

my VIM Cheatsheet

Version 0.1

These shortcuts and descriptions are inspired by the awesome guide: "*Onramp to Vim*"[4], an austrian talk script "*Vim für Fortgeschrittene*"[1] and at least the "*Vim Reference Manual*"[3]. Another good resource for starting with Vim is the online Version of the vimtutor[2].

Motions (nouns)

Motions are in the sense of movement.

See Vim's help page for motions for a full listing: `:h motion`

Moving within a line

h, **l** move left/right by character
w move forward one (**w**)ord
b move to (**b**)eginning of a word
e move forward to the (**e**)nd of a word

Jumping within a line

f<char> (**f**)ind a character forward in a line and move to it
t<char> find a character forward in a line and move un(**t**)il it
(one character before)
F<char> (**f**)ind a character backward in a line and move to it
T<char> find a character backward in a line and move un(**t**)il it
; repeat last **f**, **t**, **F**, or **T** command
, repeat last **f**, **t**, **F**, or **T** command, but in opposite direction

Moving between lines

j, **k** move up/down one line
H, **M**, **L** move (**H**)igh, (**M**)iddle, or (**L**)ow within the viewport
/ search
n repeat last search - (**n**)ext
N repeat last search in opposite direction
<ctrl>u, **<ctrl>d** moves (**u**p or (**d**own half of a page
<NN>G (**G**)o to line number *NN*

Inserting text

a (**a**)ppend text after the cursor
A (**a**)ppend text at the end of the line
i (**i**)nset text before the cursor
I (**I**)nset text before the first non-blank in the line
o (**o**)pen a new line below the current line, append text
O (**O**)pen a new line above the current line, append text

Text Objects (nouns)

Text objects allow you to run the command from anywhere inside the text object

For a full listing, see `:h text-objects`

word

aw (**a**) (**w**)ord, includes surrounding white space
iw (**i**)nner (**w**)ord, does not include surrounding white space

Others: [**a**|**i**] +

s (**s**)entence
p (**p**)aragraph
" "double quoted string"
) (parenthesized block)
t **<h1>**single (**t**)ag**</h1>**

Commands (verbs)

After pressing the key-mapping for a given command Vim will wait for you to identify the text you want the command to operate on.

The simplest commands are made by simply repeating the operator a second time to act on the current line.

For example, where **d** is the operator for "delete", **dd** will delete the whole line. Each of **yy**, **cc**, **>>**, **=** behave similarly.

d (**d**)elete
c (**c**)hange
y (**y**)ank (copy)
v (**v**)isually select
>, **<** indent, dedent
= reformat (reindent, break long lines, etc.)

The Structure of an Editing Command (sentence)

<number><command><text object or motion>

Examples: Command + Motion

The following table shows some of the many variations of the delete operation you can build by combining the **d** operator mapping with a motion:

dw (**d**)elete from current position to the next (**w**)ord
de (**d**)elete to to the (**e**)nd of the current word
d2e (**d**)elete to to the (**e**)nd of the next word
dj (**d**)elete down a line (current and one below)
dt) (**d**)elete forward un(**t**)il closing paranthesis
d/world (**d**)elete up until the first search match for *world*

References

- [1] Guckes. *Vim für Fortgeschrittene*. URL: http://www.guckes.net/talks/vim/vim_advanced.html. (accessed: 02.11.2020).
- [2] *Interactive Vim tutorial*. URL: <https://openvim.com/>. (accessed: 30.10.2020).
- [3] Bram Moolenaar. *Vim Reference Manual*. URL: <https://vimhelp.org/quickref.txt.html#quickref>. (accessed: 02.11.2020).
- [4] Ben Orenstein & Chris Toomey. *Onramp to Vim*. URL: <https://thoughtbot.com/upcase/onramp-to-vim>. (accessed: 30.10.2020).