

# Linux commands

(grep,sed,awk,gvim)

# GREP (global regular expression print)

- **Syntax :** `grep string filename`

`grep -i string` :to grep by considering case-sensitive

`egrep - "str1|str2|str3" filename` : to search multiple strings at a time in a file.

`grep -c str1 filename` : to count str1 in file

`grep -w str1 filename` :to print lines where str1 is present

`grep -v str1 filename` :to print lines where str1 is not present.

`grep -r str1 filename` :to search recursively.

# Sed (stream editor)

- Syntax : sed 's/oldword/newword' filename

sed 's/unix/linux' filename :replace unix as linux

sed 's/unix/linux/2' filename :replace 2<sup>nd</sup> occurrence of unix as linux

sed 's/unix/linux/g' filename :replace unix as linux globally(all)

sed '3s/unix/linux' filename :replace unix as linux in 3<sup>rd</sup> line

sed '1,3s/unix/linux' filename :replace unix as linux in 1<sup>st</sup> to 3<sup>rd</sup> lines

sed '2,\$s/unix/linux' filename :replace unix as linux from 2<sup>nd</sup> to end

sed '3d' filename :3<sup>rd</sup> line is deleted.

# awk

awk '{print \$1}' file : print first column in each line

awk '{print NF}' file :prints no of fields in a line

awk '{print NR}' file :to display line numbers

awk '/abc/{print}' file :print all lines which match with abc

awk '{print \$1,\$NF}' file : print 1<sup>st</sup> and last columns in all lines

# GVIM editor

- `gvim filename` – to create a file or to open an existing file
- `i` – to edit the file
- `:wq` to save file(write quit)
- `:x` to save file
- `:q` – quit the file with out saving
- `:q!` – to quit file forcefully

# Gvim

- Setting parameters

:set nu - to display linenumbers

:set nonu - to remove line numbers

:set ai -to set auto indentation

:set no ai –to remove indentation

:set ic - to set case sensitive mode

:set all - show all options for set command

:noh –to remove highlighting

:set showmode – display mode on last line of screen

:set noshowmode – turn off showmode

# gvim

## Search command

/string - search in forward manner

?string – search in backward manner in file

n – to navigate to next match

N – to navigate to previous match

f character – search single character in current line

; - to navigate to next match of character

, - to navigate to previous match of character

# Gvim

- Cursor movements

H - first line left corner of screen

M – middle line of screen

L – last line left corner of screen

gg – starting line of file

G – ending line of line

:n – cursor move to given line number

h – one character backward

j- down line

k- up

l- one character forward

zz – cursor at center of screen

Zt – cursor at top of screen

Zb – cursor at bottom of screen



# gvim (cursor movement.....)

^ - beginning of line

\$ - end of line

w- one word forward

b – one word backward

a – append at cursor

A – append at end of line

e – end of word

O – start of line

{ - beginning of paragraph

}- ending of paragraph

(- start of sentence

)- end of sentence

i – insert at cursor

# Gvim

Ctrl +d - scroll down half screen

Ctrl +u –scroll up half screen

Ctrl+f –scroll down full screen

ctrl+b –scroll up full screen

Ctrl+e – move screen down one line

Ctrl+y – move screen up one line

# gvim

- Copy(yank)

.....words.....

yw – copy word at/after cursor

yb – copy word before cursor

nyw – copy n words from cursor

nby – copy n words before cursor

.....lines.....

y0 – cursor to start of line

y\$ - cursor to end of line

{y} – copy paragraph upto new line

# gvim(copy.....)

y) - cursor to end of sentence

(y - cursor to start of sentence

yy - copy current line

nyy - copy n no of line from cursor

yg - copy from current line to start of file

yG - copy from current line to end of file

:n,m y - copy lines in range of n,m

# gvim

- paste

p - paste after cursor

P – paste before cursor

# gvim

- delete

.....character.....

x – delete single character at cursor

nx – deletes n no of characters after cursor

X – delete single character before cursor

nX – deletes n no of characters before cursor

.....word.....

dw– delete word at cursor

ndw – deletes n no of words after cursor

d^ - from cursor to start of line

d\$ - from cursor to end of line

# gvim(delete.....)

{d} – delete paragraph

dg – cursor to start of line

dG – cursor to end of file

.....lines.....

dd – delete current line

ndd – delete n lines from cursor

n,m d – deletes line in range of n,m

# gvim

v – visual mode – to select text

r – replace character at cursor

u – undo last change

U – undo all changes

o – open a new line

esc – terminate insert mode

i – insert mode



# gvim

- replace

syntax : [area]s/oldword/newword/

area

. -current line

n -nth line

.+m -current line +m lines

% - anyline

p,q - range from p line to q line

# gvim (replace.....)

`:%s/^/H/` - add H at start of every line

`:%s/.^/H/` - replace first char with H in every line

`:%s/$/H/` - add H at end of every line

`:%s/.$/H/` - replace last char with H in every line

`:%s/^/#/` - comment all lines

`:%s/^#//` - uncomment all lines

`:%s/[0-9]+//g` – search and remove all numbers

`:%s/is/of/g` – replace is with of globally

`:x,y s/old /new/` - replace old as new in range of x ,y lines

`:%s/is/of/gc` – replace is with of globally and count

`:%s/is/of/gi` – replace is with of globally by considering case sensitive

`:g/ pattern/d` – deletes lines with matched pattern

`:v/pattern/d` –deletes lines except matched patterns

# gvim

- open file from another file in gvim

:e file name

:e# - to switch between two files

- open multiple files in gvim

:tabe filename

:tabn navigate to next file

:tabp navigate to previous file

# gvim

- advanced commands

`:%s/\r\(\n\)/\1/g` – to remove ^M at end of each line in file.

`gvim diff file1 file2` – to compare two files at a time

`diff -s file1 file2` –to compare two files without opening in gvim

`ctrl +v` and use arrow keys to select text in file

`:3,6 m8` –to move 3 to 6 lines after 8<sup>th</sup> line in file