

# Intro to Robotics: Introducing Ubuntu Commands

D158 Unix lab

Logon (see username and password instructions on screen) and practice Linux (Ubuntu)

**bash** GNU Bourne-Again Shell

## TO START

- Simultaneously hitting the buttons Ctrl-Alt-T will bring up a terminal window. The Unity Launcher --> Terminal will also bring up a terminal window. If screen text is too small, use menu options View → Zoom In to make larger.
- The tilde (~) symbol stands for your home directory. If you are *user*, then the tilde (~) stands for */home/user*

## COMMAND HELP

**man** - use to view manual pages for executable program, utility or function **help**

- displays a list of shell commands

<command> **-h** or **--help** to display help for a built-in command

For any command, view the options. Pipe (|) “less” to display one screen at a time and you can scroll up and down on the screen.

**\$ man ls | less**

## FILE AND DIRECTORY MANAGEMENT

**ls** – List all of the contents (names of files and directories) in a specified directory. If no directory is specified, it will use the current directory.

**\$ ls -a, or --all** Show all entries including entries starting with . (hidden files) **.bashrc \$**

**ls -l** Show long listing format with:

- file type indicator
- read/write and execute permissions for Owner/Group/Others
- owner of the file and group the file belongs
- size and date of modification/creation

example:

```
-rwxrwx r-x 1tlharmanphd tlharmanphd 1956 Sep 28 2015 python_goforward.py
```

**cd** –Changes the current working directory in the command line console. **pwd**–Displays the current working directory for the command line terminal. Good for when you’ve lost track of where you are in your system. **mkdir / rmdir**–Creates a directory (*mkdir*) or deletes a specified directory (*rmdir*). Directories can only be created and deleted within directories that you have permission to access.

```
$ mkdir ubuntuIntro
$ cd ubuntuIntro/      ( / is optional)
<user>:~/ubuntuIntro$ (You have created a directory ubuntuIntro.)
```

```
$ cd ..      Go up one level
$ cd ~      Go to user's home directory
$ cd ../../  Go up 2 levels
$ cd /      Go to root
                (Notice the differences in ~ versus / prompt)
```

At this point, return to your home directory `$ cd ~`

## CREATE A TEXT FILE

**gedit, nano** and **emacs** are text editors in Ubuntu. **cat**

- Concatenate and displays the content of files

```
$ gedit NewFile.txt      (Type your text in the gedit window,
                        If no line numbers, use Edit → Preferences then check “Display line numbers”
                        When finished, choose Save icon of gedit.)
$ cat NewFile.txt      (Display the contents of your new file.)
```

## COPY A FILE

**cp** – Copy one or more files to another location. **rm**

– Removes one or more files.

```
$ cp NewFile.txt NewerFile.txt (Copy file to a new location with a new name.)
                                (Use the ls command to check that the command worked.) $
rm NewFile.txt                  (Delete the file.)
```

## MAKE A FILE EXECUTABLE

**chmod** – Changes the access permissions of one or more files. Only users with permission or ownership of a file can change that file’s permissions.

- **chmod 777 filename** gives owner, group and others read, write and execute privileges

- **chmod 755 filename** gives owner read, write, execute privileges; group and others get read and execute privileges

\$ gedit NewProgram.py (Create a simple program and then save.)

\$ ls -la

```
drwxrwxr-x 2tlharmanphdtlharmanphd 4096 Aug 12 16:07 . drwxr-xr-x
76 tlharmanphdtlharmanphd 4096 Aug 12 15:45 .. -rw-rw-r--
1tlharmanphdtlharmanphd 73 Aug 12 16:00 NewerFile.txt
-rw-rw-r-- 1tlharmanphdtlharmanphd 69 Aug 12 16:07 NewProgram.py
```

**(Note that NewProgram.py is not executable – no x)**

\$ chmod +x NewProgram.py

\$ ls -l

```
-rw-rw-r-- 1 tlharmanphdtlharmanphd 73 Aug 12 16:00 NewerFile.txt
-rwxrwxr-x 1 tlharmanphdtlharmanphd 69 Aug 12 16:07 NewProgram.py
```

**(The NewProgram.py is now executable according to the file permissions.)**

## MOVE AND RENAME FILE

**mv** – Moves file to another location or renames a file.

~\$ mv NewerFile.txt ubuntuIntro/NewerFile.txt

~\$ cd ubuntuIntro/

~/ubuntuIntro\$ ls NewerFile.txt

## COMPARE FILES

**diff** – Displays the differences between two files.

~/ubuntuIntro\$ cp ~/NewProgram.py .

~/ubuntuIntro\$ diff NewerFile.txt NewProgram.py

```
1,3c1,4
< Hello Ubuntu file
< This is a test of creating a text file NewFile.txt
< Bye
---
> # gedit of NewProgram.py
> This is a different line
> Another different line
>
```

## **GREP – One of the most powerful Commands**

\$ man grep (671 Lines)

<https://www.gnu.org/software/grep/manual/>

**grep** – Searches file(s) for lines that match a given text string  
options are: - r or - R for recursive

- n for line number

- w to match whole word

Find a file with a specified line of text.

```
harman@harman-VirtualBox:~$ cd Desktop
```

```
harman@harman-VirtualBox:~/Desktop$ ls | grep New
```

```
NewText1.txt
```

```
NewText2.txt
```

```
harman@harman-VirtualBox:~/Desktop$ cat NewText1.txt
```

```
NewText1 Demo
```

```
This is a test of text files
```

```
harman@harman-VirtualBox:~/Desktop$ grep -R "This is a test"
```

```
Commands3 9_12_2022: This is a test of text files
```

```
Commands3 9_12_2022: < This is a test of text files
```

```
NewText1.txt:This is a test of text filesNewerFile.txt:This is a test of creating a  
text file NewFile.txt NewProgram.py:This is a different line
```

Find references to an environmental parameter

```
harman@harman-VirtualBox:~/Desktop$ env | grep ROS
```

```
ROS_VERSION=1
```

```
ROS_PYTHON_VERSION=3
```

```
ROS_PACKAGE_PATH=/opt/ros/noetic/share
```

```
ROSLISP_PACKAGE_DIRECTORIES=
```

```
ROS_DOMAIN_ID=231
```

```
ROS_ETC_DIR=/opt/ros/noetic/etc/ros
```

```
ROS_MASTER_URI=http://localhost:11311
```

```
ROS_LOCALHOST_ONLY=0
```

```
ROS_ROOT=/opt/ros/noetic/share/ros
```

```
ROS_DISTRO=noetic
```

## OTHER USEFUL COMMANDS

**history** – displays command history **clear** –  
clears the terminal screen **source** – runs  
commands from a specified file  
**.** – runs a specified command script in the current shell  
<tab><tab> for tab completion  
<up arrow> brings up the last commands

```
harman@harman-VirtualBox:~$ locate history*  
/home/harman/history1.txt  
/home/harman/history2.txt  
(Created by history > <name.txt>)
```