Vim Cheat Sheet

viiii Cheat Sheet			
Global :h[elp] keyword - open help for keyword	Editing r - replace a single character	Macros qa - record macro a	Working with multiple files :e[dit] file - edit a file in a new buffer
:sav[eas] file - save file as	J - join line below to the current one with one space in between	${\bf q}$ - stop recording macro	:bn[ext] - go to the next buffer
:clo[se] - close current pane		@a - run macro a	:bp[revious] - go to the previous buffer
	gJ - join line below to the current one without space in between	@@ - rerun last run macro	:bd[elete] - delete a buffer (close a file)
:ter[minal] - open a terminal window	gwip - reflow paragraph	Cut and paste	:b[uffer]# - go to a buffer by index #
K - open man page for word under the cursor	g∼ - switch case up to motion	yy - yank (copy) a line	:b[uffer] file - go to a buffer by file
	gu - change to lowercase up to motion	2yy - yank (copy) 2 lines	:1s or :buffers - list all open buffers
Cursor movement h - move cursor left	gU - change to uppercase up to motion	yw - yank (copy) the characters of the word from the cursor position to the start	:sp[lit] file - open a file in a new buffer and split window
j - move cursor down	cc - change (replace) entire line	of the next word	:vs[plit] file - open a file in a new buffer and vertically split window
k - move cursor up	C - change (replace) to the end of the line	y\$ - yank (copy) to end of line	:vert[ical] ba[11] - edit all buffers as vertical windows
1 - move cursor right	c\$ - change (replace) to the end of the line	\boldsymbol{p} - put (paste) the clipboard after cursor	:tab ba[11] - edit all buffers as tabs
H - move to top of screen	ciw - change (replace) entire word	P - put (paste) before cursor	Ctrl + ws - split window
M - move to middle of screen	CW - change (replace) to the end of the	dd - delete (cut) a line	Ctrl + wv - split window vertically
L - move to bottom of screen	word	2dd - delete (cut) 2 lines	Ctrl + ww - switch windows
w - jump forwards to the start of a word	${f s}$ - delete character and substitute text	dw - delete (cut) the characters of the word from the cursor position to the start	Ctrl + wq - quit a window
W - jump forwards to the start of a word	S - delete line and substitute text (same as	6.3	Ctrl + wx - exchange current window with next one
(words can contain punctuation)	cc)	D - delete (cut) to the end of the line	Ctrl + w= - make all windows equal height & width
${f e}$ - jump forwards to the end of a word	xp - transpose two letters (delete and paste)	d\$ - delete (cut) to the end of the line	Ctrl + wh - move cursor to the left window (vertical split)
E - jump forwards to the end of a word (words can contain punctuation)	u - undo	x - delete (cut) character	Ctrl + wl - move cursor to the right window (vertical split)
b - jump backwards to the start of a word	U - restore (undo) last changed line	Indent text	Ctrl + wj - move cursor to the window below (horizontal split)
B - jump backwards to the start of a word	Ctrl + r - redo	>> - indent (move right) line one shiftwidth	Ctrl + wk - move cursor to the window above (horizontal split)
	• - repeat last command	<< - de-indent (move left) line one	Diff
% - move to matching character (default supported pairs: '()', '{}', '[]' - use: h	airs: 'O' 'O' 'T' - use : h Marking text (visual mode)	${f z}{f f}$ - manually define a fold up to motion	
	zd - delete fold under the cursor		
0 - jump to the start of the line	▼ - start linewise visual mode	>ib - indent inner block with ()	za - toggle fold under the cursor
^ - jump to the first non-blank character of the line	• - move to other end of marked area	>at - indent a block with <> tags	20 - open fold under the cursor
\$ - jump to the end of the line	Ctrl + v - start visual block mode	3== - re-indent 3 lines	zc - close fold under the cursor
g_ - jump to the last non-blank character	O - move to other corner of block aw - mark a word	=% - re-indent a block with () or {} (cursor on brace)	${f zr}$ - reduce (open) all folds by one level
of the line			zm - fold more (close) all folds by one level
$\mathbf{g}\mathbf{g}$ - go to the first line of the document		= iB - re-indent inner block with {}	

G - go to the last line of the document	ab - a block with ()	gg=G - re-indent entire buffer
5gg or 5G - go to line 5	aB - a block with {}] p - paste and adjust indent to current
	at - a block with <> tags	line
gd - move to local declaration	ib - inner block with ()	Exiting
gD - move to global declaration		: w - write (save) the file, but don't exit
$\boldsymbol{f}\boldsymbol{x}$ - jump to next occurrence of character x	<pre>iB - inner block with {} it - inner block with <> tags</pre>	:w !sudo tee % - write out the current file using sudo
${f tx}$ - jump to before next occurrence of character x	Esc - exit visual mode	:wq or :x or ZZ - write (save) and quit
$\mathbf{F}\mathbf{x}$ - jump to previous occurence of character x	Visual commands > - shift text right	∶ q - quit (fails if there are unsaved changes)
Tx - jump to after previous occurence of character x	< - shift text left	:q! or ZQ - quit and throw away unsaved changes
; - repeat previous f, t, F or T movement	y - yank (copy) marked text	: wqa - write (save) and quit on all tabs
, - repeat previous f, t, F or T movement,	d - delete marked text	Search and replace
backwards	~ - switch case	/pattern - search for pattern
} - jump to next paragraph (or function/block, when editing code)	u - change marked text to lowercase	?pattern - search backward for pattern
{ - jump to previous paragraph (or	U - change marked text to uppercase	\vpattern - 'very magic' pattern: non-
function/block, when editing code) zz - center cursor on screen	Registers :reg[isters] - show registers	alphanumeric characters are interpreted as special regex symbols (no escaping needed)
Ctrl + e - move screen down one line	content	- n - repeat search in same direction
(without moving cursor)	" xy - yank into register x	
Ctrl + y - move screen up one line (without moving cursor)	" xp - paste contents of register x	N - repeat search in opposite direction %s/old/new/g - replace all old with
Ctrl + b - move back one full screen	"+y - yank into the system clipboard register	new throughout file :%s/old/new/gc - replace all old
Ctrl + f - move forward one full screen	"+p - paste from the system clipboard register	with new throughout file with confirmations
Ctrl + d - move forward 1/2 a screen		:noh[lsearch] - remove
Ctrl + u - move back 1/2 a screen	Marks and positions :marks - list of marks	highlighting of search matches
nsert mode - inserting/appending text i - insert before the cursor	ma - set current position for mark A	Search in multiple files :vim[grep] /pattern/
I - insert at the beginning of the line	a - jump to position of mark A	{ `{file} `} - search for pattern in
a - insert (append) after the cursor	y`a - yank text to position of mark A	multiple files
A - insert (append) at the end of the line	0 - go to the position where Vim was previously exited	:cn[ext] - jump to the next match
• - append (open) a new line below the current line	" - go to the position when last editing this file	:cp[revious] - jump to the previous match
O - append (open) a new line above the current line	• - go to the position of the last change in this file	: cope[n] - open a window containing the list of matches
ea - insert (append) at the end of the	- go to the position before the last	:ccl[ose] - close the quickfix window
ea - insert (append) at the end of the word	- go to the position before the last jump : ju[mps] - list of jumps	Tabs

zi - toggle folding functionality
]c - jump to start of next change
[c - jump to start of previous change
do or :diffg[et] - obtain (get) difference (from other buffer)
dp or :diffpu[t] - put difference (to other buffer)
:diffthis - make current window part of diff
:dif[fupdate] - update differences
:diffo[ff] - switch off diff mode for current window

	$\textbf{Ctrl} + \textbf{i} - go \ to \ newer \ position \ in \ jump \\ list$	<pre>:tabnew or :tabnew {page.words.file} - open a file in a new tab Ctrl + wT - move the current split window into its own tab gt or :tabn[ext] - move to the next tab</pre>
$\label{eq:ctrl} \textbf{Ctrl} + \textbf{w} - \text{delete word before the cursor} \\ \text{during insert mode}$	Ctrl + o - go to older position in jump list	
Ctrl + j - begin new line during insert mode	: changes - list of changes	
Ctrl + t - indent (move right) line one	g, - go to newer position in change list	
shiftwidth during insert mode	g; - go to older position in change list	gT or :tabp[revious] - move to
Ctrl + d - de-indent (move left) line one shiftwidth during insert mode	Ctrl +] - jump to the tag under cursor	#gt - move to tab number #
$ \begin{array}{l} \textbf{Ctrl} + \textbf{n} \text{ - insert (auto-complete) next} \\ \text{match before the cursor during insert} \\ \text{mode} \end{array} $:tabm[ove] # - move current tab to the #th position (indexed from o)
Ctrl + p - insert (auto-complete) previous match before the cursor during		:tabc[lose] - close the current tab and all its windows
insert mode		: tabo[nly] - close all tabs except for the current one
Ctrl + rx - insert the contents of register x		: tabdo command - run the command
Esc - exit insert mode		on all tabs (e.g. :tabdo q-closes all opened tabs)