

SED

* SED : Stream Editor *

- For filtering and transforming text.
- Works in a single pass - Efficient!
- Use in command line or in a script.

Any trick to correct multiple typos in a script w/a single line of code?

Jeez! Heard of SED Command! Do 'man sed' in bash.



`sed -e '/pattern/command' my-file`

to execute script regexp s → search & replace txt file

By default, all lines are printed. Use '-n' to suppress.

- p → print
- d → delete
- i → insert
- a → append



`>> ls -L | sed -n -e '/^d/p'` → Listing all subdirs

`>> sed -e '/^#/d' my-file` → deleting lines starting w/ '#'

`>> sed -e 's,100 p' my-file` → print lines 1-100

can use '\$' for end of file

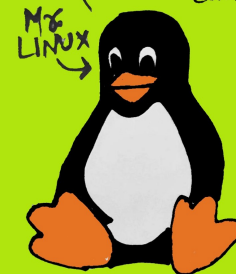
How to make changes to current file using sed?

Use '-i' option for inplace editing.



The 's' command [substitute command & flags]

`'s/regexp/replacement/flags'`



What in replacement?

- n [n → number 1 to 9, inclusive] Can also be a special sequence
- \L: Turn replacement to lower case
- \I: Turn next char to lower case
- \U: Turn replacement to upper case
- \u: Turn next char to upper case

extension added to make case insensitive replace version w/ story!

`# sed 's/version/story/gi' file.txt`
g - for all instances, i - case insensitive

Sed can replace words, characters & even regular expressions!

Use Sed!
sed 's/[zZ]/[pP]/g' my-file.txt
g is used for replacing all instances



SOME ADDITIONAL COMMANDS & EXAMPLES.

`>> sed G testfile.txt` → add a blank line after every non-blank line

append 'bar' after lines containing 'foo' → `>> sed '/foo/a bar' test.txt`

`>> sed 'i30 World' test.txt` → insert 'World' on line 30

!!! CAUTION
-i → in-place editing.
`>> sed -i 's/this/that/gi' test.txt`
this is for case insensitive



* Reference: JULIA EVANS @bork

SOME MORE EXAMPLES -

`>> sed '11q' test.txt` → quits when 11th line is reached

prints only changed lines → `>> sed -n 's|bingo|glob|p' test.txt`

`>> sed -h` → lists all the available options & flags

