

vim Cheat Sheet

by mutanclan (mutanclan) via cheatography.com/79625/cs/19288/

Up-Down motions		
k	count lines upward	
j	count lines downward	
gk	count display lines upward	
gj	count display lines downward	
-	count lines upward on the firest non-blank char	
+	count lines downward on the first non-blank char	
_	count - 1 lines downward on the first non-blank char	
G	goto line count, default last line	
gg	goto line count, default first line	
:[range]	set cursor on last line number in range	
{count}%	go to count percentage in file	

Left-right	motions
Lon ngin	

Leit-i	Leit-right motions		
h	count chars to the left		
1	count chars to the right		
0	to the first char of the line		
٨	to the first non-blank char of the line		
\$	to the end of the line		
g_	to the last non-blank char of the line and <i>count</i> - 1 lines downward		
g0	wrap: to the first char of the screen line		
g^	rwap: to the first non-blank char of the screen line		
gm	like g0, but half a screenwidth to the		

Bv mutanclan	(mutanclan)

zb

Page 1 of 3.

Published 6th April, 2019.

Last updated 9th April, 2019.

Left-right motions (cont)	
g\$	wrap: to the last char of the screen line
1	to screen column <i>count</i> in current line
f <i>char</i>	to <i>count</i> occurence of <i>char</i> to the right
F <i>char</i>	to <i>count</i> occurence of <i>char</i> to the left
t <i>char</i>	till before <i>count</i> occurence of <i>char</i> to the right
T <i>char</i>	till after <i>count</i> occurence of <i>char</i> to the left
;	repeat latest f, t, F, T count times

repeat latest f, t, F, T count times in

opposite direction

Scrolling	
CTRL- U	scroll down half a screen of text
CTRL- D	scroll up half a screen of text
CTRL- E	scroll one line up
CTRL- Y	scroll one line down
CTRL- F	scroll forward a whole screen
CTRL- B	scroll backward a whole screen
ZZ	scroll to see the context of the line
zt	put the cursor line at the top

put the cursor line at the bottom

Wor	d motions
W	move forward count words
W	move forward count WORDS
b	move backward count words
В	move backward count WORDs
е	move to the next end of a word
Е	move to next end of a WORD
ge	move to the previous end of a word
gE	move to the previous end of a WORD
a WORD is white-space separated word.	
Оре	erators and motions

dd	delete whole line	
x or dl	delete char under the cursor	
X or dh	delete char left of the cursor	
D or d\$	delete to the end of the line	
C or c\$	chage to the end of the line	
s or cl	change one char	
S or	change whole line excluding indentation	
r	command to replace char under cursor	
	repeat last change	
Window	s and Panes	

CTRL-W +	increase current window height by N
CTRL-W -	decrease current window height by N

Sponsored by CrosswordCheats.com
Learn to solve cryptic crosswords!
http://crosswordcheats.com



right

cheatography.com/mutanclan/



vim Cheat Sheet

by mutanclan (mutanclan) via cheatography.com/79625/cs/19288/

Windows and Panes (cont)

CTRL-W = evensize all windows CTRL-W < decrease current window width by N

CTRL-W > increase current window width by N

Jumps	
CTRL-O	Go to [count] Older cursor position in jump list
CTRL-I	go to [count] newer cursor position in jump list
:ju[mps]	print the jump list
:cle[arjumps]	clear the jump list of the current window
g;	go to <i>count</i> older position in change list
g,	go to <i>count</i> newer position in change list
:changes	print the change list
%	find the next item in this line and jump to its match
[(go to <i>count</i> previous unmatched (
])	go to <i>count</i> next unmatched
[{	go to count previous unmatched {
]}	go to count next unmatched }
]m	go to count next start of a method
]M	go to count next end of a Method
[m	go to count previous start of a methdo

Jumps (cont)

go to count previous end of a method move to first visible line move to middle visible line move to last visible line

The following commands are "jump" commands: """, "`", "G", "/", "?", "n", "N", "-%", "(", ")", "[[", "]]", "{", "}", ":s", ":tag", "L", "-M", "H" and the commands that start editing a new file.

Using marks	
`` or "	to the position of the latest jump
CTRL-O	jump to older position
CTRL-I	jump back to newer position
:jumps	Give list of positions of jumps
m <i>char</i>	mark the place under cursor with mark <i>char</i>
`{a-z} or '{a-z}	move to the mark char in current buffer
`{ <i>A-Z0-9</i> } or '{ <i>A-Z0-9</i> }	move to the mark char in other file
`char	move to the mark char
'char	move to first non- blank char of line with mark <i>char</i>
:marks	get a list of marks
п	cursor position when last editing the file
[start of the last

change

marks

end of the last change

Delete the specified

Using marks (cont)

Delete all marks for current buffer " or `" to the position when last exiting the current buffer

'^ or to the position where the cursor was the last time when Insert mode was stopped

`. or '. to the position of the last change

'(or `(to the start of the current sentence

to the end of the current sentence ') or `)

to the start of the current '{ or `{ paragraph

When making jumps to positions further than within the same line, vim remembers the position before the jump and sets a mark.

Search mode

/	activate forward search mode
?	activate backward search
	mode
n	continue search forward
N	continue search backward
<up></up>	Search through history
	upwards
<down></down>	Search through history
	downwards
*	search <i>count</i> words under
	cursor forward
#	search count word under
	cursor backward
/>	match the end of a word
\<	match the beginning of a word

By mutanclan (mutanclan) cheatography.com/mutanclan/

Published 6th April, 2019. Last updated 9th April, 2019. Page 2 of 3.

:delm[arks] {marks}

Sponsored by CrosswordCheats.com Learn to solve cryptic crosswords! http://crosswordcheats.com



vim Cheat Sheet

by mutanclan (mutanclan) via cheatography.com/79625/cs/19288/

Visual mode	
٧	start visual mode
V	visual mode on whole lines
CTRL- V	visual block mode
0	move to the other end of selection
0	in block mode: move to the other corner of the same line

Operators	
С	change
d	delete
У	yank into register
~	swap case under cursor
g~[motion]	swap case within motion
gu[motion]	make lowercase within motion
gU[motion]	make uppercase within motion
!	filter through external program
g?	ROT-13 encoding
>	shift right
<	shift left
zf	define a fold
g@	call function set with the
	'operatorfunc' option

Moving text		
р	paste back last deleted text	
P	past back last deleted text before the cursor	
<i>number</i> p	like p but <i>number</i> times	

Moving text (cont)		
<i>number</i> P	like P but <i>number</i> times	
хр	swap the next char with the char under cursor	
У	yank char under cursor	
yw	yank word	
Y or yy	yank whole line	
*уу	yank whole line to clipboard	
*р	put text from the clipboard back	

Text object selection		
aw	a word	
iw	inner word	
aW	a WORD	
as	a sentence includes white space after the sentence	
ар	a paragraph	
a] or a[a [] block	
a) or a(or ab	a () block	
a< or a>	a <> block	
at	a tag block	
a} or a{ or aB	a {} block	
a" or a' or a`	a quoted string	
a blocks may be inner blocks with i instead of a. These commands can only be used while in		

Text object motions		
(count sentences backward	
)	count sentences forward	
{	count paragraphs backward	
}	count paragraphs forward	
]]	<pre>count sections forward or to the next { in the first column</pre>	
[[<pre>count sections backward or to the previous { in the first column</pre>	
][<pre>count sections forward or to the next { in the first column</pre>	
	count sections backward or to the previous } in the first column	



By mutanclan (mutanclan) cheatography.com/mutanclan/

Published 6th April, 2019. Last updated 9th April, 2019. Page 3 of 3.

Visual mode or after an operator

Sponsored by CrosswordCheats.com Learn to solve cryptic crosswords! http://crosswordcheats.com