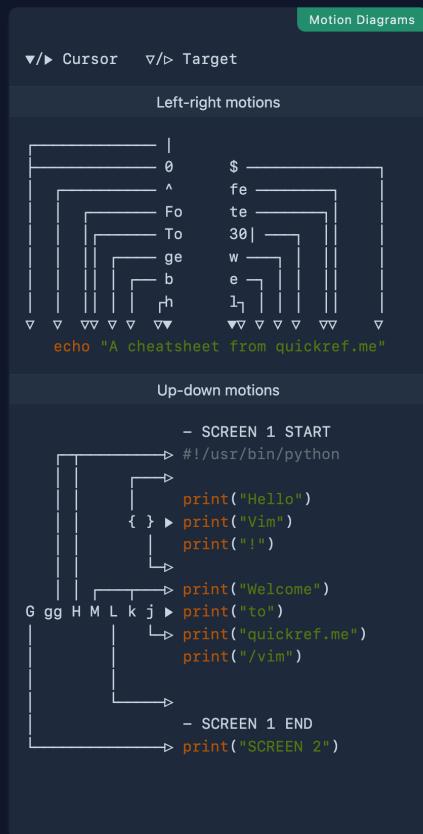




Vim cheatsheet

A useful collection of Vim 8.2 quick reference cheat sheets to help you learn vim editor faster.

Getting Started



Motions

h j k l	Arrow keys
<C-u> <C-d>	Half-page up/down
<C-b> <C-f>	Page up/down
b / w	Words
ge / e	Previous/Next end of word
0 (zero) / \$	Line
Start/End of line	
^	Start of line (non-blank)
Character	
Fe / fe	Move to previous/next e
To / to	Move before/after previous/next o
I / nl	Go to first/nth column
Document	
gg / G	First/Last line
:n / nG	Go to line n
{ / }	Next/Previous empty line
Window	
H M L	Top/Middle/Bottom screen
zt zZ zb	Top/Center/Bottom this line

Insert Mode

i / a	Insert before/after cursor
I / A	Insert start/end of line
o / O	Insert new line below/above
s / S	Delete char/line and insert
C / cc	Change to end of/current line
gi	Insert at last insert point
Esc <C-[>	Exit insert mode

Saving and Exiting

:w	Save
:q	Close file
:wq :x ZZ	Save and quit
:wqa	Save and quit all files
:q! ZQ	Force quit
:qa	Close all files
:qall	Force quit all files
:w now.txt	Write to now.txt
:sav new.txt	Save and edit new.txt
:w !sudo tee %	Write to readonly file

Normal Mode

r	Replace one character
R	Enter Replace mode
u / 3u	Undo changes 1 / 3 times
U	Undo changes on one line
J	Join with next line
<C-r> / 5 <C-r>	Redo changes 1 / 5 times

Cut and paste

x	Delete character (Cut)
p / P	Paste after/before
xp	Swap two characters
D	Delete to end of line (Cut)
dw	Delete word (Cut)
dd	Delete line (Cut)
ddp	Swap two lines
yy	Yank line (Copy)
"*p "+p	Paste from system clipboard
"*y "+y	Paste to system clipboard

Repeating

.	Repeat last command
;	Repeat latest f, t, F or T
,	Repeat latest f, t, F or T reversed
&	Repeat last :s
@:	Repeat a command-line command

Visual mode

v	Enter visual mode
V	Enter visual line mode
<C-v>	Enter visual block mode
ggVG	Select all text
> <	Shift text right/left

In visual mode

d x	Delete selection (Cut)
s	Replace selection
y	Yank selection (Copy)

Macros

qi	Record macro i
q	Stop recording macro
@i	Run macro i
7@i	Run macro i 7 times
@@	Repeat last macro

You can save macro for any letters not just i

Vim Operators

d	w
Operator	Motion
Combine operators with motions to use them	

Counts		
[count] <operator> <motion>		
<operator> [count] <motion>		
2dd	Delete 2 lines	
6yy	Copy 6 lines	
d3w	Delete 3 words	
d5j	Delete 5 lines downwards	
>4k	Indent 4 lines upwards	

d	Delete
y	Yank (copy)
c	Change (delete then insert)
p	Paste
=	Formats code
g~	Toggle case
gU	Uppercase
gu	Lowercase
>	Indent right
<	Indent left
!	Filter through external program

Combination	Description
dd	Delete current line
dj	Delete two lines
dw	Delete to next word
db	Delete to beginning of word
dfa	Delete until a char
d/hello	Delete until hello
cc	Change current line, synonym with S
yy	Copy current line
>j	Indent 2 lines
ggdG	Delete a complete document
gg=G	Indent a complete document
ggyG	Copy a whole document

Vim Text objects

Usage		
v	i / a	p
Operator inner / around Text object		

Selections	
vi"	Select inner quotes " ... "
va"	Select quotes "..."
vi[Select inner brackets [...]
va[Select brackets [...]
viw	Select inner word
vip	Select inner paragraph
vi <p>ip</p>	Select more paragraph

Text objects	
p	Paragraph
w	Word
W	WORD (surrounded by whitespace)
s	Sentence
[{{<	A [], (), or {} block
}}}>	A [], (), or {} block
'' '	A quoted string
b	A block [(
B	A block in [(
t	A HTML tag block

See :help text-objects

Delete	
diw	Delete inner word
dis	Delete inner sentence
di"	Delete in quotes
da"	Delete in quotes (including quotes)
di <p>ip</p>	Delete a paragraph

Misc	
ciw	Change inner word
ci"	Change inner quotes
cit	Change inner tags (HTML)
cip	Change inner paragraph
yip	Yank inner paragraph
yap	Yank paragraph (including newline)

Vim Multiple files

Buffers	
:e file	Edit a file in a new buffer
:bn	Go to the next buffer
:bp	Go to the previous buffer
:bd	Remove file from buffer list
:b 5	Open buffer #5
:b file	Go to a buffer by file
:ls	List all open buffers
:sp file	Open and split window
:vs file	Open and vertically split window
:hid	Hide this buffer
:wn	Write file and move to next
:tab ba	Edit all buffers as tabs

Windows	
<C-w> s	Split window
<C-w> v	Split window vertically
<C-w> w	Switch windows
<C-w> q	Quit a window
<C-w> T	Break out into a new tab
<C-w> x	Swap current with next
<C-w> - / +	Decrease/Increase height
<C-w> < / >	Decrease/Increase width
<C-w> I	Max out the width
<C-w> =	Equally high and wide
<C-w> h / l	Go to the left/right window
<C-w> j / k	Go to the up/down window

Tabs	
:tabe [file]	Edit file in a new tab
:tabf [file]	Open if exists in new tab
:tabc	Close current tab
:tabo	Close other tabs
:tabs	List all tabs
:tabr	Go to first tab
:tabl	Go to last tab
:tabm 0	Move to position 0
:tabn	Go to next tab
:tabp	Go to previous tab
Normal mode	
gt	Go to next tab
gT	Go to previous tab
2gt	Go to tab number 2

Vim Search and Replace

Search		Replace LINE		Replace FILE	
/foo	Search forward	:[range]s/{pattern}/{str}/[flags]		:%s/{pattern}/{str}/[flags]	
/foo\c	Search forward (case insensitive)	:s/old/new	Replace first	:%s/old/new	Replace first
?foo	Search backward	:s/old/new/g	Replace all	:%s/old/new/g	Replace all
\v\d+	Search with regex	:s/\vold/new/g	Replace all with regex	:%s/old/new/gc	Replace all (Confirm)
n	Next matching search pattern	:s/old/new/gc	replace all (Confirm)	:%s/old/new/gi	Replace all (ignore case)
N	Previous match	:s/old/new/i	Ignore case replace first	:%s/\vold/new/g	Replace all with regex
*	Search for current word forward	:2,6s/old/new/g	Replace between lines 2-6		
#	Search for current word backward				

Ranges		Global command		Inverse :g	
%	Entire file	:[range]g/{pattern}/{command}		:[range]v/{pattern}/{command}	
'<,'>	Current selection	:g/foo/d	Delete lines containing foo	:v/foo/d	Delete lines not containing foo (also :g!/foo/d)
5	Line 5	:g!/foo/d	Delete lines not containing foo		
5,10	Lines 5 to 10	:g/^s*\$d	Delete all blank lines		
\$	Last line	:g/foo/t\$	Copy lines containing foo to EOF		
2,\$	Lines 2 to Last	:g/foo/m\$	Move lines containing foo to EOF		
.	Current line	:g/^m0	Reverse a file		
,3	Next 3 lines	:g/^t.	Duplicate every line		
-3,	Forward 3 lines				

Substitute expression (magic)		Flags		Examples	
& \0	Replace with the whole matched	:s/a\ b/xxx\0xxx/g	# Modifies "a b" to "xxxxxxx xxxxxxx"		
\1... \9	Replace with the group 0-9	:s/test/\U& file/	# Modifies "test" to "TEST FILE"		
\u	Uppercase next letter	:s/(\test\)/\U\1\ae file/	# Modifies "test" to "TEST file"		
\U	Uppercase following characters	:s/\v[abc](\efg)\/\2\1/g	# Modifies "af fa bg" to "fa fa gb"		
\1	Lowercase next letter	:s/\vw+\/\u\0/g	# Modifies "bla bla" to "Bla Bla"		
\L	Lowercase following characters	:s/\v([ab]) ([cd])/\1x/g	# Modifies "a b c d" to "ax bx x x"		
\e	End of \u, \U, \1 and \L	:s/.*\L/&	# Modifies "HTML" to "html"		
\E	End of \u, \U, \1 and \L	:s/\v<(.)(\w*)/\u\1\L\2/g	# Make every first letter of a word uppercase		
		:%s/^(\.*\)(\r\?\n\1)\+\$	# Remove duplicate lines		
		/v^(\.*\)(\r?\n\1)+\$	# Convert HTML-Tags to uppercase		
		/^\(.*\)(\r?\n\1)+\$	# Find and append text to the end		
		/g/pattern/norm! @i	# Run a macro on matching lines		
		/^\(.*\)(\r?\n\1)\+\$	# View the duplicates lines		
		/v//./.-j	# View the duplicates lines (very magic)		
		/g<p1>/,<p2>/d	# Compress blank lines into a blank line		
			# Delete inclusively from <p1> to <p2>		

Vimdiff

Usage		Editing		Folds	
\$ vimdiff file1 file2 [file3]	\$ vim -d file1 file2 [file3]	:[range]diffget [bufspec]	# Open	:zo / zo	Open
		:[range]diffput [bufspec]		:zc / zC	Close
		do / :diffget	Obtain (get) difference	za / zA	Toggle
		dp / :diffput	Put difference	zv	Open folds for this line
		:dif	Re-scan differences	zM	Close all
		:difo	Switch off diff mode	zR	Open all
		:1,\$+1diffget	Get all differences	zm	Fold more (foldlevel += 1)
		zQ	Quit without changes	zr	Fold less (foldlevel -= 1)
		See: Ranges		zx	Update folds

Miscellaneous

Case	Jumping	Misc command-lines

vU	Uppercase character
vu	Lowercase character
~	Toggle case character
viw U	Uppercase word
viw u	Lowercase word
viw ~	Toggle case word
VU / gUU	Uppercase line
Yu / guu	Lowercase line
V~ / g~	Toggle case line
gggUG	Uppercase all text
ggguG	Lowercase all text
ggg~G	Toggle case all text

<C-o>	Go back to previous
<C-i>	Go forward
gf	Go to file in cursor
ga	Display hex, ascii value

:h	Help open help view
:edit!	Reload current file
:2,8m0	Move lines 2-8 to 0
:noh	Clear search highlights
:sort	Sort lines
:ter	Open a terminal window
:set paste	Enable Insert Paste sub-mode
:set nopaste	disable Insert Paste sub-mode
:cq	Exiting with an error (aborting Git)

Counters	
<C-a>	Increase number
<C-x>	Decrease number
Formatting	
:ce 8	Center lines between 8 columns
:ri 4	Right-align lines at 4 columns
:le	Left-align lines
See :help formatting	

:tag Classname	Jump to first definition of Classname
<C-]>	Jump to definition
g]	See all definitions
<C-t>	Go back to last tag
<C-o> <C-i>	Back/forward
:tselect Classname	Find definitions of Classname
:tjump Classname	Find definitions of Classname (auto-select 1st)

Marks	
^	Last position of cursor in insert mode
.	Last change in current buffer
^"	Last exited current buffer
^0	In last file edited
"	Back to line in current buffer where jumped from
^"	Back to position in current buffer where jumped from
['	To beginning of previously changed or yanked text
']	To end of previously changed or yanked text
'<	To beginning of last visual selection
'>	To end of last visual selection
ma	Mark this cursor position as a
`a	Jump to the cursor position a
'a	Jump to the beginning of the line with position a
d'a	Delete from current line to line of mark a
d'a	Delete from current position to position of mark a
c'a	Change text from current line to line of a
y'a	Yank text from current position to position of a
:marks	List all current marks
:delm a	Delete mark a
:delm a-d	Delete marks a, b, c, d
:delm abc	Delete marks a, b, c

Calculator	
<C-r> = 7*7	Shows the result
<C-r> = 10/2	Shows the result
Do this in INSERT mode	

Shell	
:!<shell>	Interpret Shell Command
:r!<shell>	Read in output of shell
:r!date	Insert date
:!!date	Replace current line with date

Command line	
<C-r><C-w>	Insert current word into the command line
<C-r>"	Paste from " register
<C-x><C-i>	Auto-completion of path in insert mode

Tricks	
Remove duplicate lines	
:sort %!uniq -u	
To number the lines in the file	
:%!cat -n	
Copy whole doc to clipboard	
:%w !pbcopy	# Mac OS X
:%w !xclip -i -sel c	# GNU/Linux
:%w !xsel -i -b	# GNU/Linux

Also see



[Devhints \(devhints.io\)](#)
[Vim cheatsheet \(vim.rotrr.com\)](#)
[Vim documentation \(vimdoc.sourceforge.net\)](#)
[Interactive Vim tutorial \(openvim.com\)](#)

Related Cheatsheet

[Emacs Cheatsheet](#)
Quick Reference

[VSCode Cheatsheet](#)
Quick Reference

Recent Cheatsheet

[Google Search Cheatsheet](#)
Quick Reference

[Kubernetes Cheatsheet](#)
Quick Reference

[Sed Cheatsheet](#)
Quick Reference

[Awk Cheatsheet](#)
Quick Reference

[ES6 Cheatsheet](#)
Quick Reference

[ASCII Code Cheatsheet](#)
Quick Reference



Share quick reference and cheat sheet for developers.

中文版 #Notes

f t i



Get infrastructure at software speed, powerful DevOps integrations, and hybrid cloud solutions.

ADS VIA CARBON

© 2023 QuickRef.ME, All rights reserved.