

Vim Cheat Sheet

Global	Cursor movement	Marking text (visual mode)	Macros	Working with multiple files
:help keyword - open help for keyword	; - repeat previous f, t, F or T movement	v - start visual mode, mark lines, then do a command (like y-yank)	qa - record macro a	:e file - edit a file in a new buffer
:o file - open file	, - repeat previous f, t, F or T movement, backwards	V - start linewise visual mode	q - stop recording macro	:bnext or :bn - go to the next buffer
:saveas file - save file as	} - jump to next paragraph (or function/block, when editing code)	o - move to other end of marked area	@a - run macro a	:bprev or :bp - go to the previous buffer
:close - close current pane	{ - jump to previous paragraph (or function/block, when editing code)	Ctrl + v - start visual block mode	@@ - rerun last run macro	:bd - delete a buffer (close a file)
K - open man page for word under the cursor	zz - center cursor on screen	O - move to other corner of block		:ls - list all open buffers
Cursor movement	Ctrl + b - move back one full screen	aw - mark a word	Cut and paste	:sp file - open a file in a new buffer and split window
h - move cursor left	Ctrl + f - move forward one full screen	ab - a block with ()	yy - yank (copy) a line	:vsp file - open a file in a new buffer and vertically split window
j - move cursor down	Ctrl + d - move forward 1/2 a screen	aB - a block with {}	2yy - yank (copy) 2 lines	Ctrl + ws - split window
k - move cursor up	Ctrl + u - move back 1/2 a screen	ib - inner block with ()	yw - yank (copy) the characters of the word from the cursor position to the start of the next word	Ctrl + ww - switch windows
l - move cursor right		iB - inner block with {}	y\$ - yank (copy) to end of line	Ctrl + wq - quit a window
H - move to top of screen	Insert mode - inserting/appendng text	Esc - exit visual mode	p - put (paste) the clipboard after cursor	Ctrl + ww - split window vertically
M - move to middle of screen	i - insert before the cursor	Visual commands	P - put (paste) before cursor	Ctrl + wh - move cursor to the left window (vertical split)
L - move to bottom of screen	I - insert at the beginning of the line	> - shift text right		Ctrl + wl - move cursor to the right window (vertical split)
w - jump forwards to the start of a word	a - insert (append) after the cursor	< - shift text left	dd - delete (cut) a line	
W - jump forwards to the start of a word (words can contain punctuation)	A - insert (append) at the end of the line	y - yank (copy) marked text	2dd - delete (cut) 2 lines	Ctrl + wj - move cursor to the window below (horizontal split)
e - jump forwards to the end of a word	o - append (open) a new line below the current line	d - delete marked text	dw - delete (cut) the characters of the word from the cursor position to the start of the next word	Ctrl + wj - move cursor to the window below (horizontal split)
E - jump forwards to the end of a word (words can contain punctuation)	O - append (open) a new line above the current line	~ - switch case	D - delete (cut) to the end of the line	Ctrl + wk - move cursor to the window above (horizontal split)
b - jump backwards to the start of a word	ea - insert (append) at the end of the word	Registers	d\$ - delete (cut) to the end of the line	
B - jump backwards to the start of a word (words can contain punctuation)	Esc - exit insert mode	:reg - show registers content	x - delete (cut) character	
% - move to matching character (default supported pairs: '()', '{}', '[]' - use :h matchpairs in vim for more info)	Editing	"xy - yank into register x		Search and replace
	r - replace a single character	"xp - paste contents of register x		/pattern - search for pattern
0 - jump to the start of the line	J - join line below to the current one with one space in between			?pattern - search backward for pattern
^ - jump to the first non-blank character of the line	gJ - join line below to the current one without space in between	Marks		\vpattern - 'very magic' pattern: non-alphanumeric characters are interpreted as special regex symbols (no escaping needed)
\$ - jump to the end of the line	cc - change (replace) entire line	:marks - list of marks		n - repeat search in same direction
g_ - jump to the last non-blank character of the line	cw - change (replace) to the end of the word	ma - set current position for mark A		N - repeat search in opposite direction
gg - go to the first line of the document	c\$ - change (replace) to the end of the line	`a - jump to position of mark A		:s/old/new/g - replace all old with new throughout file
G - go to the last line of the document	s - delete character and substitute text	y`a - yank text to position of mark A		:s/old/new/gc - replace all old with new throughout file with confirmations
5G - go to line 5	S - delete line and substitute text (same as cc)	Exiting		:noh - remove highlighting of search matches
fx - jump to next occurrence of character x	xp - transpose two letters (delete and paste)	:w - write (save) the file, but don't exit		
tx - jump to before next occurrence of character x	u - undo	:w !sudo tee % - write out the current file using sudo		Search in multiple files
Fx - jump to previous occurrence of character x	Ctrl + r - redo	:wq or :x or ZZ - write (save) and quit		:vimgrep /pattern/ {file} - search for pattern in multiple files
Tx - jump to after previous occurrence of character x	.	:q - quit (fails if there are unsaved changes)		:cn - jump to the next match
		:q! or ZQ - quit and throw away unsaved changes		:cp - jump to the previous match
		:wqa - write (save) and quit on all tabs		:copen - open a window containing the list of matches