

Manual for VIM/Terminal

Ibrohim Nosirov

April 8, 2020

1 Terminal/BASH

The terminal is in essence, the computer. Here you can do pretty much anything you like, from running inventory on your computer to writing malware to take down the 1% ☹. I don't think we'll go that far with this manual. Let's just stick to the basics.

`ls` lists files

`cd` go to main directory

`cd ..` go back one directory

`cd [name of directory]`

go to the following directory (must be below the directory you are in)

`cp [files to be copied] [copy name/target directory/both]`

copies a file to designated location/file name.

Usage might be something like `cp manual.tex Documents/my_manual.tex`, which would create a copy of this document in your `Documents` folder and title it `my_manual.tex`

`mv [file name] [new location OR new file name]`

used to move or rename a file depending on whether you declare a new location or a new file name

`mkdir` make new directory (aka folder)

`vi` or `vim [file name].[file type]`

opens a file or creates a new file with the entered name. File extensions are important with every command, but especially in VI, if you misspell a file ending like `.jav` instead of `.java`, it will create an empty file that will not run.

`rm [file name or directory]` removes a file.

2 VIM

If terminal is your hand that moves around the workspace, then VIM is your pen used to write documents.

`:w` 'write' or 'save' the file.

`:q` 'quit' the file.

`:wq` 'write then quit' the file. One of VI's advantages is in being able to string together commands.

`e` move to end of next word.

`b` move to beginning of previous word.