

Summary of VI commands

This list is a summary of VI commands, categorized by function. There may be other commands available, so check the [on-line manual on VI](#). For easy reference, you can save this file as text and delete any commands you don't think you would use and print out the resulting shorter file.

Cutting and Pasting/Deleting text

- " Specify a buffer to be used any of the commands using buffers. Follow the " with a letter or a number, which corresponds to a buffer.
- D Delete to the end of the line from the current cursor position.
- P Paste the specified buffer before the current cursor position or line. If no buffer is specified (with the " command.) then 'P' uses the general buffer.
- X Delete the character before the cursor.
- Y Yank the current line into the specified buffer. If no buffer is specified, then the general buffer is used.
- d Delete until *where*. "dd" deletes the current line. A count deletes that many lines. Whatever is deleted is placed into the buffer specified with the " command. If no buffer is specified, then the general buffer is used.
- p Paste the specified buffer after the current cursor position or line. If no buffer is specified (with the " command.) then 'p' uses the general buffer.
- x Delete character under the cursor. A count tells how many characters to delete. The characters will be deleted after the cursor.
- y Yank until

Inserting New Text

- A Append at the end of the current line.
- I Insert from the beginning of a line.
- O (letter oh) Enter *insert* mode in a new line above the current cursor position.
- a Enter *insert* mode, the characters typed in will be inserted after the current cursor position. A count inserts all the text that had been inserted that many times.
- i Enter *insert* mode, the characters typed in will be inserted before the current cursor position. A count inserts all the text that had been inserted that many times.
- o Enter *insert* mode in a new line below the current cursor position.

Moving the Cursor Within the File

- ^B Scroll backwards one page. A count scrolls that many pages.
- ^D Scroll forwards half a window. A count scrolls that many lines.

- ^F Scroll forwards one page. A count scrolls that many pages.
- ^H Move the cursor one space to the left. A count moves that many spaces.
- ^J Move the cursor down one line in the same column. A count moves that many lines down.
- ^M Move to the first character on the next line.
- ^N Move the cursor down one line in the same column. A count moves that many lines down.
- ^P Move the cursor up one line in the same column. A count moves that many lines up.
- ^U Scroll backwards half a window. A count scrolls that many lines.
- \$ Move the cursor to the end of the current line. A count moves to the end of the following lines.
- % Move the cursor to the matching parenthesis or brace.
- ^ Move the cursor to the first non-whitespace character.
- (Move the cursor to the beginning of a sentence.
-) Move the cursor to the beginning of the next sentence.
- { Move the cursor to the preceding paragraph.
- } Move the cursor to the next paragraph.
- | Move the cursor to the column specified by the count.
- + Move the cursor to the first non-whitespace character in the next line.
- Move the cursor to the first non-whitespace character in the previous line.
- 0 (Zero) Move the cursor to the first column of the current line.
- B Move the cursor back one word, skipping over punctuation.
- E Move forward to the end of a word, skipping over punctuation.
- G Go to the line number specified as the count. If no count is given, then go to the end of the file.
- H Move the cursor to the first non-whitespace character on the top of the screen.
- L Move the cursor to the first non-whitespace character on the bottom of the screen.
- M Move the cursor to the first non-whitespace character on the middle of the screen.
- W Move forward to the beginning of a word, skipping over punctuation.
- b Move the cursor back one word. If the cursor is in the middle of a word, move the cursor to the first character of that word.
- e Move the cursor forward one word. If the cursor is in the middle of a word, move the cursor to the last character of that word.
- h Move the cursor to the left one character position.
- j Move the cursor down one line.
- k Move the cursor up one line.
- l Move the cursor to the right one character position.
- w Move the cursor forward one word. If the cursor is in the middle of a word, move the cursor to the first character of the next word.

Moving the Cursor Around the Screen

- ^E Scroll forwards one line. A count scrolls that many lines.

- ^Y** Scroll backwards one line. A count scrolls that many lines.
- z** Redraw the screen with the following options. "z<return>" puts the current line on the top of the screen; "z." puts the current line on the center of the screen; and "z-" puts the current line on the bottom of the screen. If you specify a count before the 'z' command, it changes the current line to the line specified. For example, "16z." puts line 16 on the center of the screen.

Replacing Text

- C** Change to the end of the line from the current cursor position.
- R** Replace characters on the screen with a set of characters entered, ending with the Escape key.
- S** Change an entire line.
- c** Change until
Replace one character under the cursor. Specify a count to replace a number of characters.
- s** Substitute one character under the cursor, and go into insert mode. Specify a count to substitute a number of characters. A dollar sign (\$) will be put at the last character to be substituted.

Searching for Text or Characters

- ,** Repeat the last f, F, t or T command in the reverse direction.
- /** Search the file downwards for the string specified after the /.
- ;** Repeat the last f, F, t or T command.
- ?** Search the file upwards for the string specified after the ?.
- F** Search the current line backwards for the character specified after the 'F' command. If found, move the cursor to the position.
- N** Repeat the last search given by '/' or '?', except in the reverse direction.
- T** Search the current line backwards for the character specified after the 'T' command, and move to the column after the if it's found.
- f** Search the current line for the character specified after the 'f' command. If found, move the cursor to the position.
- n** Repeat last search given by '/' or '?'.
- t** Search the current line for the character specified after the 't' command, and move to the column before the character if it's found.

Manipulating Character/Line Formatting

- ~** Switch the case of the character under the cursor.
- <** Shift the lines up to *where* to the left by one shiftwidth. "<<" shifts the current line to the left, and can be specified with a count.
- >** Shift the lines up to *where* to the right by one shiftwidth. ">>" shifts the current line to the right, and

can be specified with a count.

J Join the current line with the next one. A count joins that many lines.

Saving and Quitting

^\
Quit out of "VI" mode and go into "EX" mode. The EX editor is the line editor VI is build upon. The EX command to get back into VI is ":vi".

Q Quit out of "VI" mode and go into "EX" mode. The ex editor is a line-by-line editor. The EX command to get back into VI is ":vi".

ZZ Exit the editor, saving if any changes were made.

Miscellany

^G Show the current filename and the status.

^L Clear and redraw the screen.

^R Redraw the screen removing false lines.

^[Escape key. Cancels partially formed command.

^^ Go back to the last file edited.

! Execute a shell. If a

Repeat the previous ":s" command.

. Repeat the last command that modified the file.

: Begin typing an EX editor command. The command is executed once the user types return. (See section below.)

@ Type the command stored in the specified buffer.

U Restore the current line to the state it was in before the cursor entered the line.

m Mark the current position with the character specified after the 'm' command.

u Undo the last change to the file. Typing 'u' again will re-do the change.

EX Commands

The VI editor is built upon another editor, called EX. The EX editor only edits by line. From the VI editor you use the : command to start entering an EX command. This list given here is not complete, but the commands given are the more commonly used. If more than one line is to be modified by certain commands (such as ":s" and ":w") the range must be specified before the command. For example, to substitute lines 3 through 15, the command is ":3,15s/from/this/g".

:ab string strings

Abbreviation. If a word is typed in VI corresponding to string1, the editor automatically inserts the corresponding words. For example, the abbreviation ":ab usa United States of America" would insert the words, "United States of America" whenever the word "usa" is typed in.

`:map keys new_seq`

Mapping. This lets you map a key or a sequence of keys to another key or a sequence of keys.

`:q`

Quit VI. If there have been changes made, the editor will issue a warning message.

`:q!`

Quit VI without saving changes.

`:s/pattern/to_pattern/options`

Substitute. This substitutes the specified pattern with the string in the `to_pattern`. Without options, it only substitutes the first occurrence of the pattern. If a 'g' is specified, then all occurrences are substituted. For example, the command `:1,$s/Dwayne/Dwight/g` substitutes all occurrences of "Dwayne" to "Dwight".

`:set [all]`

Sets some customizing options to VI and EX. The `":set all"` command gives all the possible options. (See the section on customizing VI for some options.)

`:una string`

Removes the abbreviation previously defined by `":ab"`.

`:unm keys`

Removes the remove mapping defined by `":map"`.

`:vi filename`

Starts editing a new file. If changes have not been saved, the editor will give you a warning.

`:w`

Write out the current file.

`:w filename`

Write the buffer to the filename specified.

`:w >> filename`

Append the contents of the buffer to the filename.

`:wq`

Write the buffer and quit.

Author: Ben Yoshino (ben@wiliki.eng.hawaii.edu)

VI Quick Reference Sheet

To enter command mode press the **ESC** key

Command mode format [count] command [where] []indicates optional

Check Pren	%	Finds open pareses when on close pareses
Delete	x	Delete the character under the cursor
	dd	Delete current line
	d^	Deletes to beginning of line
	d\$	Deletes to end of line
	dw	Deletes to end of word
Down	j	Move the cursor down one line
Indent	<<	Revoke Indent
	>>	Indent
Insert	a	Enter insert mode char after cursor
	i	Enter insert mode char before cursor
Left	h	Move the cursor one position left
Paste	p	Past after cursor
	P	Past before cursor
Quit	:q	Quit
	:q!	Quit without saving
	:wq	Write to file and quit
	zz	[Write to file and quit]
Next search	n	Find next search same direction
	N	Find next search reverse direction
Right	l	Move the cursor right one position
Search	/word	Forward search word
	?word	Reverse search word
	/^word	For/Rev search word at the start of each line
	/word\$	For/Rev search word at the end of each line
	/*word	For/Rev search word wildcard
	f char	Right search for character on line
	F char	Left search for character on line
	\	for special characters used in search
Set Swift Width	:set sw=?	Set Swift Width (indent) to a value (?)
Up	k	Move the cursor up one line
Undo	u	Undo the last change [u again will redoit]
Write	:w	Write to file
	:w filename	Write to a new filename

There are 36 buffers a-z and 0-9.

The General purpose buffer is standard.

Placing a number or letter before the command also copies changes to a specific buffer.

Author Sam Watson