**NeoVim Cheatsheet**

**h l:** ← →

**j k:** ↓ ↑

**Command:**

x: delete a character

u: undo (one change)

U: undo for the whole line

r: replace the character

R: replace mode

p: paste

P: paste, the other direction

G: to the bottom of file

gg: to the beginning

<Ctr-g>: show position in file

<ctr-o>: go backward to last position

m[a-z]: make mark within a file

m[A-Z]: make mark globally

`[a-z]: jump to mark

‘[a-z]: jump to the line start of mark

`[: Start of the last change

`]: End of the last change

``:go to last position

<ctr-i>: go forward

<ctr-T>: pop tag

?[phrase]: search phrase backward

:#1,#2s/old/new/g: change all old to new between #1 and #2.

:! [cmd]: execute external command

:r [file/cmd]: retrieve output txt and paste it

o/O: insert newline below/above current ln.

n: next search

:set no[opt]: turn off an option

:set ic: ‘ignorecase’

.: repeat last CHANGE command, not working with u, <ctr-R>, cmds start with :

J: delete a line break

<ctr-]>:jump to tag (|tag|)

<ctr-U/F>: scroll up half/full of the screen

<ctr-D/B>: scroll dn half/full of the screen

z[t/z/b]: move cursor line to the top/mid/bottom of screen

D: = d$

C: = c$

:saveas: save as a new file

:xvsplit [file]: split window [file] with x lines

<ctr-W/w h/l>: jump between window

<ctr-W +/->: change the size of window

<ctr-W H/L>: move widows around

:close: close window

:only: close all other windows

:new: open an empty new window

nvim –O: open files in windows

nvim –d: open files in diff mode

:vertical diffsplit file: open another file in diff mode.

do: apply other window diff to current one

dp: put diff to other window

zo: open fold

zc: close fold

:tabedit file: open a new tab

**Operator:**

**d:** delete

c: change [delete + insert]

f: find character in the same line

F: backward of f

t: similar as f

T: backward of t

x%: go to x% of the file

“[a-z]: register a-z

all: all windows

**Motion:**

aw: a word

as: a sentence

w: until the start of the next word, EXCLUDING its first character.

b: opposite direction of w.

e - to the end of the current word, the last character.

ge: opposite direction of e

$ - to the end of the line, INCLUDING the last character.

0 –to the begin of the line.

**Visual Mode:**

Select certain part of text.

Regrex:

“\>” match the end of the word

“\<” match the beginning of the word

\***05.3**\* Simple mappings

A [mapping](https://neovim.io/doc/user/map.html#mapping) enables you to bind a set of Vim commands to a single key. Suppose,

for example, that you need to surround certain words with curly braces. In

other words, you need to change a [word](https://neovim.io/doc/user/motion.html#word) such [as](https://neovim.io/doc/user/motion.html#as) "amount" into "{amount}". With

the [:map](https://neovim.io/doc/user/map.html#:map) command, you can tell Vim that the F5 key does this job. The command

is [as](https://neovim.io/doc/user/motion.html#as) follows:

**:map <F5> i{<Esc>ea}<Esc>**

Note:

When entering this command, you must enter <F5> by typing four

characters. Similarly, [<Esc>](https://neovim.io/doc/user/intro.html#%3CEsc%3E) is not entered by pressing the [<Esc>](https://neovim.io/doc/user/intro.html#%3CEsc%3E)

key, but by typing five characters. Watch out for this difference

when reading the manual!

Let's break this down:

<F5> The F5 function key. This is the trigger key that causes the

command to be executed [as](https://neovim.io/doc/user/motion.html#as) the key is pressed.

i{<Esc> [Insert](https://neovim.io/doc/user/insert.html#Insert) the { character. The [<Esc>](https://neovim.io/doc/user/intro.html#%3CEsc%3E) key ends [Insert](https://neovim.io/doc/user/insert.html#Insert) mode.

[e](https://neovim.io/doc/user/motion.html#e) Move to the end of the [word](https://neovim.io/doc/user/motion.html#word).

a}<Esc> Append the } to the [word](https://neovim.io/doc/user/motion.html#word).

After you execute the "[:map](https://neovim.io/doc/user/map.html#:map)" command, all you have to [do](https://neovim.io/doc/user/diff.html#do) to put [{}](https://neovim.io/doc/user/intro.html#{}) around a

[word](https://neovim.io/doc/user/motion.html#word) is to put the cursor on the first character and press F5.

In this example, the trigger is a single key; [it](https://neovim.io/doc/user/motion.html#it) can be any [string](https://neovim.io/doc/user/eval.html#string). But when

you use an existing Vim command, that command will no longer be available.

You better avoid that.

One key that can be used with mappings is the [backslash](https://neovim.io/doc/user/intro.html#backslash). Since you

probably want to define more than one [mapping](https://neovim.io/doc/user/map.html#mapping), add another character. You

could map "\p" to add parentheses around a [word](https://neovim.io/doc/user/motion.html#word), and "\c" to add curly braces,

for example:

**:map \p i(<Esc>ea)<Esc>**

**:map \c i{<Esc>ea}<Esc>**

You need to type the \ and the [p](https://neovim.io/doc/user/change.html#p) quickly after another, so that Vim knows they

belong together.

The "[:map](https://neovim.io/doc/user/map.html#:map)" command (with no arguments) lists your current mappings. At

least the ones for [Normal](https://neovim.io/doc/user/intro.html#Normal) mode. More about mappings in section |[40.1](https://neovim.io/doc/user/usr_40.html#40.1)|.

USING REGISTERS

When you want to copy several pieces of text from one file to another, having

to switch between the files and [writing](https://neovim.io/doc/user/editing.html#writing) the target file takes a lot of time.

To avoid this, copy each piece of text to its own [register](https://neovim.io/doc/user/sponsor.html#register).

A [register](https://neovim.io/doc/user/sponsor.html#register) is a place where Vim stores text. Here we will use the

[registers](https://neovim.io/doc/user/change.html#registers) named a to [z](https://neovim.io/doc/user/vimindex.html#z) (later you will find out there are others). Let's copy

a [sentence](https://neovim.io/doc/user/motion.html#sentence) to the [f](https://neovim.io/doc/user/motion.html#f) [register](https://neovim.io/doc/user/sponsor.html#register) (f for First):

**"fyas**

The "yas" command yanks a [sentence](https://neovim.io/doc/user/motion.html#sentence) like before. It's the "[f](https://neovim.io/doc/user/motion.html#f) that tells Vim

the text should be placed in the [f](https://neovim.io/doc/user/motion.html#f) [register](https://neovim.io/doc/user/sponsor.html#register). This must come just before the

[yank](https://neovim.io/doc/user/change.html#yank) command.

Now [yank](https://neovim.io/doc/user/change.html#yank) three whole lines to the [l](https://neovim.io/doc/user/motion.html#l) [register](https://neovim.io/doc/user/sponsor.html#register) (l for line):

**"l3Y**

The [count](https://neovim.io/doc/user/intro.html#count) could be before the "[l](https://neovim.io/doc/user/motion.html#l) just [as](https://neovim.io/doc/user/motion.html#as) well. To [yank](https://neovim.io/doc/user/change.html#yank) a block of text to the

[b](https://neovim.io/doc/user/motion.html#b) (for block) [register](https://neovim.io/doc/user/sponsor.html#register):

**CTRL-Vjjww"by**

Notice that the [register](https://neovim.io/doc/user/sponsor.html#register) specification "[b](https://neovim.io/doc/user/motion.html#b) is just before the "[y](https://neovim.io/doc/user/change.html#y)" command.

This is required. If you would have put [it](https://neovim.io/doc/user/motion.html#it) before the "[w](https://neovim.io/doc/user/motion.html#w)" command, [it](https://neovim.io/doc/user/motion.html#it) would

not have worked.

Now you have three pieces of text in the [f](https://neovim.io/doc/user/motion.html#f), [l](https://neovim.io/doc/user/motion.html#l) and [b](https://neovim.io/doc/user/motion.html#b) [registers](https://neovim.io/doc/user/change.html#registers). Edit

another file, move around and place the text where you want [it](https://neovim.io/doc/user/motion.html#it):

**"fp**

Again, the [register](https://neovim.io/doc/user/sponsor.html#register) specification "[f](https://neovim.io/doc/user/motion.html#f) comes before the "[p](https://neovim.io/doc/user/change.html#p)" command.

You can put the [registers](https://neovim.io/doc/user/change.html#registers) in any order. And the text stays in the [register](https://neovim.io/doc/user/sponsor.html#register)

until you [yank](https://neovim.io/doc/user/change.html#yank) something else into [it](https://neovim.io/doc/user/motion.html#it). Thus you can put [it](https://neovim.io/doc/user/motion.html#it) [as](https://neovim.io/doc/user/motion.html#as) many times [as](https://neovim.io/doc/user/motion.html#as)

you like.

When you delete text, you can also specify a [register](https://neovim.io/doc/user/sponsor.html#register). Use this to move

several pieces of text around. For example, to delete-a-word and write [it](https://neovim.io/doc/user/motion.html#it) in

the [w](https://neovim.io/doc/user/motion.html#w) [register](https://neovim.io/doc/user/sponsor.html#register):

**"wdaw**

Again, the [register](https://neovim.io/doc/user/sponsor.html#register) specification comes before the delete command "[d](https://neovim.io/doc/user/change.html#d)".