

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

:%!xxd - convert current work into hex edition format (no updates on string interpreter)

:%!xxd -g4 - group bytes into 4B words

:%!xxd -r - convert back into ordinary file - must be done, before save (file write)

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

#system-wide clipboard for copying between vi/vim sessions:

```
#sudo apt-get install -y vim-gtk
```

#copy in Visual mode:

```
"+y
```

#paste in other terminal with different vim session:

```
"+p
```

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 (does not correctly handle with enums):
#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

```

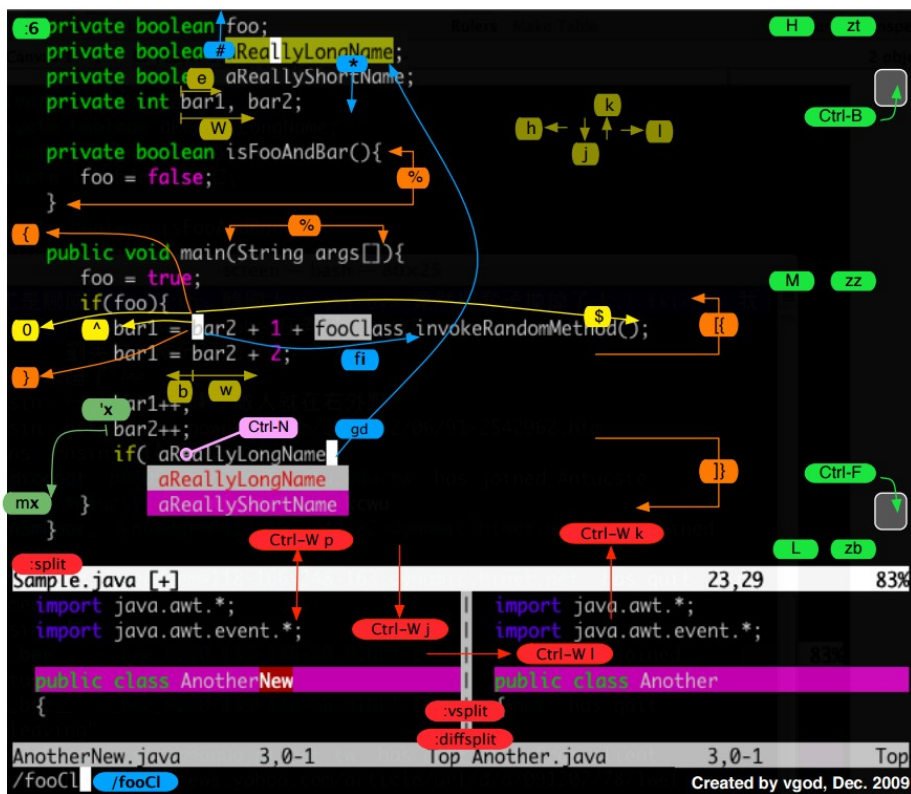
cscope:

```

#sudo aptitude install cscope
https://cscope.sourceforge.net/cscope\_vim\_tutorial.html
add cscope_maps.vim to .vimrc via added line „source ~/cscope_maps.vim”
Ctrl+\+s – show tag list within files
Ctrl+] - go to next tag from tag list
Ctrl+T – go backward
vim -t main
:cscope find symbol main = :cs f s main
Ctrl+spacebar+s - search and open in split horizontally
Ctrl+W+left/right – jump to another view

```

Vim Visual Cheat Sheet



Movement/Range	
Character	
h	j
k	l
←	→
↓	↑
word, WORD(all non-blank ch)	
w	b
W	B
e	E
Line	
0	\$
^	
Paragraph, Block	
{	}
[]
%	
Window, File	
H	zt
M	zz
L	zb
Ctrl-B	Ctrl-F
gg	G
mx	'x
Search	
*	#
fx	
gd	
/xxx	
n	N

Mode Commands	
ESC	C-]
V	
i	
R	
a	
A	
General Commands	
y	
d	
c	
x	
D	
C	
p	
J	
r	
>	
<	
.	
u	
EX Commands	
:w	
:q	
:e x	
:n	
:h	
:xx	
Auto-completion [insert mode]	
C-N	C-P
C-X	C-F
Split window	
:vsp	:sp
:diffs	
C-W p	
C-W w	

shortcut:

```

:nnoremap <keystroke> vimCommand

```