

The background of the entire page is a repeating pattern of teal-colored pine needles or branches, rendered in a sketchy, hand-drawn style. These elements are scattered across the white background, with some appearing behind the dark blue box.

Raspberry Pi for Beginners

LESSON 4

MAKERHOUSE
EMPOWERING MAKERS

TABLE OF CONTENTS

| | | |
|----------|--|-----------|
| 1 | INTRODUCTION | 2 |
| 1.1 | Terminal | 2 |
| 1.1.1 | Getting Started | 3 |
| 2 | Terminal Exploration | 4 |
| 2.1 | Navigate Folders and List Files | 4 |
| 2.1.1 | List (ls) | 5 |
| 2.1.2 | Change Directory (cd) | 5 |
| 2.1.3 | Present Working Directory (pwd) | 7 |
| 2.2 | Read and Write Text | 7 |
| 2.2.1 | Create Folder (mkdir) | 7 |
| 2.2.2 | Editing (nano) | 8 |
| 2.2.3 | Concatenate (cat) | 9 |
| 2.2.4 | Remove FOLDER/Files (rm) | 9 |
| 2.3 | Permission to Install | 10 |
| 2.3.1 | Install | 10 |
| 2.3.2 | Search | 11 |
| 2.3.3 | Clear / Remove | 12 |
| 3 | Miscellaneous | 13 |
| 3.1 | Find Help | 13 |
| 3.2 | Tab Completion | 14 |
| 3.3 | Wildcard | 14 |

MASTERING COMMAND LINE

1 INTRODUCTION

As you are going through Lesson 1 – 3, Raspbian OS offers both GUI (Graphical User Interface) and Terminal – opening a software or programs. Undeniable, GUI's provided a friendly visual in exploring the Raspbian desktop – mouse-click on files and folders in the file manager, etc. - however somehow it loose power and expressiveness. On the other side, Terminal – vice-versa -- you get full control, and it is a fast way to do everything – opening software, managing files and folders, updating and installing software; almost anything with privilege.

A good example, and if you still remember, in Lesson 3; Turtle module, we have been introduced with the power of Terminal – finding the help instruction (method) for controlling our Turtle.

Can GUI do that?

Sadly, can't but even if GUI can give us, it's a long journey – click here and there, open this and that file – tedious.

1.1 TERMINAL

Bash, Shell, or Terminal and whatever its name, referring to the command line – a black television screen looks likes. For windows user – beginners -- you might feel obscured. It's normal.

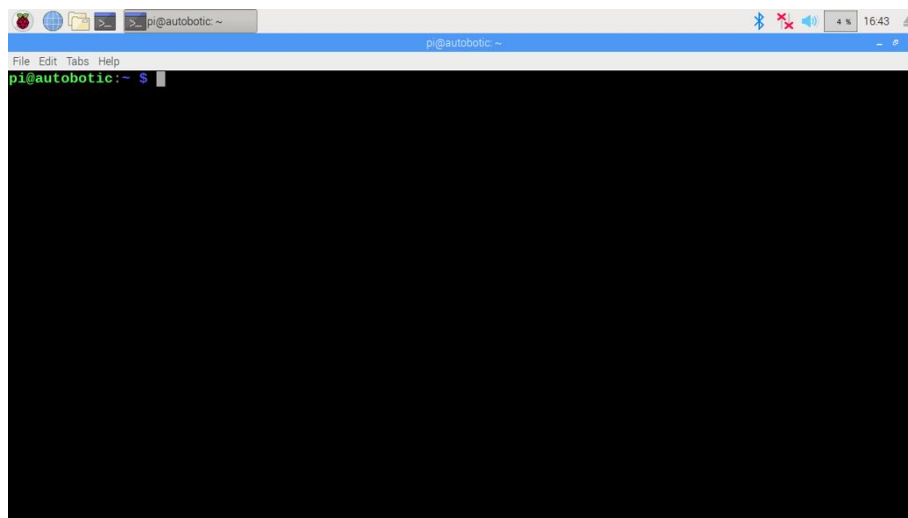


Figure 1: You may familiar with this -- Terminal

Command line often the best way (Thanks! Raspbian; Unix System) – the user only needs to type in the instruction where the computer will respond to it. It's clear, quick and unambiguous – simplify complex tasks, or instantly carry out simple ones.

Did you agree?

1.1.1 GETTING STARTED

From the beginning, we have been configured our Raspbian to boot up straight to the Desktop mode. So, we have a beautiful and sleek look of Raspbian Desktop.

How to open the Terminal?

You shall be an expert in doing that now.

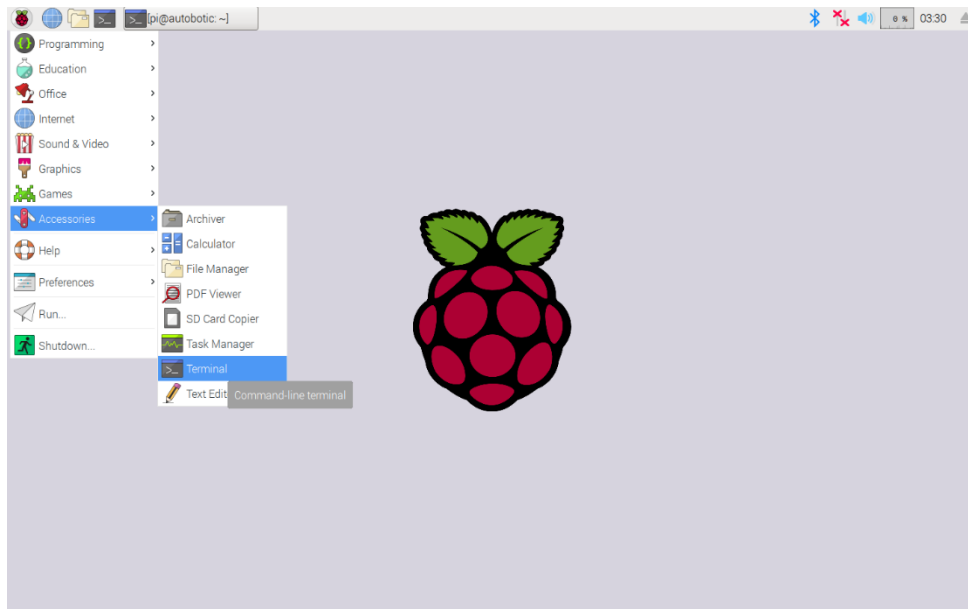


Figure 2: **Main menu > Accessories > Terminal**

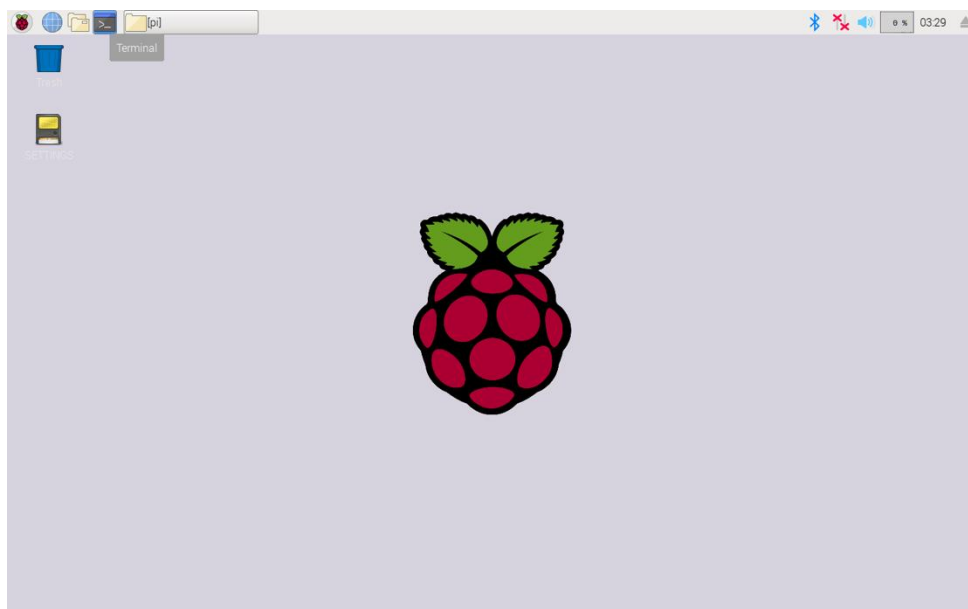
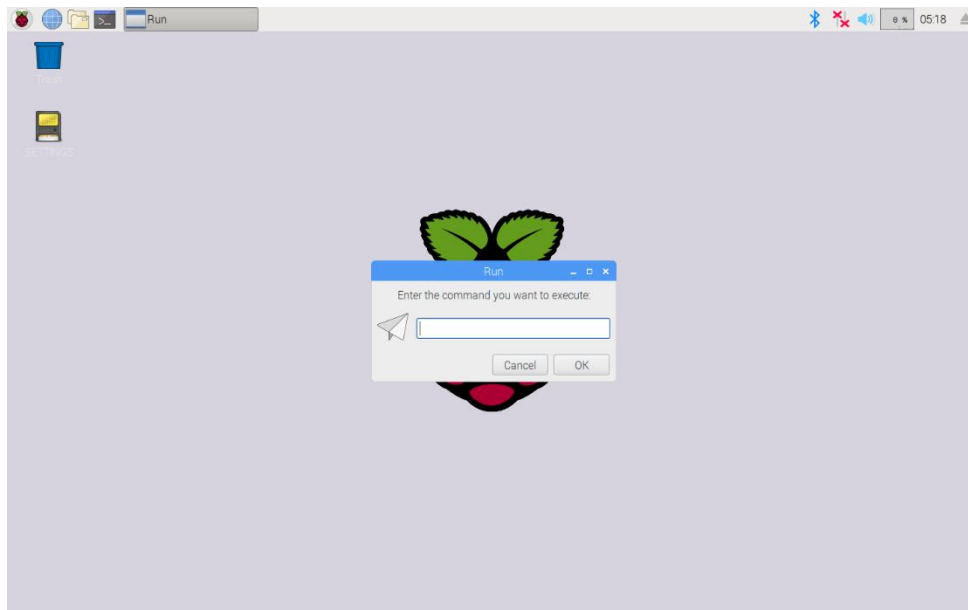


Figure 3: Terminal Icon (short-cuts)

Figure 4: Hit **alt + F7** -- **ltxterminal**

Interestingly, you can try other methods. Try – a combination of **ctrl+ alt + [F1 until F6]** – you manageably to switch between Terminal and GUI in full-screen mode.

| No | Instruction | Description |
|----|-----------------|--------------------------------------|
| 1 | Ctrl + Alt + F1 | Terminal |
| 2 | Ctrl + Alt + F2 | Further Terminal ; password required |
| 3 | Ctrl + Alt + F3 | Further Terminal ; password required |
| 4 | Ctrl + Alt + F4 | Further Terminal ; password required |
| 5 | Ctrl + Alt + F5 | Further Terminal ; password required |
| 6 | Ctrl + Alt + F6 | GUI |

Table 1: Switch between Terminal and GUI.

2 TERMINAL EXPLORATION

For the beginners, you may face with \$ prompt most of the time in the terminal. But don't panic! This lesson will guide you to find your way around the terminal on the Raspberry Pi. Now prepare, pen and or paper if need. Let's begin our terminal exploration.

**** These lessons only will focus on the basics**

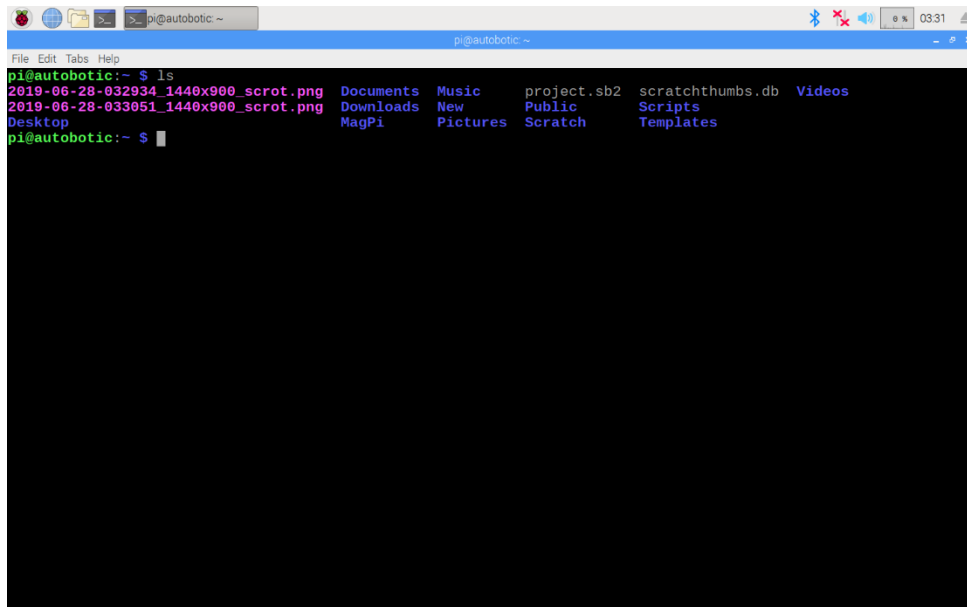
2.1 NAVIGATE FOLDERS AND LIST FILES

You might say that using the GUI to navigate folders and list file through the file manager is much intuitive. Yes, it is true, but give a try and feel how the command line give you much more power compare to the GUI.

2.1.1 LIST (LS)

ls – list directory content

ls [option] [file ...] – list directory content

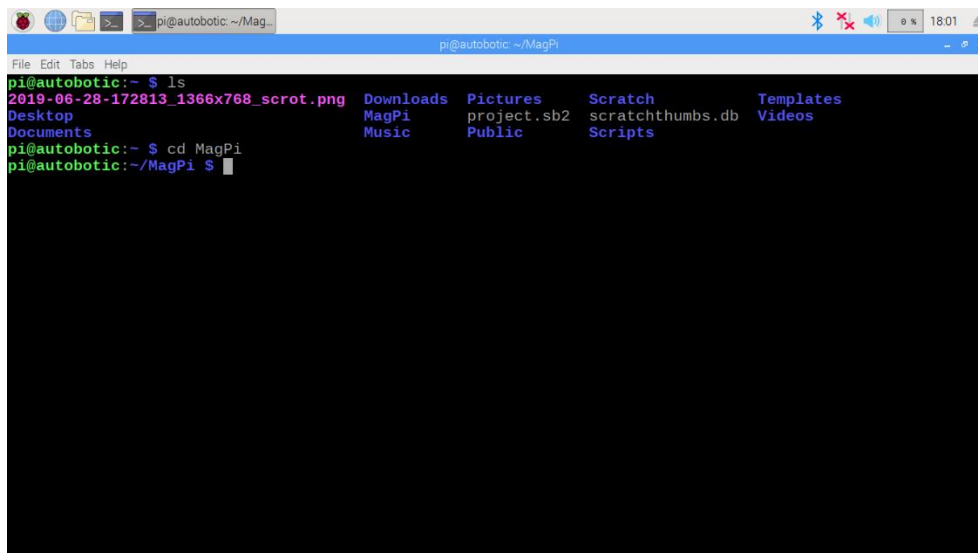


```
pi@autobotic: ~  
pi@autobotic:~$ ls  
2019-06-28-032934_1440x900_screenshot.png  Desktop  Documents  Music  project.sb2  scratchthumbs.db  Videos  
2019-06-28-033051_1440x900_screenshot.png  Downloads  MagPi      New     Public      Scripts  
pi@autobotic:~$
```

Figure 5: Using **ls** in “home” pi directory

2.1.2 CHANGE DIRECTORY (CD)

cd [file ...] – change working directory



```
pi@autobotic: ~  
pi@autobotic:~$ ls  
2019-06-28-172813_1366x768_screenshot.png  Desktop  Documents  Downloads  MagPi      Music  Pictures  project.sb2  Public  Scratch  scratchthumbs.db  Templates  Videos  
pi@autobotic:~$ cd MagPi  
pi@autobotic:~/MagPi$
```

Figure 6: Using **cd** from “home” pi into “MagPi” directory

You can type “pwd” to confirm you have successfully move from home pi to MagPi directory (see 2.1.3).

2.1.2.1 HOME DIRECTORY (~)

Home directory is abbreviated as “~”.

cd ~ – change to “home” working directory

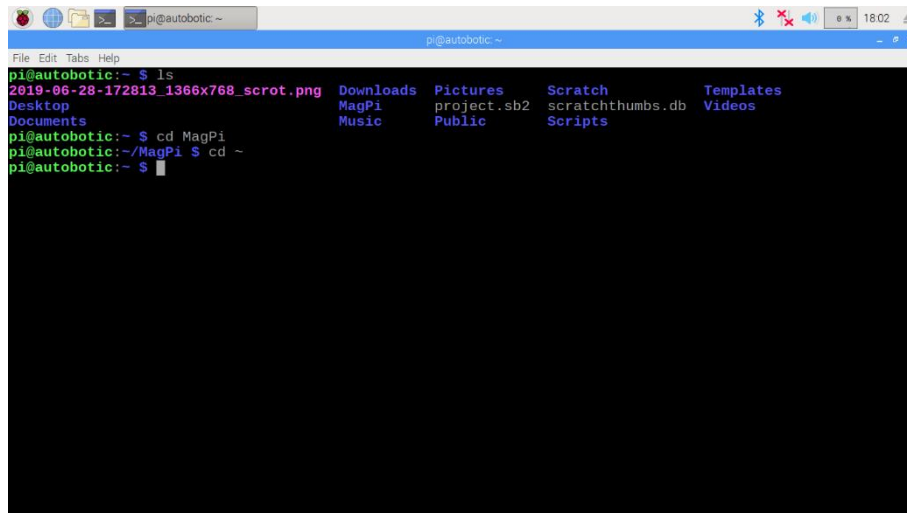


Figure 7: Using **cd** from “MagPi” pi into “home” directory

You can type “pwd” to confirm you have successfully move from MagPi to home pi directory (see 2.1.3).

2.1.2.2 LAST/PREVIOUS WORKING DIRECTORY (-)

“-” is a shortcut for “wherever I was before I came here” – last previous working directory.

cd - – change to last previous working directory

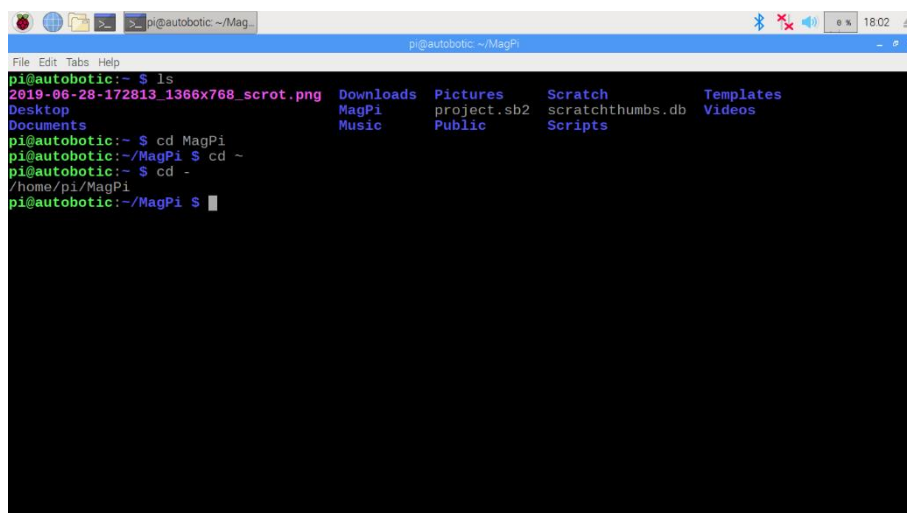


Figure 8: Using **cd** from “home” pi into “MagPi” directory – “MagPi” is last previous working directory

You can type “pwd” to confirm you have successfully move from home pi to MagPi directory (see 2.1.3).

2.1.3 PRESENT WORKING DIRECTORY (**PWD**)

pwd – return working directory name

```

pi@autobotic: ~/MagPi
File Edit Tabs Help
pi@autobotic:~$ ls
2019-06-28-172813_1366x768_scrot.png Desktop Pictures Scratch Templates
MagPi project.sb2 scratchthumbs.db Videos
Documents Music Public Scripts
pi@autobotic:~$ cd MagPi
pi@autobotic:~/MagPi$ cd -
pi@autobotic:~$ cd -
/home/pi/MagPi
pi@autobotic:~/MagPi$ pwd
/home/pi/MagPi
pi@autobotic:~/MagPi$

```

Figure 9: Ensure working in correct directory by using **pwd**

2.2 READ AND WRITE TEXT

In GUI, create-view-alter files or folders require you to mouse-click, naming, write on the dedicate text file, etc. Terminal offer more simple way – for sure fast.

2.2.1 CREATE FOLDER (**MKDIR**)

mkdir [file ...] – make directories

```

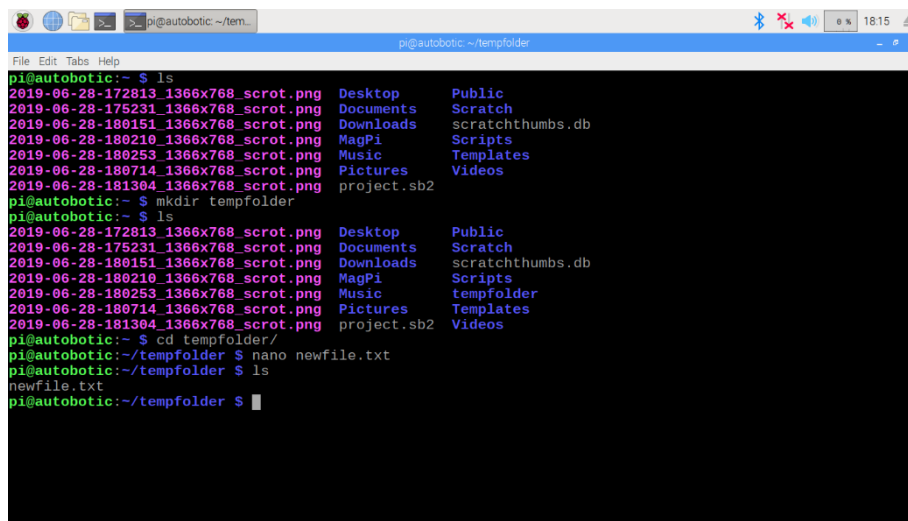
pi@autobotic: ~
File Edit Tabs Help
pi@autobotic:~$ ls
2019-06-28-172813_1366x768_scrot.png Desktop Public
2019-06-28-175231_1366x768_scrot.png Documents Scratch
2019-06-28-180151_1366x768_scrot.png Downloads scratchthumbs.db
2019-06-28-180210_1366x768_scrot.png MagPi Scripts
2019-06-28-180253_1366x768_scrot.png Music Templates
2019-06-28-180714_1366x768_scrot.png Pictures Videos
2019-06-28-181304_1366x768_scrot.png project.sb2
pi@autobotic:~$ mkdir tempfolder
pi@autobotic:~$ ls
2019-06-28-172813_1366x768_scrot.png Desktop Public
2019-06-28-175231_1366x768_scrot.png Documents Scratch
2019-06-28-180151_1366x768_scrot.png Downloads scratchthumbs.db
2019-06-28-180210_1366x768_scrot.png MagPi Scripts
2019-06-28-180253_1366x768_scrot.png Music tempfolder
2019-06-28-180714_1366x768_scrot.png Pictures Templates
2019-06-28-181304_1366x768_scrot.png project.sb2 Videos
pi@autobotic:~$

```

Figure 10: Create a new directory – tempfolder – in “home” pi directory

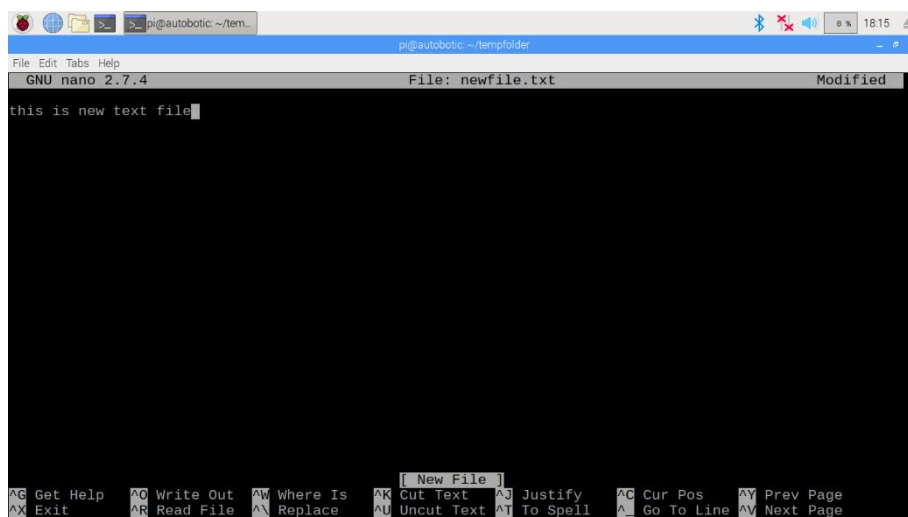
2.2.2 EDITING (NANO)

nano is a small, free and friendly editor – ready use in terminal.



```
pi@autobotic: ~ $ ls
2019-06-28-172813_1366x768_scrot.png Desktop Public
2019-06-28-175231_1366x768_scrot.png Documents Scratch
2019-06-28-180151_1366x768_scrot.png Downloads scratchthumbs.db
2019-06-28-180210_1366x768_scrot.png MagPi Scripts
2019-06-28-180253_1366x768_scrot.png Music Templates
2019-06-28-180714_1366x768_scrot.png Pictures Videos
2019-06-28-181304_1366x768_scrot.png project.sb2
pi@autobotic:~ $ mkdir tempfolder
pi@autobotic:~ $ ls
2019-06-28-172813_1366x768_scrot.png Desktop Public
2019-06-28-175231_1366x768_scrot.png Documents Scratch
2019-06-28-180151_1366x768_scrot.png Downloads scratchthumbs.db
2019-06-28-180210_1366x768_scrot.png MagPi Scripts
2019-06-28-180253_1366x768_scrot.png Music tempfolder
2019-06-28-180714_1366x768_scrot.png Pictures Templates
2019-06-28-181304_1366x768_scrot.png project.sb2 Videos
pi@autobotic:~ $ cd tempfolder/
pi@autobotic:~/tempfolder $ nano newfile.txt
pi@autobotic:~/tempfolder $ ls
newfile.txt
pi@autobotic:~/tempfolder $
```

(a) Create new file in tempfolder – **cd tempfolder > nano newfile.txt**



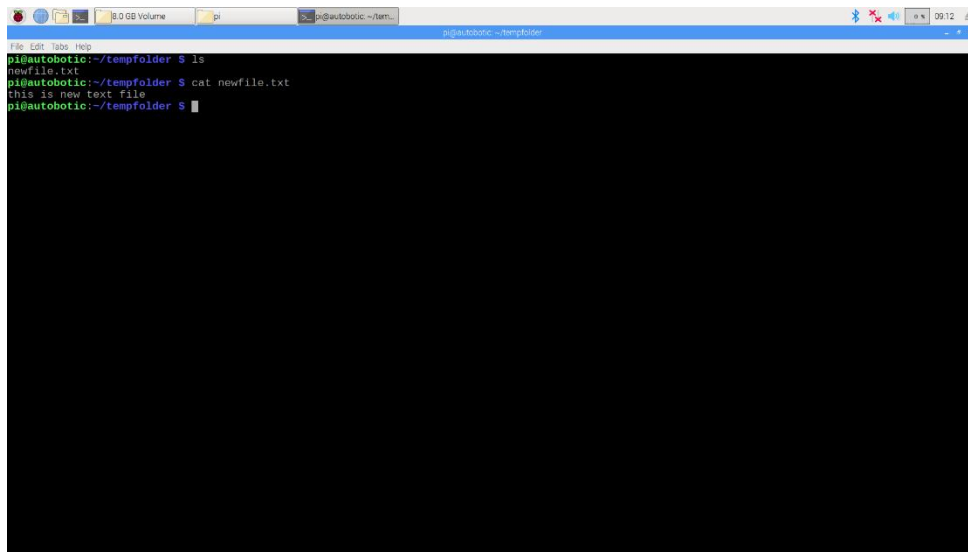
```
GNU nano 2.7.4 File: newfile.txt Modified
this is new text file
[New File]
Get Help Write Out Where Is Cut Text Justify Cur Pos Prev Page
Exit Read File Replace Uncut Text To Spell Go To Line Next Page
```

(b) Edit file in nano editor; once done – **ctrl+x > Enter > y**

Figure 11: Add and edit new file using **nano**

2.2.3 CONCATENATE (**CAT**)

cat [option] [file ...] – concatenate and print files

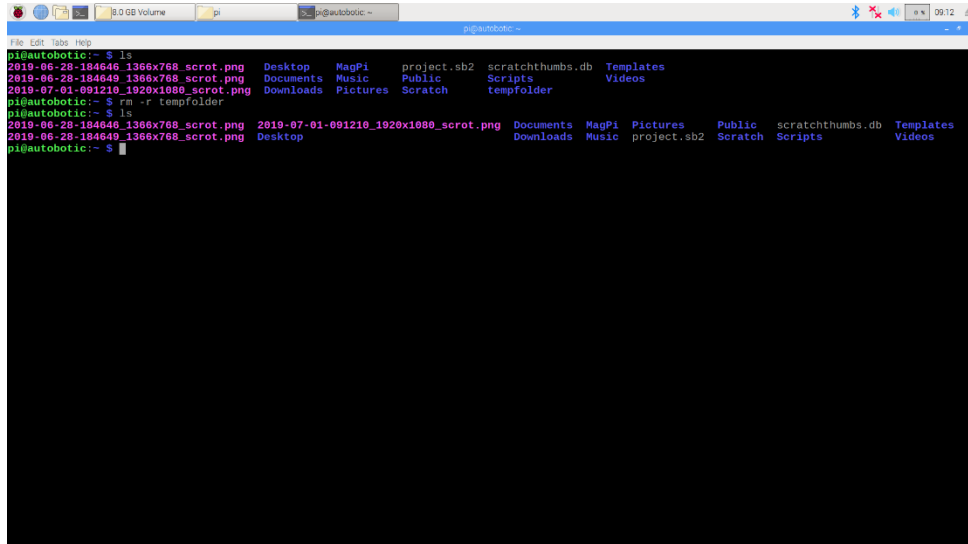


```
pi@autobotic: ~/tempfolder
$ ls
newfile.txt
pi@autobotic:~/tempfolder $ cat newfile.txt
this is new text file
pi@autobotic:~/tempfolder $
```

Figure 12: Review what's inside newfile.txt using **cat**

2.2.4 REMOVE FOLDER/FILES (**RM**)

rm [option] [file ...] – remove directory entries



```
pi@autobotic:~$ ls
2019-06-28-184646_1386x768_screenshot.png  Desktop  MagPi  project.sb2  scratchthumbs.db  Templates
2019-06-28-184649_1386x768_screenshot.png  Documents Music  Public  Scripts  Videos
2019-07-01-091210_1920x1080_screenshot.png Downloads Pictures Scratch tempfolder
pi@autobotic:~$ rm -r tempfolder
pi@autobotic:~$ ls
2019-06-28-184646_1386x768_screenshot.png  2019-07-01-091210_1920x1080_screenshot.png  Documents  MagPi  Pictures  Public  scratchthumbs.db  Templates
2019-06-28-184649_1386x768_screenshot.png  Desktop  Downloads  Music  project.sb2  Scratch  Scripts  Videos
pi@autobotic:~$
```

Figure 13: removing **tempfolder** using **rm -r tempfolder**

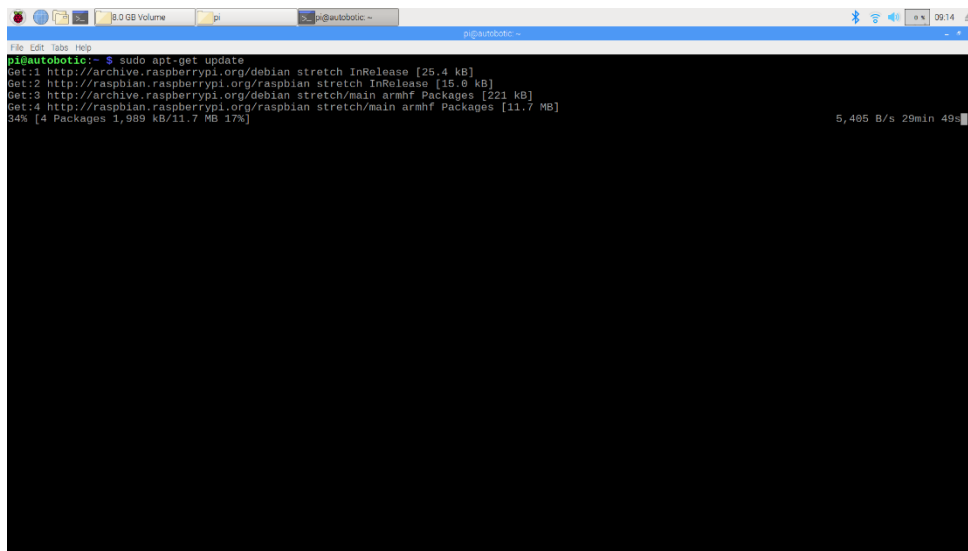
2.3 PERMISSION TO INSTALL

2.3.1 INSTALL

sudo apt-get update – updating Raspberry Pi software etc

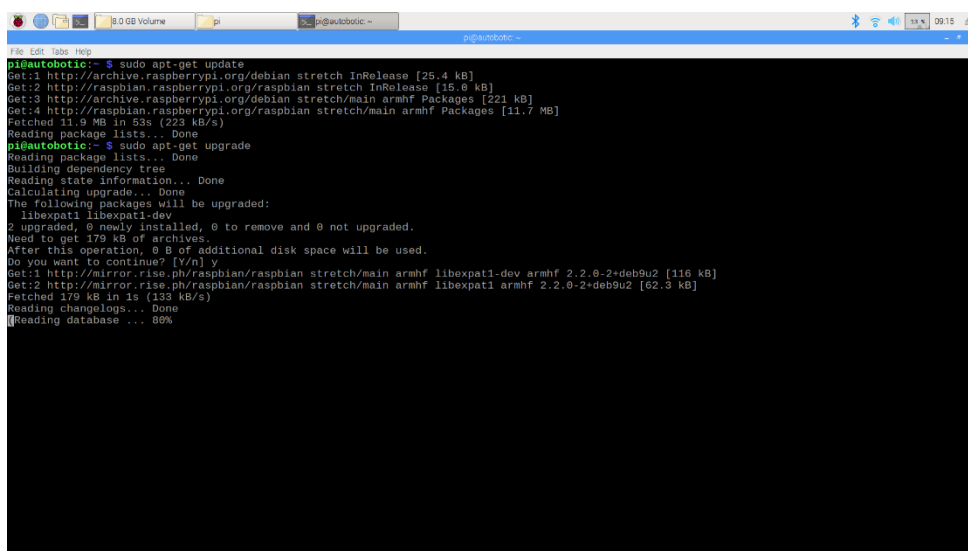
sudo apt-get upgrade – upgrading Raspberry Pi system

sudo apt-get install [file ...] – install specific software



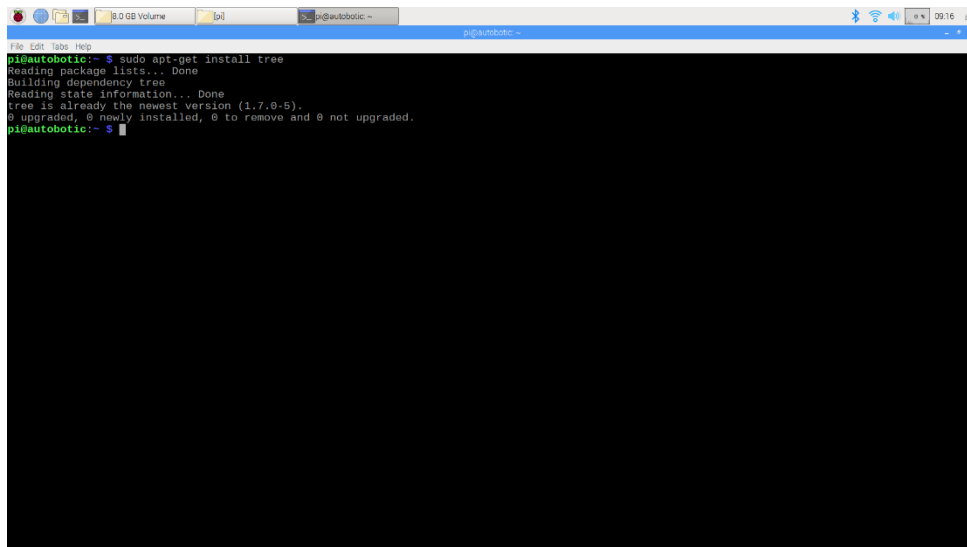
```
pi@autobotic:~$ sudo apt-get update
Get:1 http://archive.raspberrypi.org/debian stretch InRelease [25.4 kB]
Get:2 http://raspbrian.raspberrypi.org/raspbian stretch InRelease [15.0 kB]
Get:3 http://archive.raspberrypi.org/debian stretch/main armhf Packages [221 kB]
Get:4 http://raspbrian.raspberrypi.org/raspbian stretch/main armhf Packages [11.7 MB]
34% [4 Packages 1,989 kB/11.7 MB 17%]
5,485 B/s 29min 49s
```

(a) Updating apps and dependencies using **sudo apt-get update**



```
pi@autobotic:~$ sudo apt-get upgrade
Reading package lists... Done
Reading package lists... Done
Building dependency tree
Reading state information... Done
Calculating upgrade... Done
The following packages will be upgraded:
  libxpat1 libxpat1-dev
2 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
Need to get 179 kB of archives.
After this operation, 0 B of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://mirror.rise.ph/raspbian/raspbian stretch/main armhf libxpat1-dev armhf 2.2.0-2+deb9u2 [116 kB]
Get:2 http://mirror.rise.ph/raspbian/raspbian stretch/main armhf libxpat1 armhf 2.2.0-2+deb9u2 [62.3 kB]
Fetched 179 kB in 1s (133 kB/s)
Reading changelogs... Done
Reading database ... 88%
```

(b) Updating apps and dependencies using **sudo apt-get upgrade**



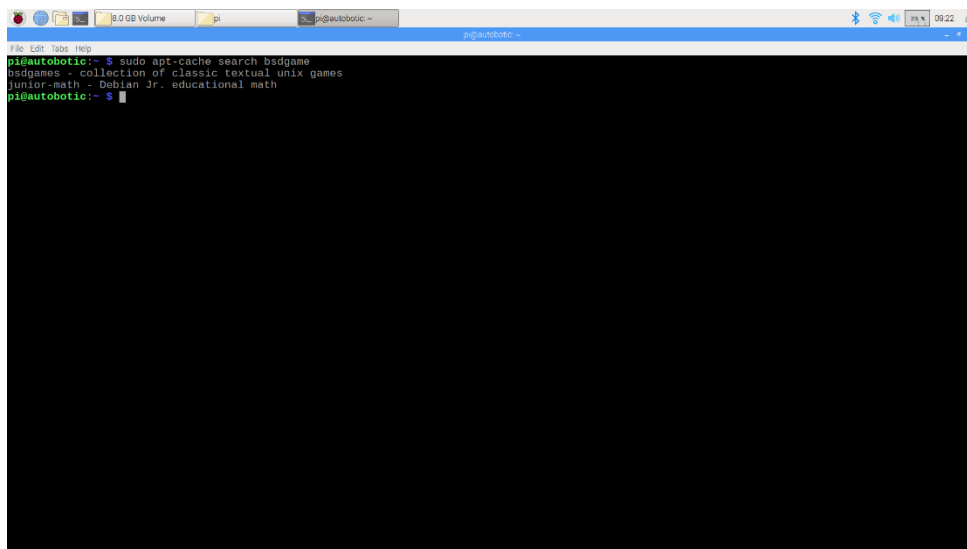
```
pi@autobotic:~$ sudo apt-get install tree
Reading package lists... Done
Building dependency tree
Reading state information... Done
tree is already the newest version (1.7.0-5).
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
pi@autobotic:~$
```

(c) Add and edit new file using **nano**

Figure 15: Updating apps, dependencies, and install an apps

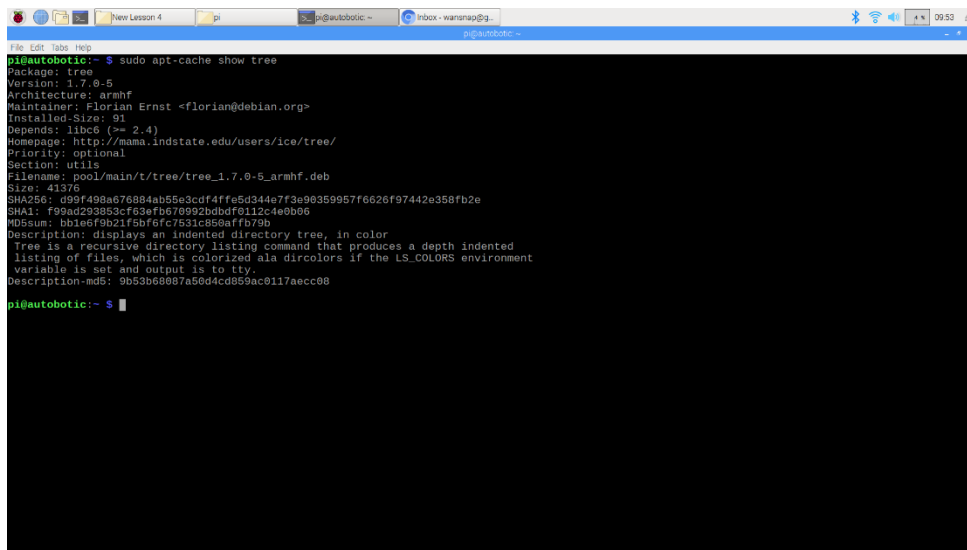
2.3.2 SEARCH

sudo apt-cache search [file ...] – find particular apps



```
pi@autobotic:~$ sudo apt-cache search bsdgame
bsdgames - collection of classic textual unix games
junior-math - Debian Jr. educational math
pi@autobotic:~$
```

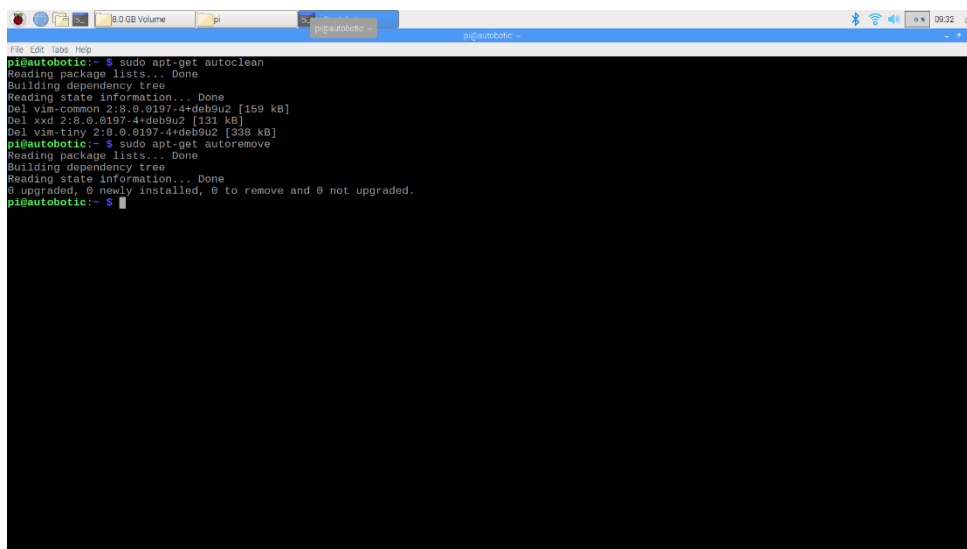
Figure 16: Using **apt-cache search** to find an apps or games etc

sudo apt-cache show [file ...] – examine individual packages


```

pi@autobotic:~$ sudo apt-cache show tree
Package: tree
Version: 1.7.0-5
Architecture: armhf
Maintainer: Florian Ernst <florian@debian.org>
Installed-Size: 91
Depends: libc6 (>= 2.4)
Homepage: http://mama.indstate.edu/users/ice/tree/
Priority: optional
Section: utils
Filename: pool/main/t/tree/tree_1.7.0-5_armhf.deb
Size: 41376
SHA256: d99f498a676884ab55e3cdf4ffe5d344e7f3e90359957f6626f97442e358fb2e
SHA1: f99ad293853cf63efb670992b0bdf0112c4e0b06
MD5sum: bb1e6f9b21f5bf6fc7531c859affb79b
Description: displays an indented directory tree, in color
 tree is a recursive directory listing command that produces a depth indented
 listing of files, which is colorized ala dircolors if the LS_COLORS environment
 variable is set and output is to tty.
Description-md5: 9b53b6007a50d4cd059ac0117a6cc08
pi@autobotic:~$

```

Figure 17: Using **apt-cache show** to find an specific apps or games etc**2.3.3 CLEAR / REMOVE****sudo apt-get autoclean** – auto clean the removed apps and the dependencies**sudo apt-get autoremove** – auto remove apps and the dependencies obsolete


```

pi@autobotic:~$ sudo apt-get autoclean
Reading package lists... Done
Building dependency tree
Reading state information... Done
Del vim-common 2:8.0.0197-4-deb9u2 [159 kB]
Del xxd 2:8.0.0197-4-deb9u2 [131 kB]
Del vim-tiny 2:8.0.0197-4-deb9u2 [338 kB]
pi@autobotic:~$ sudo apt-get autoremove
Reading package lists... Done
Building dependency tree
Reading state information... Done
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
pi@autobotic:~$

```

Figure 18: Auto-clean and remove – to remove obsoletes or apps and dependencies

3 MISCELLANEOUS

I have listed here some of tips and tricks for finding help, tab completion, wildcard*, etc. in strengthening using the command line.

3.1 FIND HELP

man [command] – manual for command line syntax (simple)

info [command] – manual for command line syntax (complete)

If we need to find more info on certain command line syntax, **man** (manual) and **info** is there for help.

```

File Edit Tabs Help
LS(1) User Commands LS(1)
NAME
ls - list directory contents
SYNOPSIS
ls [OPTION]... [FILE]...
DESCRIPTION
list information about the FILES (the current directory by default). Sort entries alphabetically if none of -cftuvSUX nor --sort is
specified.
Mandatory arguments to long options are mandatory for short options too.
-a, --all
do not ignore entries starting with .
-A, --almost-all
do not list implied . and ..
--author
with -l, print the author of each file
-b, --escape
print C-style escapes for nongraphic characters
--block-size=SIZE
scale sizes by SIZE before printing them; e.g., '--block-size=M' prints sizes in units of 1,048,576 bytes; see SIZE format below
-B, --ignore-backups
do not list implied entries ending with ~
-c
with -lt: sort by, and show, ctime (time of last modification of file status information); with -l: show ctime and sort by name;
otherwise: sort by ctime, newest first
-C
list entries by columns
--color[=WHEN]
colorize the output; WHEN can be 'always' (default if omitted), 'auto', or 'never'; more info below
-d, --directory
Manual page ls(1) line 1 (press h for help or q to quit)

```

(a) Using **man** (manual)

```

File Edit Tabs Help
LS(1) User Commands LS(1)
NAME
ls - list directory contents
SYNOPSIS
ls [OPTION]... [FILE]...
DESCRIPTION
list information about the FILES (the current directory by default). Sort entries alphabetically if none of -cftuvSUX nor --sort is
specified.
Mandatory arguments to long options are mandatory for short options too.
-a, --all
do not ignore entries starting with .
-A, --almost-all
do not list implied . and ..
--author
with -l, print the author of each file
-b, --escape
print C-style escapes for nongraphic characters
--block-size=SIZE
scale sizes by SIZE before printing them; e.g., '--block-size=M' prints sizes in units of 1,048,576 bytes; see SIZE format below
-B, --ignore-backups
do not list implied entries ending with ~
-c
with -lt: sort by, and show, ctime (time of last modification of file status information); with -l: show ctime and sort by name;
otherwise: sort by ctime, newest first
-C
list entries by columns
--color[=WHEN]
colorize the output; WHEN can be 'always' (default if omitted), 'auto', or 'never'; more info below
-d, --directory
Manual page ls(1) line 1 (press h for help or q to quit)

```

(b) Using **info**

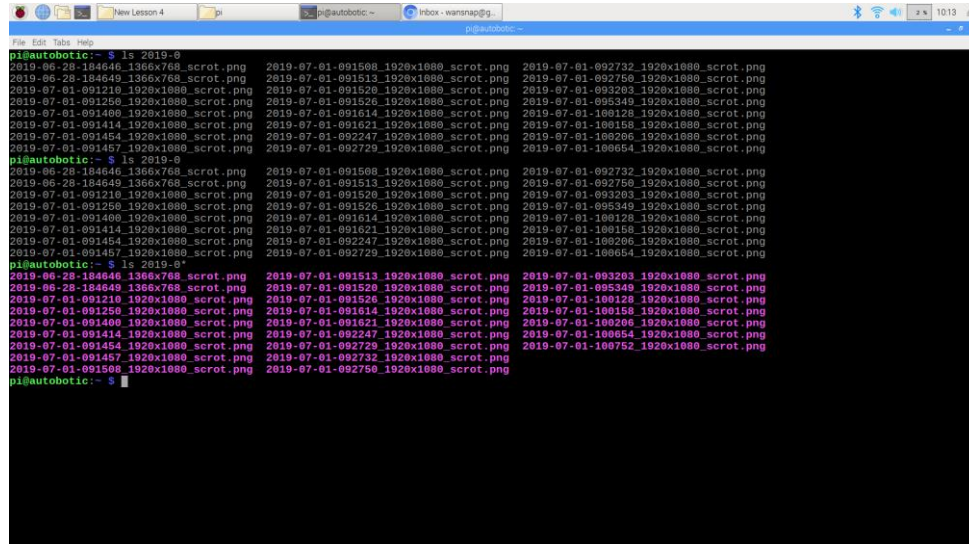
Figure 19: Finding help

3.2 TAB COMPLETION

This is a lazy way for typing long text name for folders. If you've more than one file beginning with p, they'll all be listed, and you can type more letters and hit Tab again.

3.3 WILDCARD

"*" or **asterisk** is known as a wildcard, and this one represents any number of characters (including none). Powerful for listing up all the files or folders has identically same as

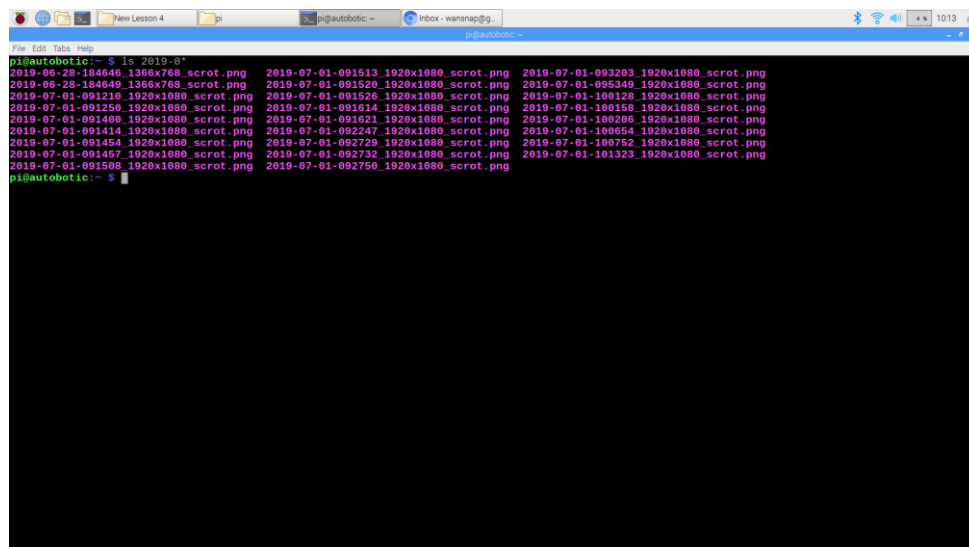


```

pi@autobotic:~$ ls 2019-0
2019-06-28-184646-136x768_scrut.png  2019-07-01-091508-1920x1080_scrut.png  2019-07-01-092732-1920x1080_scrut.png
2019-06-28-184649-136x768_scrut.png  2019-07-01-091513-1920x1080_scrut.png  2019-07-01-092750-1920x1080_scrut.png
2019-07-01-091210-1920x1080_scrut.png  2019-07-01-091520-1920x1080_scrut.png  2019-07-01-093203-1920x1080_scrut.png
2019-07-01-091250-1920x1080_scrut.png  2019-07-01-091526-1920x1080_scrut.png  2019-07-01-095349-1920x1080_scrut.png
2019-07-01-091400-1920x1080_scrut.png  2019-07-01-091614-1920x1080_scrut.png  2019-07-01-100128-1920x1080_scrut.png
2019-07-01-091414-1920x1080_scrut.png  2019-07-01-091621-1920x1080_scrut.png  2019-07-01-100158-1920x1080_scrut.png
2019-07-01-091454-1920x1080_scrut.png  2019-07-01-092247-1920x1080_scrut.png  2019-07-01-100206-1920x1080_scrut.png
2019-07-01-091457-1920x1080_scrut.png  2019-07-01-092729-1920x1080_scrut.png  2019-07-01-100654-1920x1080_scrut.png
pi@autobotic:~$ ls 2019-0*
2019-06-28-184646-136x768_scrut.png  2019-07-01-091508-1920x1080_scrut.png  2019-07-01-092732-1920x1080_scrut.png
2019-06-28-184649-136x768_scrut.png  2019-07-01-091513-1920x1080_scrut.png  2019-07-01-092750-1920x1080_scrut.png
2019-07-01-091210-1920x1080_scrut.png  2019-07-01-091520-1920x1080_scrut.png  2019-07-01-093203-1920x1080_scrut.png
2019-07-01-091250-1920x1080_scrut.png  2019-07-01-091526-1920x1080_scrut.png  2019-07-01-095349-1920x1080_scrut.png
2019-07-01-091400-1920x1080_scrut.png  2019-07-01-091614-1920x1080_scrut.png  2019-07-01-100128-1920x1080_scrut.png
2019-07-01-091414-1920x1080_scrut.png  2019-07-01-091621-1920x1080_scrut.png  2019-07-01-100158-1920x1080_scrut.png
2019-07-01-091454-1920x1080_scrut.png  2019-07-01-092247-1920x1080_scrut.png  2019-07-01-100206-1920x1080_scrut.png
2019-07-01-091457-1920x1080_scrut.png  2019-07-01-092729-1920x1080_scrut.png  2019-07-01-100654-1920x1080_scrut.png
pi@autobotic:~$

```

(a)



```

pi@autobotic:~$ ls 2019-0*
2019-06-28-184646-136x768_scrut.png  2019-07-01-091513-1920x1080_scrut.png  2019-07-01-093203-1920x1080_scrut.png
2019-06-28-184649-136x768_scrut.png  2019-07-01-091520-1920x1080_scrut.png  2019-07-01-095349-1920x1080_scrut.png
2019-07-01-091210-1920x1080_scrut.png  2019-07-01-091526-1920x1080_scrut.png  2019-07-01-100128-1920x1080_scrut.png
2019-07-01-091250-1920x1080_scrut.png  2019-07-01-091614-1920x1080_scrut.png  2019-07-01-100158-1920x1080_scrut.png
2019-07-01-091400-1920x1080_scrut.png  2019-07-01-091621-1920x1080_scrut.png  2019-07-01-100206-1920x1080_scrut.png
2019-07-01-091414-1920x1080_scrut.png  2019-07-01-092247-1920x1080_scrut.png  2019-07-01-100654-1920x1080_scrut.png
2019-07-01-091454-1920x1080_scrut.png  2019-07-01-092729-1920x1080_scrut.png  2019-07-01-100752-1920x1080_scrut.png
2019-07-01-091457-1920x1080_scrut.png  2019-07-01-092732-1920x1080_scrut.png  2019-07-01-101323-1920x1080_scrut.png
2019-07-01-091508-1920x1080_scrut.png  2019-07-01-092750-1920x1080_scrut.png
pi@autobotic:~$

```

(b)

Figure 20: List (**ls**) using the wildcard