

Anchors		Quantifiers		Groups and Ranges	
^	Start of string	*	0 or more	.	Any character except new line (\n)
\A	Start of string	+	1 or more	(a b)	a or b
\$	End of string	?	0 or 1	(...)	Group
\Z	End of string	{3}	Exactly 3	(?:...)	Passive Group
\b	Word boundary	{3,}	3 or more	[abc]	Range (a or b or c)
\B	Not word boundary	{3,5}	3, 4 or 5	[^abc]	Not a or b or c
\<	Start of word			[a-q]	Letter between a and q
\>	End of word			[A-Q]	Upper case letter between A and Q
		Quantifier Modifiers		[0-7]	Digit between 0 and 7
Character Classes		"x" below represents a quantifier		\n	nth group/subpattern
\c	Control character	x?	Ungreedy version of "x"		Note: Ranges are inclusive.
\s	White space			Pattern Modifiers	
\S	Not white space	Escape Character		g	Global match
\d	Digit	\	Escape Character	i	Case-insensitive
\D	Not digit			m	Multiple lines
\w	Word	Metacharacters (must be escaped)		s	Treat string as single line
\W	Not word			x	Allow comments and white space in pattern
\x	Hexadecimal digit	^	[.	
\O	Octal digit	\$	{	*	
		(\	+	
POSIX)		?	
[:upper:]	Upper case letters	<	>		
[:lower:]	Lower case letters				
[:alpha:]	All letters	Special Characters		String Replacement (Backreferences)	
[:alnum:]	Digits and letters	\n	New line	\$n	nth non-passive group
[:digit:]	Digits	\r	Carriage return	\$2	"xyz" in /^(abc(xyz))\$/
[:xdigit:]	Hexadecimal digits	\t	Tab	\$1	"xyz" in /^(?:abc)(xyz)\$/
[:punct:]	Punctuation	\v	Vertical tab	\$`	Before matched string
[:blank:]	Space and tab	\f	Form feed	\$'	After matched string
[:space:]	Blank characters	\xxx	Octal character xxx	\$+	Last matched string
[:cntrl:]	Control characters	\xhh	Hex character hh	\$&	Entire matched string
[:graph:]	Printed characters				
[:print:]	Printed characters and spaces	Sample Patterns			
[:word:]	Digits, letters and underscore	Pattern		Will Match	
Assertions		([A-Za-z0-9-]+)		Letters, numbers and hyphens	
?=	Lookahead assertion	(\d{1,2}\V\d{1,2}\V\d{4})		Date (e.g. 21/3/2006)	
?!	Negative lookahead	([^\s]+(?:=\.(jpg gif png))\.\2)		jpg, gif or png image	
?<=	Lookbehind assertion	(^[1-9]{1}\$ ^[1-4]{1}[0-9]{1}\$ ^[50]\$)		Any number from 1 to 50 inclusive	
?!= or ?<!	Negative lookbehind	(#?([A-Fa-f0-9]){3}((([A-Fa-f0-9]){3})?))		Valid hexadecimal colour code	
?>	Once-only Subexpression	((?=[.*\d])(?=[.*[a-z])(?=[.*[A-Z])).{8,15})		String with at least one upper case letter, one lower case letter, and one digit (useful for passwords).	
?()	Condition [if then]	(\w+@[a-zA-Z_]+?\. [a-zA-Z]{2,6})		Email addresses	
?()	Condition [if then else]	(\<(/?[^\>]+\>)		HTML Tags	
?#	Comment				
					</