pattern> is in () it will get captured  
join <expression> <list> - joins the <list> into a single string separated by the <expression>