sed Stream editor

CONCEPTS

sed reads input one line at a time, runs your script on it, and prints the result. Without -i the file itself is never changed.

Read line into pattern space
↓ `run script`
Commands address? command
↓ `auto-print`
Output stdout, or -i file
Repeated for every input line

macOS ships BSD sed: -i needs an argument (-i ''), and \U, \n in replacements and 0~2 are not supported. This sheet is GNU sed.

SUBSTITUTE

`sed 's/<re>/<new>/' <file>`
Replace first match on each line

`sed 's/<re>/<new>/g' <file>`
Replace every match

`sed 's/<re>/<new>/<N>' <file>`
Replace only the Nth match

`sed 's/<re>/<new>/gI' <file>`
Ignore case

`sed 's/<re>/&!/' <file>`
& inserts the whole match

`sed 's|<path>|<new>|g' <file>`
Pick another delimiter for slashes

`sed -E 's/(<a>)(<b>)/\2\1/' <file>`
Swap groups with backreferences

OPTIONS

`sed -n`
Print only when told to (p)

`sed -i[<suffix>]`
Edit in place, optional backup

`sed -E`
Extended regex: (), +, ?, |

`sed -e '<cmd>' -e '<cmd>'`
Combine several commands

`sed -f <script.sed> <file>`
Read commands from a file

ADDRESSES

`sed '<N>cmd'`
Act on line N

`sed '$cmd'`
Act on the last line

`sed '<N>,<M>cmd'`
Act on lines N to M

`sed '/<re>/cmd'`
Act on lines matching a regex

`sed '/<re>/!cmd'`
Act on lines not matching

`sed '/<a>/,/<b>/cmd'`
From the first match to the next

`sed '<N>,+<K>cmd'`
Line N and the K lines after

`sed '<N>~<K>cmd'`
Every Kth line, starting at N

COMMANDS

`p`
Print the pattern space

`d`
Delete line, start next cycle

`q / Q`
Quit (Q without printing)

`a\ <text>`
Append text after the line

`i\ <text>`
Insert text before the line

`c\ <text>`
Replace the line with text

`{ cmd; cmd }`
Group commands under one address

REGEX NOTES

`^ $ . *`
Start, end, any char, repeat

`[abc] [^abc] [a-z]`
Character classes

`\+ \? \| \(\)`
BRE needs backslashes (GNU)

`+ ? | () {n,m}`
Plain in -E mode

`\. \* \/ \[`
Escape a special character