

TECHNOLOGY



Ubuntu Fundamentals

Introduction to CLI



Learning Objectives

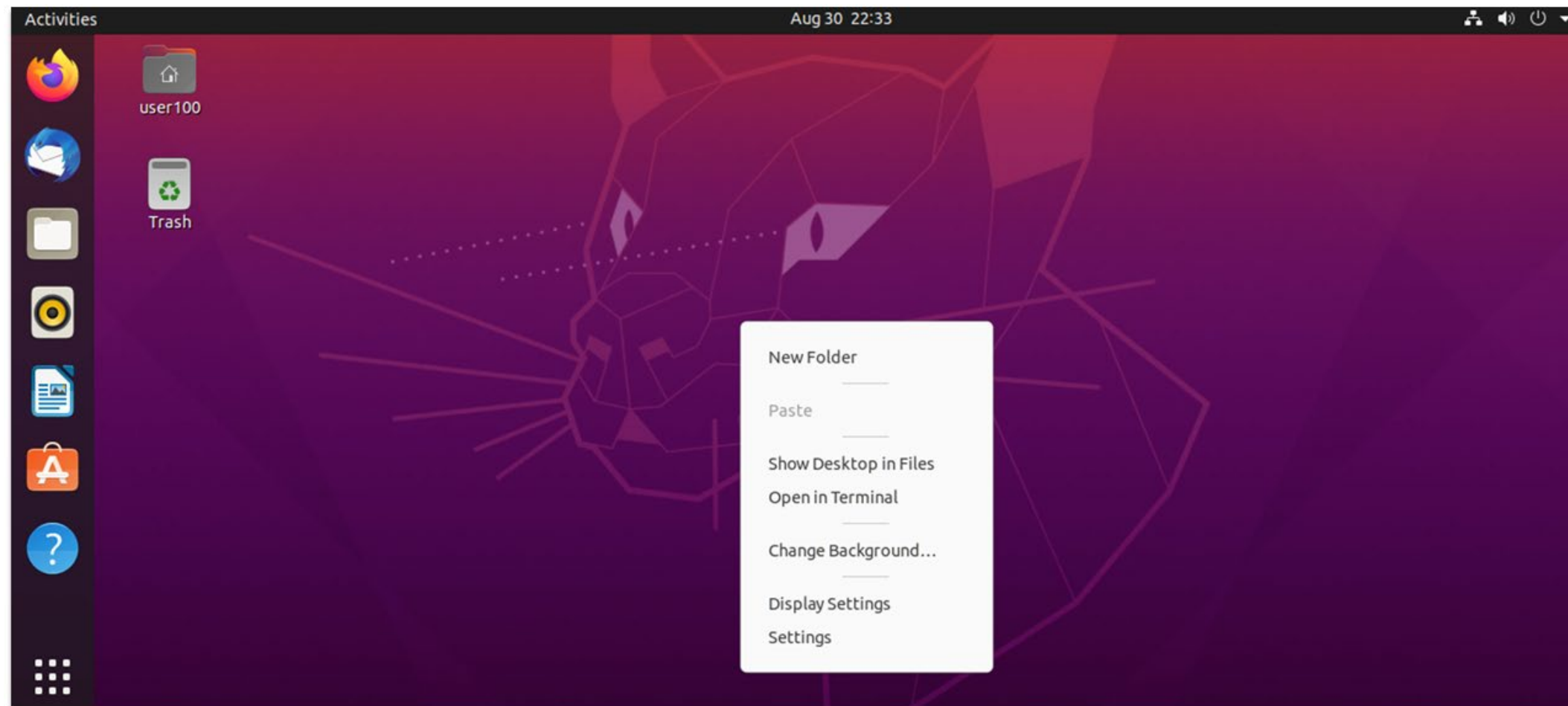
By the end of this lesson, you will be able to:

- 🕒 Execute various commands in the terminal
- 🕒 Apply superuser commands using the sudo command
- 🕒 Find help for commands using manual pages
- 🕒 Navigate through the file system
- 🕒 Manipulate files and folders
- 🕒 Execute system commands



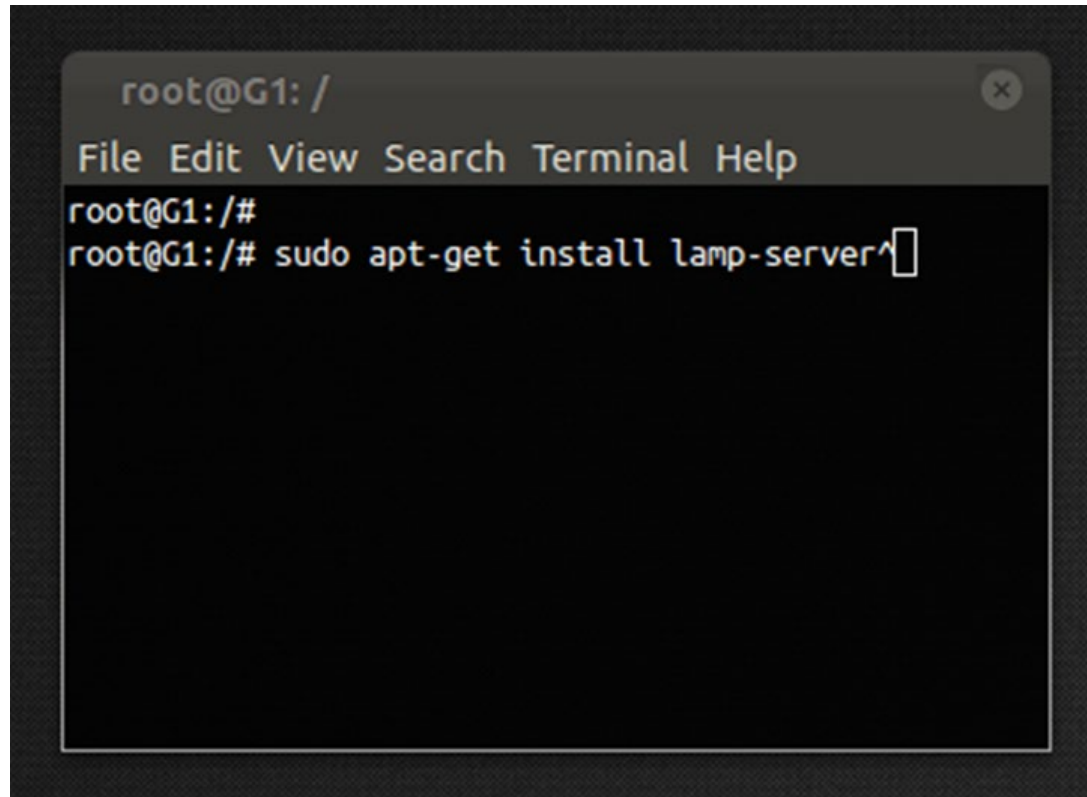
Starting Up the Terminal

To start the terminal, right click on the mouse and select the option open terminal.



Running Commands as Superuser

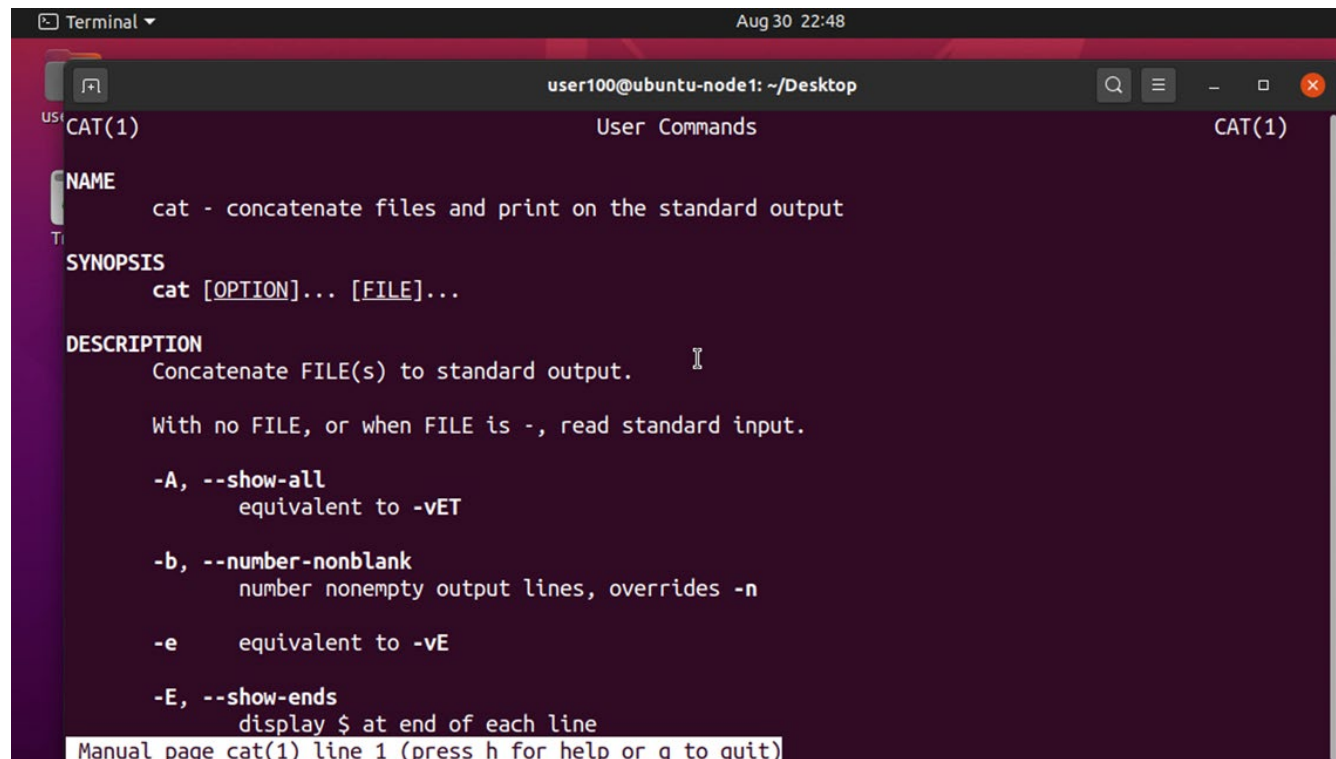
A normal user does not have the privilege to perform certain tasks.

A terminal window titled 'root@G1: /' with a menu bar containing 'File', 'Edit', 'View', 'Search', 'Terminal', and 'Help'. The prompt is 'root@G1: /#'. The command 'sudo apt-get install lamp-server' is entered and followed by a carriage return. The terminal has a dark background and a light-colored text cursor.

```
root@G1: /  
File Edit View Search Terminal Help  
root@G1: /#  
root@G1: /# sudo apt-get install lamp-server^
```

- The **sudo** command in Linux elevates the privilege of a normal user to perform certain restricted tasks.
- In Ubuntu, by default, a user can join the sudo group to get these facilities.
- The user doesn't require a root password of the root user to execute the privilege commands.

Manual Pages



The screenshot shows a terminal window titled 'Terminal' with a timestamp of 'Aug 30 22:48'. The window displays the manual page for the 'cat' command. The page is titled 'CAT(1)' and 'User Commands'. It includes sections for 'NAME', 'SYNOPSIS', and 'DESCRIPTION'. The 'NAME' section states 'cat - concatenate files and print on the standard output'. The 'SYNOPSIS' section shows 'cat [OPTION]... [FILE]...'. The 'DESCRIPTION' section explains that 'cat' concatenates FILE(s) to standard output. It also lists several options: '-A, --show-all' (equivalent to -vET), '-b, --number-nonblank' (number nonempty output lines, overrides -n), '-e' (equivalent to -vE), and '-E, --show-ends' (display \$ at end of each line). At the bottom, it says 'Manual page cat(1) line 1 (press h for help or q to quit)'.

```
Terminal Aug 30 22:48
user100@ubuntu-node1: ~/Desktop
CAT(1) User Commands CAT(1)
NAME
  cat - concatenate files and print on the standard output
SYNOPSIS
  cat [OPTION]... [FILE]...
DESCRIPTION
  Concatenate FILE(s) to standard output.

  With no FILE, or when FILE is -, read standard input.

  -A, --show-all
      equivalent to -vET

  -b, --number-nonblank
      number nonempty output lines, overrides -n

  -e
      equivalent to -vE

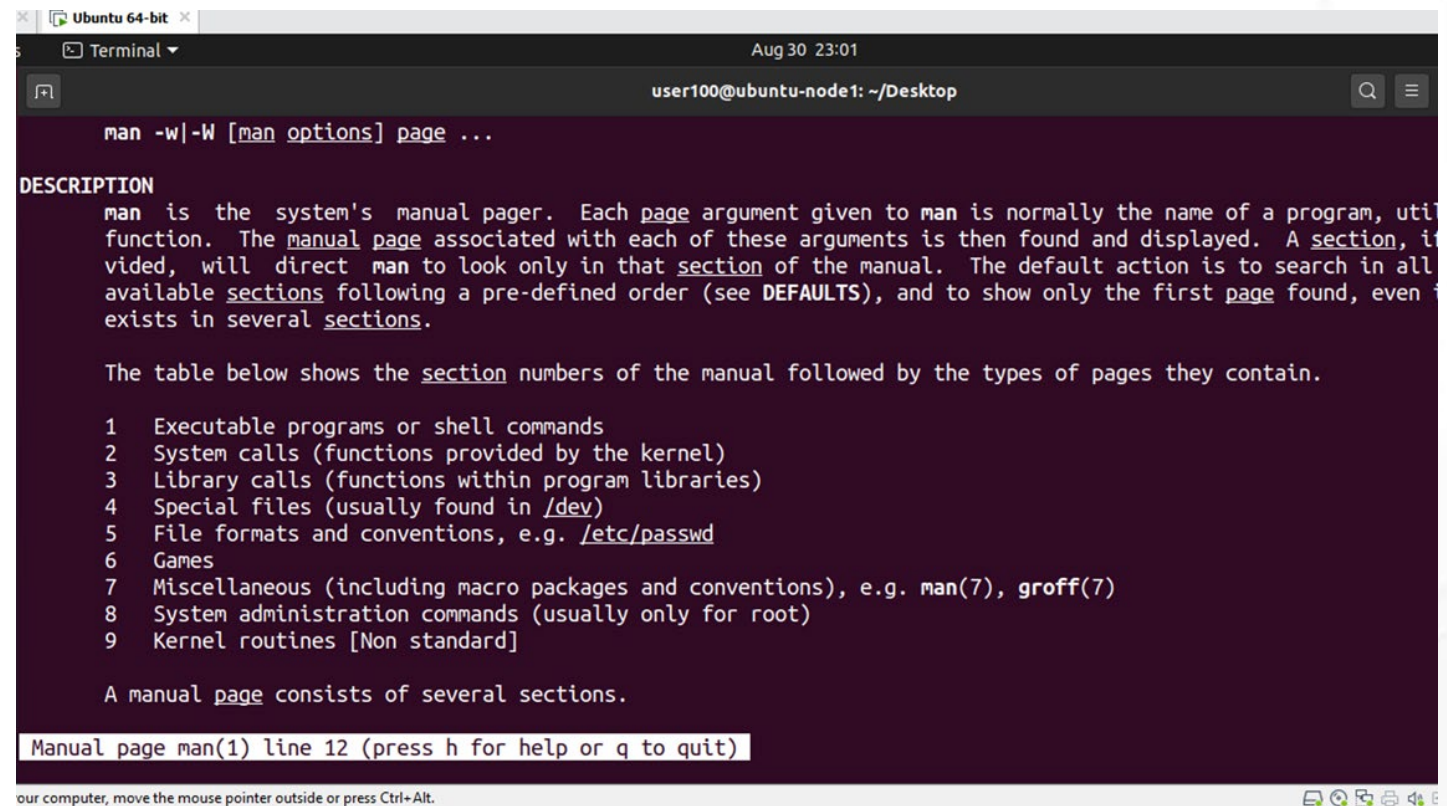
  -E, --show-ends
      display $ at end of each line
Manual page cat(1) line 1 (press h for help or q to quit)
```

- Manual pages have all the information about the commands.
- They are also called man pages.
- Example: **man cat** command to view the function, description, and options of the cat command.

Manual Sections

The manual pages are further divided into several sections. They are :

- The Executable programs or shell commands
- System calls
- Library calls
- Special files
- File formats
- Miscellaneous
- Games
- System admin commands
- Kernel routines



The screenshot shows a terminal window titled 'Ubuntu 64-bit' with a 'Terminal' tab. The prompt is 'user100@ubuntu-node1: ~/Desktop'. The command 'man -w|-W [man options] page ...' has been entered. The output displays the 'DESCRIPTION' section of the 'man' manual page. It explains that 'man' is the system's manual pager and lists the nine manual sections and their corresponding page types. A table is provided for reference. The terminal also shows the status 'Manual page man(1) line 12 (press h for help or q to quit)' and a footer instruction to move the mouse pointer outside or press Ctrl+Alt.

```
man -w|-W [man options] page ...

DESCRIPTION
man is the system's manual pager. Each page argument given to man is normally the name of a program, utility, or function. The manual page associated with each of these arguments is then found and displayed. A section, if provided, will direct man to look only in that section of the manual. The default action is to search in all available sections following a pre-defined order (see DEFAULTS), and to show only the first page found, even if it exists in several sections.

The table below shows the section numbers of the manual followed by the types of pages they contain.

1 Executable programs or shell commands
2 System calls (functions provided by the kernel)
3 Library calls (functions within program libraries)
4 Special files (usually found in /dev)
5 File formats and conventions, e.g. /etc/passwd
6 Games
7 Miscellaneous (including macro packages and conventions), e.g. man(7), groff(7)
8 System administration commands (usually only for root)
9 Kernel routines [Non standard]

A manual page consists of several sections.

Manual page man(1) line 12 (press h for help or q to quit)
```


Manual Captions

A command's manual page has several captions:

- Name
- Synopsis
- Description
- Options
- Exit values
- Files

```
NAME
    useradd - create a new user or update default new user information

SYNOPSIS
    useradd [options] LOGIN

    useradd -D

    useradd -D [options]

DESCRIPTION
    useradd is a low level utility for adding users. On Debian, administrators should usually

    When invoked without the -D option, the useradd command creates a new user account using the
    command line plus the default values from the system. Depending on command line options, it
    update system files and may also create the new user's home directory and copy initial files.

    By default, a group will also be created for the new user (see -g, -N, -U, and USERGROUPS_

OPTIONS
    The options which apply to the useradd command are:

    /etc/subuid
        Per user subordinate user IDs.

    /etc/login.defs
        Shadow password suite configuration.

EXIT VALUES
    The useradd command exits with the following values:

    0
        success

    1
        can't update password file

    2
        invalid command syntax

    3
        invalid argument to option

    4

    USERGROUPS_ENAB (boolean)
        If set to yes, userdel will remove the user's group if it contains no more members, and useradd will
        default a group with the name of the user.

FILES
    /etc/passwd
        User account information.

    /etc/shadow
        Secure user account information.

    /etc/group
        Group account information.

    /etc/gshadow
        Secure group account information.

    /etc/default/useradd
        Default values for account creation.

    /etc/skel/
        Directory containing default files.

Manual page useradd(8) line 323 (press h for help or q to quit)
```


man -k Command

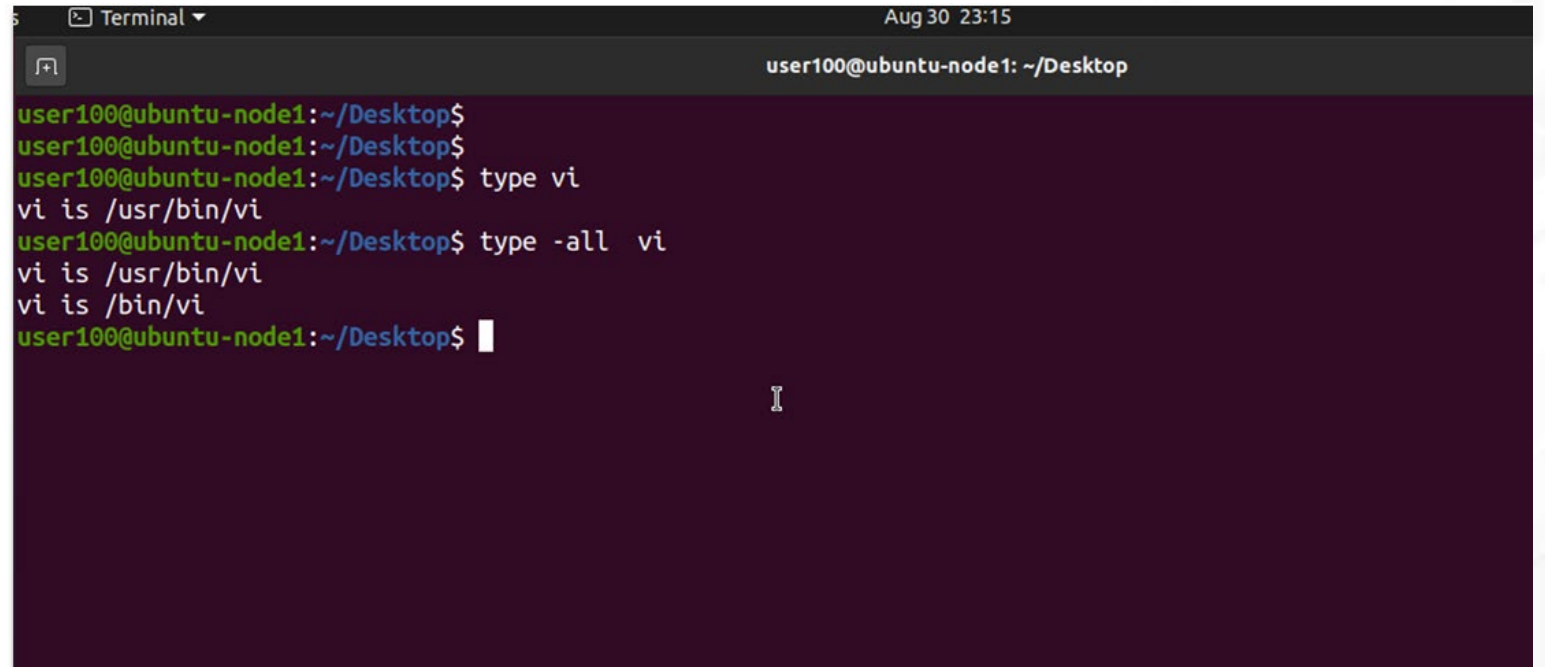
- Use the command `man -k <keyword>`, keyword can be anything.
- Example: Check for `printf`. This command will extract all the text information that matches one or more characters from the keyword.
- After extracting all the content, find the command from the list. It is `printf` in this case.
- The number next to the command indicates the section of the man page this command belongs to.

```
Terminal Aug 30 23:08
user100@ubuntu-node1: ~/Desktop
user100@ubuntu-node1:~$ man -k printf
asprintf (3)      - print to allocated string
dprintf (3)      - formatted output conversion
fprintf (3)      - formatted output conversion
fwprintf (3)     - formatted wide-character output conversion
printf (1)       - format and print data
printf (3)       - formatted output conversion
snprintf (3)     - formatted output conversion
sprintf (3)      - formatted output conversion
swprintf (3)     - formatted wide-character output conversion
vasprintf (3)    - print to allocated string
vdprintf (3)     - formatted output conversion
vfprintf (3)     - formatted output conversion
vfwprintf (3)    - formatted wide-character output conversion
vprintf (3)      - formatted output conversion
vsnprintf (3)    - formatted output conversion
vsprintf (3)     - formatted output conversion
vswprintf (3)    - formatted wide-character output conversion
vwprintf (3)     - formatted wide-character output conversion
wprintf (3)      - formatted wide-character output conversion
```

Find Command

The **type** command or **which** command is used to find the version of a command.

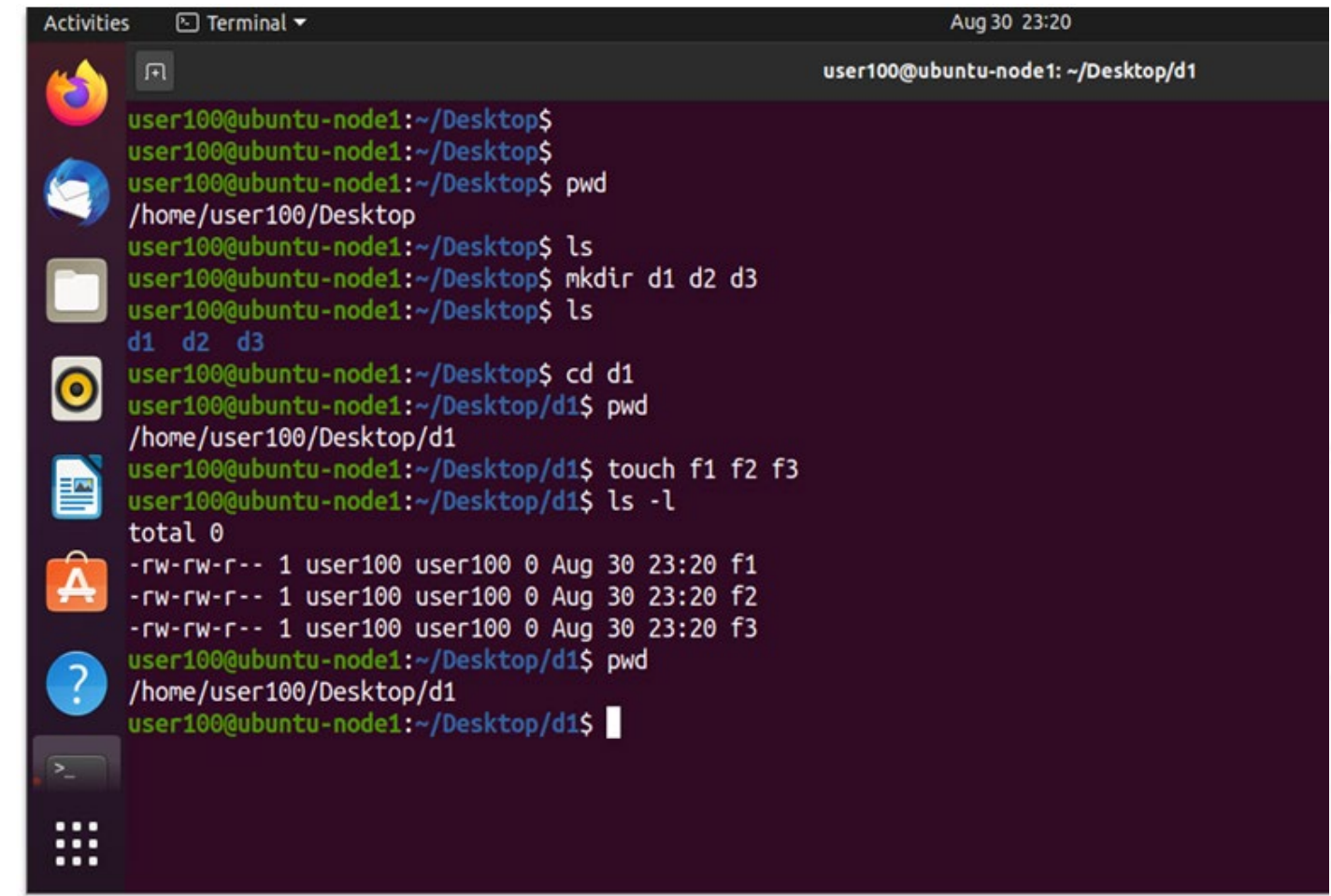
- The version of the command is selected based on the path variable.



```
Terminal Aug 30 23:15
user100@ubuntu-node1: ~/Desktop
user100@ubuntu-node1:~/Desktop$
user100@ubuntu-node1:~/Desktop$
user100@ubuntu-node1:~/Desktop$ type vi
vi is /usr/bin/vi
user100@ubuntu-node1:~/Desktop$ type -all vi
vi is /usr/bin/vi
vi is /bin/vi
user100@ubuntu-node1:~/Desktop$
```

Moving Around the File System

- A directory is a file that contains the names of the other files.
- It is also called a pointer as it points to the files you are looking for.
- The command `$ ls-l` can be used to list the contents of the directory and `$ pwd` to view the current working directory.

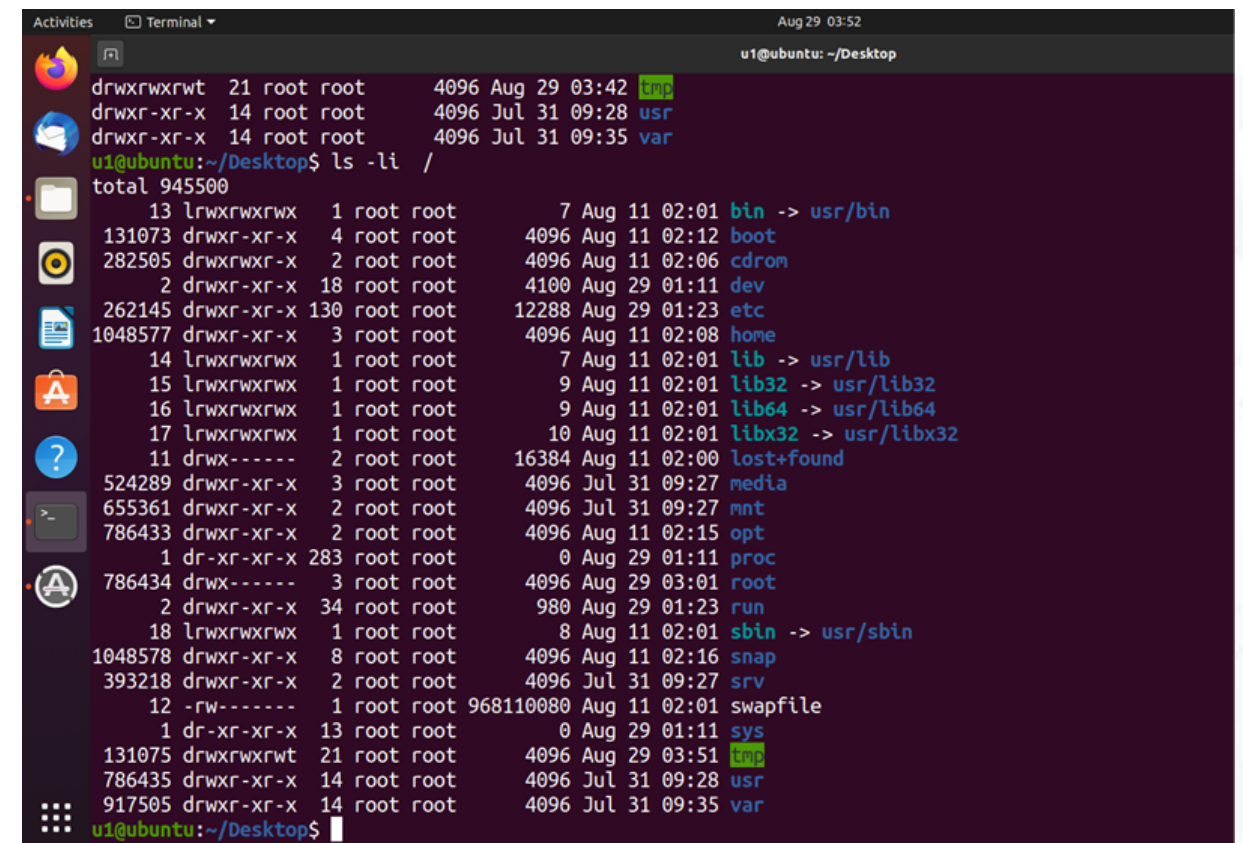


```
Activities Terminal Aug 30 23:20 user100@ubuntu-node1: ~/Desktop/d1
user100@ubuntu-node1:~/Desktop$
user100@ubuntu-node1:~/Desktop$
user100@ubuntu-node1:~/Desktop$ pwd
/home/user100/Desktop
user100@ubuntu-node1:~/Desktop$ ls
user100@ubuntu-node1:~/Desktop$ mkdir d1 d2 d3
user100@ubuntu-node1:~/Desktop$ ls
d1 d2 d3
user100@ubuntu-node1:~/Desktop$ cd d1
user100@ubuntu-node1:~/Desktop/d1$ pwd
/home/user100/Desktop/d1
user100@ubuntu-node1:~/Desktop/d1$ touch f1 f2 f3
user100@ubuntu-node1:~/Desktop/d1$ ls -l
total 0
-rw-rw-r-- 1 user100 user100 0 Aug 30 23:20 f1
-rw-rw-r-- 1 user100 user100 0 Aug 30 23:20 f2
-rw-rw-r-- 1 user100 user100 0 Aug 30 23:20 f3
user100@ubuntu-node1:~/Desktop/d1$ pwd
/home/user100/Desktop/d1
user100@ubuntu-node1:~/Desktop/d1$
```


Moving Around the File System

A file is defined as the space occupied by data on the hard disk.

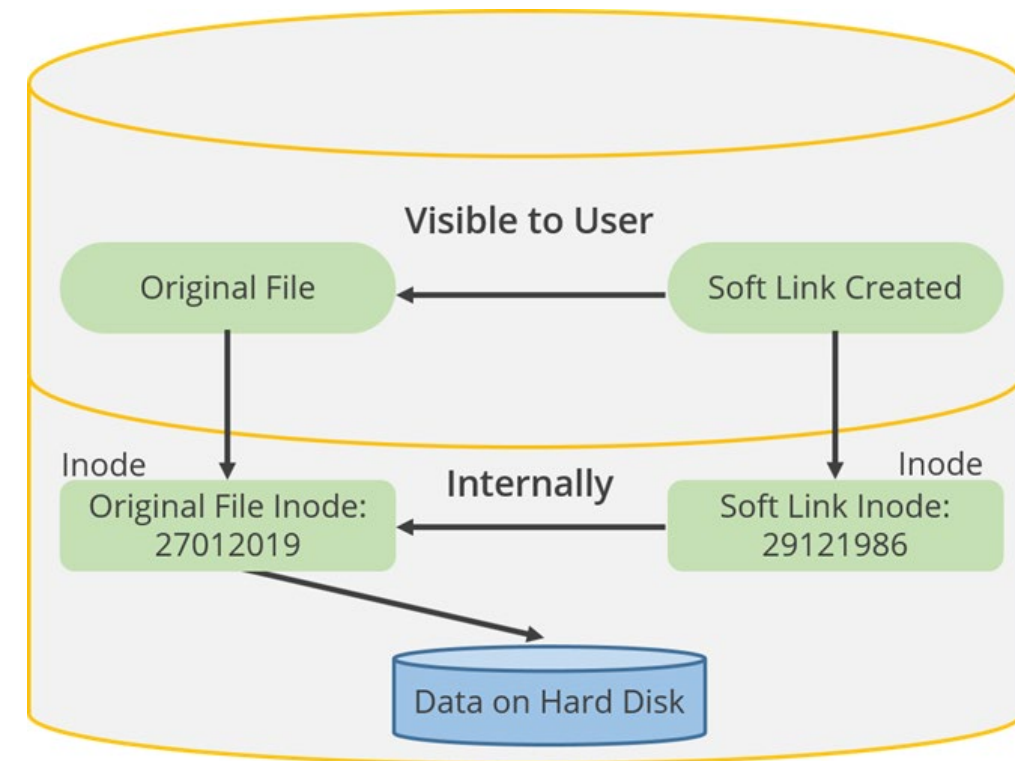
- Every file or directory in Linux is assigned a numerical value called **inode**.
- The system will not identify the files without the inode numbers.
- Inode numbering will start from 2, as 0 and 1 are reserved numbers.
- The default directory is root; its inode number is 2.



```
Aug 29 03:52
u1@ubuntu: ~/Desktop
drwxrwxrwt 21 root root 4096 Aug 29 03:42 tmp
drwxr-xr-x 14 root root 4096 Jul 31 09:28 usr
drwxr-xr-x 14 root root 4096 Jul 31 09:35 var
u1@ubuntu:~/Desktop$ ls -li /
total 945500
13 lrwxrwxrwx 1 root root 7 Aug 11 02:01 bin -> usr/bin
131073 drwxr-xr-x 4 root root 4096 Aug 11 02:12 boot
282505 drwxrwxr-x 2 root root 4096 Aug 11 02:06 cdrom
2 drwxr-xr-x 18 root root 4100 Aug 29 01:11 dev
262145 drwxr-xr-x 130 root root 12288 Aug 29 01:23 etc
1048577 drwxr-xr-x 3 root root 4096 Aug 11 02:08 home
14 lrwxrwxrwx 1 root root 7 Aug 11 02:01 lib -> usr/lib
15 lrwxrwxrwx 1 root root 9 Aug 11 02:01 lib32 -> usr/lib32
16 lrwxrwxrwx 1 root root 9 Aug 11 02:01 lib64 -> usr/lib64
17 lrwxrwxrwx 1 root root 10 Aug 11 02:01 libx32 -> usr/libx32
11 drwx----- 2 root root 16384 Aug 11 02:00 lost+found
524289 drwxr-xr-x 3 root root 4096 Jul 31 09:27 media
655361 drwxr-xr-x 2 root root 4096 Jul 31 09:27 mnt
786433 drwxr-xr-x 2 root root 4096 Aug 11 02:15 opt
1 dr-xr-xr-x 283 root root 0 Aug 29 01:11 proc
786434 drwx----- 3 root root 4096 Aug 29 03:01 root
2 drwxr-xr-x 34 root root 980 Aug 29 01:23 run
18 lrwxrwxrwx 1 root root 8 Aug 11 02:01 sbin -> usr/sbin
1048578 drwxr-xr-x 8 root root 4096 Aug 11 02:16 snap
393218 drwxr-xr-x 2 root root 4096 Jul 31 09:27 srv
12 -rw----- 1 root root 968110080 Aug 11 02:01 swapfile
1 dr-xr-xr-x 13 root root 0 Aug 29 01:11 sys
131075 drwxrwxrwt 21 root root 4096 Aug 29 03:51 tmp
786435 drwxr-xr-x 14 root root 4096 Jul 31 09:28 usr
917505 drwxr-xr-x 14 root root 4096 Jul 31 09:35 var
u1@ubuntu:~/Desktop$
```

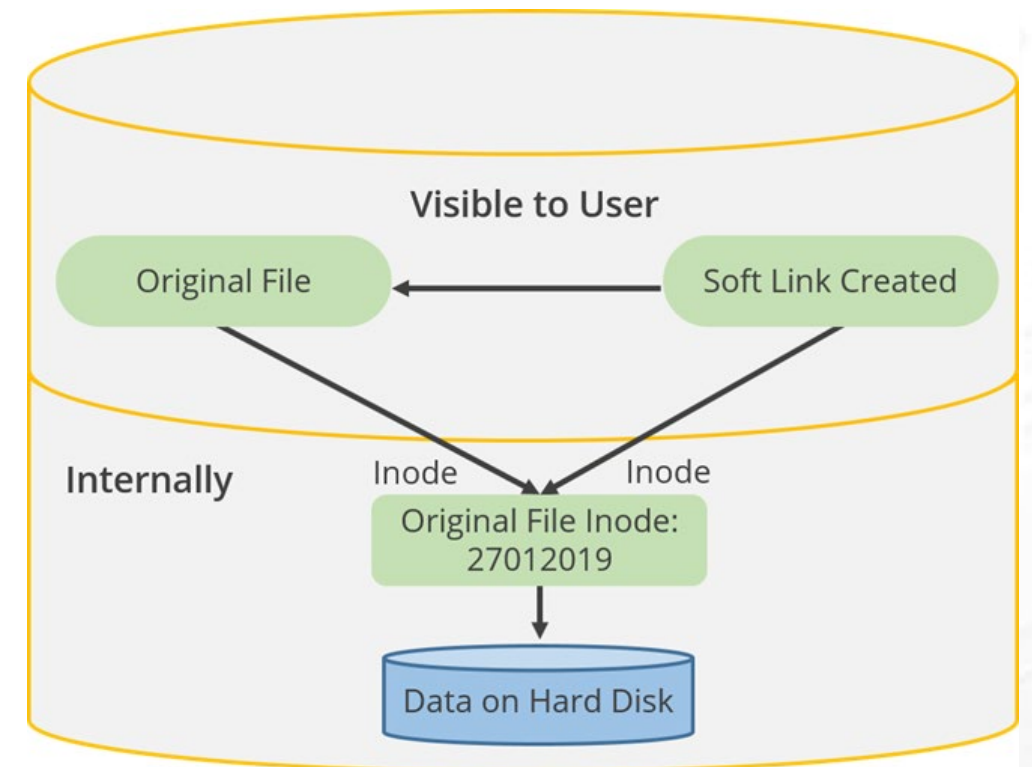
Soft and Hard Links

- As some files have long names, it might be difficult to refer to them. Soft links help you refer to the same files with a shorter version of the name.
- You can use a soft link to create two or more names for a single file and use any one to refer to the file.



Soft and Hard Links

- Hard links are used to have a backup copy of the file without having to occupy any additional space.
- This is useful in case of accidental deletion of files. So, even if one file is removed, you can retrieve the same data using the copied version.
- Any change in one file gets reflected in all the copied files.
- The copied files have the same inode number.



Manipulating Commands

Some of the commands used to manipulate files are:

cp

Copy

mv

Rename

rm

Remove

ls

List files and directories

mkdir

Create a directory

chmod

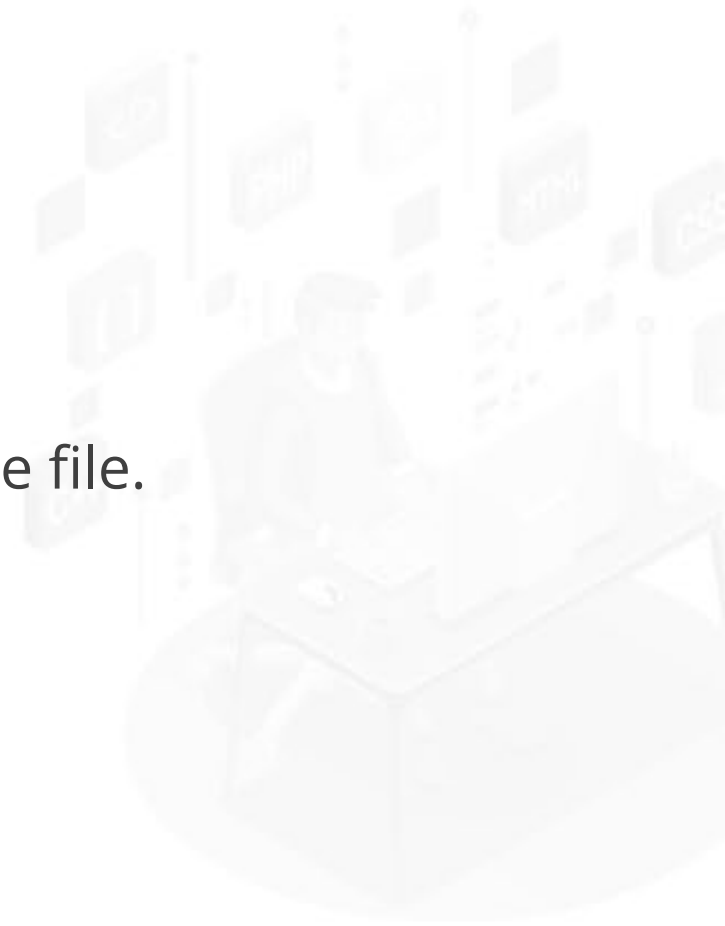
Change permissions

chown

Change ownership

Creating Files and Directories

- The three ways to create a file in Ubuntu are:
 - touch command
 - cat command
 - vi editor.
- The file created by any of these three methods occupies zero bytes of space.
- After entering the required data in the created file, use the **control + d** keys to save the file.

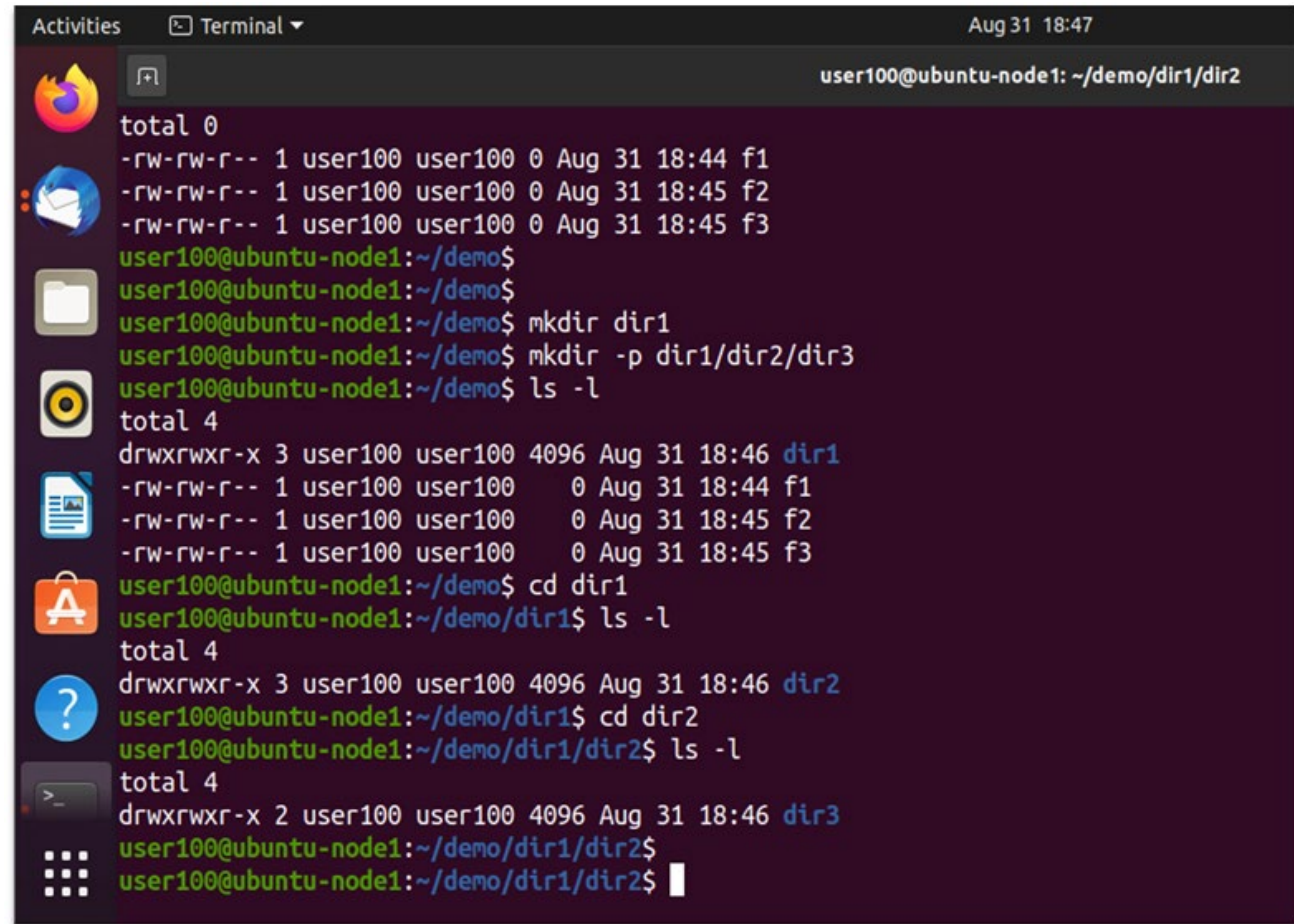


Creating Files and Directories

The `mkdir` command is used to create a directory.

Syntax:

```
mkdir <directory_name>
```

A terminal window titled 'Terminal' showing a series of commands and their outputs. The user is in a directory ~/demo. They first list files f1, f2, and f3. Then they create a directory 'dir1'. Next, they create a nested directory structure 'dir1/dir2/dir3' using the -p flag. Finally, they navigate into each directory and list the contents, showing the hierarchy is correctly created.

```
Aug 31 18:47
user100@ubuntu-node1: ~/demo/dir1/dir2

total 0
-rw-rw-r-- 1 user100 user100 0 Aug 31 18:44 f1
-rw-rw-r-- 1 user100 user100 0 Aug 31 18:45 f2
-rw-rw-r-- 1 user100 user100 0 Aug 31 18:45 f3
user100@ubuntu-node1:~/demo$
user100@ubuntu-node1:~/demo$
user100@ubuntu-node1:~/demo$ mkdir dir1
user100@ubuntu-node1:~/demo$ mkdir -p dir1/dir2/dir3
user100@ubuntu-node1:~/demo$ ls -l
total 4
drwxrwxr-x 3 user100 user100 4096 Aug 31 18:46 dir1
-rw-rw-r-- 1 user100 user100 0 Aug 31 18:44 f1
-rw-rw-r-- 1 user100 user100 0 Aug 31 18:45 f2
-rw-rw-r-- 1 user100 user100 0 Aug 31 18:45 f3
user100@ubuntu-node1:~/demo$ cd dir1
user100@ubuntu-node1:~/demo/dir1$ ls -l
total 4
drwxrwxr-x 3 user100 user100 4096 Aug 31 18:46 dir2
user100@ubuntu-node1:~/demo/dir1$ cd dir2
user100@ubuntu-node1:~/demo/dir1/dir2$ ls -l
total 4
drwxrwxr-x 2 user100 user100 4096 Aug 31 18:46 dir3
user100@ubuntu-node1:~/demo/dir1/dir2$
user100@ubuntu-node1:~/demo/dir1/dir2$
```

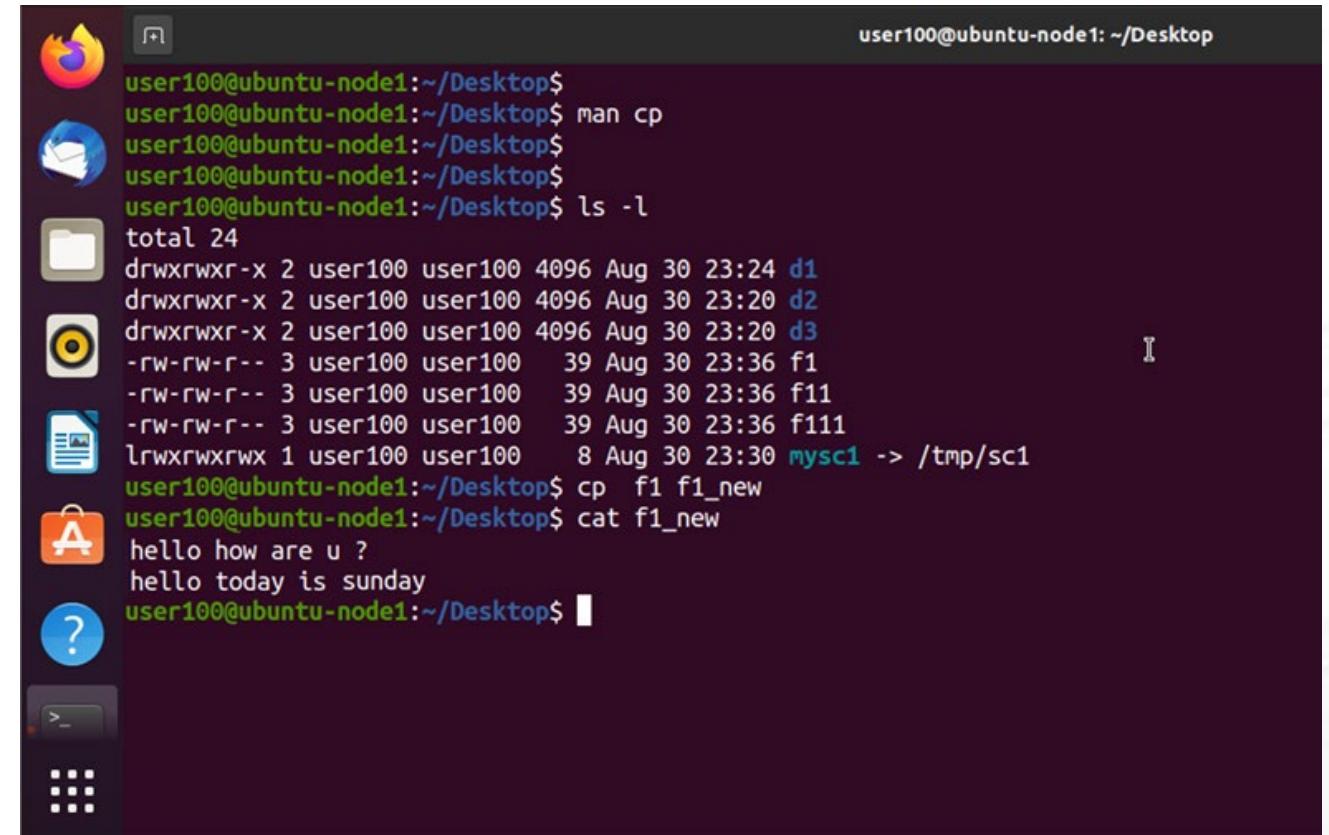
Hierarchy of directories can be created using this command.

Copying Files and Directories

In Linux, the command **cp** is used to copy a file or its contents from one folder to another folder.

Syntax:

```
cp <source_filename> <destination_filename>
```

A terminal window titled 'user100@ubuntu-node1: ~/Desktop' showing a series of commands and their outputs. The commands include 'man cp', 'ls -l', 'cp f1 f1_new', and 'cat f1_new'. The output of 'ls -l' shows a directory listing with permissions, owner, group, size, date, and file names. The output of 'cat f1_new' shows the contents of the copied file.

```
user100@ubuntu-node1:~/Desktop$  
user100@ubuntu-node1:~/Desktop$ man cp  
user100@ubuntu-node1:~/Desktop$  
user100@ubuntu-node1:~/Desktop$  
user100@ubuntu-node1:~/Desktop$ ls -l  
total 24  
drwxrwxr-x 2 user100 user100 4096 Aug 30 23:24 d1  
drwxrwxr-x 2 user100 user100 4096 Aug 30 23:20 d2  
drwxrwxr-x 2 user100 user100 4096 Aug 30 23:20 d3  
-rw-rw-r-- 3 user100 user100   39 Aug 30 23:36 f1  
-rw-rw-r-- 3 user100 user100   39 Aug 30 23:36 f11  
-rw-rw-r-- 3 user100 user100   39 Aug 30 23:36 f111  
lrwxrwxrwx 1 user100 user100    8 Aug 30 23:30 mysc1 -> /tmp/sc1  
user100@ubuntu-node1:~/Desktop$ cp f1 f1_new  
user100@ubuntu-node1:~/Desktop$ cat f1_new  
hello how are u ?  
hello today is sunday  
user100@ubuntu-node1:~/Desktop$
```

Copying Files and Directories

The different options available for the copy command are:

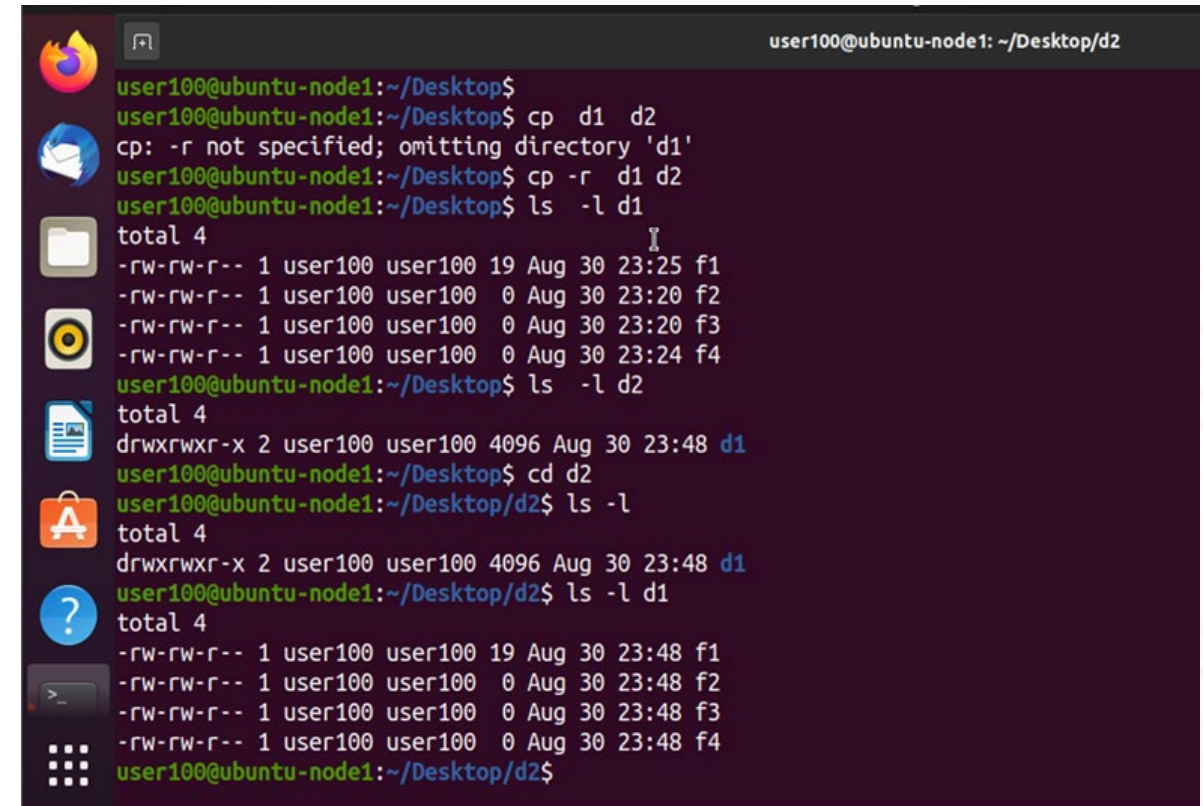
| Option | Description |
|--------|---|
| -i | Ask for user permission before writing to the destination file |
| -b | Create a backup copy of the destination file |
| -f | Copy file to the destination file forcefully if you don't have permission |
| -p | Preserve file attributes if possible |

Copying Files and Directories

- The **copy** command of a directory is similar to the copy command of a file.
- The option **-r** will recursively copy the contents of the source directory to the destination directory.

Syntax:

```
cp -r <source_dir> <destination_dir>
```



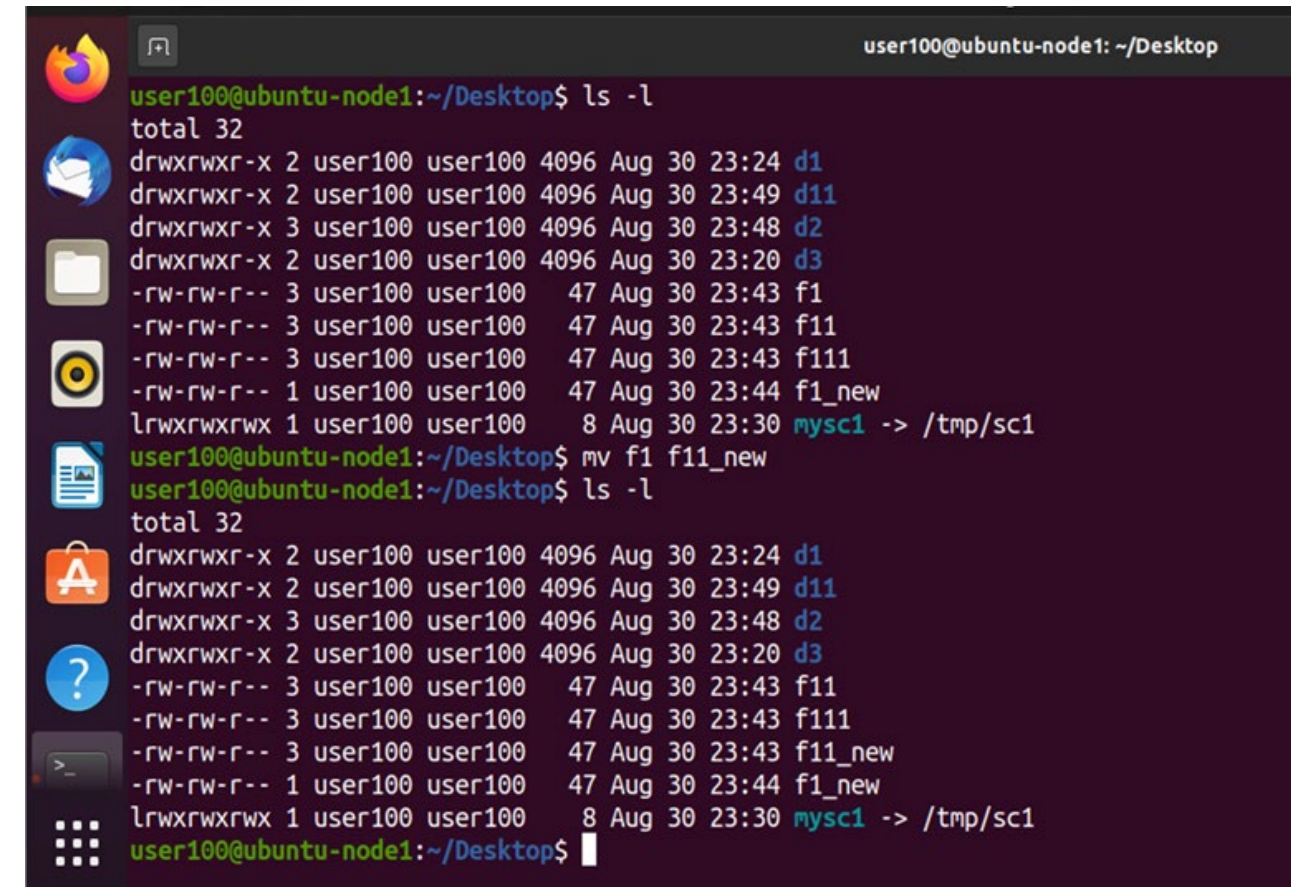
```
user100@ubuntu-node1: ~/Desktop/d2
user100@ubuntu-node1:~/Desktop$
user100@ubuntu-node1:~/Desktop$ cp d1 d2
cp: -r not specified; omitting directory 'd1'
user100@ubuntu-node1:~/Desktop$ cp -r d1 d2
user100@ubuntu-node1:~/Desktop$ ls -l d1
total 4
-rw-rw-r-- 1 user100 user100 19 Aug 30 23:25 f1
-rw-rw-r-- 1 user100 user100 0 Aug 30 23:20 f2
-rw-rw-r-- 1 user100 user100 0 Aug 30 23:20 f3
-rw-rw-r-- 1 user100 user100 0 Aug 30 23:24 f4
user100@ubuntu-node1:~/Desktop$ ls -l d2
total 4
drwxrwxr-x 2 user100 user100 4096 Aug 30 23:48 d1
user100@ubuntu-node1:~/Desktop$ cd d2
user100@ubuntu-node1:~/Desktop/d2$ ls -l
total 4
drwxrwxr-x 2 user100 user100 4096 Aug 30 23:48 d1
user100@ubuntu-node1:~/Desktop/d2$ ls -l d1
total 4
-rw-rw-r-- 1 user100 user100 19 Aug 30 23:48 f1
-rw-rw-r-- 1 user100 user100 0 Aug 30 23:48 f2
-rw-rw-r-- 1 user100 user100 0 Aug 30 23:48 f3
-rw-rw-r-- 1 user100 user100 0 Aug 30 23:48 f4
user100@ubuntu-node1:~/Desktop/d2$
```


Renaming Files and Directories

The **mv** command in Linux is used to rename a given file.

Syntax:

```
mv <old_file_name> <new_file_name>
```



A terminal window titled 'user100@ubuntu-node1: ~/Desktop' showing the execution of the 'mv' command. The user first runs 'ls -l' to list files in the Desktop directory. The output shows several files and directories, including 'd1', 'd11', 'd2', 'd3', 'f1', 'f11', 'f111', 'f1_new', and 'mysc1'. The user then runs 'mv f1 f11_new' to rename the file 'f1' to 'f11_new'. Finally, the user runs 'ls -l' again, and the output shows that 'f1' has been replaced by 'f11_new'.

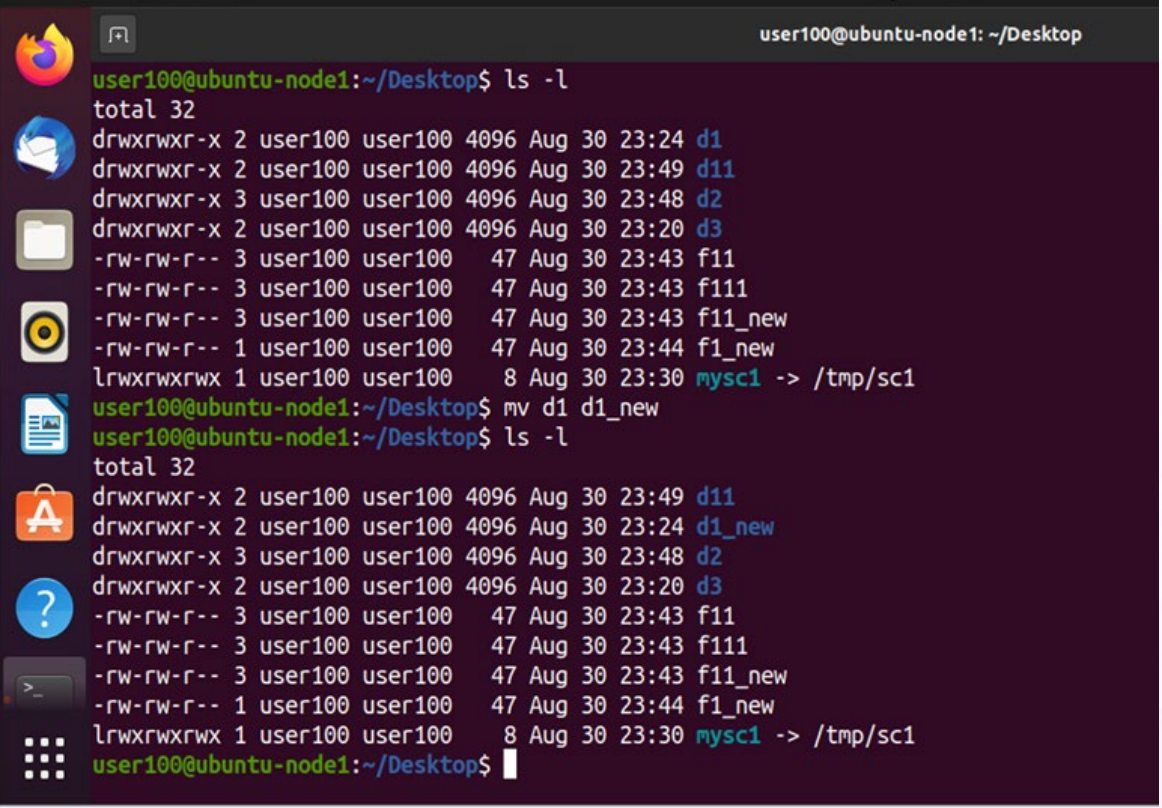
```
user100@ubuntu-node1:~/Desktop$ ls -l
total 32
drwxrwxr-x 2 user100 user100 4096 Aug 30 23:24 d1
drwxrwxr-x 2 user100 user100 4096 Aug 30 23:49 d11
drwxrwxr-x 3 user100 user100 4096 Aug 30 23:48 d2
drwxrwxr-x 2 user100 user100 4096 Aug 30 23:20 d3
-rw-rw-r-- 3 user100 user100  47 Aug 30 23:43 f1
-rw-rw-r-- 3 user100 user100  47 Aug 30 23:43 f11
-rw-rw-r-- 3 user100 user100  47 Aug 30 23:43 f111
-rw-rw-r-- 1 user100 user100  47 Aug 30 23:44 f1_new
lrwxrwxrwx 1 user100 user100   8 Aug 30 23:30 mysc1 -> /tmp/sc1
user100@ubuntu-node1:~/Desktop$ mv f1 f11_new
user100@ubuntu-node1:~/Desktop$ ls -l
total 32
drwxrwxr-x 2 user100 user100 4096 Aug 30 23:24 d1
drwxrwxr-x 2 user100 user100 4096 Aug 30 23:49 d11
drwxrwxr-x 3 user100 user100 4096 Aug 30 23:48 d2
drwxrwxr-x 2 user100 user100 4096 Aug 30 23:20 d3
-rw-rw-r-- 3 user100 user100  47 Aug 30 23:43 f11
-rw-rw-r-- 3 user100 user100  47 Aug 30 23:43 f111
-rw-rw-r-- 3 user100 user100  47 Aug 30 23:43 f11_new
-rw-rw-r-- 1 user100 user100  47 Aug 30 23:44 f1_new
lrwxrwxrwx 1 user100 user100   8 Aug 30 23:30 mysc1 -> /tmp/sc1
user100@ubuntu-node1:~/Desktop$
```

Renaming Files and Directories

The same `mv` command is used to rename a directory.

Syntax:

```
mv <old_dir_name> <new_dir_name>
```

A terminal window titled 'user100@ubuntu-node1: ~/Desktop' showing the process of renaming a directory. The user runs 'ls -l' to list files, then 'mv d1 d1_new' to rename directory 'd1' to 'd1_new', and finally 'ls -l' again to confirm the change. The directory 'd1' is replaced by 'd1_new' in the listing.

```
user100@ubuntu-node1:~/Desktop$ ls -l
total 32
drwxrwxr-x 2 user100 user100 4096 Aug 30 23:24 d1
drwxrwxr-x 2 user100 user100 4096 Aug 30 23:49 d11
drwxrwxr-x 3 user100 user100 4096 Aug 30 23:48 d2
drwxrwxr-x 2 user100 user100 4096 Aug 30 23:20 d3
-rw-rw-r-- 3 user100 user100 47 Aug 30 23:43 f11
-rw-rw-r-- 3 user100 user100 47 Aug 30 23:43 f111
-rw-rw-r-- 3 user100 user100 47 Aug 30 23:43 f11_new
-rw-rw-r-- 1 user100 user100 47 Aug 30 23:44 f1_new
lrwxrwxrwx 1 user100 user100 8 Aug 30 23:30 mysc1 -> /tmp/sc1
user100@ubuntu-node1:~/Desktop$ mv d1 d1_new
user100@ubuntu-node1:~/Desktop$ ls -l
total 32
drwxrwxr-x 2 user100 user100 4096 Aug 30 23:49 d11
drwxrwxr-x 2 user100 user100 4096 Aug 30 23:24 d1_new
drwxrwxr-x 3 user100 user100 4096 Aug 30 23:48 d2
drwxrwxr-x 2 user100 user100 4096 Aug 30 23:20 d3
-rw-rw-r-- 3 user100 user100 47 Aug 30 23:43 f11
-rw-rw-r-- 3 user100 user100 47 Aug 30 23:43 f111
-rw-rw-r-- 3 user100 user100 47 Aug 30 23:43 f11_new
-rw-rw-r-- 1 user100 user100 47 Aug 30 23:44 f1_new
lrwxrwxrwx 1 user100 user100 8 Aug 30 23:30 mysc1 -> /tmp/sc1
user100@ubuntu-node1:~/Desktop$
```

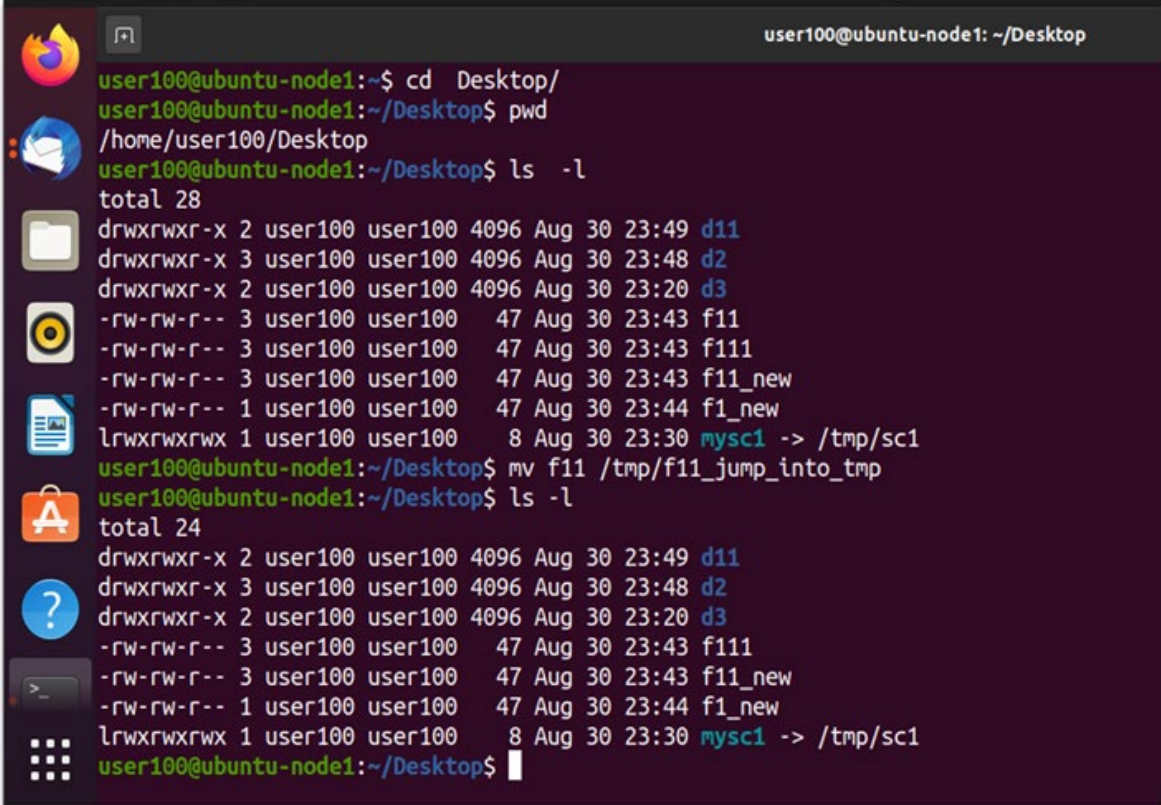
Moving Files and Directories

The **mv** command used for renaming files and directories can be used to move them to a different location.

- The only change in the syntax is that instead of file or directory name, you use the file location.
- If you compare the mv command with the cp command, you see that the mv command deletes the source location when it moves the files or directories to the destination location.

Syntax:

```
mv <source_location> <destination_location>
```



```
user100@ubuntu-node1: ~/Desktop
user100@ubuntu-node1:~$ cd Desktop/
user100@ubuntu-node1:~/Desktop$ pwd
/home/user100/Desktop
user100@ubuntu-node1:~/Desktop$ ls -l
total 28
drwxrwxr-x 2 user100 user100 4096 Aug 30 23:49 d11
drwxrwxr-x 3 user100 user100 4096 Aug 30 23:48 d2
drwxrwxr-x 2 user100 user100 4096 Aug 30 23:20 d3
-rw-rw-r-- 3 user100 user100 47 Aug 30 23:43 f11
-rw-rw-r-- 3 user100 user100 47 Aug 30 23:43 f111
-rw-rw-r-- 3 user100 user100 47 Aug 30 23:43 f11_new
-rw-rw-r-- 1 user100 user100 47 Aug 30 23:44 f1_new
lrwxrwxrwx 1 user100 user100 8 Aug 30 23:30 mysc1 -> /tmp/sc1
user100@ubuntu-node1:~/Desktop$ mv f11 /tmp/f11_jump_into_tmp
user100@ubuntu-node1:~/Desktop$ ls -l
total 24
drwxrwxr-x 2 user100 user100 4096 Aug 30 23:49 d11
drwxrwxr-x 3 user100 user100 4096 Aug 30 23:48 d2
drwxrwxr-x 2 user100 user100 4096 Aug 30 23:20 d3
-rw-rw-r-- 3 user100 user100 47 Aug 30 23:43 f111
-rw-rw-r-- 3 user100 user100 47 Aug 30 23:43 f11_new
-rw-rw-r-- 1 user100 user100 47 Aug 30 23:44 f1_new
lrwxrwxrwx 1 user100 user100 8 Aug 30 23:30 mysc1 -> /tmp/sc1
user100@ubuntu-node1:~/Desktop$
```

Moving Files and Directories

The different options available for the mv command are:

| Option | Description |
|--------|--|
| -i | Ask user's permission before performing the task |
| -f | Ask user's permission while writing protected files |
| -n | Prevent overwriting of an existing file at destination |
| -b | Backup the destination file |

Removing Files and Directories

The `rm` command which stands for remove is used to delete or remove files and directories.

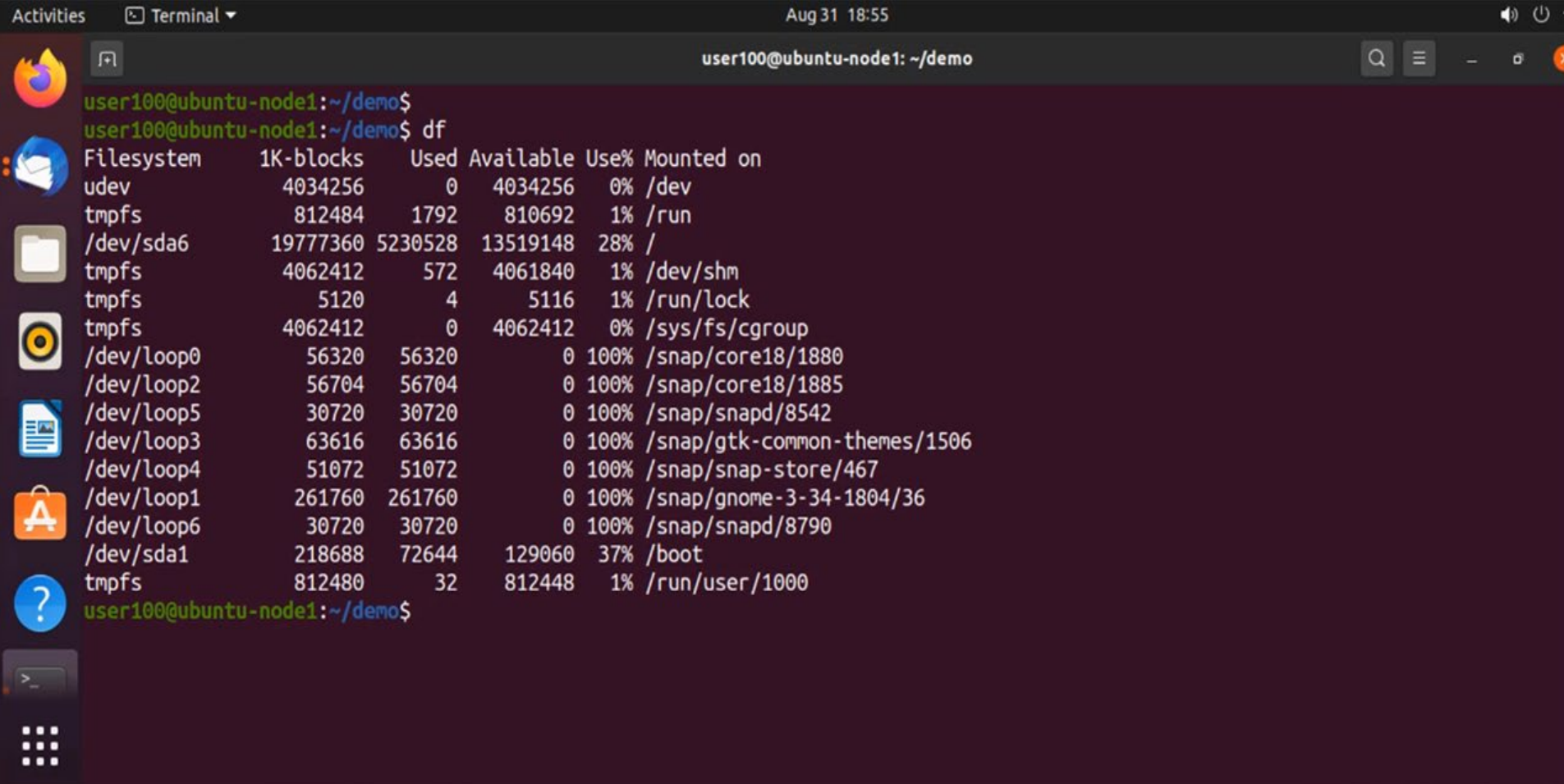
```
user100@ubuntu-node1: ~/demo
user100@ubuntu-node1:~/demo/dir1/dir2$
user100@ubuntu-node1:~/demo/dir1/dir2$
user100@ubuntu-node1:~/demo/dir1/dir2$
user100@ubuntu-node1:~/demo/dir1/dir2$
user100@ubuntu-node1:~/demo/dir1/dir2$ cd dir3
user100@ubuntu-node1:~/demo/dir1/dir2/dir3$ pwd
/home/user100/demo/dir1/dir2/dir3
user100@ubuntu-node1:~/demo/dir1/dir2/dir3$
user100@ubuntu-node1:~/demo/dir1/dir2/dir3$ cd ..
user100@ubuntu-node1:~/demo/dir1/dir2$ cd ..
user100@ubuntu-node1:~/demo/dir1$ cd ..
user100@ubuntu-node1:~/demo$ pwd
/home/user100/demo
user100@ubuntu-node1:~/demo$ ls -l
total 4
drwxrwxr-x 3 user100 user100 4096 Aug 31 18:46 dir1
-rw-rw-r-- 1 user100 user100  0 Aug 31 18:44 f1
-rw-rw-r-- 1 user100 user100  0 Aug 31 18:45 f2
-rw-rw-r-- 1 user100 user100  0 Aug 31 18:45 f3
user100@ubuntu-node1:~/demo$
user100@ubuntu-node1:~/demo$ rm f1
user100@ubuntu-node1:~/demo$
```

```
user100@ubuntu-node1: ~/demo
/home/user100/demo/dir1/dir2
user100@ubuntu-node1:~/demo/dir1/dir2$ ls -l
total 4
drwxrwxr-x 2 user100 user100 4096 Aug 31 18:46 dir3
user100@ubuntu-node1:~/demo/dir1/dir2$ rmdir dir3
user100@ubuntu-node1:~/demo/dir1/dir2$ cd ..
user100@ubuntu-node1:~/demo/dir1$ ls -l
total 4
drwxrwxr-x 2 user100 user100 4096 Aug 31 18:52 dir2
user100@ubuntu-node1:~/demo/dir1$ rmdir dir2
user100@ubuntu-node1:~/demo/dir1$ ls -l
total 0
user100@ubuntu-node1:~/demo/dir1$ cd ..
user100@ubuntu-node1:~/demo$ ls -l
total 4
drwxrwxr-x 2 user100 user100 4096 Aug 31 18:52 dir1
-rw-rw-r-- 1 user100 user100  0 Aug 31 18:45 f2
-rw-rw-r-- 1 user100 user100  0 Aug 31 18:45 f3
user100@ubuntu-node1:~/demo$ rmdir dir1
user100@ubuntu-node1:~/demo$ ls -l
total 0
-rw-rw-r-- 1 user100 user100  0 Aug 31 18:45 f2
-rw-rw-r-- 1 user100 user100  0 Aug 31 18:45 f3
user100@ubuntu-node1:~/demo$
```

A directory can be removed only after all the files and directories within it are removed.

df Command

The **df** command is used to view the file system usage and the mount point of the file system.

A screenshot of a Linux terminal window titled 'Terminal' with a dark background. The prompt is 'user100@ubuntu-node1: ~/demo\$'. The user has entered the 'df' command, and the terminal displays a table of disk space usage for various filesystems. The table has columns for Filesystem, 1K-blocks, Used, Available, Use%, and Mounted on. The filesystems listed include udev, tmpfs, /dev/sda6, /dev/shm, /run/lock, /sys/fs/cgroup, several loop devices (loop0 through loop6), /dev/sda1, and /run/user/1000. The /dev/sda6 filesystem shows 28% usage, while /dev/sda1 shows 37% usage. Most other filesystems are at 0% or 1% usage. The terminal window has a sidebar on the left with application icons and a top bar showing the date and time as 'Aug 31 18:55'.

```
user100@ubuntu-node1:~/demo$ df
Filesystem      1K-blocks    Used Available Use% Mounted on
udev            4034256      0    4034256  0% /dev
tmpfs           812484     1792    810692   1% /run
/dev/sda6       19777360 5230528 13519148  28% /
tmpfs           4062412      572    4061840   1% /dev/shm
tmpfs            5120         4        5116   1% /run/lock
tmpfs           4062412      0    4062412   0% /sys/fs/cgroup
/dev/loop0       56320     56320          0 100% /snap/core18/1880
/dev/loop2       56704     56704          0 100% /snap/core18/1885
/dev/loop5       30720     30720          0 100% /snap/snapd/8542
/dev/loop3       63616     63616          0 100% /snap/gtk-common-themes/1506
/dev/loop4       51072     51072          0 100% /snap/snap-store/467
/dev/loop1       261760    261760          0 100% /snap/gnome-3-34-1804/36
/dev/loop6       30720     30720          0 100% /snap/snapd/8790
/dev/sda1        218688     72644    129060   37% /boot
tmpfs           812480        32    812448   1% /run/user/1000
user100@ubuntu-node1:~/demo$
```

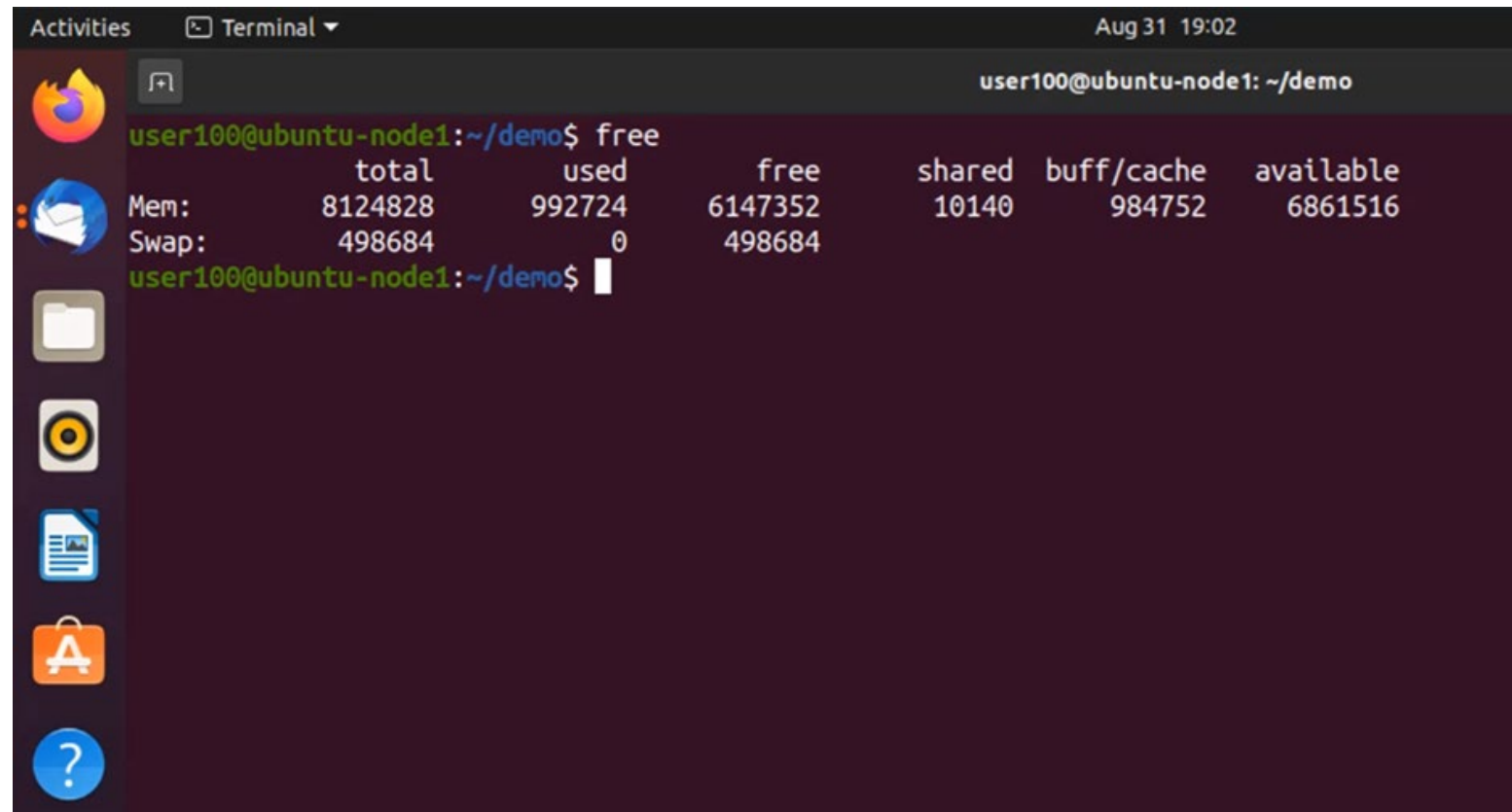
df Command

The different options available for the df command are:

| Option | Description |
|--------|---|
| -a | Display the information about the file systems |
| -h | Display the same file system information in a human readable format |
| -i | List the information of inode |

free Command

The **free** command is used to display the information of memory used by the system.

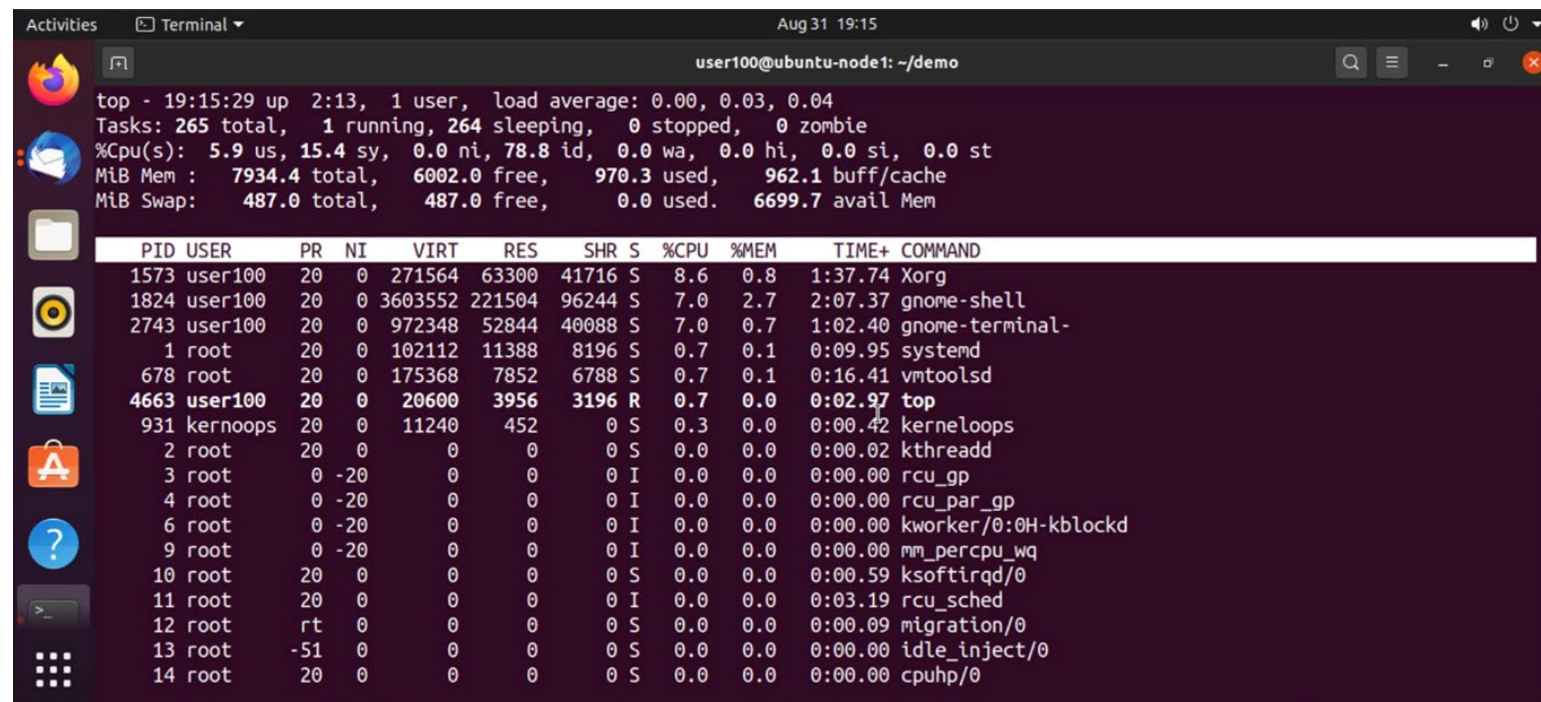
A terminal window titled 'Terminal' with a date and time of 'Aug 31 19:02'. The prompt is 'user100@ubuntu-node1: ~/demo'. The command 'free' has been executed, displaying a table of memory statistics. The table has columns: Mem, Swap, total, used, free, shared, buff/cache, and available. The data for Mem is: total 8124828, used 992724, free 6147352, shared 10140, buff/cache 984752, available 6861516. The data for Swap is: total 498684, used 0, free 498684.

```
user100@ubuntu-node1:~/demo$ free
              total        used        free      shared  buff/cache   available
Mem:      8124828      992724      6147352       10140       984752      6861516
Swap:      498684           0       498684
```

- There are two types of memories. Physical memory, represented by Mem, and swap memory.
- The information displayed by the free command includes: total, used, free, shared, buff/cache, and available.

top Command

The **top** command is used to display the information of processes running in Linux.



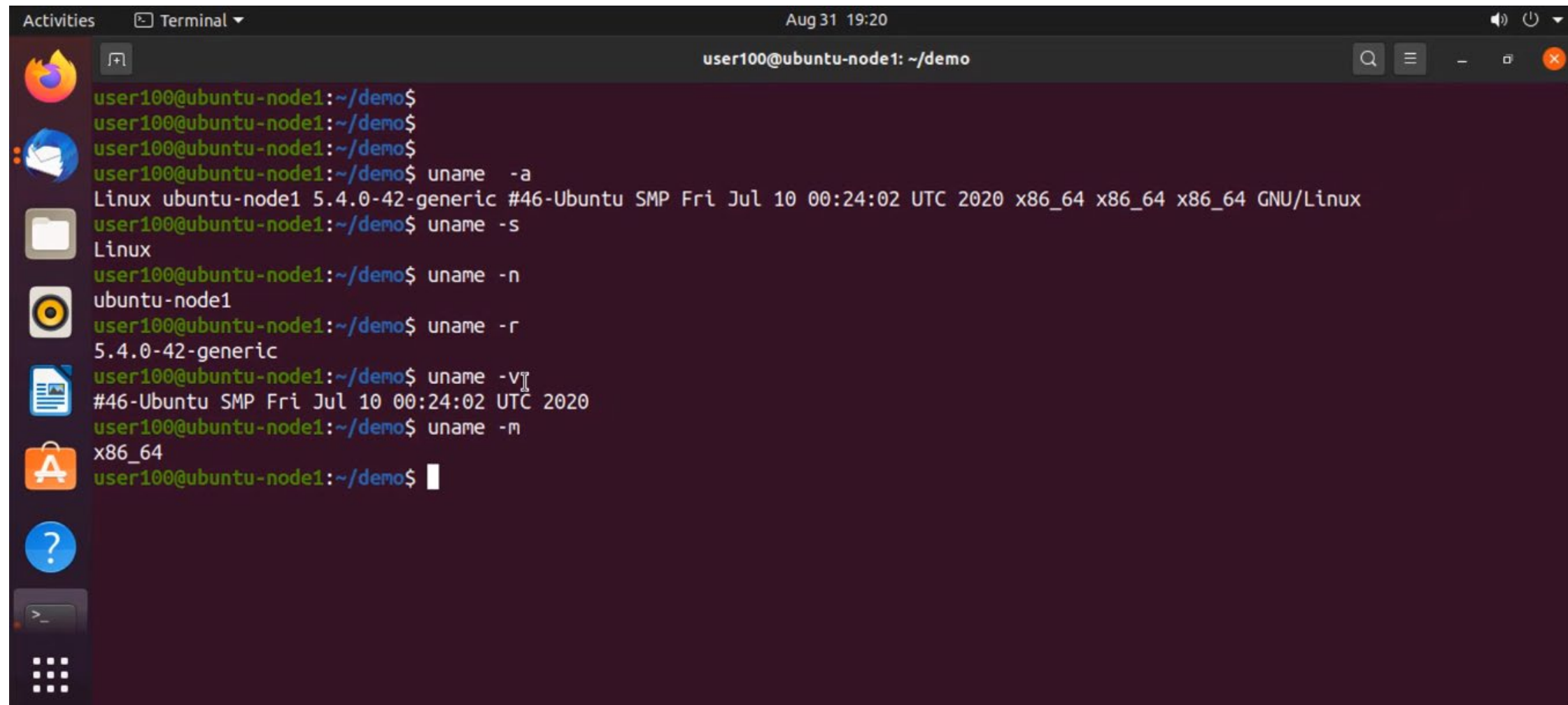
```
top - 19:15:29 up 2:13, 1 user, load average: 0.00, 0.03, 0.04
Tasks: 265 total, 1 running, 264 sleeping, 0 stopped, 0 zombie
%Cpu(s): 5.9 us, 15.4 sy, 0.0 ni, 78.8 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 7934.4 total, 6002.0 free, 970.3 used, 962.1 buff/cache
MiB Swap: 487.0 total, 487.0 free, 0.0 used. 6699.7 avail Mem

  PID USER      PR  NI   VIRT   RES   SHR  S  %CPU  %MEM    TIME+  COMMAND
 1573 user100   20   0  271564 63300 41716 S   8.6   0.8   1:37.74 Xorg
 1824 user100   20   0 3603552 221504 96244 S   7.0   2.7   2:07.37 gnome-shell
 2743 user100   20   0  972348 52844 40088 S   7.0   0.7   1:02.40 gnome-terminal-
    1 root      20   0  102112 11388  8196 S   0.7   0.1   0:09.95 systemd
   678 root      20   0  175368  7852  6788 S   0.7   0.1   0:16.41 vmtoolsd
  4663 user100   20   0   20600  3956  3196 R   0.7   0.0   0:02.97 top
   931 kernoops   20   0   11240    452    0 S   0.3   0.0   0:00.42 kerneloops
    2 root      20   0      0      0    0 S   0.0   0.0   0:00.02 kthreadd
    3 root      0 -20      0      0    0 I   0.0   0.0   0:00.00 rcu_gp
    4 root      0 -20      0      0    0 I   0.0   0.0   0:00.00 rcu_par_gp
    6 root      0 -20      0      0    0 I   0.0   0.0   0:00.00 kworker/0:0H-kblockd
    9 root      0 -20      0      0    0 I   0.0   0.0   0:00.00 mm_percpu_wq
   10 root      20   0      0      0    0 S   0.0   0.0   0:00.59 ksoftirqd/0
   11 root      20   0      0      0    0 I   0.0   0.0   0:03.19 rcu_sched
   12 root      rt    0      0      0    0 S   0.0   0.0   0:00.09 migration/0
   13 root     -51   0      0      0    0 S   0.0   0.0   0:00.00 idle_inject/0
   14 root      20   0      0      0    0 S   0.0   0.0   0:00.00 cpuhp/0
```

- The information provided is dynamic.
- The different information displayed includes, PID, PR, VIRT, RES, SHR, USER, %CPU, TIME, NI, and %MEM.

uname Command

uname command will provide the information regarding the OS distribution, release version, CPU architecture, and kernel version.

A screenshot of a Linux terminal window titled 'user100@ubuntu-node1: ~/demo'. The window shows a series of commands and their outputs. The left sidebar of the terminal window displays various application icons. The terminal output is as follows:

```
user100@ubuntu-node1:~/demo$  
user100@ubuntu-node1:~/demo$  
user100@ubuntu-node1:~/demo$  
user100@ubuntu-node1:~/demo$ uname -a  
Linux ubuntu-node1 5.4.0-42-generic #46-Ubuntu SMP Fri Jul 10 00:24:02 UTC 2020 x86_64 x86_64 x86_64 GNU/Linux  
user100@ubuntu-node1:~/demo$ uname -s  
Linux  
user100@ubuntu-node1:~/demo$ uname -n  
ubuntu-node1  
user100@ubuntu-node1:~/demo$ uname -r  
5.4.0-42-generic  
user100@ubuntu-node1:~/demo$ uname -v  
#46-Ubuntu SMP Fri Jul 10 00:24:02 UTC 2020  
user100@ubuntu-node1:~/demo$ uname -m  
x86_64  
user100@ubuntu-node1:~/demo$
```

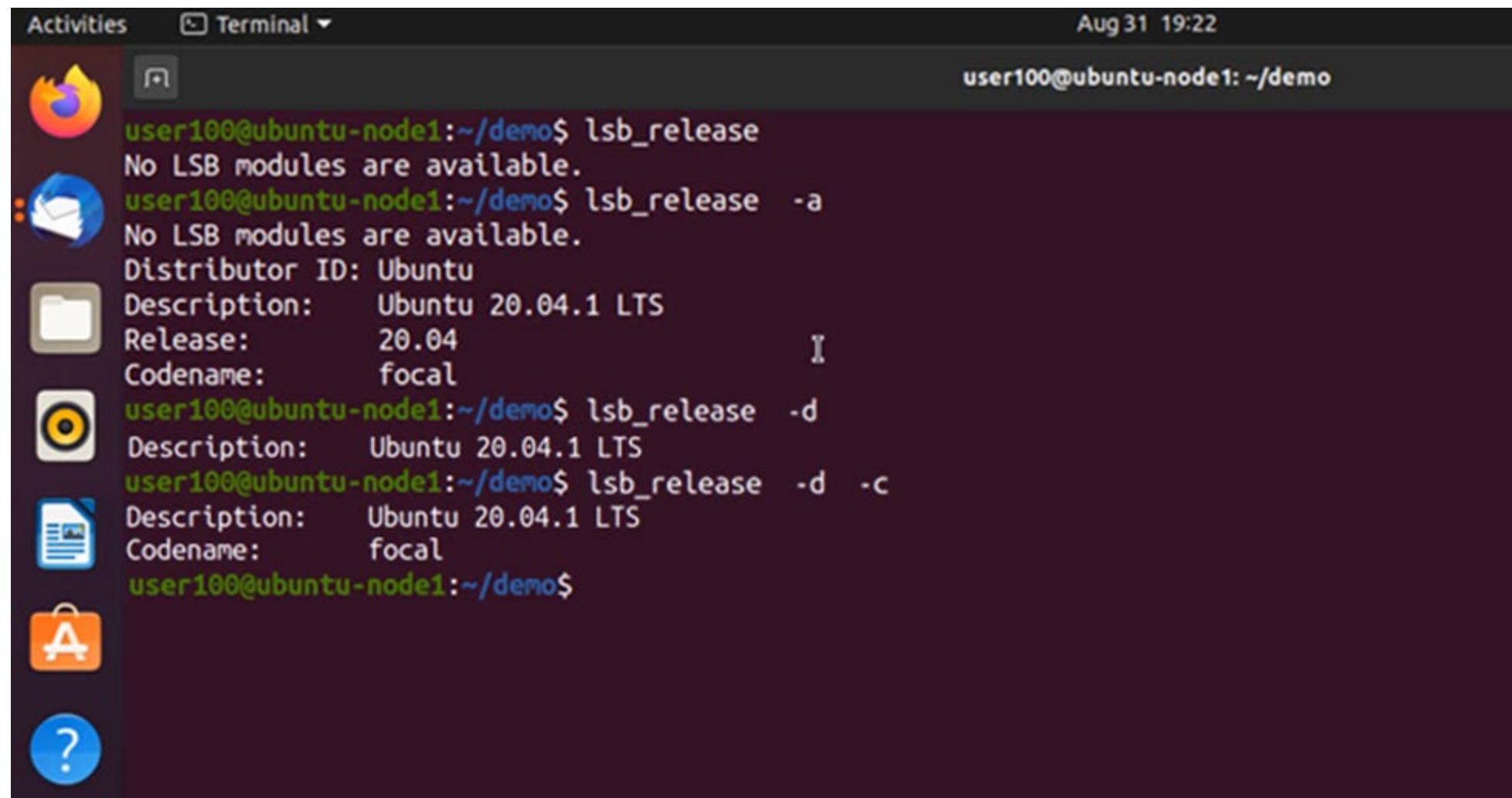
uname Command

The different options available for the uname command are:

| Option | Description |
|--------|--|
| -a | Print the system information in a specific order |
| -s | Print the kernel name |
| -n | Print the hostname |
| -r | Print the kernel version |
| -v | Print the current kernel version |
| -m | Print the machine hardware information |

lsb_release Command

The `lsb_release` command is used to display the installed Linux distributor ID, release version, and codename.

A screenshot of a Linux terminal window. The title bar shows 'Activities', 'Terminal', and the date/time 'Aug 31 19:22'. The terminal prompt is 'user100@ubuntu-node1: ~/demo'. The user enters 'lsb_release', which returns 'No LSB modules are available.'. Then the user enters 'lsb_release -a', which returns 'Distributor ID: Ubuntu', 'Description: Ubuntu 20.04.1 LTS', 'Release: 20.04', and 'Codename: focal'. Next, the user enters 'lsb_release -d', which returns 'Description: Ubuntu 20.04.1 LTS'. Finally, the user enters 'lsb_release -d -c', which returns 'Codename: focal'. The terminal window has a dark background and a sidebar on the left with icons for various applications.

```
user100@ubuntu-node1:~/demo$ lsb_release
No LSB modules are available.
user100@ubuntu-node1:~/demo$ lsb_release -a
No LSB modules are available.
Distributor ID: Ubuntu
Description:    Ubuntu 20.04.1 LTS
Release:       20.04
Codename:      focal
user100@ubuntu-node1:~/demo$ lsb_release -d
Description:    Ubuntu 20.04.1 LTS
user100@ubuntu-node1:~/demo$ lsb_release -d -c
Description:    Ubuntu 20.04.1 LTS
Codename:      focal
user100@ubuntu-node1:~/demo$
```


lsb_release Command

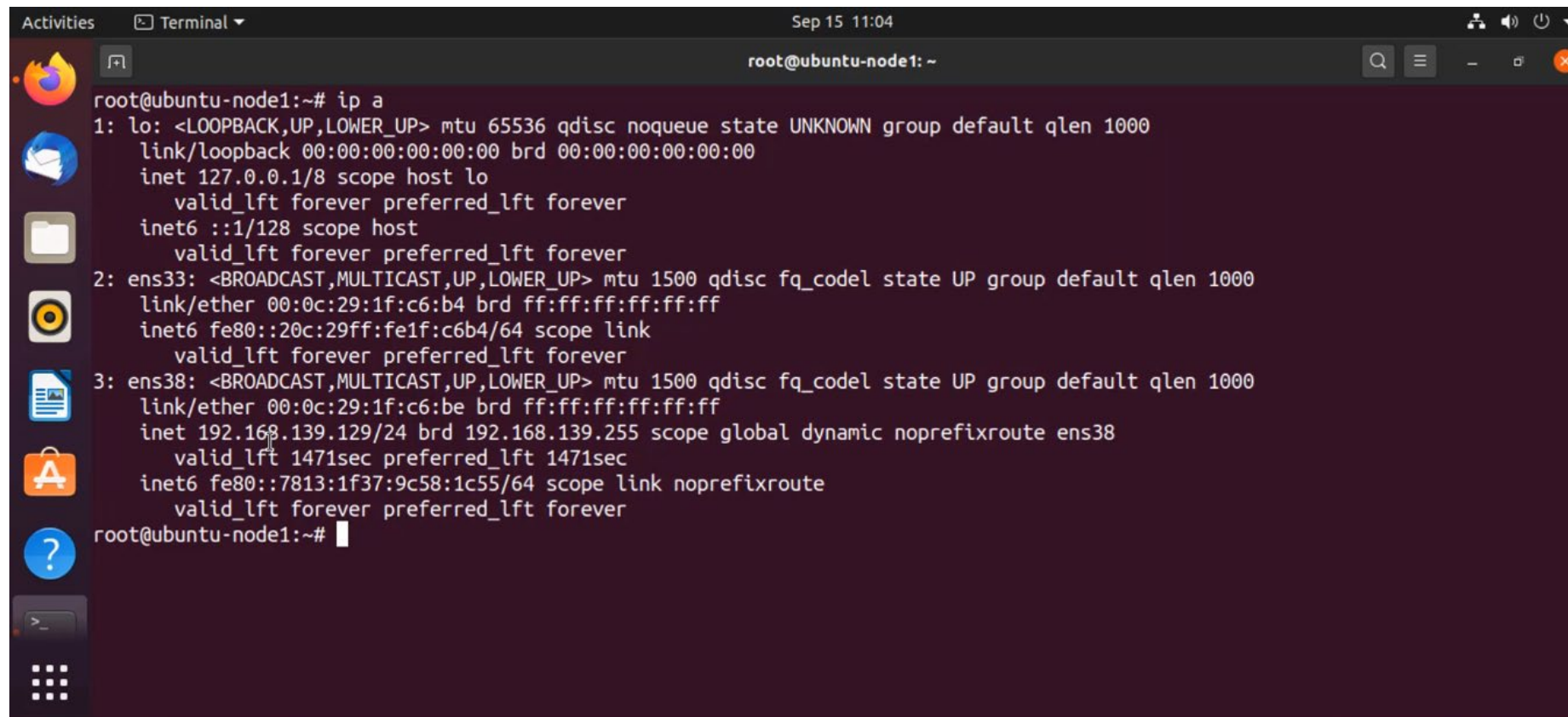
The different options available for the lsb_release command are:

| Option | Description |
|--------|--|
| -a | Print the OS distribution and releases information |
| -d | Display the description |
| -c | Display the code |



ip Command

The **ip** command is used to display the IP related details of the system.

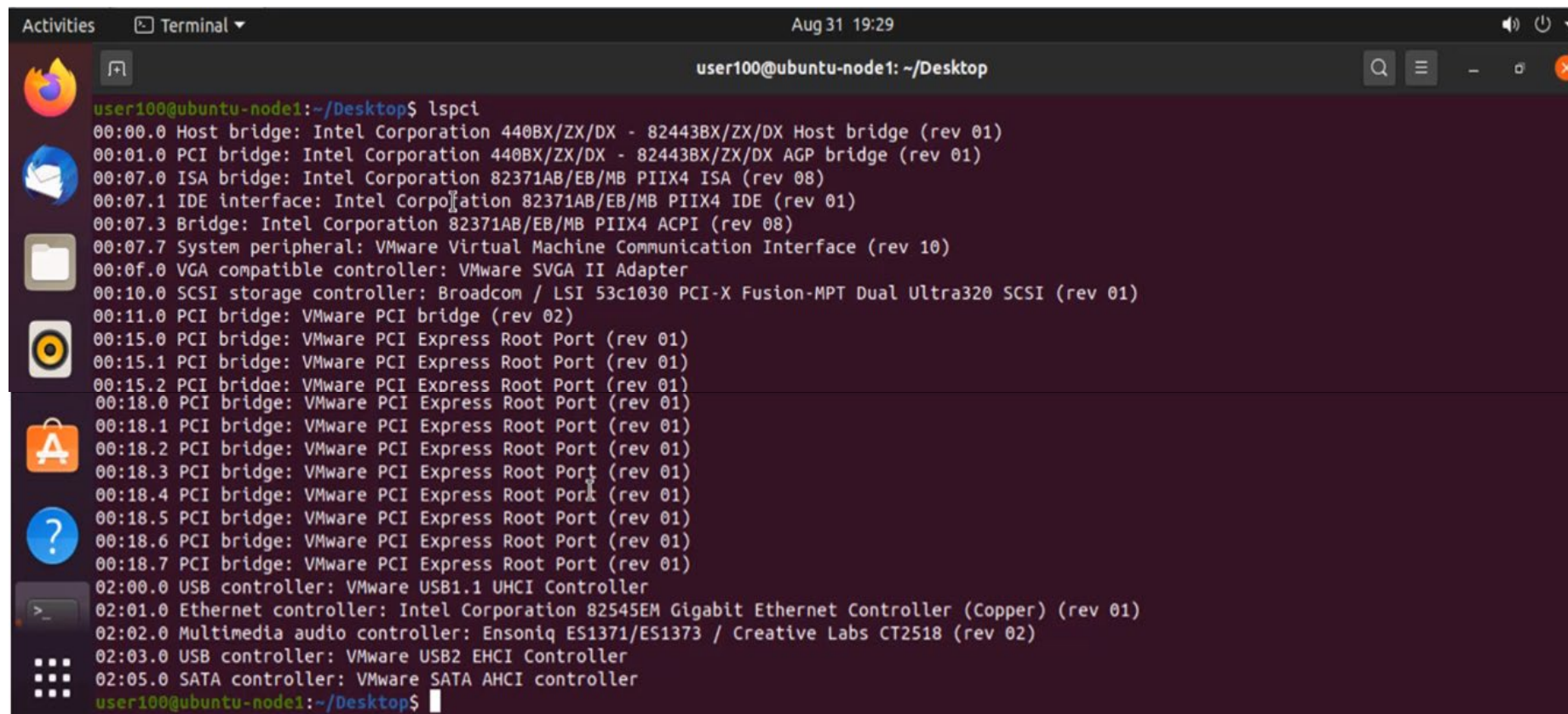
A terminal window titled 'Terminal' with a date and time of 'Sep 15 11:04'. The prompt is 'root@ubuntu-node1: ~'. The command 'ip a' has been executed, showing details for three network interfaces: 'lo' (loopback), 'ens33' (ethernet), and 'ens38' (ethernet). The output for each interface includes its state, MTU, QoS discipline, link type, MAC address, and IP address(es) with their respective scopes and lifetimes.

```
root@ubuntu-node1:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: ens33: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:1f:c6:b4 brd ff:ff:ff:ff:ff:ff
    inet6 fe80::20c:29ff:fe1f:c6b4/64 scope link
        valid_lft forever preferred_lft forever
3: ens38: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:1f:c6:be brd ff:ff:ff:ff:ff:ff
    inet 192.168.139.129/24 brd 192.168.139.255 scope global dynamic noprefixroute ens38
        valid_lft 1471sec preferred_lft 1471sec
    inet6 fe80::7813:1f37:9c58:1c55/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
root@ubuntu-node1:~#
```

The different information displayed includes interface name, ip address, network mask, and mac address.

lspci Command

The **lspci** command displays the machine hardware information based on the PCI architecture.

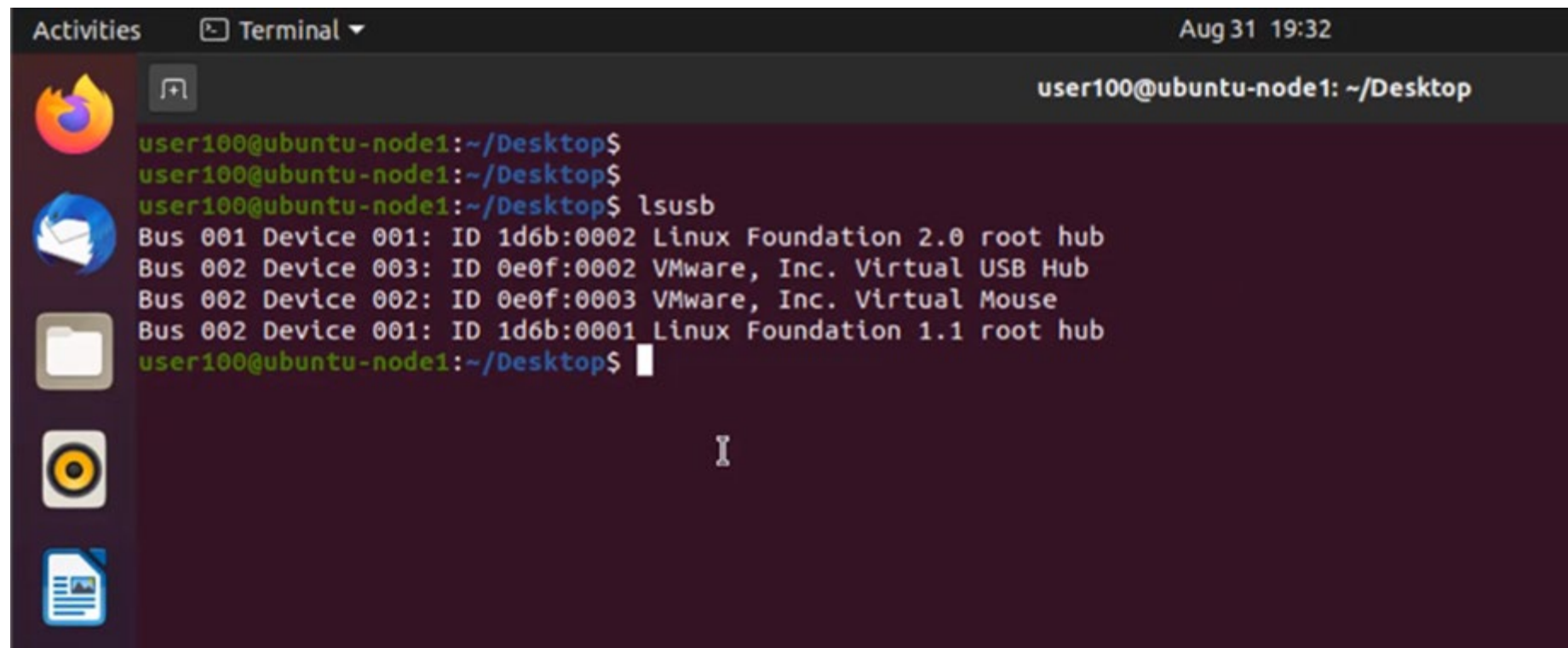


```
user100@ubuntu-node1: ~/Desktop
user100@ubuntu-node1:~/Desktop$ lspci
00:00.0 Host bridge: Intel Corporation 440BX/ZX/DX - 82443BX/ZX/DX Host bridge (rev 01)
00:01.0 PCI bridge: Intel Corporation 440BX/ZX/DX - 82443BX/ZX/DX AGP bridge (rev 01)
00:07.0 ISA bridge: Intel Corporation 82371AB/EB/MB PIIX4 ISA (rev 08)
00:07.1 IDE interface: Intel Corporation 82371AB/EB/MB PIIX4 IDE (rev 01)
00:07.3 Bridge: Intel Corporation 82371AB/EB/MB PIIX4 ACPI (rev 08)
00:07.7 System peripheral: VMware Virtual Machine Communication Interface (rev 10)
00:0f.0 VGA compatible controller: VMware SVGA II Adapter
00:10.0 SCSI storage controller: Broadcom / LSI 53c1030 PCI-X Fusion-MPT Dual Ultra320 SCSI (rev 01)
00:11.0 PCI bridge: VMware PCI bridge (rev 02)
00:15.0 PCI bridge: VMware PCI Express Root Port (rev 01)
00:15.1 PCI bridge: VMware PCI Express Root Port (rev 01)
00:15.2 PCI bridge: VMware PCI Express Root Port (rev 01)
00:18.0 PCI bridge: VMware PCI Express Root Port (rev 01)
00:18.1 PCI bridge: VMware PCI Express Root Port (rev 01)
00:18.2 PCI bridge: VMware PCI Express Root Port (rev 01)
00:18.3 PCI bridge: VMware PCI Express Root Port (rev 01)
00:18.4 PCI bridge: VMware PCI Express Root Port (rev 01)
00:18.5 PCI bridge: VMware PCI Express Root Port (rev 01)
00:18.6 PCI bridge: VMware PCI Express Root Port (rev 01)
00:18.7 PCI bridge: VMware PCI Express Root Port (rev 01)
02:00.0 USB controller: VMware USB1.1 UHCI Controller
02:01.0 Ethernet controller: Intel Corporation 82545EM Gigabit Ethernet Controller (Copper) (rev 01)
02:02.0 Multimedia audio controller: Ensoniq ES1371/ES1373 / Creative Labs CT2518 (rev 02)
02:03.0 USB controller: VMware USB2 EHCI Controller
02:05.0 SATA controller: VMware SATA AHCI controller
user100@ubuntu-node1:~/Desktop$
```

The information includes PCI bridge, VGA controller, Ethernet controller, USB controller, audio device, and IDE interface.

lsusb Command

lsusb command is used to display the information of various USB devices connected to the system.

A screenshot of a Linux terminal window. The title bar shows 'Activities', 'Terminal', and the date/time 'Aug 31 19:32'. The terminal prompt is 'user100@ubuntu-node1: ~/Desktop'. The user has entered the command 'lsusb'. The output shows four lines of USB device information: 'Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub', 'Bus 002 Device 003: ID 0e0f:0002 VMware, Inc. Virtual USB Hub', 'Bus 002 Device 002: ID 0e0f:0003 VMware, Inc. Virtual Mouse', and 'Bus 002 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub'. The prompt is now 'user100@ubuntu-node1:~/Desktop\$' with a cursor.

```
user100@ubuntu-node1:~/Desktop$  
user100@ubuntu-node1:~/Desktop$  
user100@ubuntu-node1:~/Desktop$ lsusb  
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub  
Bus 002 Device 003: ID 0e0f:0002 VMware, Inc. Virtual USB Hub  
Bus 002 Device 002: ID 0e0f:0003 VMware, Inc. Virtual Mouse  
Bus 002 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub  
user100@ubuntu-node1:~/Desktop$
```



Key Takeaways

- The sudo command in Linux elevates the privilege of a normal user to perform certain restricted tasks as a normal user.
- File system can be defined as a method of organizing and managing data in the form of files and directories.
- Soft links help you refer to a file with a long name with a shorter version of the name.
- mkdir command is used to create directories in Linux.





Knowledge Check

Knowledge Check

1

Which command is used to view the information of running tasks or processes?

- A. df
- B. mv
- C. top
- D. rm



Knowledge
Check

1

Which command is used to view the information of running tasks or processes?

- A. df
- B. mv
- C. top
- D. rm



The correct answer is **C**

The top command is used to view the information of running tasks or processes.

Knowledge
Check

2

Which command is used to display the manual pages of a command?

- A. man
- B. manual
- C. help
- D. show



Knowledge
Check

2

Which command is used to display the manual pages of a command?

- A. man
- B. manual
- C. help
- D. show



The correct answer is **A**

The man command is used to display the manual pages of a command.

Knowledge Check

3

Which command is used to view the Linux release version?

- A. cp
- B. help cp
- C. man cp
- D. lsb_release



Knowledge
Check

3

Which command is used to view the Linux release version?

- A. cp
- B. help cp
- C. man cp
- D. lsb_release



The correct answer is **D**

The lsb_release command is used to view the Linux release version.

Knowledge
Check

4

Which command is used to know the mount point of a file system?

- A. df
- B. dd
- C. cd
- D. mv



Knowledge
Check

4

Which command is used to know the mount point of a file system?

- A. df
- B. dd
- C. cd
- D. mv



The correct answer is **A**

The df command is used to view the mount point information of a file system.