

UNIX vi EDITOR COMMAND SUMMARY

This page can be folded on the line and propped on your keyboard as a handy reference when using the vi (visual) text editor. Hash in square brackets [#] indicates an optional numerical value. If omitted, the default is 1 (one). [m,n] indicates optional line numbers. \$ may be used in place of m or n inside square brackets to indicate the last line of the file.

SAVING/QUITTING

ZZ save changes and quit
:x<cr> save changes and quit
:wq [file]<cr> save changes to file and quit
:wq! [file]<cr> overwrite file with changes and quit
:[m,n]w [file]<cr> save lines m thru n to file w/o quit
:[m,n]w! [file]<cr> overwrite file with changes w/o quit
:q!<cr> quit w/o saving changes
:q<cr> quit after no changes
:e! [file] edit file
:e! reread the current file
:n edit next file

MARK COMMANDS

mc mark position with c
'c return to marked position c
d'c delete partial lines from mark c to cursor
d'c delete whole lines from mark c to cursor
y'c yank partial lines from mark c to cursor
y'c yank whole lines from mark c to cursor
c'c change partial lines from mark c to cursor
c'c change whole lines from mark c to cursor

MISCELLANEOUS COMMANDS

:set nu<cr> number the lines
:set nonu<cr> un-number the lines
u undo the last command
!command<cr>run command
!!command<cr>replace the current line with command's output
~ repeat the last change/delete/yank
& repeat the last :[m,n]s string
J substitution on current line only
J join the next line to end of this line
:nu display current line number

INSERTING TEXT

i insert text before cursor until <esc>
I insert text before first non-blank until <esc>
a insert text after cursor until <esc>
A insert text after last non-blank until <esc>
o open a line below the current one until <esc>
O open a line above the current one until <esc>
:i file<cr>insert text from file after the current line
P insert buffer contents after cursor
P insert buffer contents before cursor

DELETING TEXT

[#]dd delete # whole lines into buffer
D delete to end of line into buffer
[#]dw delete # words into buffer
:[m,n]d<cr> delete lines m through n
[#]x delete # characters into buffer

YANKING TEXT

[#]yy yank # whole lines into buffer
y\$ yank to end of line into buffer
[#]yw yank # words into buffer

CHANGING TEXT

[#]cc change # whole lines until <esc> into buffer
C change to end of line until <esc> into buffer
[#]cw change # words until <esc> into buffer
r change this character to another
s string change this character to string until <esc>
:[m,n]s/s1/s2<cr> change first occurrence of s1 to s2
:[m,n]s/s1/s2/g<cr> change all occurrences of s1 to s2

ADJUSTING THE SCREEN

z<cr> move current line to top of screen
Z move current line to middle of screen
z- move current line to bottom of screen
<ctrl-E> move forward one line
<ctrl-F> move forward one screen
<ctrl-B> move backward one screen
<ctrl-D> move forward 1/2 screen
<ctrl-U> move backward 1/2 screen
<ctrl-L> refresh the screen
<ctrl-R> refresh the screen
<ctrl-G> show file status

MOVING THE CURSOR

/string<cr> move cursor forward to string
?string<cr> move cursor backward to string
n move cursor to next instance of string
[#]h or ← move cursor left # characters
[#]l or → move cursor right # characters
[#]j or ↓ move cursor down # characters
[#]k or ↑ move cursor up # characters
w move cursor to beginning of next word
b move cursor back one word
e move cursor to end of word
[#]G move cursor to line #
G move cursor to last line of file
H move cursor to home line of screen
M move cursor to middle of screen
L move cursor to last line of screen
O move cursor to left margin
^ move cursor to first non-blank character
\$ move cursor to last non-blank character
% move cursor to matching parenthesis