

BBEdit grep

quick reference

support@barebones.com
www.barebones.com/support

by Charles Poynton, PhD
www.poynton.ca

Characters in grep patterns generally match themselves. Metacharacters, to the right, must be preceded by backslash (\) to match literally.

A character class [abc] matches at most one character, but may be qualified by +, *, ?, {n} etc. A-Z indicates a range. Within a character class, characters such as . * | match themselves with no special meaning. To match a hyphen, place it first or backslash-escape it. Backslash escape \] matches], \\ matches \.

PRIMITIVES			
\A	beginning of file	\z	end of file
\Z	end of file (or line end immediately preceding end of file)		
\b	word boundary	\B	not a word boundary
^	beginning of line		
\$	end of line		
.	(period) any character except lineend		
\\ \^ \.	literal backslash, caret, period; also use for \$() * ? { } [] +		
\r \n	'hard' line break (linefeed), both treated identically by BBEEdit		
\t	tab	\f	page break (form feed)
\0377	(backslash, zero) octal code [deprecated]		
\x NN	hexadecimal code	\x{NNNN}	hexadecimal code
\s	whitespace character	\S	non-whitespace char
\w	word char, equivalent to [a-zA-Z0-9etc]	\W	non-word char (incl. line end)
\d	digit	\D	non-digit (incl. line end)
[abc]	any one of several characters	[a-z]	any one character in range
[^abcx-z]	any one character <i>not</i> among specified characters or range[s]		
REPLACEMENT PATTERNS		ALTERNATIVES/SUBPATTERNS	
&	entire match	p q	alternation, match p or match q, or (p q r s) etc.
\1	k th subpattern	(p)	subpattern (capture)
\99		\1 .. \99	backreference to subpattern capture
\P<a>	named subpattern	(?:p)	subpattern (noncapture / cluster-only)
\u	make next character upper/lowercase	(?P<a>p)	name a for subpattern p (named capture)
\l	(case specifier)	(?P=a)	backreference to named capture
MULTIPLES			
\U	make characters upper/lowercase	p?	zero or one (greedy)
\L	until case specifier	p**	zero or one (nongreedy)
\E	end case transform	p*	zero or more (greedy)
		p+?	zero or more (nongreedy)
		p{n}	one or more (greedy)
		p{n}?	one or more (nongreedy)
		p{m,}	exactly n (greedy)
		p{m,}?	exactly n (nongreedy)
		p{m,n}	at least m (greedy)
		p{m,n}?	at least m (nongreedy)
		p{m,n}	at least m, but no more than n
		p{m,n}?	at least m, but no more than n (nongreedy)
POSITIONAL ASSERTIONS			
(?!x)	negative lookbehind: match subpattern only if <i>not</i> preceded by x	(?!x)	negative lookahead: match subpattern only if <i>not</i> followed by x
(?<=x)	positive lookbehind: match subpattern only if preceded by x	(?=x)	positive lookahead: match subpattern only if followed by x
CONDITIONAL SUBPATTERNS			
(?(k)y)	if subpattern number k matched, attempt match of y-clause; else, skip. for named subpattern, (?(<a>)y)		
(?(k)y n)	if subpattern number k matched, attempt match of y-clause; else, attempt match of n-clause		
ONCE-ONLY SUBPATTERNS (NONCAPTURING)			
(?>p)	If p matches, prevent backtracking in p		