

GREP & Metacharacters 2017

in the InDesign CC 2017 Find/Change dialog

Description	Text	GREP	Text	GREP	Text	GREP	Text	GREP	
Tab	^t	^t	Any Double Quotation Mark	"	"	Locations		GREP	
Forced Line Break	^n	^n	Any Single Quotation Mark	'	'	Beginning of Word	^<	Found Text	\$0
End of Paragraph	^p	^r	Straight Double Quotation Mark	"	"	End of Word	^>	Found 1	\$1
Any Line End Character (NEW!)		^R	Double Left Quotation Mark	^{"	~{"	Word Boundary	^b	Found 2	\$2
Symbols	Text	GREP	Double Right Quotation Mark	^}"	~}"	Beginning of Paragraph	^	Found 3	\$3
Bullet Character	^8	~8	Straight Single Quotation Mark	^{'	~{'	End of Paragraph	^\$	Found 4	\$4
Japanese Bullet	^5	~5	Single Left Quotation Mark	^[{'	~[{'	Beginning of Story	^A	Found 5	\$5
Carat Character	^^	^A	Single Right Quotation Mark	^['	~['	End of Story	^Z	Found 6	\$6
Backslash Character	^ \	^ \	Break Character	Text	GREP	Repeat	GREP	Found 7	\$7
Copyright Symbol	^2	~2	Standard Carriage Return	^b	~b	Zero or One Time	?	Found 8	\$8
Ellipsis	^e	~e	Column Break	^M	~M	Zero or More Times	*	Found 9	\$9
Tilde	~	^~	Frame Break	^R	~R	One or More Times	+	Undocumented	GREP
Paragraph Symbol	^7	~7	Page Break	^P	~P	Zero or One Time (Shortest Match)	??	Letter	^p{letter}
Registered Trademark Symbol	^r	~r	Odd Page Break	^L	~L	Zero or More Times (Shortest Match)	^*	Lowercase letter	^p{lowercase_letter}
Section Symbol	^6	~6	Even Page Break	^E	~E	One or More Times (Shortest Match)	+?	Uppercase letter	^p{uppercase_letter}
Trademark Symbol	^d	~d	Discretionary Line Break	^k	~k	Repeat Expression 1-9 Times	^ \1 ... \9	Titlecase letter	^p{titlecase_letter}
Open Parenthesis	(^(\	Variable	Text	GREP	Match	GREP	Spacing modifier letter	^p{modifier_letter}
Closed Parenthesis)	^)\	Any Variable	^v	~v	Marking Subexpression	()	Non-case non-modifier letters	^p{letter_other}
Open Brace Character	{	^{\	Running Header (Paragraph Style)	^Y	~Y	Non-Marking Subexpression	^(?)	Marks	^p{mark}
Closed Brace Character	}	^}\	Running Header (Character Style)	^Z	~Z	Character Set	^[]	Diacritical and tone marks	^p{non_spacing_mark}
Open Bracket Character	[^[Custom Text Variable	^u	~u	Or		Vowels SE Asian languages	^p{spacing_combining_mark}
Closed Bracket Character]	^]\	Last Page Number Variable	^T	~T	Positive Lookbehind	^(?<=)	Circles, squares, keycaps	^p{enclosing_mark}
Markers	Text	GREP	Chapter Number Variable	^H	~H	Negative Lookbehind	^(?!)	Spaces, returns, 2028, 2029	^p{separator}
Any Page Number	^#	~#	Creation Date Variable	^O	~O	Positive Lookahead	^(?=)	All spaces except tab & return	^p{space_separator}
Current Page Number	^N	~N	Modification Date Variable	^o	~o	Negative Lookahead	^(?!)	Line separator character 2028	^p{line_separator}
Next Page Number	^X	~X	Output Date Variable	^D	~D	Lookbehind (NEW!)	^k	Paragraph separator 2029	^p{paragraph_separator}
Previous Page Number	^V	~V	File Name Variable	^1	~1	(Keep text found so far out of overall Regex match)		Math and Wingdings	^p{symbol}
Section Marker	^x	~x	Metadata Caption	^J	~J	Modifiers	GREP	Math symbols	^p{math_symbol}
Anchored Object Marker	^a	~a	Other	Text	GREP	Case-Insensitive On	^(?)	Currency symbols	^p{currency_symbol}
Footnote Reference Marker	^F	~F	Right Indent Tab	^y	~y	Case-Insensitive Off	^(?-i)	Modifier symbols	^p{modifier_symbol}
Index Reference Marker	^I	~I	Indent to Here	^i	~i	Multiline On or Multiline Off	^(?m) or (?-m)	Wingdings, dingbats	^p{other_symbol}
Hyphens and Dashes	Text	GREP	End Nested Style Here	^h	~h	Single-Line On or Single-Line Off	^(?s) or (?-s)	Numbers	^p{number}
Em Dash	^_	~_	Non-Joiner	^j	~j	Ignore Whitespace On	^(?w)	Digits 0 to 9	^p{decimal_digit_number}
En Dash	^=	~=	Clipboard Contents Formatted	^c	~c	Ignore Whitespace Off	^(?-w)	Roman numerals	^p{letter_number}
Discretionary Hyphen	^~	~~	Clipboard Contents Unformatted	^C	~C	Posix	GREP	Other numbers	^p{other_number}
Nonbreaking Hyphen	^~	~~	Wildcards	Text	GREP	Any alphanumeric character	^[[:alnum:]]	Punctuation	^p{punctuation}
White Space	Text	GREP	Any Digit	^9	^d	Any alphabetic character	^[[:alpha:]]	Hyphens and dashes	^p{dash_punctuation}
Em Space	^m	~m	Any Character Not a Digit		^D	Only digits 0 to 9	^[[:digit:]]	Opening parentheses, braces	^p{open_punctuation}
En Space	^>	~>	Any Letter	^S	^[^u]	Any lowercase character a to z	^[[:lower:]]	Closing parentheses, brackets	^p{close_punctuation}
Flush Space	^f	~f	Any Character	^?	^.	Any punctuation and symbols	^[[:punct:]]	Opening quotes	^p{initial_punctuation}
Hair Space	^	~	Any White Space	^w	^s	Any whitespace and break	^[[:space:]]	Closing quotes	^p{final_punctuation}
Nonbreaking Space	^S	~S	Any Character Not a White Space	^s	^S	Any uppercase character A to Z	^[[:upper:]]	Underscore, 203F, 2040, 2054	^p{connector_punctuation}
Nonbreaking Fixed Width Space	^s	~s	Any Vertical Space (hard and soft returns)	^v		Any letter, number, and underscore	^[[:word:]]	Other punctuation	^p{other_punctuation}
Ideographic Space	^(<	~(<	Any Horizontal Space (all but hard returns)	^h		Any blank character, ie space or tab	^[[:blank:]]	Other	^p{other}
Thin Space	^<	~<	Any Word Character	^w		Any control character	^[[:control:]]	Control characters	^p{control}
Figure Space	^/	~/	Any Character Not a Word Character	^W		Any visible character	^[[:graph:]]	Formatting character markers	^p{format}
Punctuation Space	^.	~.	Any Lowercase Letter	^l		Any visible characters and spaces	^[[:print:]]	E000-F8FF	^p{private_use}
Third Space	^3	~3	Any Character Not a Lowercase Letter	^L		Any character code > 255	^[[:unicode:]]	Unassigned Unicode range	^p{unassigned}
Quarter Space	^4	~4	Any Uppercase Letter	^u		(applies only to the wide character traits classes)			
Sixth Space	^%	~%	Any Character Not Uppercase	^U					

Id

GREP and InDesign CC 2017 some useful search examples

Id

1. Find ALL CAPS paragraphs

Find: `\u\w+(\.|\?|\r)?$`

Change with formatting: paragraph style

Note: When your untagged text has occasional paragraphs typed in all caps, this script finds two or more uppercase letters followed by a period and a hard return. This is enough to find such paragraphs and tag them. Step 2, run `ChangeCaseofSelectedStyle.jsx`, a script by Dave Saunders of PDSAssoc.com. This converts case.

2. Find ALL CAPS words

Find: `\<[A-Z]{2,}\>`

Change: with formatting: OpenType All Caps

Note: The minimal length {2,} is to prevent it picking up all single capitals (remove if you want those as well). Ideally, you search full caps with this regular expression and replace them with OpenType All Caps. Any other way will need handwork or a script. For international text, you should search for `\<\u{2,}\>` (the `\u` uppercase wildcard also finds capital "Ü")

3. Find capital letters like acronyms

Find: `\u\w+` or Find: `>\<\u\w+\>`

Change: with formatting; such as AllSmallCaps

Note: This finds 2 or more uppercase letters. The `\<` and `\>` are word delimiters, so it won't alter anything else than caps words. If the Replace field is empty and it has formatting, it applies the formatting to the found text. Only if there is also no formatting, it replaces with nothing.

4. Swap List of First Names and Last Names

Find: `(+) (+)`

Change: `$2, $1`

Note: This finds something to the left and right of a space; and puts them back in reversed with a comma and space

5. Thin spaces around em dashes

Find: `(?<!~<)(~_)(?!~<)`

Change: `~<$0~<`

This finds em dashes without thin spaces and puts them in.

6. Find and delete soft return line breaks

Find: `\s?n`

Change: with single spacebar space

Note: Soft returns replaced by spacebar space

7. Find quoted words and change to italic

Find: `(~){(\u*\{*\s*\w*\d*)(~)}`

Change: `$2` and italic character style formatting

Note: This finds the quotes but leaves them out since it only changes to found group 2; not 1 and 3.

8. Find all email addresses

Find: `[\\u\d_%-]+@[\\u\d_%-]+`

Change: with formatting

Note: The `_`, `%`, and hyphen were added because those characters are not included in the wildcard codes. The square brackets group the codes together. The `+` indicates that these characters can appear any number of times, or not at all. The `@` sign is outside the group to appear only once. The group is then repeated for the rest of the address.

9. Capitalize first letter of word after bullet

Find: `(.)(\b)(\l)`

Change: `$1$2$3`

Note: In the Replace formatting set text to uppercase

10. Search for words with brackets

Find: `\\w{4,20}\`

Change: with formatting

Note: This finds with brackets (could be parentheses)

11. Search for words within brackets

Find: `(?<=\\)\w{4,20}(?=\\)`

Change: with formatting

Note: This finds within but not including brackets

12. Search for 3 of the same letters; reduce to 2

Find: `(\w)\1\1`

Change: `$1$1`

13. Search for phone numbers to regularize

Find: `\d?[-.]?(\d{3})\d?[-.]?(\d{3})[-.]?(\d{4})`

Change: `$1.$2.$3`

Or Change: `$1-$2-$3` or Change: `($1) $2-$3`

Note: This finds a sequence of 3, 3, and 4 digits. The `?` symbol indicates that the element can exist zero or one time in the string, so the code `[-.]?` means that a hyphen, dot, or space may or may not exist between the digits. Replace can use periods, parentheses, or hyphens, as you wish.

14. Search for dates to format

Find: `\d\d?\-?\d\d?\-?\d\d)?\d\d`

Change: with formatting from the character styles panel

Notes: This sequence finds dates formatted like 09-04-2008 as well as 9-4-08, and allows you to apply a character style.

15. Replace multiple terms with one term

Find: `illustration | graph | map | chart`

Change: figure

Note: The list of alternative words must be separated by the vertical bar, and each alternative is replaced with the replace term. GREP is case sensitive by default; so to replace case-insensitively, add `(?)` before the expression.

16. Find or remove trailing characters after numbers, as in 1st, 2nd, 3rd, 4th

Find: `(?<=\d)+st|(?<=\d)+th|(?<=\d)+nd|(?<=\d)+rd`

Change: leave blank, to strip out; or apply formatting of superscript character style

Note: In this example, we are looking for the two-letter trailing characters to strip off the number. The pipe character is an OR statement. So, the search repeats itself four times, saying that if the characters occur where there is a digit preceding it, then the characters will be removed. If you surround the text with parentheses, you could superscript the found text instead of stripping it out.

17. Search for duplicate words; reduce to 1 word

Find: `((\b\w+))(\ \1)+\b`

Change: `$1`

18. TitleCase applied except for small words

Find: `:\s\|[-\s]\|(?!(f|he|nd?|o|y|rom|or|ut)?\b)`

Change: formatting to All Caps styling

Note: Search the selection. Leave the Change what field blank. This won't work on all uppercase text! This also converts to uppercase any word that follows a colon or a hyphen.

19. Undocumented negation in Wildcards

Not documented in the Wildcards flyout is that the uppercase version of the wildcards negates them.

Note: Thus, `\D` finds everything that is not a digit; `\U` finds everything that is not an uppercase letter; and `\L` finds everything that is not a lowercase letter, etc.

20. Find underlining and replace with italics

Find Formatting: Underline

Change Formatting: Italic character style

This uses Text; not GREP, and is good because character styles control better than direct formatting. Note: Have pre-built character styles for bold, italic, and bold italic. Also, have superscript, smallcaps, and other character styles defining swatch colors pre-built for use in Find/Change as well as nested styles.

21. Interpret GREP into English

RegexBuddy (<http://www.regexbuddy.com/>)

Notes: To see the translation, click the Create tab after entering a regular expression. You will see that the program has listed your expression as a sequence of icons/explanations. Windows-only.

Michael Witherell
mikewitherell@jetsetcom.net

Urszula Witherell
urszulawitherell@jetsetcom.net

Join us for Adobe Certified Expert training in:
InDesign | Illustrator | Photoshop | Acrobat | InCopy

Book us online at: www.jetsetcom.net
and www.trainingonsite.com