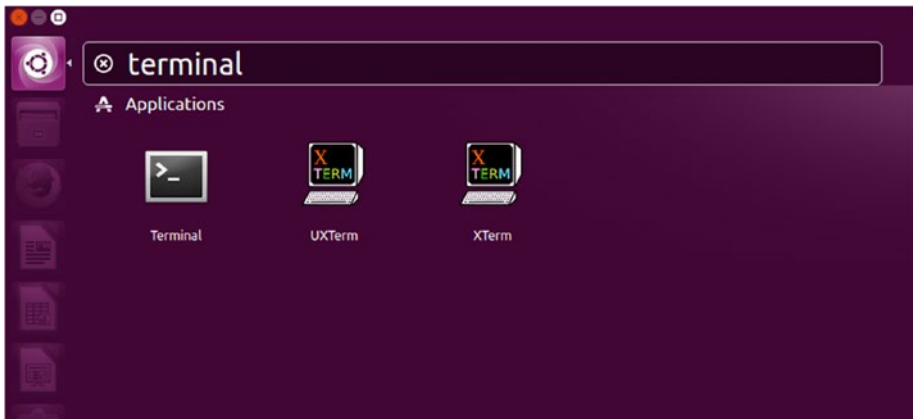


## Getting Started with Shell Commands

The graphical tools in Ubuntu are very easy to use, but if you want to perform advanced tasks in Linux, you may need to learn the Ubuntu command-line interface (CLI). The command-line tools are faster and used often in debugging the system. The command-line interface in Linux can be compared to the disk operating system (DOS) in Windows.

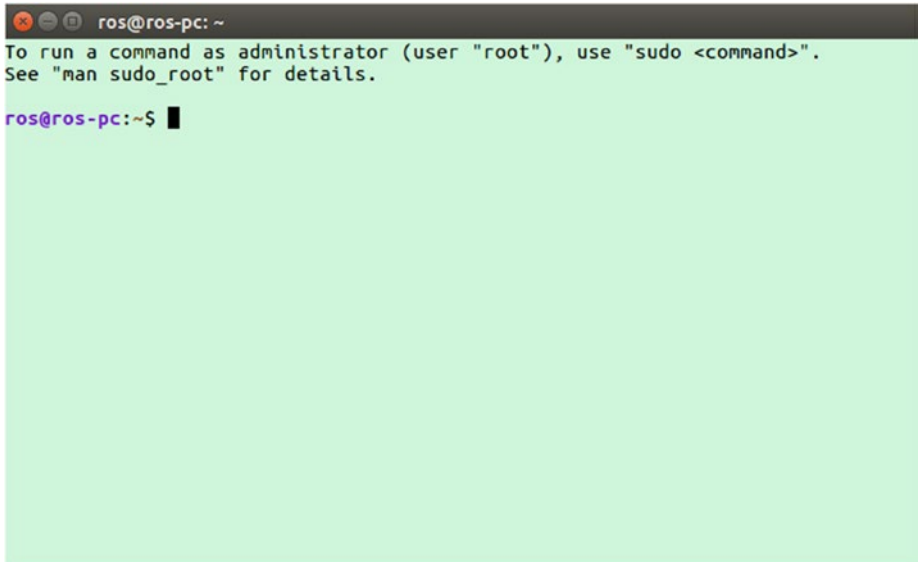
We mainly use the command line when we work with ROS. Knowledge of the Linux terminal commands is a prerequisite for working with ROS.

The Ubuntu command-line interface is in a tool called Terminal. Use the Ubuntu Dash search to find the Terminal application. Figure 1-31 shows an example.



**Figure 1-31.** Searching for the Terminal application

Click Terminal to open the application, which is shown in Figure 1-32.



**Figure 1-32.** *The Ubuntu terminal*

## Terminal Commands Cheat Sheet

This section covers useful shell commands for working with robots and ROS. The following are the popular commands that you want to explore.

### man: Manual Pages for Shell Commands

The `man` command stands for *manual*. This command provides the manual page of a given command.

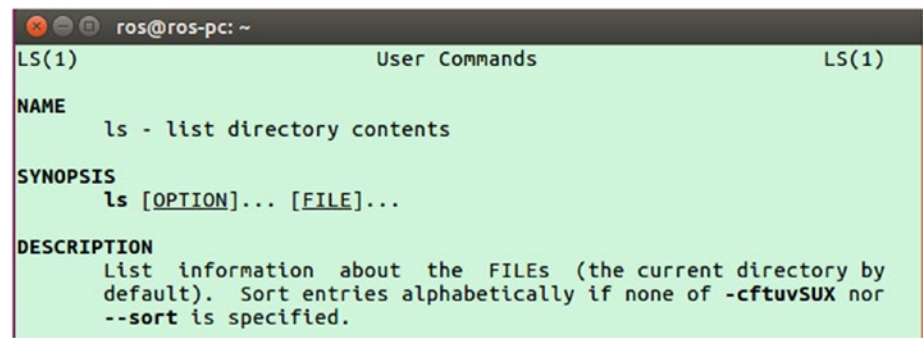
---

Usage: `man <shell command>`

Example: `man ls`

---

The preceding asks for the manual page of `ls`. Figure 1-33 shows the output of `man ls`.



*Figure 1-33. The manual page of `ls`*

## ls: List Directory Content

The `ls` command lists the content of files and folders in the current directory.

---

Usage: `ls`

---

The output of `ls` is shown in Figure 1-34.



*Figure 1-34. List of files in the current path*

## cd: Change Directory

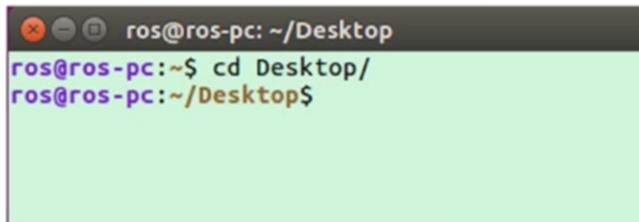
The `cd` command switches from one folder to another (see Figure 1-35).

---

Usage: `cd <Directory_path>`

Example: `cd Desktop`

---

A terminal window with a dark title bar showing 'ros@ros-pc: ~/Desktop'. The prompt is 'ros@ros-pc:~\$'. The user enters 'cd Desktop/' and the prompt changes to 'ros@ros-pc:~/Desktop\$'.

**Figure 1-35.** *Changing folders*

## pwd: Current Terminal Path

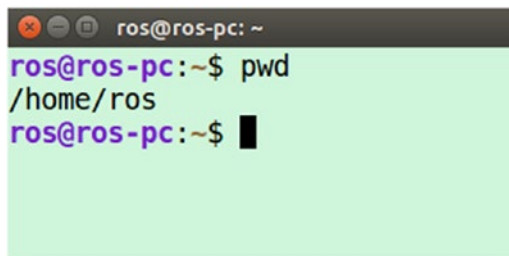
The `pwd` command returns the current path of the terminal. This is useful for getting the absolute path.

---

Usage: `pwd`

---

Figure 1-36 shows the output of the `pwd` command.

A terminal window with a dark title bar showing 'ros@ros-pc: ~'. The prompt is 'ros@ros-pc:~\$'. The user enters 'pwd' and the output is '/home/ros'. The prompt returns to 'ros@ros-pc:~\$'.

**Figure 1-36.** *Command to get current path*

## mkdir: Create a Folder

The `mkdir` command creates an empty folder or directory.

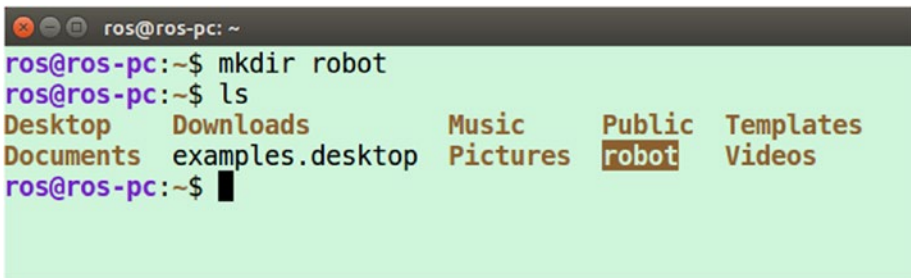
---

Usage: `mkdir <folder_name>`

Example: `mkdir robot`

---

Figure 1-37 shows how to create and list folders.



```
ros@ros-pc: ~  
ros@ros-pc:~$ mkdir robot  
ros@ros-pc:~$ ls  
Desktop    Downloads    Music        Public    Templates  
Documents  examples.desktop  Pictures    robot    Videos  
ros@ros-pc:~$
```

**Figure 1-37.** *Creating a new folder*

## rm: Delete a File

The `rm` command deletes a file.

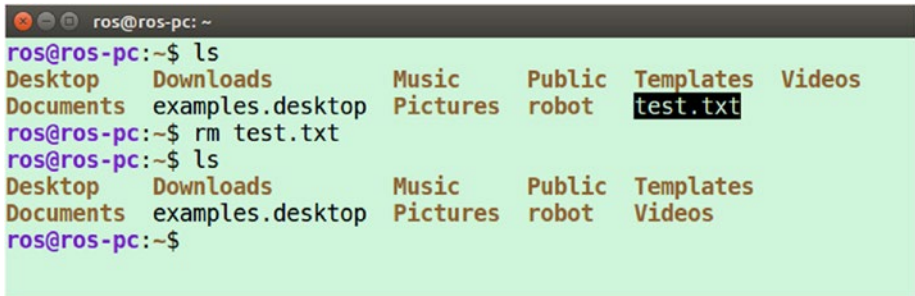
---

Usage: `rm <file_path>`

Example: `rm test.txt`

---

An example is shown Figure 1-38. The files are listed before deletion and after deletion to confirm that the files were actually deleted.



```

ros@ros-pc: ~
ros@ros-pc:~$ ls
Desktop  Downloads      Music    Public  Templates  Videos
Documents examples.desktop Pictures  robot   test.txt
ros@ros-pc:~$ rm test.txt
ros@ros-pc:~$ ls
Desktop  Downloads      Music    Public  Templates
Documents examples.desktop Pictures  robot   Videos
ros@ros-pc:~$

```

**Figure 1-38.** *Deleting a file*

To delete a folder by recursively deleting its files, use the following command.

```
$ rm -r <folder_name>
```

To delete a file inside the root (/) file system, use `sudo` before the `rm` command.

```
$ sudo rm <file_name>
```

## rmmdir: Delete a Folder

The `rmmdir` command deletes an empty folder. You may need to delete files before using this command.

---

Usage: `rmmdir <folder_name>`

Example: `rmmdir robot`

---

Figure 1-39 shows an example of this command.

```
ros@ros-pc: ~  
ros@ros-pc:~$ ls  
Desktop      Downloads      Music      Public      Templates  
Documents    examples.desktop Pictures     robot       Videos  
ros@ros-pc:~$ rmdir robot  
ros@ros-pc:~$ ls  
Desktop      Downloads      Music      Public      Videos  
Documents    examples.desktop Pictures     Templates  
ros@ros-pc:~$
```

*Figure 1-39. Deleting an empty folder*

## mv: Move a File from One Place to Another

The mv command moves a file from one location to another and then renames the file.

---

Usage: mv source\_file destination/destination\_file

Example: mv test.txt test\_2.txt

---

In Figure 1-40, test.txt is moved into the same folder under a different name (i.e., test\_2.txt).

It is moving the file by renaming the file.

```
ros@ros-pc: ~  
ros@ros-pc:~$ ls  
Desktop      Downloads      Music      Public      test.txt  
Documents    examples.desktop Pictures     Templates  Videos  
ros@ros-pc:~$ mv test.txt test_2.txt  
ros@ros-pc:~$ ls  
Desktop      Downloads      Music      Public      test_2.txt  
Documents    examples.desktop Pictures     Templates  Videos  
ros@ros-pc:~$ █
```

*Figure 1-40. Moving a file*

## cp: Copy a File from One Path to Another

The `cp` command copies files from one location to another.

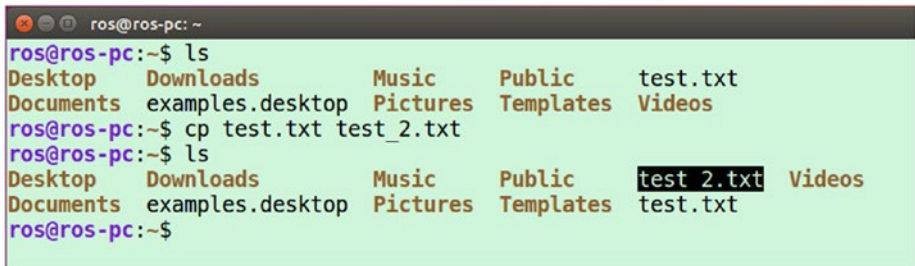
---

Usage: `cp source_file destination_folder/destination_file`

Example: `cp test.txt test_2.txt`

---

Figure 1-41 demonstrates this example.



```
ros@ros-pc: ~  
ros@ros-pc:~$ ls  
Desktop    Downloads      Music          Public         test.txt  
Documents  examples.desktop Pictures        Templates      Videos  
ros@ros-pc:~$ cp test.txt test_2.txt  
ros@ros-pc:~$ ls  
Desktop    Downloads      Music          Public         test_2.txt  Videos  
Documents  examples.desktop Pictures        Templates      test.txt  
ros@ros-pc:~$
```

**Figure 1-41.** Copying a file

## dmesg: Display a Kernel Message

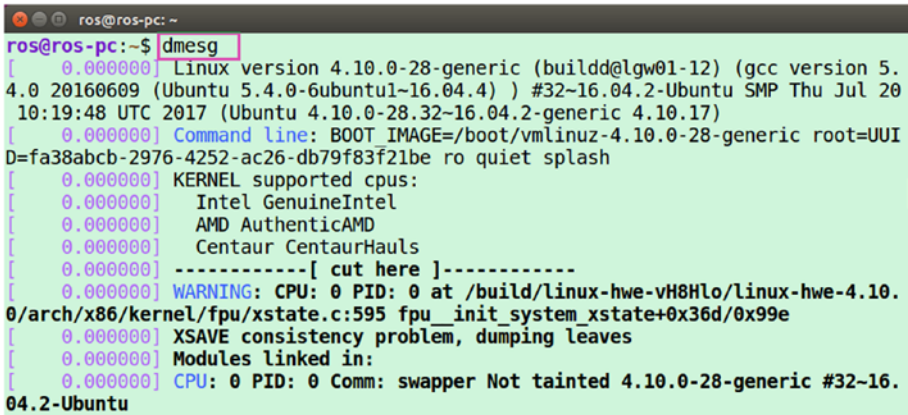
The `dmesg` command is very useful for debugging the system. It displays the kernel logs (see Figure 1-42). From these logs, you can debug the problem.

---

Usage: `dmesg`

---





```

ros@ros-pc: ~$ dmesg
[ 0.000000] Linux version 4.10.0-28-generic (build@lgw01-12) (gcc version 5.
4.0 20160609 (Ubuntu 5.4.0-6ubuntu1~16.04.4) ) #32~16.04.2-Ubuntu SMP Thu Jul 20
10:19:48 UTC 2017 (Ubuntu 4.10.0-28.32~16.04.2-generic 4.10.17)
[ 0.000000] Command line: BOOT_IMAGE=/boot/vmlinuz-4.10.0-28-generic root=UUI
D=fa38abcb-2976-4252-ac26-db79f83f21be ro quiet splash
[ 0.000000] KERNEL supported cpus:
[ 0.000000] Intel GenuineIntel
[ 0.000000] AMD AuthenticAMD
[ 0.000000] Centaur CentaurHauls
[ 0.000000] -----[ cut here ]-----
[ 0.000000] WARNING: CPU: 0 PID: 0 at /build/linux-hwe-vH8Hlo/linux-hwe-4.10.
0/arch/x86/kernel/fpu/xstate.c:595 fpu_init_system_xstate+0x36d/0x99e
[ 0.000000] XSAVE consistency problem, dumping leaves
[ 0.000000] Modules linked in:
[ 0.000000] CPU: 0 PID: 0 Comm: swapper Not tainted 4.10.0-28-generic #32~16.
04.2-Ubuntu

```

*Figure 1-42. Checking the kernel logs*

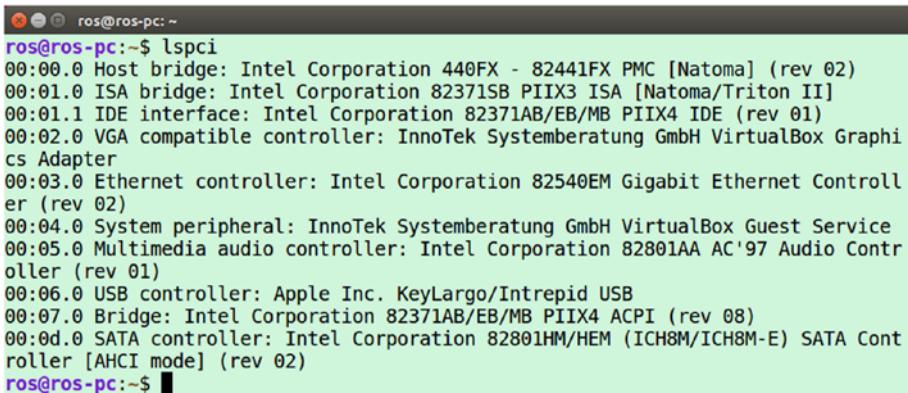
## lspci: List of PCI Devices in the System

The `lspci` command also debugs the PC. This command lists the PCI devices in the PC (see Figure 1-43).

---

Usage: `lspci`

---



```

ros@ros-pc: ~$ lspci
00:00.0 Host bridge: Intel Corporation 440FX - 82441FX PMC [Natoma] (rev 02)
00:01.0 ISA bridge: Intel Corporation 82371SB PIIX3 ISA [Natoma/Triton II]
00:01.1 IDE interface: Intel Corporation 82371AB/EB/MB PIIX4 IDE (rev 01)
00:02.0 VGA compatible controller: InnoTek Systemberatung GmbH VirtualBox Graphi
cs Adapter
00:03.0 Ethernet controller: Intel Corporation 82540EM Gigabit Ethernet Controll
er (rev 02)
00:04.0 System peripheral: InnoTek Systemberatung GmbH VirtualBox Guest Service
00:05.0 Multimedia audio controller: Intel Corporation 82801AA AC'97 Audio Contr
oller (rev 01)
00:06.0 USB controller: Apple Inc. KeyLargo/Intrepid USB
00:07.0 Bridge: Intel Corporation 82371AB/EB/MB PIIX4 ACPI (rev 08)
00:0d.0 SATA controller: Intel Corporation 82801HM/HEM (ICH8M/ICH8M-E) SATA Cont
roller [AHCI mode] (rev 02)
ros@ros-pc: ~$

```

*Figure 1-43. Listing the PCI devices*

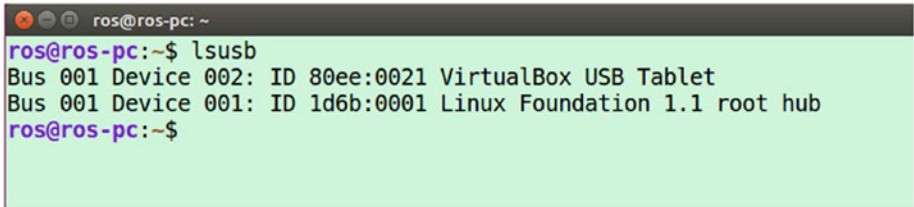
## lsusb: List of USB Devices in the System

The `lsusb` command lists all USB devices (see Figure 1-44).

---

Usage: `lsusb`

---

A terminal window with a dark title bar showing 'ros@ros-pc: ~'. The prompt is 'ros@ros-pc:~\$' and the command 'lsusb' has been entered. The output shows two USB devices: 'Bus 001 Device 002: ID 80ee:0021 VirtualBox USB Tablet' and 'Bus 001 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub'. The prompt is now 'ros@ros-pc:~\$'.

```
ros@ros-pc:~$ lsusb
Bus 001 Device 002: ID 80ee:0021 VirtualBox USB Tablet
Bus 001 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
ros@ros-pc:~$
```

**Figure 1-44.** *Listing the USB devices*

## sudo: Run a Command in Administrative Mode

The `sudo` command is one of the most important. We use it regularly. It runs a command with administrative privileges (see Figure 1-45). We can also completely switch to root (administrator) mode using this command.

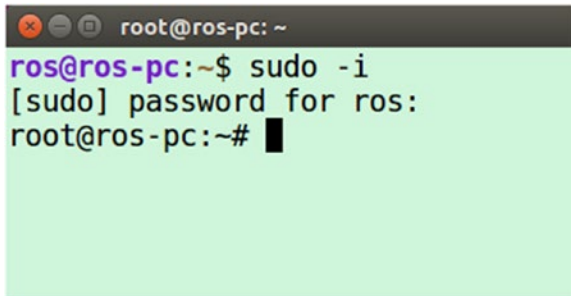
---

Usage: `sudo <parameter> <command>`

Example: `sudo -i`

---

This example command switches to root mode.



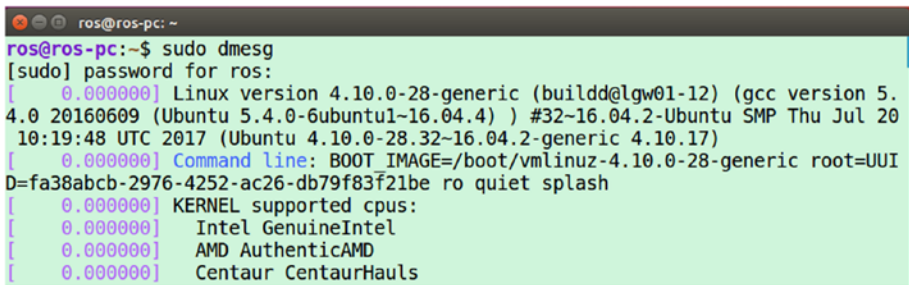
```

root@ros-pc: ~
ros@ros-pc:~$ sudo -i
[sudo] password for ros:
root@ros-pc:~#

```

**Figure 1-45.** *Switching to administrator mode*

Figure 1-46 shows the results of executing a command in root mode.



```

ros@ros-pc: ~
ros@ros-pc:~$ sudo dmesg
[sudo] password for ros:
[ 0.000000] Linux version 4.10.0-28-generic (build@lgw01-12) (gcc version 5.
4.0 20160609 (Ubuntu 5.4.0-6ubuntu1~16.04.4) ) #32~16.04.2-Ubuntu SMP Thu Jul 20
10:19:48 UTC 2017 (Ubuntu 4.10.0-28.32~16.04.2-generic 4.10.17)
[ 0.000000] Command line: BOOT_IMAGE=/boot/vmlinuz-4.10.0-28-generic root=UUI
D=fa38abcb-2976-4252-ac26-db79f83f21be ro quiet splash
[ 0.000000] KERNEL supported cpus:
[ 0.000000] Intel GenuineIntel
[ 0.000000] AMD AuthenticAMD
[ 0.000000] Centaur CentaurHauls

```

**Figure 1-46.** *Running a command with administrative privilege*

## ps: List the Running Process

The `ps` command lists the running process in your system.

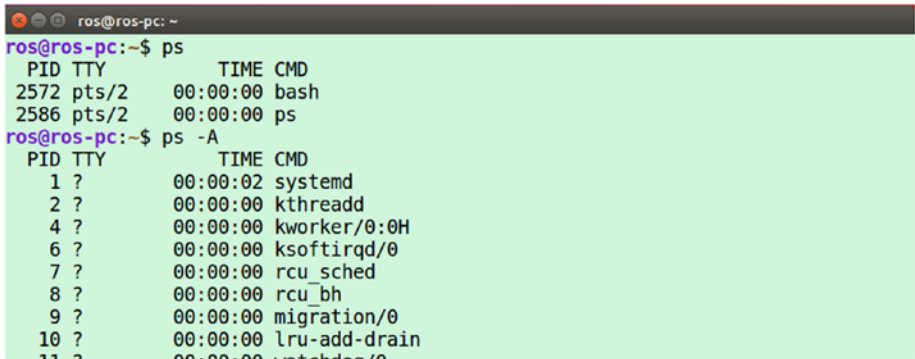
---

Usage: `ps <command arguments>`

Example: `ps -A`

---

When we execute the `ps` command, it lists the process in the current terminal. If we run `ps -A`, it lists all the processes running in the system. Both results are shown in Figure 1-47. PID is the process ID, which identifies the running process. TTY is the terminal type.



```

ros@ros-pc: ~
ros@ros-pc:~$ ps
  PID TTY          TIME CMD
 2572 pts/2    00:00:00 bash
 2586 pts/2    00:00:00 ps
ros@ros-pc:~$ ps -A
  PID TTY          TIME CMD
    1 ?           00:00:02 systemd
    2 ?           00:00:00 kthreadd
    4 ?           00:00:00 kworker/0:0H
    6 ?           00:00:00 ksoftirqd/0
    7 ?           00:00:00 rcu_sched
    8 ?           00:00:00 rcu_bh
    9 ?           00:00:00 migration/0
   10 ?           00:00:00 lru-add-drain
   11 ?           00:00:00 watchdog/0

```

*Figure 1-47. Listing the processes running on the system*

## kill: Kill a Process

To end a process running in the system, use the `kill` command.

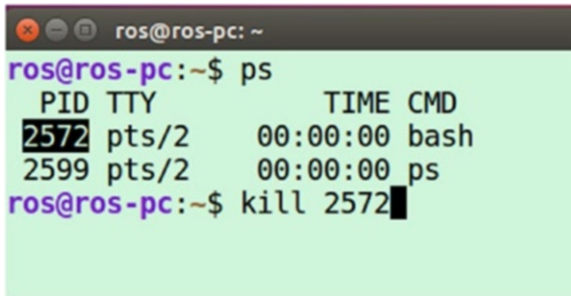
---

Usage: `kill <PID>`

Usage: `kill 2573`

---

To kill a process, we have to identify the PID of process and provide it with the command. The results of the command are shown in Figure 1-48.

A terminal window titled 'ros@ros-pc: ~' with a light green background. The user enters 'ps' and the output shows two processes: PID 2572 (bash) and PID 2599 (ps). Then the user enters 'kill 2572' and the prompt returns.

```
ros@ros-pc: ~  
ros@ros-pc:~$ ps  
  PID TTY          TIME CMD  
 2572 pts/2        00:00:00 bash  
 2599 pts/2        00:00:00 ps  
ros@ros-pc:~$ kill 2572
```

**Figure 1-48.** *Killing a process*

## apt-get: Install a Package in Ubuntu

The apt-get command is important and very useful when working with Ubuntu and ROS. It installs an Ubuntu package that is either in the Ubuntu repositories or on the local system. The packages are called Debian packages, which have .deb extensions. Installing a package requires root permission, so we have to use sudo before the command. We can also update the list of packages in the repositories using this command.

---

Usage: \$ sudo apt-get <command\_argument> <package\_name>

Example: \$ sudo apt-get update

Example: \$ sudo apt-get install htop

Example: \$ sudo apt-get remove htop

---

Figure 1-49 shows the Ubuntu package update using sudo apt-get update. This command updates the package download location in the local system.

```

ros@ros-pc: ~$ sudo apt-get update
[sudo] password for ros:
Hit:1 http://in.archive.ubuntu.com/ubuntu xenial InRelease
Get:2 http://security.ubuntu.com/ubuntu xenial-security InRelease [102 kB]
Get:3 http://in.archive.ubuntu.com/ubuntu xenial-updates InRelease [102 kB]
Get:4 http://in.archive.ubuntu.com/ubuntu xenial-backports InRelease [102 kB]
Fetched 306 kB in 3s (80.5 kB/s)
Reading package lists... Done
ros@ros-pc:~$ █

```

**Figure 1-49.** Updating the Ubuntu software repository

Figure 1-50 shows how to install a package. We are installing a tool called `htop`. It is a terminal process viewer.

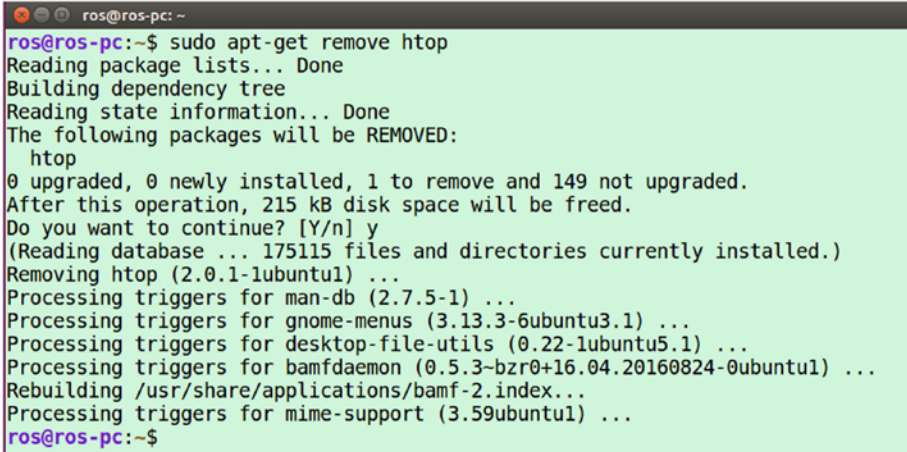
```

ros@ros-pc: ~$ sudo apt-get install htop
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following NEW packages will be installed:
  htop
0 upgraded, 1 newly installed, 0 to remove and 149 not upgraded.
Need to get 76.4 kB of archives.
After this operation, 215 kB of additional disk space will be used.
Get:1 http://in.archive.ubuntu.com/ubuntu xenial-updates/universe amd64 htop amd
64 2.0.1-1ubuntu1 [76.4 kB]
Fetched 76.4 kB in 2s (36.1 kB/s)
Selecting previously unselected package htop.
(Reading database ... 175107 files and directories currently installed.)
Preparing to unpack .../htop 2.0.1-1ubuntu1_amd64.deb ...
Unpacking htop (2.0.1-1ubuntu1) ...
Processing triggers for gnome-menus (3.13.3-6ubuntu3.1) ...
Processing triggers for desktop-file-utils (0.22-1ubuntu5.1) ...
Processing triggers for bamfdaemon (0.5.3-bzr0+16.04.20160824-0ubuntu1) ...
Rebuilding /usr/share/applications/bamf-2.index...
Processing triggers for mime-support (3.59ubuntu1) ...
Processing triggers for man-db (2.7.5-1) ...
Setting up htop (2.0.1-1ubuntu1) ...
ros@ros-pc:~$ █

```

**Figure 1-50.** Installing a package on Ubuntu

The `sudo apt-get remove htop` command in Figure 1-51 shows how to remove a package. We have to use the `remove` argument to delete it.

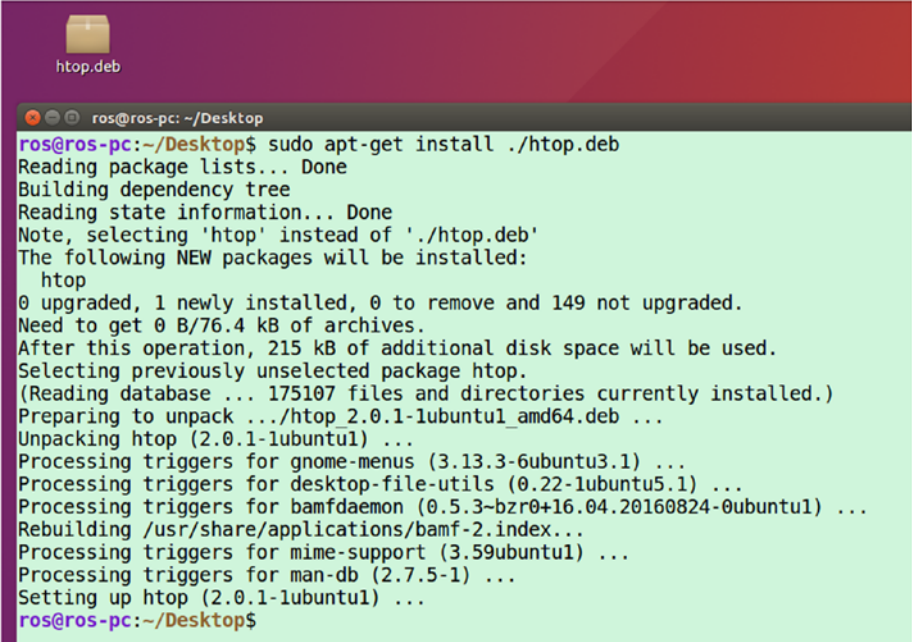
A terminal window with a dark title bar showing 'ros@ros-pc: ~'. The terminal output is as follows:

```
ros@ros-pc:~$ sudo apt-get remove htop
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages will be REMOVED:
  htop
0 upgraded, 0 newly installed, 1 to remove and 149 not upgraded.
After this operation, 215 kB disk space will be freed.
Do you want to continue? [Y/n] y
(Reading database ... 175115 files and directories currently installed.)
Removing htop (2.0.1-1ubuntu1) ...
Processing triggers for man-db (2.7.5-1) ...
Processing triggers for gnome-menus (3.13.3-6ubuntu3.1) ...
Processing triggers for desktop-file-utils (0.22-1ubuntu5.1) ...
Processing triggers for bamfdaemon (0.5.3-bzr0+16.04.20160824-0ubuntu1) ...
Rebuilding /usr/share/applications/bamf-2.index...
Processing triggers for mime-support (3.59ubuntu1) ...
ros@ros-pc:~$
```

**Figure 1-51.** *Removing a package from Ubuntu*

Figure 1-52 shows how to install a local Debian package using the `apt-get` command. The local file is on the same path of the terminal, and the name of the Debian file is `htop.deb`, so we can use the following:

```
$ sudo apt-get install ./htop.deb
```



```

ros@ros-pc: ~/Desktop
ros@ros-pc:~/Desktop$ sudo apt-get install ./htop.deb
Reading package lists... Done
Building dependency tree
Reading state information... Done
Note, selecting 'htop' instead of './htop.deb'
The following NEW packages will be installed:
  htop
0 upgraded, 1 newly installed, 0 to remove and 149 not upgraded.
Need to get 0 B/76.4 kB of archives.
After this operation, 215 kB of additional disk space will be used.
Selecting previously unselected package htop.
(Reading database ... 175107 files and directories currently installed.)
Preparing to unpack .../htop 2.0.1-lubuntu1_amd64.deb ...
Unpacking htop (2.0.1-lubuntu1) ...
Processing triggers for gnome-menus (3.13.3-6ubuntu3.1) ...
Processing triggers for desktop-file-utils (0.22-lubuntu5.1) ...
Processing triggers for bamfdaemon (0.5.3-bzr0+16.04.20160824-0ubuntu1) ...
Rebuilding /usr/share/applications/bamf-2.index...
Processing triggers for mime-support (3.59ubuntu1) ...
Processing triggers for man-db (2.7.5-1) ...
Setting up htop (2.0.1-lubuntu1) ...
ros@ros-pc:~/Desktop$

```

**Figure 1-52.** Installing a Debian package in Ubuntu

## dpkg -i: Install a Package in Ubuntu

The dpkg command is another way to install a Debian package.

---

Usage: dpkg <command\_arguments> debian file name

Example: dpkg -i htop.deb

---

Figure 1-53 shows the results of the dpkg command.





**Figure 1-53.** *Installing a Debian package in Ubuntu*

## reboot: Reboot the System

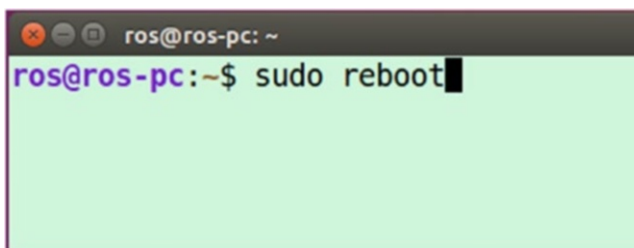
We can restart the system using the terminal command (see Figure 1-54).

---

Usage: `sudo reboot`

---

This instantly reboots the system.



**Figure 1-54.** *Rebooting PC*

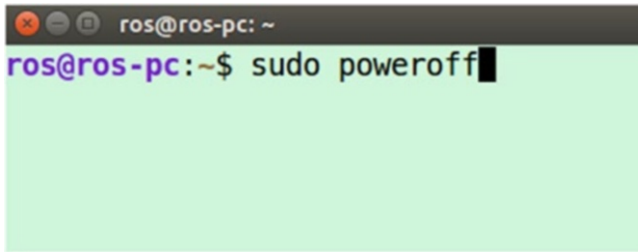
## poweroff: Switch off the System

If you want to instantly shut down the system, use the `poweroff` command (see Figure 1-55).

---

Usage: `$ sudo poweroff`

---



*Figure 1-55. Shutting down the PC*

## htop: Terminal Process View

The `htop` is a process viewer in Linux (see Figure 1-56). It is not installed in the system by default. You have to install it using `apt-get`. This command is very useful for managing process.

---

Usage: `htop`

---

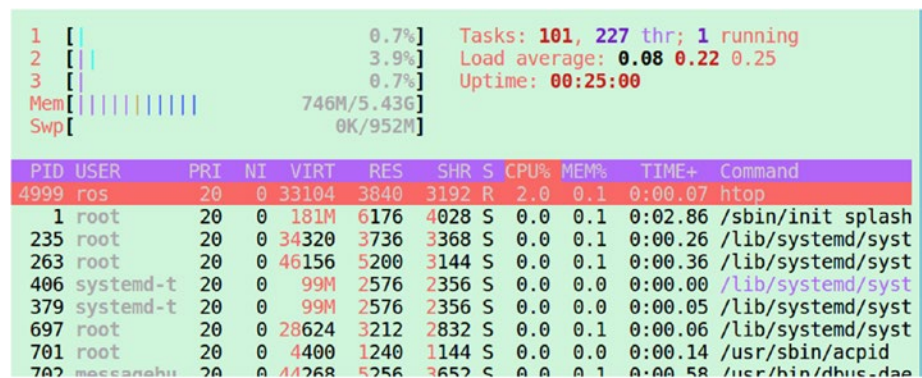


Figure 1-56. Terminal process viewer

## nano: Text Editor in Terminal

There is a useful text editor that you can use while working in the terminal. You can create code inside the terminal (see Figure 1-57).

```
Usage:$ nano file_name

Example:$ nano test.txt
```

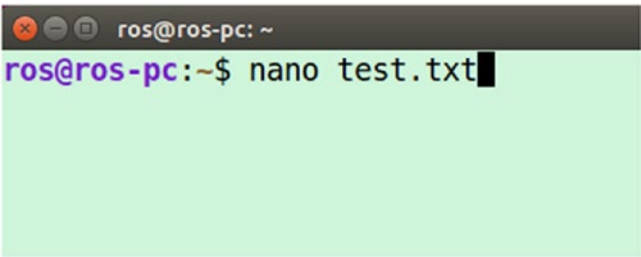
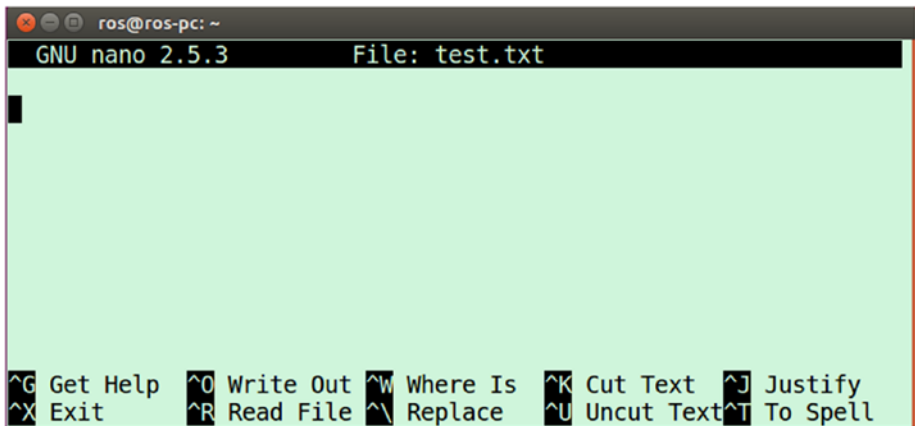


Figure 1-57. Text editor in the terminal

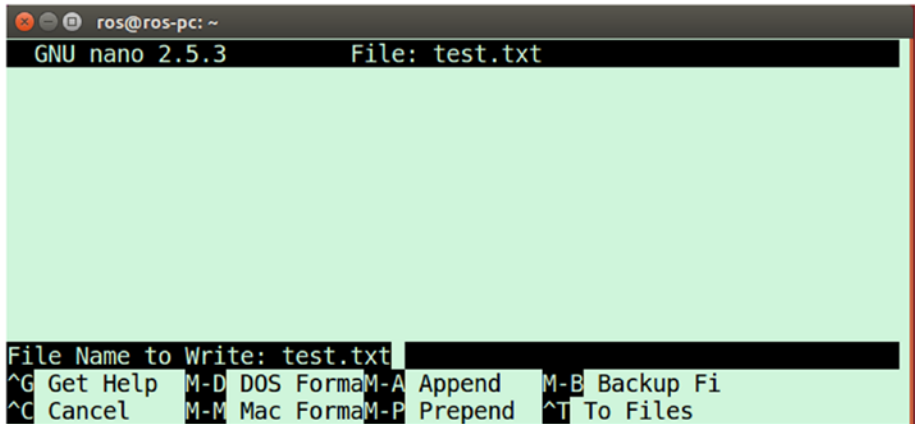
Figure 1-58 shows the resulting screen. In this editor, you can enter your code.



**Figure 1-58.** Nano text editor in terminal

After completing the code, press Ctrl+O to save the file. You are asked to enter the file name. You can enter a new file name or use an existing name. Press Enter to save (see Figure 1-59).

Press Ctrl+X to exit from the editor. To open the file again, use nano file\_name.



**Figure 1-59.** Saving a file in the nano text editor in the terminal

## Summary

This chapter discussed the fundamentals of the Ubuntu operating system, its installation, and the important shell commands that we need for working with robots. This chapter is important because, before working with ROS-based applications, you should have a basic understanding of Linux and its commands. Understanding the Linux environment and its commands is one of the prerequisites for learning ROS. This book discusses all the prerequisites needed for learning ROS. This chapter is the first step in learning ROS.