

VI TEXT EDITOR BRIEF START

1) **vi/vim** (slightly modernized vi) is a common text editor in majority of UNIX distributions, it is fully configurable and useful – it needs only a few commands to know to start editing:

```
#touch someTextFile.txt && vim someTextFile.txt
Esc – starting base command mode
a – insert text ( append mode )
Esc – return to command ( normal ) mode
:w – save document
:q – close document without saving
:q! - force closing document without saving changes
:wq – save and close
Ctrl + p – word/variables auto filling
Ctrl+o :w – save command :w-save for saving by shortcut Ctrl+o during insert mode
v – visual mode ( highline text fragment ):
    y – copy highlined
    p – paste highlined after cursor
Esc – return to command ( normal mode )
u – undo last change ( 1 change in vi; more in vim )
Ctrl+r/. – redo last change
?keyword – find keyword
:lineNumber – go to file line
```

2) vim configuration file (add after line „syntax on”):

```
#sudo vi /etc/vim/vimrc
    set autoindent
    set ts=4
    set sw=4
    set mouse=a
    set number
    set background=dark
    color evening
    map <F5> :w <bar> exec
    \ '!clear && g++ ' .shellescape('%').
    \ ' -o ' .shellescape('%:r').
    \ ' && ./' .shellescape('%:r') <CR>
```

It sets line autoindent, tab space number, block autoindent, mouse handling and line numbers.

3) view – vi file viewer

4) there are other editors, with different controls like **nano**.

5) terminal colorization – edit file in home directory ~/bashrc:

```
uncomment line:
#force_color_prompt=yes
```

6) different stuff:

```
split window
    :sp
    :vsp
split window, and open file
    :sp file.txt
switch view to neighbour window:
    Ctrl+w arrow
run shell command:
    :! ls
    :! ./RUN_COMMANDS.sh
```

ctags:

```
#sudo aptitude install -y exuberant-ctags
    :! ctags -f `pwd`/tags -R *
    :Ctrl+]      #jump to the tag
    :tn          #go to next tag definition
    :tp          #go to previous tag definition
    :ts          #list all tag definitions
```

:Ctrl+T #jump back in tag stack

shortcut:

:nnoremap <keystroke> vimCommand

Vim Visual Cheat Sheet

Sample.java [+] 23,29 83%

```

import java.awt.*;
import java.awt.event.*;

public class AnotherNew
{
    private boolean foo;
    private boolean aReallyLongName;
    private boolean aReallyShortName;
    private int bar1, bar2;

    private boolean isFooAndBar(){
        foo = false;
    }

    public void main(String args[]){
        foo = true;
        if(foo){
            bar1 = bar2 + 1 + fooClass.invokeRandomMethod();
            bar1 = bar2 + 2;
            bar1++;
            bar2++;
            if( aReallyLongName
                aReallyLongName
                aReallyShortName
            )
        }
    }
}

```

AnotherNew.java 3,0-1 Top Another.java 3,0-1 Top

Created by vgod, Dec. 2009

Movement/Range

Character

h j k l

← ↓ ↑ →

word, WORD(all non-blank ch)

w b next/prev word

W B next/prev WORD

e E end of word/WORD

Line

0 \$ begin/end of line

^ begin (non-blank) of line

Paragraph, Block

{ } prev/ next paragraph

[] begin/end of block

% matching parenthesis

Window, File

H top of win zt scroll to top

M mid of win zz scroll to middle

L btm of win zb scroll to bottom

C-B C-F prev/next page

gg G begin/end of file

mx 'x mark/jump to x

Search

* # find current word backward/forward

fx to character x to right

gd to definition of current word

/xxx search xxx

n N next/prev search result

Mode Commands

(ESC) (C-L) enter normal mode

v enter visual mode

V enter visual line mode

(C-v) enter visual block mode

i enter insert mode

R enter replace mode

a append

A append at end of line

General Commands

y yank/copy (range)

d delete/cut (range)

c modify (range)

x delete/cut (character)

D delete to end of line

C modify to end of line

p paste after cursor

J join lines

r replace (character)

> indent

< indent leftward

. redo

u undo

EX Commands

:w save(:wq save and quit)

:q quit(:q! quit anyway)

:e x edit file x

:n new window

:h vim help

:xx jump to line #xx

Auto-completion [insert mode]

C-N C-P auto-complete

C-X C-F next/prev keyword

C-X C-F auto-complete file name

Split window

:vsp :sp vertically/horizontally split

:diffs split and diff

C-W p to last accessed window

C-W w to next window

