

sed Quick Reference

Stream editing, substitution, addresses, hold space, in-place transforms

Basics

Running sed

```
sed 's/old/new/' file.txt # substitute first match per line
sed 's/old/new/g' file.txt # substitute all matches per line
sed -n '5p' file.txt # print only line 5
sed '3d' file.txt # delete line 3
echo "hello" | sed 's/hello/hi/' # pipe input
```

Command-Line Flags

-n	Suppress automatic printing; only print with p
-e 'cmd'	Execute a sed command (chain multiple with -e)
-f script.sed	Read commands from a file
-i [suffix]	Edit file in-place (optional backup suffix)
-E / -r	Use extended regular expressions

Substitution

Substitution Syntax

```
sed 's/foo/bar/' f # first occurrence per line
sed 's/foo/bar/g' f # all occurrences
sed 's/foo/bar/3' f # 3rd occurrence only
sed 's/foo/bar/gi' f # all, case-insensitive
sed 's|usr/bin|opt/bin|g' f # alternate delimiter
```

Substitution Flags

g	Replace all occurrences on the line
N (number)	Replace only the Nth occurrence
p	Print line if substitution was made
w file	Write substituted lines to file
i / I	Case-insensitive match (GNU)

Addresses

Address Examples

```
sed '3s/a/b/' f # only on line 3
sed '2,5s/a/b/' f # lines 2 through 5
sed '/^#/d' f # delete lines starting with #
sed '/start/,/end/d' f # delete range between patterns
sed '1~2d' f # delete odd-numbered lines (GNU)
```

Address Types

N	Line number N
\$	Last line
N,M	Line range from N to M
/regex/	Lines matching regex
/regex1/,/regex2/	Range from first match to second match
N~step	Starting at N, every step-th line (GNU)
addr!	Negate — apply to non-matching lines

Delete & Print

Delete & Print Commands

```
sed '5d' f # delete line 5
sed '/^$/d' f # delete blank lines
sed -n '10,20p' f # print lines 10-20
sed -n '/error/p' f # print lines matching pattern
sed '/debug!d' f # keep only matching lines
```

Command Reference

d	Delete pattern space, start next cycle
D	Delete up to first newline in pattern space
p	Print pattern space
P	Print up to first newline in pattern space
q	Quit after printing current pattern space
Q	Quit without printing (GNU)

Insert & Append

Insert, Append, Change

```
sed '3i\inserted line' f # insert before line 3
sed '3a\appended line' f # append after line 3
sed '3c\replaced line' f # replace line 3
sed '/marker/a\new line' f # append after pattern match
```

Commands

i\text	Insert text before current line
a\text	Append text after current line
c\text	Replace current line with text
r file	Read and append contents of file
R file	Read and append one line from file (GNU)
w file	Write pattern space to file

Hold Space

Hold Space Commands

h	Copy pattern space to hold space
H	Append pattern space to hold space
g	Copy hold space to pattern space
G	Append hold space to pattern space
x	Exchange pattern and hold spaces

Hold Space Examples

```
sed -n '1!G;h;$p' f # reverse lines (tac)
sed '/^$/{ x; s/\n//; x; }' f # collapse hold on blank
sed -n 'H;$x;s/\n/ /g;p;' f # join all lines with space
```

Multiple Commands

Chaining Commands

```
sed -e 's/foo/bar/g' -e 's/baz/qux/g' f
sed 's/foo/bar/g; s/baz/qux/g' f
sed '/header/{ s/old/new; s/foo/bar; }' f
sed -f commands.sed input.txt
```

Grouping & Branching

{ cmd1; cmd2; }	Group commands for same address
:label	Define a branch label
b label	Branch (jump) to label
t label	Branch if last s/// succeeded
T label	Branch if last s/// failed (GNU)

In-place Editing

In-place Examples

```
sed -i 's/old/new/g' file.txt # edit in place (GNU)
sed -i.bak 's/old/new/g' file.txt # backup as file.txt.bak
sed -i '' 's/old/new/g' file.txt # macOS in-place (no backup)
sed -i '/^#/d' config.txt # remove comments in place
```

Platform Notes

GNU sed -i	Suffix is optional; -i alone edits without backup
BSD/macOS sed -i	Requires suffix argument; use -i '' for no backup
-i.bak	Creates backup file with .bak extension
Multiple files	sed -i 's/a/b/g' *.txt edits all matching files

Regex

Regex in sed

.	Any single character
*	Zero or more of preceding
\+	One or more (BRE) — + in ERE
\?	Zero or one (BRE) — ? in ERE
^	Start of line
\$	End of line
[abc]	Character class
\(\)	Capture group (BRE) — () in ERE
\1, \2	Back-reference to capture group
&	Entire matched string (in replacement)

Regex Examples

```
sed 's/[0-9]\+/NUM/g' f # replace numbers
sed -E 's/(foo)(bar)/\2\1/g' f # swap groups (ERE)
sed 's/.*/(&)/' f # wrap line in parens
sed 's/[ \t]*$//' f # strip trailing whitespace
```

Common Patterns

One-Liners

```
sed -n '1p' f # first line (head -1)
sed '$!d' f # last line (tail -1)
sed '/^$/d' f # remove blank lines
sed 's/^[ \t]*/' f # strip leading whitespace
sed '=' f | sed 'N;s/\n/\t/' f # number lines
```

Recipes

Double-space	sed G — append blank line after each line
Remove HTML tags	sed 's/<[>]*>///g'
Extract emails	sed -nE 's/.*([a-z]+@[a-z.]+).*/\1/p'
Comment lines	sed 's/^/# /' — prefix each line with #
Trim blank lines at EOF	sed -e :a -e '/^\n*\$/d'N;ba -e '}'
Replace nth line	sed 'Nc\new text' — replace line N