

maketechasier

# Vim Keyboard Shortcuts Cheatsheet

By Mayank Sharma – Posted on Dec 27, 2013 in Linux



*This article is part of the VIM User Guide series:*

- [The Beginner's Guide to Start Using Vim](#)
- [Vim Keyboard Shortcuts Cheatsheet](#)
- [5 Vim Tips and Tricks for Experienced Users](#)
- [3 Useful VIM Editor Tips and Tricks for Advanced Users](#)
- [How to Turn Vim into a Word Processor](#)

The Vim editor is a command-line based tool that's an enhanced version of the venerable vi editor. Despite the abundance of graphical rich text editors, familiarity with Vim will help every Linux user — from an experienced system administrator to a newbie Raspberry Pi user.

The light-weight editor is a very powerful tool. In the hands of an experienced operator, it can do wonders. Besides regular text editing functions, the editor also supports advanced features such as find & replace based on regular expressions and encoding conversion as well as programming features such as syntax highlighting and code folding.

One important thing to note when using Vim, is that the function of a key depends on the “mode” the editor is in. For example, pressing the alphabet “j” will move the cursor down one line in the “command mode”. You'll have to switch to the “insert mode” to make the keys input the character they represent.

Here's a cheatsheet to help you get the most out of Vim.

Shortcut Keys	Function
<b>Main</b>	
Escape key	Gets out of the current mode into the “command mode”. All keys are bound of commands.
i	“Insert mode” for inserting text.

Keys behave as expected.

:	"Last-line mode" where Vim expects you to enter a command such as to save the document.
---	---

## Navigation keys

h	moves the cursor one character to the left.
j or Ctrl + J	moves the cursor down one line.
k or Ctrl + P	moves the cursor up one line.
l	moves the cursor one character to the right.
0	moves the cursor to the beginning of the line.
\$	moves the cursor to the end of the line.
^	moves the cursor to the first non-empty character of the line
w	move forward one word (next alphanumeric word)
W	move forward one word

(delimited by a white space)

5w	move forward five words
b	move backward one word (previous alphanumeric word)
B	move backward one word (delimited by a white space)
5b	move backward five words
G	move to the end of the file
gg	move to the beginning of the file.

## Navigate around the document

(	jumps to the previous sentence
)	jumps to the next sentence
{	jumps to the previous paragraph
}	jumps to the next paragraph
[[	jumps to the previous section
]]	jumps to the next section
[]	jump to the end of the previous section

<b>]]</b>	jump to the end of the next section
-----------	-------------------------------------

---

## Insert text

---

<b>a</b>	Insert text after the cursor
<b>A</b>	Insert text at the end of the line
<b>i</b>	Insert text before the cursor
<b>o</b>	Begin a new line below the cursor
<b>O</b>	Begin a new line above the cursor

---

## Special inserts

---

<b>:r [filename]</b>	Insert the file [filename] below the cursor
<b>:r ![command]</b>	Execute [command] and insert its output below the cursor

---

## Delete text

---

<b>x</b>	delete character at cursor
----------	----------------------------

dw	delete a word.
d0	delete to the beginning of a line.
d\$	delete to the end of a line.
d)	delete to the end of sentence.
dgg	delete to the beginning of the file.
dG	delete to the end of the file.
dd	delete line
3dd	delete three lines

## Simple replace text

r{text}	Replace the character under the cursor with {text}
R	Replace characters instead of inserting them

## Copy/Paste text

yy	copy current line into storage buffer
----	---------------------------------------

<code>["x]yy</code>	Copy the current lines into register x
<code>p</code>	paste storage buffer after current line
<code>P</code>	paste storage buffer before current line
<code>["x]p</code>	paste from register x after current line
<code>["x]P</code>	paste from register x before current line

## Undo/Redo operation

<code>u</code>	undo the last operation.
<code>Ctrl+r</code>	redo the last undo.

## Search and Replace keys

<code>/search_text</code>	search document for search_text going forward
<code>?search_text</code>	search document for search_text going backward

n	move to the next instance of the result from the search
N	move to the previous instance of the result
:%s/original/replacement	Search for the first occurrence of the string “original” and replace it with “replacement”
:%s/original/replacement/g	Search and replace all occurrences of the string “original” with “replacement”
:%s/original/replacement/gc	Search for all occurrences of the string “original” but ask for confirmation before replacing them with “replacement”

## Bookmarks

m {a-z A-Z}	Set bookmark {a-z A-Z} at the current cursor position
:marks	List all bookmarks
`{a-z A-Z}	Jumps to the bookmark {a-z A-Z}

## Select text



---

v	Enter visual mode per character
V	Enter visual mode per line
Esc	Exit visual mode

---

## Modify selected text

---

~	Switch case
d	delete a word.
c	change
y	yank
>	shift right
<	shift left
!	filter through an external command

---

## Save and quit

---

:q	Quits Vim but fails when file has been changed
----	--

---

<code>:W</code>	Save the file
<code>:w new_name</code>	Save the file with the new_name filename
<code>:wq</code>	Save the file and quit Vim.
<code>:q!</code>	Quit Vim without saving the changes to the file.
<code>ZZ</code>	Write file, if modified, and quit Vim
<code>ZQ</code>	Same as <code>:q!</code> Quits Vim without writing changes

## Download VIM Keyboard Shortcuts Cheatsheet

Can't get enough of this? We have prepared a downloadable cheat sheet for you so you can access to it when you need it.

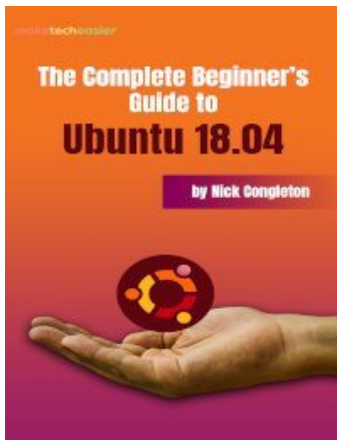
[Download it here!](#)

Is this article useful?

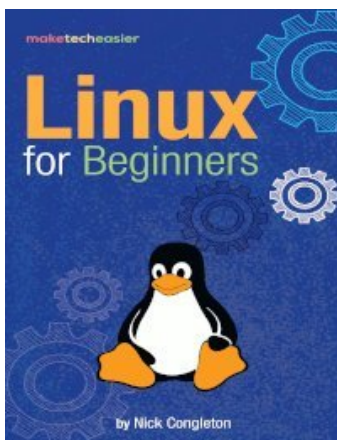
### Ebooks



## The Complete Beginner's Guide to Linux Mint



## The Complete Beginner's Guide to Ubuntu 18.04



## Linux for Beginners

[More ebooks »»](#)[Comments \(1\)](#)[f Facebook](#)[🐦 Tweet](#)

## One comment

**dragonmouth**

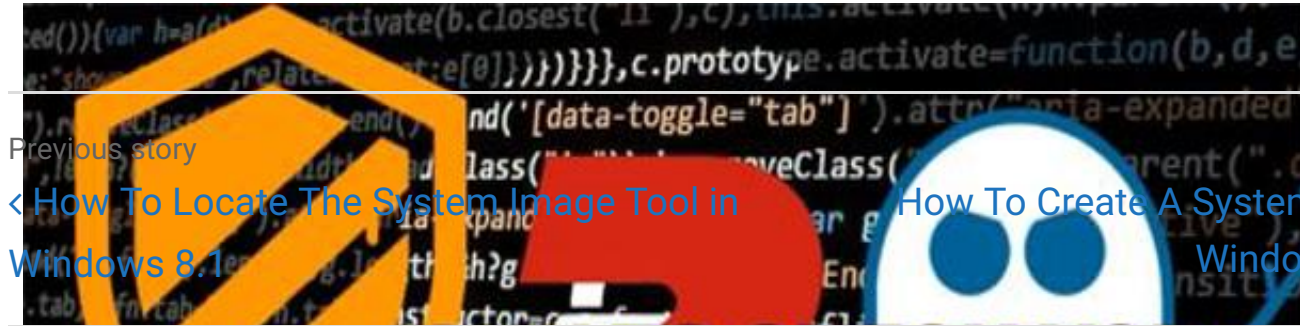
Dec 28, 2013 at 7:06 am

“In the hands of an experienced operator, it can do wonders.”

The problem is that it takes a long, frustrating time to become an “expert operator”. I prefer “nano”. The commands and key strokes are less arcane and esoteric.

---

Comments are closed.



Previous story

< [How To Locate The System Image Tool in Windows 8.1](#)

Next story

[How To Create A System Image in Windows 8/8.1](#) >

Featured Posts



[FlatHub vs. Snap Store: Which App Store Should You Use?](#)



[How to Manage the Priority of I/O Processes in Linux](#)



[How to Set and List Environment Variables in Linux](#)



## How to Manage and Restore Tmux Sessions in Linux



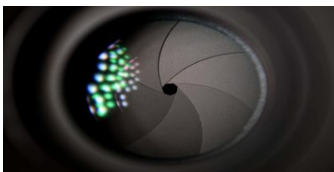
## How to Sync Linux Time with NTP Server



## How to Install a Dictionary for Use in Linux Terminal



## How to Run Mac Apps on Linux



## [How to Reduce Video File Sizes Using FFMPEG from Linux Terminal](#)



[About](#) [Contact](#) [Advertise](#) [Write For Us](#) [Terms of Use](#) [Privacy Policy](#)

[RSS Feed](#) [Terms](#)

© 2007 - 2019 Uqnic Network Pte Ltd. All rights reserved.