

Vi Cheat Sheet

[Click here for the Advanced VI Cheatsheet](#)

Modes

Vi has two modes insertion mode and command mode. The editor begins in command mode, where the cursor movement and text deletion and pasting occur. Insertion mode begins upon entering an insertion or change command. [ESC] returns the editor to command mode (where you can quit, for example by typing :q!). Most commands execute as soon as you type them except for "colon" commands which execute when you press the return key.

Quitting

:x Exit, saving changes
:q Exit as long as there have been no changes
ZZ Exit and save changes if any have been made
:q! Exit and ignore any changes

Inserting Text

i Insert before cursor

I Insert before line
a Append after cursor
A Append after line
o Open a new line after current line
O Open a new line before current line
r Replace one character
R Replace many characters

Motion

h Move left
j Move down
k Move up
l Move right
w Move to next word
W Move to next blank delimited word
b Move to the beginning of the word
B Move to the beginning of blank delimited word
e Move to the end of the word
E Move to the end of Blank delimited word
(Move a sentence back
) Move a sentence forward
{ Move a paragraph back
} Move a paragraph forward
0 Move to the begining of the line
\$ Move to the end of the line
1G Move to the first line of the file
G Move to the last line of the file
nG Move to nth line of the file
:n Move to nth line of the file
fc Move forward to c
Fc Move back to c
H Move to top of screen
M Move to middle of screen
L Move to botton of screen
% Move to associated (), { }, []

Deleting Text

Almost all deletion commands are performed by typing d followed by a motion. For example, dw deletes a word. A few other deletes are:

x Delete character to the right of cursor
X Delete character to the left of cursor
D Delete to the end of the line

```

68 dd Delete current line
69 :d Delete current line
70
71
72 Yanking Text
73
74 Like deletion, almost all yank commands are performed by typing y followed by a
motion. For example, y$ yanks to the end of the line. Two other yank commands are:
75 yy Yank the current line
76 :y Yank the current line
77
78
79 Changing text
80
81 The change command is a deletion command that leaves the editor in insert mode. It
is performed by typing c followed by a motion. For wxample cw changes a word. A few
other change commands are:
82 C Change to the end of the line
83 cc Change the whole line
84
85
86 Putting text
87
88 p Put after the position or after the line
89 P Put before the poition or before the line
90
91
92 Buffers
93
94 Named buffers may be specified before any deletion, change, yank or put command. The
general prefix has the form "c where c is any lowercase character. for example, "adw
deletes a word into buffer a. It may thereafter be put back into text with an
appropriate "ap.
95
96
97 Markers
98
99 Named markers may be set on any line in a file. Any lower case letter may be a
marker name. Markers may also be used as limits for ranges.
100 mc Set marker c on this line
101 `c Go to beginning of marker c line.
102 'c Go to first non-blank character of marker c line.
103
104
105 Search for strings
106
107 /string Search forward for string
108 ?string Search back for string
109 n Search for next instance of string
110 N Search for previous instance of string
111
112
113 Replace
114
115 The search and replace function is accomplished with the :s command. It is commonly
used in combination with ranges or the :g command (below).
116 :s/pattern/string/flags Replace pattern with string according to flags.
117 g Flag - Replace all occurrences of pattern
118 c Flag - Confirm replaces.
119 & Repeat last :s command
120
121
122 Regular Expressions
123
124 . (dot) Any single character except newline
125 * zero or more occurances of any character
126 [...] Any single character specified in the set
127 [^...] Any single character not specified in the set
128 ^ Anchor - beginning of the line
129 $ Anchor - end of line
130 \< Anchor - begining of word
131 \> Anchor - end of word
132 \(...\) Grouping - usually used to group conditions

```

```

133 \n Contents of nth grouping
134
135 [...] - Set Examples
136 [A-Z] The SET from Capital A to Capital Z
137 [a-z] The SET from lowercase a to lowercase z
138 [0-9] The SET from 0 to 9 (All numerals)
139 [./=+] The SET containing . (dot), / (slash), =, and +
140 [-A-F] The SET from Capital A to Capital F and the dash (dashes must be specified
first)
141 [0-9 A-Z] The SET containing all capital letters and digits and a space
142 [A-Z][a-zA-Z] In the first position, the SET from Capital A to Capital Z
143 In the second character position, the SET containing all letters
144
145 Regular Expression Examples
146 /Hello/ Matches if the line contains the value Hello
147 /^TEST$/ Matches if the line contains TEST by itself
148 /^[a-zA-Z]/ Matches if the line starts with any letter
149 /^[a-z].*/ Matches if the first character of the line is a-z and there is at least
one more of any character following it
150 /2134$/ Matches if line ends with 2134
151 /\(21|35\) / Matches if the line contains 21 or 35
152 Note the use of ( ) with the pipe symbol to specify the 'or' condition
153 /[0-9]*/ Matches if there are zero or more numbers in the line
154 /^[^#]/ Matches if the first character is not a # in the line
155 Notes:
156 1. Regular expressions are case sensitive
157 2. Regular expressions are to be used where pattern is specified
158
159 Counts
160
161 Nearly every command may be preceded by a number that specifies how many times it is
to be performed. For example, 5dw will delete 5 words and 3fe will move the cursor
forward to the 3rd occurrence of the letter e. Even insertions may be repeated
conveniently with this method, say to insert the same line 100 times.
162
163
164 Ranges
165
166 Ranges may precede most "colon" commands and cause them to be executed on a line or
lines. For example :3,7d would delete lines 3-7. Ranges are commonly combined with
the :s command to perform a replacement on several lines, as with
:.,$s/pattern/string/g to make a replacement from the current line to the end of the
file.
167 :n,m Range - Lines n-m
168 :. Range - Current line
169 :$ Range - Last line
170 :'c Range - Marker c
171 :% Range - All lines in file
172 :g/pattern/ Range - All lines that contain pattern
173
174
175 Files
176
177 :w file Write to file
178 :r file Read file in after line
179 :n Go to next file
180 :p Go to previous file
181 :e file Edit file
182 !!program Replace line with output from program
183
184
185 Other
186
187 ~ Toggle upp and lower case
188 J Join lines
189 . Repeat last text-changing command
190 u Undo last change
191 U Undo all changes to line

```