

Basic use of command lines

Alise Ponsero

Abstract

Learn the basic commands to navigate in a command-line terminal.

Citation: Alise Ponsero Basic use of command lines. **protocols.io**

dx.doi.org/10.17504/protocols.io.s9reh56

Published: 05 Sep 2018

Before start

For Windows users :

You will need to install a shell, I personally recommend Cygwin. To install Cygwin, go to [this page and follow the installation instructions for windows](#).

For MacOS and Linux users :

A shell is provided in your OS, no need to install anything.

MacOS users can however choose to use iTerm2 instead of the built in shell. To download and install iTerm2, go to [this page and follow the instructions](#).

Protocol

Open a command-line terminal

Step 1.

Note : This protocol requires you to have a command-line interface installed on your laptop. Please read the 'Before starting' section in 'Guidelines and Warnings' of this protocol to learn more about the installation.

For Windows users :

Launch Cygwin terminal from the application menu

For MacOS users :

Launch iTerm2, and open a new window.

For linux (Ubuntu users) :

Open the Dash by clicking the Ubuntu icon in the upper-left, type 'terminal', and select the Terminal application from the results that appear.

You can also use the shortcut Alt+tab+T.

Basic command lines

Step 2.

Here are the most common command lines that you will need to use.

Those commands work on Cygwin, iTerm and Linux shell.

command	example	effect
ls	ls /applications	will show the user all the folders and files stored in a given folder (here the 'applications' folder).
cd	cd /user/iMicrobe	the cd command (change directory), allows the user to navigate between directories. You can either provide an absolute path (the path to reach a directory from the root) or a relative path (the path to reach the directory from the directory that you are currently in).
mv	mv /user/fileA /user/folderB/	The mv command (move) allows the user to move a file to another folder. The first part of the command (/user/fileA) lists the object to move. The second part of the command (/user/folderB/) lists where the object should be moved to.
man	man cd	the man command (manual) shows the manual for the inputted command. The man command is your best friend.
mkdir	mkdir folderC	The mkdir (make directory) command allows the user to create a new folder (here the name of the new folder is 'folderC')
rm	rm fileA rm -r folderB	the rm (remove) command allows you to delete a file or a directory. Please note that the 'rm -r' command will delete the given folder, and any files/folder contained in it. Be careful!
touch	touch my_file.txt	The touch command allows the user to create an empty file (here the name of the file will be 'my_file.txt').
clear	clear	The clear command allows the user to clear the previous commands in the terminal.
cp	cp fileA /user/folderB cp -r /user/folderC /user/folderB/	The cp (copy) command allows the user to copy a file or a folder. the 'cp -r' command will allow the copy of a folder and its content. The first part of the command is the object that you want to copy, the second part of the command is the path you want to copy the object into.

Other important shortcuts :

shortcut	example	effect
..	cd ..	Directory parent to the current one. The command 'cd ..' will change the user's current working directory one one level up.
.	./programA	Current directory. The command './programA' will execute the file 'programA' located in the user's current working directory.

