Regular Expressions

Ramesh S

Characters

Character	Description
.	a single character
\s	a whitespace character (space, tab, newline)
\S	non-whitespace character
\d	a digit (0-9)
\D	a non-digit
\w	a word character (a-z, A-Z, 0-9, _)
\W	a non-word character
[aeiou]	matches a single character in the given set
[^aeiou]	matches a single character outside given set
(foo|bar|baz)	matches any of the alternatives specified

Quantifiers

Character	Description
*	zero or more of the previous thing
+	one or more of the previous thing
?	zero or one of the previous thing
{3}	matches exactly 3 of the previous thing
{3,6}	matches between 3 and 6 of the previous thing
{3,}	matches 3 or more of the previous thing

m?ethane

- would match either "ethane" or "methane".

comm?a

- would match either "coma" or "comma".

ab*c

- would match any string that starts with an "a", is followed by a sequence of "b"'s, and ends with a "c".
- would match "ac", "abc", "abbc", "abbbc", "abbbbbbbbc"

ab+c

- would not match "ac",
- but it would match "abc", "abbc", "abbbc", "abbbbbbbbc" and so on.

Working with back slash

- Note that the placement of backslash is important.

a\.*z

- Matches any string starting with "a", followed by a series of periods (including the "series" of length zero), and terminated by "z". Thus, "az", "a.z", "a..z", "a...z" and so forth are all matched.

a.*z

- (Note that the backslash and period are reversed in this regular expression.)
- Matches any string starting with an "a", followed by one arbitrary character, and terminated with "*z". Thus, "ag*z", "a5*z" and "a@*z" are all matched. Only strings of length four, where the first character is "a", the third "*", and the fourth "z", are matched.

a\++z

- Matches any string starting with "a", followed by a series of plus signs, and terminated by "z". There must be at least one plus sign between the "a" and the "z". Thus, "az" is not matched, but "a+z", "a++z", "a+++z", etc. will be matched.

a\+\+z

- Matches only the string "a++z".

a+\+z

- Matches any string starting with a series of "a"'s, followed by a single plus sign and ending with a "z". There must be at least one "a" at the start of the string. Thus "a+z", "aa+z", "aaa+z" and so on will match, but "+z" will not.

a.?e

- Matches "ace", "ale", "axe" and any other three-character string beginning with "a" and ending with "e"; will also match "ae".

a\.?e

- Matches "ae" and "a.e". No other string is matched.

a.\?e

- Matches any four-character string starting with "a" and ending with "?e". Thus, "ad?e", "a1?e" and "a%?e" will all be matched.

a\.\?e

- Matches only "a.?e" and nothing else.

1\.\d\d\d\d\d

- would match any six-digit floating-point number from 1.00000 to 1.99999 inclusive.

a\wz

- would match "abz", "aTz", "a5z", "a_z", or any three-character string starting with "a", ending with "z", and whose second character was either a letter (upper- or lower-case), a number, or the underscore.

a\Wz

- would not match "abz", "aTz", "a5z", or "a_z". It would match "a%z", "a{z", "a?z" or any three-character string starting with "a" and ending with "z" and whose second character was not a letter, number, or underscore. (This means the second character must either be a symbol or a whitespace character.)

- would match any three-character string starting with "a" and ending with "z" and whose second character was a space, tab, or newline. Likewise,

a\Sz

- would match any three-character string starting with "a" and ending with "z" whose second character was not a space, tab or newline. (Thus, the second character could be a letter, number or symbol.)

\007

- There is one other metacharacter starting with a backslash, the octal metacharacter. The octal metacharacter looks like this: "\nnn", where "n" is a number from zero to seven. This is used for specifying control characters that have no typed equivalent.
- would find all subjects with an embedded ASCII "bell" character. (The bell is specified by an ASCII value of 7.) You will rarely need to use the octal metacharacter.

ab{3,5}c

- There are three other metacharacters that may be of use. The first is the braces metacharacter. This metacharacter follows a normal character and contains two number separated by a comma (,) and surrounded by braces ({}). It is like the star metacharacter, except the length of the string it matches must be within the minimum and maximum length specified by the two numbers in braces.
- will match "abbbc", "abbbbc" or "abbbbbc". No other string is matched.

.{3,5}pentane

- will match "cyclopentane", "isopentane" or "neopentane", but not "n-pentane"

a[1234-]z or a[1-4-]z

- both do the same thing. They both match "a1z", "a2z", "a3z", "a4z" or "a-z", and nothing else.

textfile0[^02468]

- matches any ten-character string starting with "textfile0" and ending with anything except an even number.

[^f-h]ood

- Inversion and ranges can be combined, so that
- matches any four letter wording ending in "ood" except for "food", "good" or "hood". (Thus "mood" and "wood" would both be matched.)

Advanced matches

- \bcat\b # Matches 'the cat sat' but not 'cat on the mat'
- \Bcat\B # Matches 'verification' but not 'the cat on the mat'
- \bcat\B # Matches 'catatonic' but not 'polecat'
- \Bcat\b # Matches 'polecat' but not 'catatonic'

Some Examples

^$

Match a blank line

(\d\s){3}

Match three digits, each followed by a whitespace
character (eg "3 4 5 ")

(a.)+

Matches a string in which every odd-numbered letter is a (eg "abacadaf")

^\d+

string starts with a number

\d+$

string that ends with a number