

GREP and Text Metacharacters in Find/Change

InDesign CS4

Description	GREP	Text
Tab	\t	^t
Forced Line Break	\n	^n
End of Paragraph	\r	^p
Symbols		
Bullet Character •	~8	^8
Carat Character ^	\^	^^
Backslash Character \	\\	^
Copyright Symbol ©	~2	^2
Ellipsis ...	~e	^e
Paragraph Symbol ¶	~7	^7
Registered Trademark Symbol ®	~r	^r
Section Symbol §	~6	^6
Trademark Symbol ™	~d	^d
Open Parenthesis (\((
Closed Parenthesis)	\))
Open Brace Character {	\{	{
Closed Brace Character }	\}	}
Open Bracket Character [\[[
Closed Bracket Character]	\]]
Markers		
Any Page Number	~#	^#
Current Page Number	~N	^N
Next Page Number	~X	^X
Previous Page Number	~V	^V
Section Marker	~x	^x
Anchored Object Marker	~a	^a
Footnote Reference Marker	~F	^F
Index Reference Marker	~I	^I
Hyphens and Dashes		
Em Dash	~_	^_
En Dash	~=	^=
Discretionary Hyphen	~-	^-
Nonbreaking Hyphen	~-	^-
White Space		
Em Space	~m	^m
En Space	~>	^>
Flush Space	~f	^f
Hair Space	~	^
Nonbreaking Space	~S	^S
Nonbreaking Space (Fixed Width)	~s	^s
Thin Space	~<	^<
Figure Space	~/	^/
Punctuation Space	~.	^.
Third Space	~3	^3
Sixth Space	~%	^%

Quotation Marks	GREP	Text
Any Double Quotation Marks	"	"
Any Single Quotation Marks	'	'
Straight Double Quotation Marks "	~"	^"
Double Left Quotation Marks "	~{	^{
Double Right Quotation Marks "	~}	^}
Straight Single Quotation Mark '	~'	^'
Single Left Quotation Mark '	~[^[
Single Right Quotation Mark '	~]	^]
Break Character		
Standard Carriage Return	~b	^b
Column Break	~M	^M
Frame Break	~R	^R
Page Break	~P	^P
Odd Page Break	~L	^L
Even Page Break	~E	^E
Discretionary Line Break	~k	^k
Variable		
Any Variable	~v	^v
Running Header (Paragraph Style)	~Y	^Y
Running Header (Character Style)	~Z	^Z
Custom Text	~u	^u
Last Page Number	~T	^T
Chapter Number	~H	^H
Creation Date	~O	^O
Modification Date	~o	^o
Output Date	~D	^D
File Name	~I	^I
Other		
Right Indent Tab	~y	^y
Indent To Here	~i	^i
End Nested Style Here	~h	^h
Non-Joiner	~j	^j
Wildcards		
Any Digit	\d	^d
Any Letter	[\u]	^\$
Any Character	period	^?
Any White Space	\s	^w
Any Word Character	\w	
Any Lowercase Letter	\l	
Any Uppercase Letter	\u	
Locations		
Beginning of Word	~<	^<
End of Word	~>	^>
Word Boundary	\b	
Beginning of Paragraph	^	

End of Paragraph	\$		
Beginning of Story	\A		
End of Story	\Z		
Repeat			
Zero or One Time	?		
Zero or More Times	*		
One or More Times	+		
Zero or One Time (Shortest Match)	??		
Zero or More Times (Shortest Match)	*?		
One or More Times (Shortest Match)	+?		
Match			
Marking Subexpression	()		
Non-Marking Subexpression	(?)		
Character Set	[]		
Or			
Positive Lookbehind	(?<=)		
Negative Lookbehind	(?!)		
Positive Lookahead	(?=)		
Negative Lookahead	(?!)		
Modifiers			
Case-Insensitive On	(?i)		
Case-Insensitive Off	(?-i)		
Multiline On	(?m)		
Multiline Off	(?-m)		
Single-Line On	(?s)		
Single-Line Off	(?-s)		
Ignore Whitespace On	(?w)		
Ignore Whitespace Off	(?-w)		
Change to Other			
Clipboard Contents, Formatted	~c		
Clipboard Contents, Unformatted	~C		
Change to Found			
Found Text	\$0		
Found 1	\$1		
Found 2	\$2		
Found 3	\$3		
Found 4	\$4		
Found 5	\$5		
Found 6	\$6		
Found 7	\$7		
Found 8	\$8		
Found 9	\$9		

Michael Witherell ↓ mikewitherell@jetsetcom.net
 Urszula Witherell ↓ urszulawitherell@jetsetcom.net
www.jetsetcom.net ↓ Adobe Certified Experts

Useful GREP search examples

1. Find ALL CAPS paragraphs

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

Replace with formatting: paragraph style

Note: When your untagged text has occasional paragraphs typed in all caps, this script finds two 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 to Title or lowercase.

2. Find ALL CAPS words

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

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

Replace: with formatting; such as SmallCaps

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. Find and delete email quote angle brackets

Find: `>+ +`

Replace: leave blank

Note: This deletes all `>` and any spaces with them

5. Find and delete soft return line breaks

Find: `\s?\n`

Replace: with single spacebar space

Note: Soft returns replaced by spacebar space

6. Find quoted words and change to italic

Find: `(~)(\u*\|*\s*.*\w*\d*)(~)`

Replace: \$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.

7. Find all email addresses

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

Replace: 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.

8. Capitalize first letter of word after bullet

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

Replace: `$1$2$3`

Note: In the Replace formatting set text to uppercase

9. Search for words with brackets

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

Replace: with formatting

Note: This finds with brackets

10. Search for words within brackets

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

Replace: with formatting

Note: This finds within but not including brackets

11. Search for phone numbers to regularize

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

Replace: `$1.$2.$3`

Or Replace: `$1-$2-$3` or Replace: `($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.

12. Search for dates to format

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

Replace: with formatting

Notes: This sequence finds dates formatted like 09-04-2008 as well as 9-4-08

13. Replace multiple terms with one term

Find: `illustration | graph | map | chart`

Replace: 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.

14. 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.

15. Find underlining and replace with italics

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.

16. 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.