# GREP CHEAT SHEET

| WILDCARDS | |
|---|---|
| Any Digit | \d |
| Any Letter | [\l\u] |
| Any Character | . |
| Any White Space | \s |
| 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 | \ |
| End of Story | \Z |

| REPEAT | |
|---|---|
| Zero or One Time | ? |
| Zero or More Times | * |
| One or More Times | + |
| Zero or One Time (Shortest Match) | *? |
| One or More Times (Shortest Match) | +? |

### Any Digit finds each single digit:

Mary had 3 little lambs. Her whole flock was made up of these three, and 15 fully-grown sheep.

### Any Letter finds each single letter (uppercase or lowercase):

Mary had 3 little lambs. Her whole flock was made up of these three, and 15 fully-grown sheep.

### Any Character finds each single character (except line break):

Mary had 3 little lambs. Her whole flock was made up of these three, and 15 fully-grown sheep.

### Beginning of Paragraph:

Mary had 3 little lambs. Her whole flock was made up of these three, and 15 fully-grown sheep.

### End of Paragraph:

Mary had 3 little lambs. Her whole flock was made up of these three, and 15 fully-grown sheep.

### End of Story:

Mary had 3 little lambs. Her whole flock was made up of these three, and 15 fully-grown sheep.
Her sister, Shari, didn't have any sheep at all.

### Zero or One Time:

The item may appear one time in sequence, or it may not appear at all.

### Zero or More Times:

The item may appear any number of times in sequence, or it may not appear at all.

### One or More Times:

The item appears one time or more in sequence.

### Shortest Match:

Keeps the search to the first complete sequence. Without "shortest match," InDesign looks to the whole paragraph for the sequence.

# GREP CHEAT SHEET

| MATCH | |
|---|---|
| Marking Subexpression | () (parentheses) |
| Non-Marking Subexpression | (?:) |
| Character Set | [] (square brackets) |
| Or | \| (pipe character) |
| Positive Lookbehind | (?<=) |
| Negative Lookbehind | (?<!) |
| Positive Lookahead | (?=) |
| Negative Lookahead | (?!) |

## Offset text as expressions:

Use parentheses to offset an entire expression or to group items together. For instance, in Find/Change, if you are looking for a string of 10 digits, but you want to return them in a different order, or insert additional information between the groups, use parenthesis.

**Example** - format a phone number from a 10-digit string:

3035551212

In Find/Change: (\d\d\d)(\d\d\d)(\d\d\d\d)

(Note: Though \d\d\d is valid, it is more efficient to write 3 digits as \d{3})

This will see each group separately. You could then return only the 2nd and 3rd groups in the Change To, if wanted.

## Look for something OR something else:

Use the straight slash (or pipe character, located over the backslash) to indicate either this item OR that item. You can have multiple items, each divided by the pipe character, but you need to enclose this OR search in parenthesis when it appears in line with other text.

**Example** - look for the word "grey" or "gray." You could easily look for the whole word spelled both ways, but the only difference is the letter "e" or "a." The search will always look to the entire string on either side of the pipe character.

Look for: gr(e|a)y - Will find: grey AND gray

Look for: (Red|Green|Blue) - Will find: Red, Green, OR Blue

Look for: (color filled|fill color|color) - Will find each of these phrases

## Look before or after to find a string:

Use Lookahead and Lookbehind to locate a string and affect just that string.

**Example:** Find digits after a decimal point by using a Positive Lookbehind.

Looking behind the text (what appears before), if the string exists (the decimal point in this case), then do something with the searched text (the digits).

Look for: (?<=\.)\d+

To find this: 123.45

Use Positive Lookbehind (?<=) and add in the decimal point. However, since the decimal means something specific in GREP, you need to "escape" the decimal to indicate you're looking for an actual decimal. That is the backslash before the decimal. Put that inside the parentheses as well so it finds an entire string. The next part, the \d+, is looking for any digit, one or more times. You can then do something to the found portion (only the 45 in the example above), such as superscript it.

# GREP CHEAT SHEET

| SYMBOLS | |
|---|---|
| Bullet Character | ~8 |
| Backslash Character | \\ |
| Caret Character | \^ |
| Copyright Symbol | ~2 |
| Ellipsis | ~e |
| Paragraph Symbol | ~7 |
| Reg. Trademark Symbol | ~r |
| Section Symbol | ~6 |
| Trademark Symbol | ~d |
| Open Parenthesis | \( |
| Close Parenthesis | \) |
| Open Brace | \{ |
| Close Brace | \} |
| Open Bracket | \[ |
| Close Bracket | \] |

| MARKERS | |
|---|---|
| Any Page Number | ~# |
| Current Page Number | ~N |
| Next Page Number | ~X |
| Previous Page Number | ~V |
| Section Marker | ~x |
| Anchored Object Marker | ~a |
| Footnote Reference Marker | ~F |
| Index Marker | ~I |

| HYPHENS AND DASHES | |
|---|---|
| Em Dash | ~_ |
| En Dash | ~= |
| Discretionary Hyphen | ~- |
| Nonbreaking Hyphen | ~~ |

| WHITE SPACE | |
|---|---|
| Em Space | ~m |
| En Space | ~> |
| Flush Space | ~f |
| Hair Space | ~\| |
| Nonbreaking Space | ~S |
| Nonbreaking Space (Fixed Width) | ~s |
| Thin Space | ~< |
| Figure Space | ~/ |
| Punctuation Space | ~. |
| Third Space | ~3 |
| Quarter Space | ~4 |
| Sixth Space | ~% |

| QUOTATION MARKS | |
|---|---|
| Any Double Quotation Marks | " |
| Any Single Quotation Mark | ' |
| Straight Dbl Quotation Marks | ~" |
| Double Left Quotation Mark | ~{ |
| Double Right Quotation Mark | ~} |
| Straight Single Quotation Mark | ~' |
| Single Left Quotation Mark | ~[ |
| Single Right Quotation Mark | ~] |

| BREAK CHARACTER | |
|---|---|
| Standard Carriage Return | ~b |
| Column Break | ~M |
| Frame Break | ~R |
| Page Break | ~P |
| Odd Page Break | ~L |
| Even Page Break | ~E |
| Discretionary Line Break | ~k |

| VARIABLE | |
|---|---|
| Any Variable | ~v |
| Running Header (Paragraph Style) | ~Y |
| Running Header (Character Style) | ~Z |
| Custom Text | ~u |
| Last Page Number | ~T |
| Chapter Number | ~H |
| Creation Date | ~O |
| Modification Date | ~o |
| Output Date | ~D |
| File Name | ~l |
| Metadata Caption | ~J |

| OTHER | |
|---|---|
| Right Indent Tab | ~y |
| Indent to Here | ~i |
| End Nested Style Here | ~h |
| Non-joiner | ~j |

# GREP CHEAT SHEET

| MODIFIERS | |
|---|---|
| Case-insensitive On | (?i) |
| Case-insensitive Off | (?-i) |
| Multiline On | (?m) |
| Multiline Off | (?-m) |
| Single-line On | (?s) |
| Single-line Off | (?-s) |

| POSIX | |
|---|---|
| any alphnumeric char. | [[:alnum:]] |
| any alphabetical char. | [[:alpha:]] |
| any digit | [[:digit:]] |
| any lowercase char. | [[:lower:]] |
| any punctuation | [[:punct:]] |
| any whitespace | [[:space:]] |
| any uppercase char. | [[:upper:]] |
| any word char. | [[:word:]] |
| any hexidecimal digit | [[:xdigit:]] |
| all forms of ref'd char. | [[:=a=:]] |

# GREP CHEAT SHEET

| EXPRESSING OR | |
|---|---|
| (Milk\|Dark) Chocolate | Milk Chocolate or Dark Chocolate |
| (m\|M)ilk (c\|C)hocolate | Milk Chocolate, milk chocolate, milk Chocolate, Milk chocolate |
| [mM]ilk chocolate | milk chocolate or Milk chocolate |
| [^m]ilk | ailk, bilk, etc., but not milk |
| (?i)milk | Case insensitive: Milk, milk, mlLk, etc. |
| colou?r | color or colour (the ? indicates zero or more times, use * for zero or more times) |

| WILDCARDS | |
|---|---|
| [cC][\l\u]+e | Any string of letters that starts with c and ends with e, such as chocolate, Chocolate, case, coarse, etc. |
| \<c\w+e\> | Any whole word that begins with c and ends with e |
| \bc\w+e\b | Exactly the same as above, but using \b for 'word boundary' |
| \s+$ | All spaces at the end of a paragraph |
| \Z | End of story |
| [\l\u\d.]+@[\l\u\d.]+ | Simple e-mail address (e.g. david@indesignsecrets.com or d2@AB.net) |
| \d{4} | Find any four digits (the curly brace expression is not found in the InDesign @ menu) |
| [-\u\l\d._]+@[-\u\l\d_.]+\.[\u\l]{2,4} | More 'robust' email (e.g. david-blatner_2@63P.com) |
| \d+/\d+ | any fraction, such as 1/2, 22/7, or 355/113 |
| (?<!\d)(?<!/)\d+/(?!11)(?!0)(?!9)\d{1,3}(?!/)(?!\d) | Robust fraction grep: Will find fractions, but leave out dates such as 6/98, 9/11, or 6/17/2012. Unfortunately, it fails on 355/113 (which happens to be a reasonably good approximation of π, proving that nothing is perfect). |
| [[=a=]] | Find any kind of a, no matter the diacritic, including a, á, à, ä, etc. |
| (?<=\d)(rd\|th\|st) | Find st, rd, or th ordinals (such as 1st, 2nd, 3rd) that follow a digit—in order to apply superscript to just the ordinal (not the digit). |
| (?<=\().+?(?=\)) | Find any text inside parentheses, but not including the parentheses. This uses 'Positive Lookbehind' and 'Positive Lookahead' |

Adapted from David Blatner's article, "Favorite GREP Expressions You Can Use" - indesignsecrets.com

# GREP CHEAT SHEET

| ESCAPE! | |
|---|---|
| \$\d\.\d\d | Find a dollar sign followed by one digit, a period, and two digits. Note that the dollar and dot have to be 'escaped' because they normally mean something different. |
| \(.+\) | Find text and surrounding parentheses. To search for literal parentheses, use \( and \). Parentheses by themselves have special meaning. |
| \Q(a+b*c?)\E | Escape everything; that is, find (a+b*c) exactly, without having to worry about escaping each special character. |

| FUN TRICKS | | |
|---|---|---|
| Description | Find | Change to |
| Add an anchored object at the beginning of each paragraph (cut object to clipboard first) | ^(.) | ~C$1 |
| Reverse first and last name in a list (but watch out for middle names or initials, too) | ^(\w+)([ .\w]+)? (\w+) | $3, $1$2 |
| After reversing the names (see above), fix any last names that started with Van or De. | ^(\w+, [ .\w]+) (Van\|De) | $2 $1 |
| Find and remove duplicate words | \b(\w+)\b \1 | $1 |
| Find and remove duplicate paragraphs/lines in a list | ^(.+\r)\1+ | $1 |
| Remove spaces at end of story | \s+\Z | (leave blank) |
| Find lists in which people have actually typed a number followed by a period followed by a space, and apply automatic numbered list style. | ^\d+\. ?(.) | $1 (and apply paragraph style that includes auto numbers) |