

Sed Command Summary

Command	Address or Range	Modifications to:			
		Input Stream	Output Stream	Pattern Space	Hold Buffer
=	–	–	√	–	–
a	1	–	√	–	–
b	2	–	–	–	–
c	2	–	√	–	–
d	2	√	–	√	–
D	2	√	–	√	–
g	2	–	–	√	–
G	2	–	–	√	–
h	2	–	–	–	√
H	2	–	–	–	√
i	1	–	√	–	–
l	1	–	√	–	–
n	2	√	□	–	–
N	2	√	–	√	–
p	2	–	√	–	–
P	2	–	√	–	–
q	1	–	–	–	–
r	1	–	√	–	–
s	2	–	–	√	–
t	2	–	–	–	–
w	2	–	√	–	–
x	2	–	–	√	√
y	2	–	–	√	–

- 1 Command takes single address or pattern.
- 2 Command takes pair of addresses.
- Command does not modify the buffer.
- √ Command modifies the buffer.
- The “**n**” command may or may not generate output depending on the “-n” command option.

© 2007 Peteris Kruminis http://www.catonmat.net	peter@catonmat.net good coders code, great reuse
------------------------------------------------------------------------------------------	------------------------------------------------------------

Command line argument summary:

Argument	Description
-n --quiet --silent	suppress automatic printing of pattern space
-e script --expression=script	add the script to the commands to be executed
-f script-file --file=script-file	add the contents of script-file to the commands to be executed
-i[suffix] --in-place[=suffix]	edit files in place (makes backup if extension supplied)
-l N --line-length=N	specify the desired line-wrap length for the `l' command
-r --regexp-extended	use extended regular expressions in the script.
-s --separate	consider files as separate rather than as a single continuous long stream.
-u --unbuffered	load minimal amounts of data from the input files and flush the output buffers more often
--help	display this help and exit
-V --version	output version information and exit

Command	Description
#	Adds a comment
=	The "=" command prints the current line number to standard output.
a \ <i>text</i>	The "a" command appends <i>text</i> after the range or pattern.
b <i>label</i>	The "b" command branches to the <i>label</i> . You can specify a <i>label</i> with a text string followed by a colon. If no <i>label</i> is there, branch to the end of the script.
c \ <i>text</i>	The "c" command changes the current line with <i>text</i>
d	The "d" command deletes the current pattern space, reads in the next line, puts the new line into the pattern space, and aborts the current command, and starts execution at the first <i>sed</i> command.
D	The "D" command deletes the first portion of the pattern space, up to the new line character, leaving the rest of the pattern alone.
g	Instead of exchanging (the "x" command) the hold space with the pattern space, you can copy the hold space to the pattern space with the "g" command.
G	If you want to append to the pattern space, use the "G" command.
h	The "h" command copies the pattern buffer into the hold buffer.
H	The "H" command allows you to combine several lines in the hold buffer. It acts like the "N" command as lines are appended to the buffer, with a "\n" between the lines. You can save several lines in the hold buffer, and print them only if a particular pattern is found later.
i \ <i>text</i>	You can insert <i>text</i> before the pattern with the "i" command.
I	The "I" command prints the current pattern space. It is therefore useful in debugging <i>sed</i> scripts. It also converts unprintable characters into printing characters by outputting the value in octal preceded by a "\" character.
n	The "n" command will print out the current pattern space (unless the "-n" flag is used), empty the current pattern space, and read in the next line of input.
N	The "N" command does not print out the current pattern space and does not empty the pattern space. It reads in the next line, but appends a new line character along with the input line itself to the pattern space.
p	Another useful command is the print command: "p". If <i>sed</i> wasn't started with an "-n" option, the "p" command will duplicate the input. The "p" command prints the entire pattern space.
P	The "P" command only prints the first part of the pattern space, up to the NEWLINE character.
q	There is one more simple command that can restrict the changes to a set of lines. It is the "q" command: quit. This command is most useful when you wish to abort the editing after some condition is reached.
r <i>filename</i>	The "r" command will append text from <i>filename</i> after the range or pattern.
s / <i>regex/repl</i> /	The substitute command replaces all occurrences of the regular expression (<i>regex</i>) with <i>repl</i> (acement)
t <i>label</i>	You can execute a branch if a pattern is found. You may want to execute a branch only if a substitution is made. The command "t <i>label</i> " will branch to the <i>label</i> if the last substitute command modified the pattern space.
w <i>filename</i>	With this command, you can specify a <i>filename</i> that will receive the modified data.
x	The "x" command exchanges the hold buffer and the pattern buffer.
y / <i>source/dest</i> /	Transliterate the characters in the pattern space, which appear in <i>source</i> to the corresponding character in <i>dest</i> (ination).

Extensions:

Command	Description
Q	Immediately quit the sed script without processing any more input. (zero or one address command)
R <i>filename</i>	Append a line read from <i>filename</i> . (zero or one address command)
T <i>label</i>	If no s/// has done a successful substitution since the last input line was read and since the last t or T command, then branch to <i>label</i> ; if <i>label</i> is omitted, branch to end of script. (accepts address range)
W <i>filename</i>	Write the first line of the current pattern space to <i>filename</i> . (accepts address range)

Address Ranges:

Format	Description	Format (ext)	Description
<i>number</i>	Match only the specified line <i>number</i> .	first~step	Match every <i>step</i> 'th line starting with line <i>first</i> .
\$	Match the last line.	0, <i>addr2</i>	Start out in "matched first address" state, until <i>addr2</i> is found.
/regex/	Match lines matching the regular expression <i>regex</i> .	<i>addr1</i>,+N	Will match <i>addr1</i> and the <i>N</i> lines following <i>addr1</i> .
\cregexc	Match lines matching the regular expression <i>regex</i> . The c may be any character.	<i>addr1</i>,~N	Will match <i>addr1</i> and the lines following <i>addr1</i> until the next line whose input line number is a multiple of <i>N</i> .

