

Vim Helper

Mode Overview

```
graph TD;
    C[Command] -- return --> N[Normal];
    N -- aio --> I[Insert];
    N -- v --> V[Visual];
    N -- : --> C;
    I -- esc --> N;
    V -- esc --> N;
```

enter diff mode

`nvim -d file1 file2 nvim -d -o file2 file2` – for horizontal split

control page view - nvim default

ctrl	comment
d	1/2 page down
u	1/2 page up
a	increment number under cursor by one
x	decrement number under cursor by one
^	toggle to last buffer
w +	open new split, close split
[sc]	
w +	move between panes in vim
[hjkl]	
v	vertical select for group action like <code>\$A</code> ; for appending ; to each line end

control page view - settings with Telescope

ctrl	comment
e	exit nvim
q	open neo-tree view
o	URLOpenunder Cursor
f	find file with telescope in popup
g	search files with grep in popup
m	open marks in popup
b	open buffers in popup

Command

enter *Command* mode with `:` from *Normal* mode

command	comment
:w	write file
:q!	exit file without save
:wq	write file and exit
:qa	quit all buffers
:0	jump to first line
:\$	jump to last line
:%	use full file for operation
:!	perform Unix operator from shell
:ls	shows a list of all open buffers
:b2	displays buffer 2
:%s/search/subst/ge	use file, substitute <i>search</i> with <i>subst</i> globally with confirmation
:bufdo	substitute in all files in buffer and write changed files
%s/search/subst/ge	
update	
:reg	show registers
:norm Ivar	adds <i>var</i> to the beginning of each line, in visual mode
:set	change line numbering to relative
relativenumber	
:put q	paste register q to cursor position
:so	source current file
:split	horizontal split of buffer
:vsplit	vertical split of buffer
:sort	sort all lines in visual mode
:sort! u	sort in reverse order and make unique
:tab all	shows all opened files an tabs
:tabnew	opens the <i>file</i> in a new tab
<i>filename</i>	
:Lazy	open lazy paket manager
:e	open new file with given name in a new buffer
:InspectTree	open parser for file with tree
:Telescope	shows pop up with all marks - ctrl-m
marks	
:Mason	open LSP server config
:LspInfo	provide info on installed language servers
:h vim.lsp.buf	provide help pages on LSP functionality

marks

key	comment
mx	set mark x
dmx	delete mark x

key	comment
m]	move to next mark
m[move to prev mark

language server

	comment
gf	update style with none_ls
gh	hover information for content below cursor
gd	show definition
ca	code action

NeoTree

key	comment
a	create new file/folder
d	delete file
r	rename file
H	toggle hidden files
o	open file in system application, e.g. Firefox

Normal

Enter *Normal* mode with esc from *Insert* or __ Visual __ mode

Enter *Normal* mode with return from *Command* mode

motion	comment
gt	switch to next tab
gg	jump to first line
G	jump to last line
H	jump to top line
L	jump to bottom line
h j k l	left, down, up, right
w	next word
W	next whitespace
b	prev word
0	jump to first column in line
_	jumps to the first character in the current line
\$	jump to last character in line
/	search all occurrences of string

motion	comment
*	search all occurrences of word under cursor
n	next occurrence
N	prev occurrence
f	find first element in current line
%	jump to matching bracket ({

Enter Insert Mode	comment
a	append after cursor + enter insert mode
A	append at eol + enter insert mode
i	insert before cursor + enter insert mode
I	insert on left most column + enter insert mode
o	new line after + enter insert mode
O	new line before + enter insert mode

Enter Visual Mode	comment
v	enter visual mode for character selection
V	enter visual mode for line selection
vG	enter visual mode from current line till file end
vi(visual mode for everything inside bracket
vt=	visual mode for everything up to =
ctrl+v	start visual block and use I# to start all lines with #

Editing	comment
dd	delete line + put in register 0-9
yy	yank line + put in register 0-9
x	delete char after cursor + put in register
X	delete char before cursor
~	toggle upper/lower case under cursor
c	change mode
p	paste after cursor
P	paste before cursor
u	undo last action
.	redo last action
J	joins current and next line to one line with space
gv	reselect previous visual selection
ga	provide ASCII under cursor
gx	open URL under cursor in browser
gf	open text file under cursor

Editing	comment
guu	make line lower case
gUU	make line upper case
gg=G	jump to first line + correct indentation until last line

delete/yank/change area	comment
diw	delete inner word
yiw	yank inner word
ciw	change inner word
di(delete inner bracket
yi(yank inner bracket
ci(change inner bracket
da”	delete outer “”
ya”	yank outer “”
dt=	delete to =
ct=	change to =
yt=	yank to =
cW	change all up to next whitespace
d\$	delete from current position to line end

Markers	comment
mm	set mark m at current cursor
’m	jump to m mark
’.	jump to position of last change
”	jump to position before last jump

Register	comment
“1diw	delete word into register 1
“*d	delete visual mode into clipboard
“1yiw	yank word into register 1
“*y	yank into clipboard
“1p	paste from register 1
“*p	paste from clipboard

Makros	comment
qa.q	start makro with name <i>a</i> and record .
@a	replay makro with name <i>a</i>

Makros	comment
3@a	replay makro with name <i>a</i> 3 times

Visual

command	comment
>	shift one tab right
<	shift one tab left
3>	shift 3 tabs right
3<	shift 3 tabs left
d	delete + put in register
y	yank + put in register
J	joins lines in visual mode to one line