| **Learning Linux sed command with examples** | |
| --- | --- |
| **Linux command syntax** | **Linux command description** |
|  | |
| sed 's/Linux|linux/RHEL/g' report.txt | Replace every occurrence of Linux or linux with RHEL. |
| sed 's/^/ /' file.txt > file\_new.txt | Add 8 spaces to the left of a text for pretty printing. |
| sed -n '/Of course/, /attention you \ pay/p' myfile | Display only one paragraph, starting with "Of course"  and ending in "attention you pay" |
| sed -n 12,18p file.txt | Show only lines 12-18 of file.txt |
| sed 12,18d file.txt | Show all of file.txt except for lines from 12 to 18 |
| sed G file.txt | Double-space file.txt |
| sed -f script.sed file.txt | Write all commands in script.sed and execute them |
| sed '5!s/ham/cheese/' file.txt | Replace ham with cheese in file.txt except in the 5th line |
| sed '$d' file.txt | Delete the last line |
| sed '/[0-9]\{3\}/p' file.txt | Print only lines with three consecutive digits |
| sed '/boom/!s/aaa/bb/' file.txt | Unless boom is found replace aaa with bb |
| sed '17,/disk/d' file.txt | Delete all lines from line 17 to 'disk' |
| echo ONE TWO | sed "s/one/unos/I" | Replaces one with unos in a case-insensitive manner,  so it will print "unos TWO" |
| sed 'G;G' file.txt | Triple-space a file |
| sed 's/.$//' file.txt | A way to replace dos2unix :) |
| sed 's/^[ ^t]\*//' file.txt | Delete all spaces in front of every line of file.txt |
| sed 's/[ ^t]\*$//' file.txt | Delete all spaces at the end of every line of file.txt |
| sed 's/^[ ^t]\*//;s/[ ^]\*$//' file.txt | Delete all spaces in front and at the end of every line  of file.txt |
| sed 's/foo/bar/' file.txt | Replace foo with bar only for the first instance in a line. |
| sed 's/foo/bar/4' file.txt | Replace foo with bar only for the 4th instance in a line. |
| sed 's/foo/bar/g' file.txt | Replace foo with bar for all instances in a line. |
| sed '/baz/s/foo/bar/g' file.txt | Only if line contains baz, substitute foo with bar |
| sed '/./,/^$/!d' file.txt | Delete all consecutive blank lines except for EOF |
| sed '/^$/N;/\n$/D' file.txt | Delete all consecutive blank lines, but allows  only top blank line |
| sed '/./,$!d' file.txt | Delete all leading blank lines |
| sed -e :a -e '/^\n\*$/{$d;N;};/\n$/ba' \ file.txt | Delete all trailing blank lines |
| sed -e :a -e '/\$/N; s/\\n//; ta' \ file.txt | If a file ends in a backslash, join it with the next (useful  for shell scripts) |
| sed '/regex/,+5/expr/' | Match regex plus the next 5 lines |
| sed '1~3d' file.txt | Delete every third line, starting with the first |
| sed -n '2~5p' file.txt | Print every 5th line starting with the second |
| sed 's/[Nn]ick/RHEL/g' report.txt | Another way to write some example above.  Can you guess which one? |
| sed -n '/RE/{p;q;}' file.txt | Print only the first match of  RE (regular expression) |
| sed '0,/RE/{//d;}' file.txt | Delete only the first match |
| sed '0,/RE/s//to\_that/' file.txt | Change only the first match |
| sed 's/^[^,]\*,/9999,/' file.csv | Change first field to 9999 in a CSV file |
| s/^ \*\(.\*[^ ]\) \*$/||/; s/" \*, \*/"|/g; : loop s/| \*\([^",|][^,|]\*\) \*, \*/||/g; s/| \*, \*/||/g; t loop s/ \*|/|/g; s/| \*/|/g; s/^|\(.\*\)|$//; | sed script to convert CSV file to bar-separated  (works only on some types of CSV,  with embedded "s and commas) |
| sed ':a;s/\(^\|[^0-9.]\)\([0-9]\+\)\ ([0-9]\{3\}\)/,/g;ta' file.txt | Change numbers from file.txt from 1234.56 form to 1.234.56 |
| sed -r "s/\<(reg|exp)[a-z]+/\U&/g" | Convert any word starting with reg or exp to uppercase |
| sed '1,20 s/RHELson/White/g' file.txt | Do replacement of RHELson with White only on  lines between 1 and 20 |
| sed '1,20 !s/RHELson/White/g' file.txt | The above reversed (match all except lines 1-20) |
| sed '/from/,/until/ { s/\<red\>/magenta/g; \ s/\<blue\>/cyan/g; }' file.txt | Replace only between "from" and "until" |
| sed '/ENDNOTES:/,$ { s/Schaff/Herzog/g; \ s/Kraft/Ebbing/g; }' file.txt | Replace only from the word "ENDNOTES:" until EOF |
| sed '/./{H;$!d;};x;/regex/!d' file.txt | Print paragraphs only if they contain regex |
| sed -e '/./{H;$!d;}' -e 'x;/RE1/!d;\ /RE2/!d;/RE3/!d' file.txt | Print paragraphs only if they contain RE1,  RE2 and RE3 |
| sed ':a; /\$/N; s/\\n//; ta' file.txt | Join two lines in the first ends in a backslash |
| sed 's/14"/fourteen inches/g' file.txt | This is how you can use double quotes |
| sed 's/\/some\/UNIX\/path/\/a\/new\ /path/g' file.txt | Working with Unix paths |
| sed 's/[a-g]//g' file.txt | Remove all characters from a to g from file.txt |
| sed 's/\(.\*\)foo/bar/' file.txt | Replace only the last match of foo with bar |
| sed '1!G;h;$!d' | A tac replacement |
| sed '/\n/!G;s/\(.\)\(.\*\n\)/&\ /;//D;s/.//' | A rev replacement |
| sed 10q file.txt | A head replacement |
| sed -e :a -e '$q;N;11,$D;ba' \ file.txt | A tail replacement |
| sed '$!N; /^\(.\*\)\n$/!P; D' \ file.txt | A uniq replacement |
| sed '$!N; s/^\(.\*\)\n$//;\  t; D' file.txt | The opposite (or uniq -d equivalent) |
| sed '$!N;$!D' file.txt | Equivalent to tail -n 2 |
| sed -n '$p' file.txt | ... tail -n 1 (or tail -1) |
| sed '/regexp/!d' file.txt | grep equivalent |
| sed -n '/regexp/{g;1!p;};h' file.txt | Print the line before the one matching regexp, but  not the one containing the regexp |
| sed -n '/regexp/{n;p;}' file.txt | Print the line after the one matching the regexp, but  not the one containing the regexp |
| sed '/pattern/d' file.txt | Delete lines matching pattern |
| sed '/./!d' file.txt | Delete all blank lines from a file |
| sed '/^$/N;/\n$/N;//D' file.txt | Delete all consecutive blank lines  except for the first two |
| sed -n '/^$/{p;h;};/./{x;/./p;}'\  file.txt | Delete the last line of each paragraph |
| sed 's/.\x08//g' file | Remove nroff overstrikes |
| sed '/^$/q' | Get mail header |
| sed '1,/^$/d' | Get mail body |
| sed '/^Subject: \*/!d; s///;q' | Get mail subject |
| sed 's/^/> /' | Quote mail message by inserting a  "> " in front of every line |
| sed 's/^> //' | The opposite (unquote mail message) |
| sed -e :a -e 's/<[^>]\*>//g;/</N;//ba' | Remove HTML tags |
| sed '/./{H;d;};x;s/\n/={NL}=/g'\  file.txt | sort \ | sed '1s/={NL}=//;s/={NL}=/\n/g' | Sort paragraphs of file.txt alphabetically |
| sed 's@/usr/bin@&/local@g' path.txt | Replace /usr/bin with /usr/bin/local in path.txt |
| sed 's@^.\*$@<<<&>>>@g' path.txt | Try it and see :) |
| sed 's/\(\/[^:]\*\).\*//g' path.txt | Provided path.txt contains $PATH, this will  echo only the first path on each line |
| sed 's/\([^:]\*\).\*//' /etc/passwd | awk replacement - displays only the users  from the passwd file |
| echo "Welcome To The Geek Stuff" | sed \ 's/\(\b[A-Z]\)/\(\)/g' (W)elcome (T)o (T)he (G)eek (S)tuff | Self-explanatory |
| sed -e '/^$/,/^END/s/hills/\ mountains/g' file.txt | Swap 'hills' for 'mountains', but only on blocks  of text beginning  with a blank line, and ending with a line beginning  with the three characters 'END', inclusive |
| sed -e '/^#/d' /etc/services | more | View the services file without the commented lines |
| sed '$s@\([^:]\*\):\([^:]\*\):\([^:]\*\ \)@::@g' path.txt | Reverse order of items in the last line of path.txt |
| sed -n -e '/regexp/{=;x;1!p;g;$!N;p;D;}'\  -e h file.txt | Print 1 line of context before and after the line matching,  with a line number where the matching occurs |
| sed '/regex/{x;p;x;}' file.txt | Insert a new line above every line matching regex |
| sed '/AAA/!d; /BBB/!d; /CCC/!d' file.txt | Match AAA, BBB and CCC in any order |
| sed '/AAA.\*BBB.\*CCC/!d' file.txt | Match AAA, BBB and CCC in that order |
| sed -n '/^.\{65\}/p' file.txt | Print lines 65 chars long or more |
| sed -n '/^.\{65\}/!p' file.txt | Print lines 65 chars long or less |
| sed '/regex/G' file.txt | Insert blank line below every line |
| sed '/regex/{x;p;x;G;}' file.txt | Insert blank line above and below |
| sed = file.txt | sed 'N;s/\n/\t/' | Number lines in file.txt |
| sed -e :a -e 's/^.\{1,78\}$/\  &/;ta' file.txt | Align text flush right |
| sed -e :a -e 's/^.\{1,77\}$/ &/;ta' -e \ 's/\( \*\)//' file.txt | Align text center |