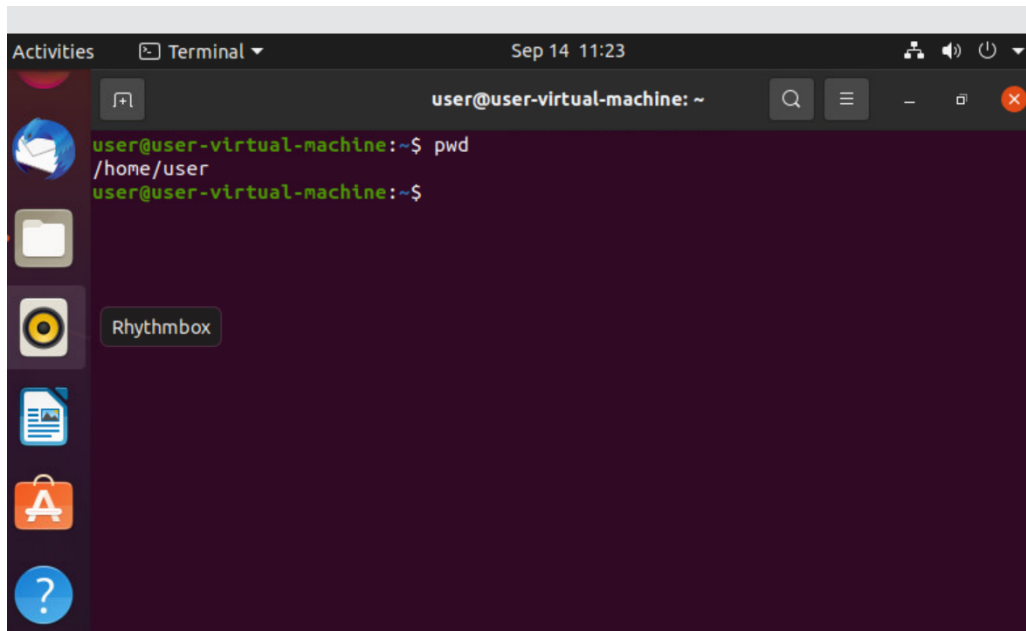


1 pwd -

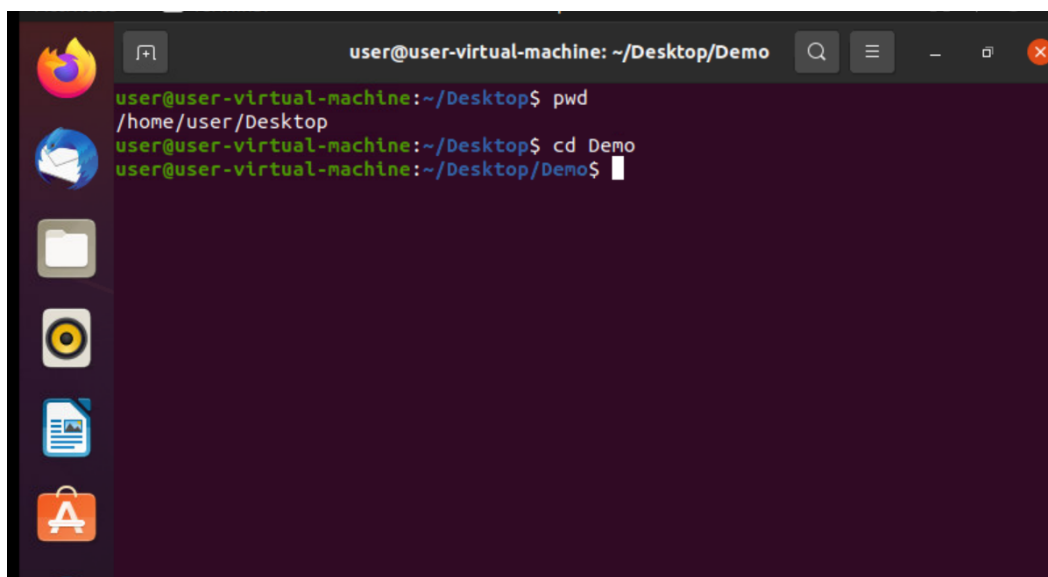
A terminal window titled "Terminal" with a date and time of "Sep 14 11:23". The window shows a user at a virtual machine prompt. The user enters the command "pwd", and the terminal outputs "/home/user". The prompt then returns to the user. The terminal window has a dark background and a sidebar with application icons on the left.

```
user@user-virtual-machine: ~  
user@user-virtual-machine:~$ pwd  
/home/user  
user@user-virtual-machine:~$
```

pwd stands for **P**rint **W**orking **D**irectory. It prints the path of the working directory, starting from the root.

pwd is shell built-in command(pwd) or an actual binary(/bin/pwd).

2 cd -

A terminal window titled "Terminal" with a date and time of "Sep 14 11:23". The window shows a user at a virtual machine prompt. The user enters the command "cd Desktop", and the terminal outputs the new path. The user then enters "pwd", and the terminal outputs the full path "/home/user/Desktop". The prompt then returns to the user. The terminal window has a dark background and a sidebar with application icons on the left.

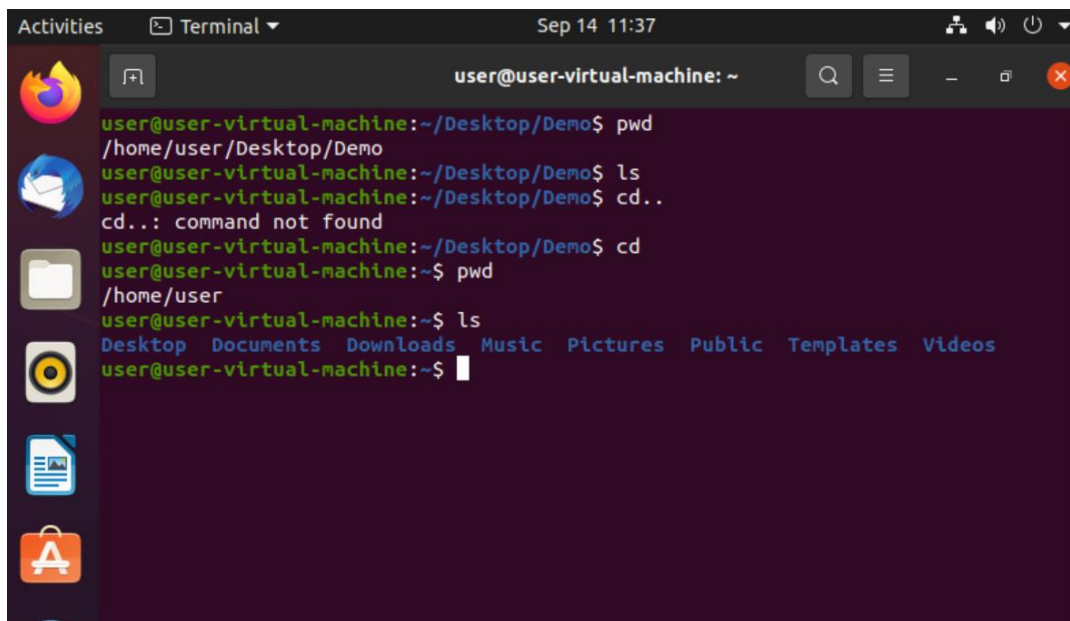
```
user@user-virtual-machine: ~/Desktop/Demo  
user@user-virtual-machine:~/Desktop$ pwd  
/home/user/Desktop  
user@user-virtual-machine:~/Desktop$ cd Demo  
user@user-virtual-machine:~/Desktop/Demo$
```

cd command in linux known as change directory command. It is used to change current working directory.

Syntax:

\$ cd [directory]

3 ls -

A screenshot of a Linux terminal window. The window title is "Terminal" and the date/time is "Sep 14 11:37". The terminal shows a user at a virtual machine prompt. The user navigates to ~/Desktop/Demo, runs 'pwd' (output: /home/user/Desktop/Demo), runs 'ls' (output: empty), runs 'cd..' (output: command not found), runs 'cd' (output: /home/user), and finally runs 'ls' (output: Desktop Documents Downloads Music Pictures Public Templates Videos).

```
user@user-virtual-machine: ~  
user@user-virtual-machine:~/Desktop/Demo$ pwd  
/home/user/Desktop/Demo  
user@user-virtual-machine:~/Desktop/Demo$ ls  
user@user-virtual-machine:~/Desktop/Demo$ cd..  
cd..: command not found  
user@user-virtual-machine:~/Desktop/Demo$ cd  
user@user-virtual-machine:~$ pwd  
/home/user  
user@user-virtual-machine:~$ ls  
Desktop Documents Downloads Music Pictures Public Templates Videos  
user@user-virtual-machine:~$
```

The **ls** is the list command in Linux. It will show the full list or content of your directory. Just type *ls* and press the enter key.

4 ls -l -

```
Activities Terminal Sep 14 11:38
user@user-virtual-machine: ~
user@user-virtual-machine:~/Desktop/Demo$ pwd
/home/user/Desktop/Demo
user@user-virtual-machine:~/Desktop/Demo$ ls
user@user-virtual-machine:~/Desktop/Demo$ cd..
cd..: command not found
user@user-virtual-machine:~/Desktop/Demo$ cd
user@user-virtual-machine:~$ pwd
/home/user
user@user-virtual-machine:~$ ls
Desktop Documents Downloads Music Pictures Public Templates Videos
user@user-virtual-machine:~$ ls -l
total 32
drwxr-xr-x 3 user user 4096 Sep 14 11:16 Desktop
drwxr-xr-x 2 user user 4096 Aug 21 17:21 Documents
drwxr-xr-x 2 user user 4096 Aug 21 17:21 Downloads
drwxr-xr-x 2 user user 4096 Aug 21 17:21 Music
drwxr-xr-x 2 user user 4096 Sep 14 11:37 Pictures
drwxr-xr-x 2 user user 4096 Aug 21 17:21 Public
drwxr-xr-x 2 user user 4096 Aug 21 17:21 Templates
drwxr-xr-x 2 user user 4096 Aug 21 17:21 Videos
user@user-virtual-machine:~$
```

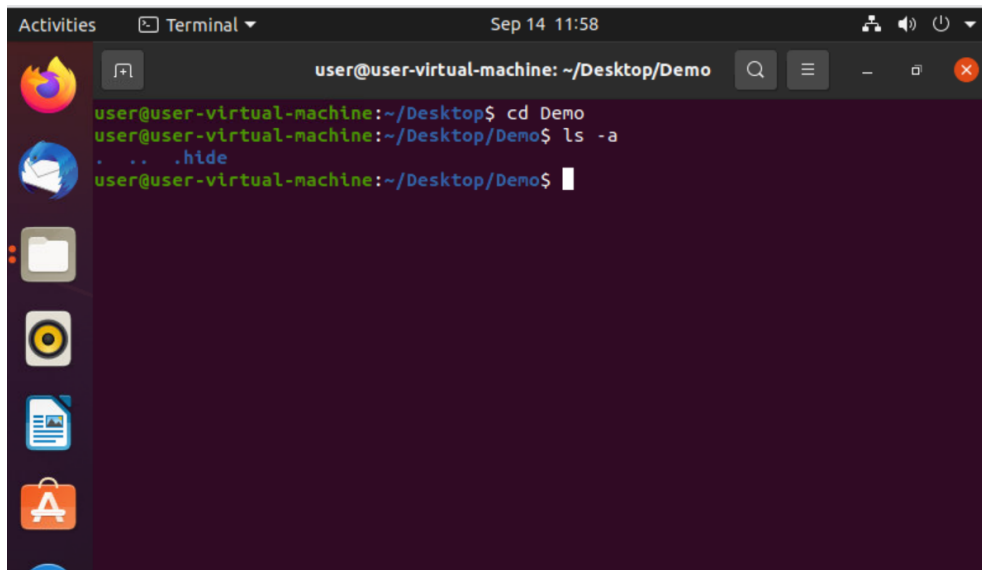
To show long listing information about the file/directory.

5 clear -

```
Activities Terminal Sep 14 11:52
user@user-virtual-machine: ~
user@user-virtual-machine:~/Desktop$ pwd
/home/user/Desktop
user@user-virtual-machine:~/Desktop$ ls -l
total 4
drwxrwxr-x 2 user user 4096 Sep 14 10:01 Demo
user@user-virtual-machine:~/Desktop$ cd
user@user-virtual-machine:~$ pwd
/home/user
user@user-virtual-machine:~$ ls -l
total 32
drwxr-xr-x 3 user user 4096 Sep 14 11:16 Desktop
drwxr-xr-x 2 user user 4096 Aug 21 17:21 Documents
drwxr-xr-x 2 user user 4096 Aug 21 17:21 Downloads
drwxr-xr-x 2 user user 4096 Aug 21 17:21 Music
drwxr-xr-x 2 user user 4096 Sep 14 11:37 Pictures
drwxr-xr-x 2 user user 4096 Aug 21 17:21 Public
drwxr-xr-x 2 user user 4096 Aug 21 17:21 Templates
drwxr-xr-x 2 user user 4096 Aug 21 17:21 Videos
user@user-virtual-machine:~$ clear
```

Clear entire Command line history using 'history' command As you might know, the 'history' command will display the last executed commands.

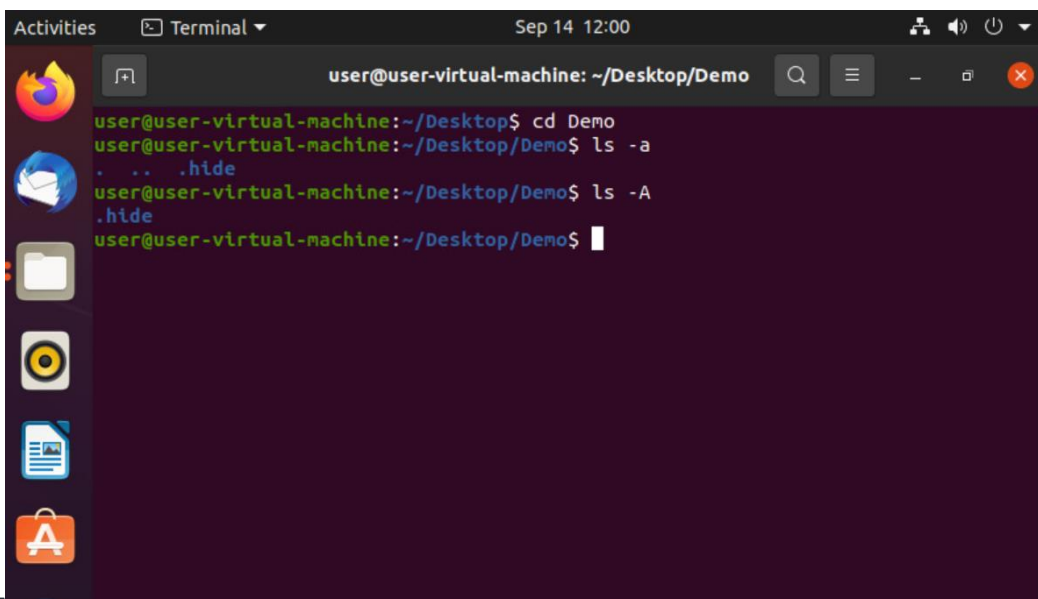
6 ls -a -

A terminal window titled 'Terminal' with a date and time of 'Sep 14 11:58'. The window shows a user at a virtual machine prompt. The user navigates to the 'Demo' directory and runs the 'ls -a' command. The output shows three items: '.', '..', and '.hide'.

```
user@user-virtual-machine: ~/Desktop/Demo
user@user-virtual-machine:~/Desktop$ cd Demo
user@user-virtual-machine:~/Desktop/Demo$ ls -a
.  ..  .hide
user@user-virtual-machine:~/Desktop/Demo$
```

To show all the hidden files in the directory, use '-a option'. Hidden files in linux starts with '.' in its file name. It will show all the files including the '.' (current directory) and '..' (parent directory).

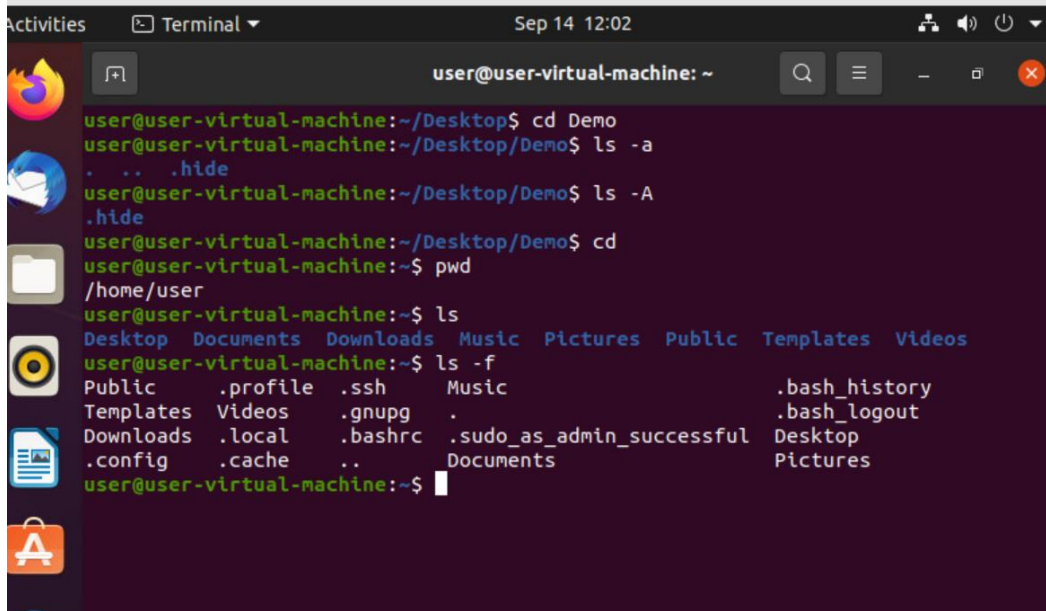
7 ls -A

A terminal window titled 'Terminal' with a date and time of 'Sep 14 12:00'. The window shows a user at a virtual machine prompt. The user navigates to the 'Demo' directory and runs the 'ls -A' command. The output shows one item: '.hide'.

```
user@user-virtual-machine: ~/Desktop/Demo
user@user-virtual-machine:~/Desktop$ cd Demo
user@user-virtual-machine:~/Desktop/Demo$ ls -a
.  ..  .hide
user@user-virtual-machine:~/Desktop/Demo$ ls -A
.hide
user@user-virtual-machine:~/Desktop/Demo$
```

To show the hidden files, but not the '.' (current directory) and '..' (parent directory).

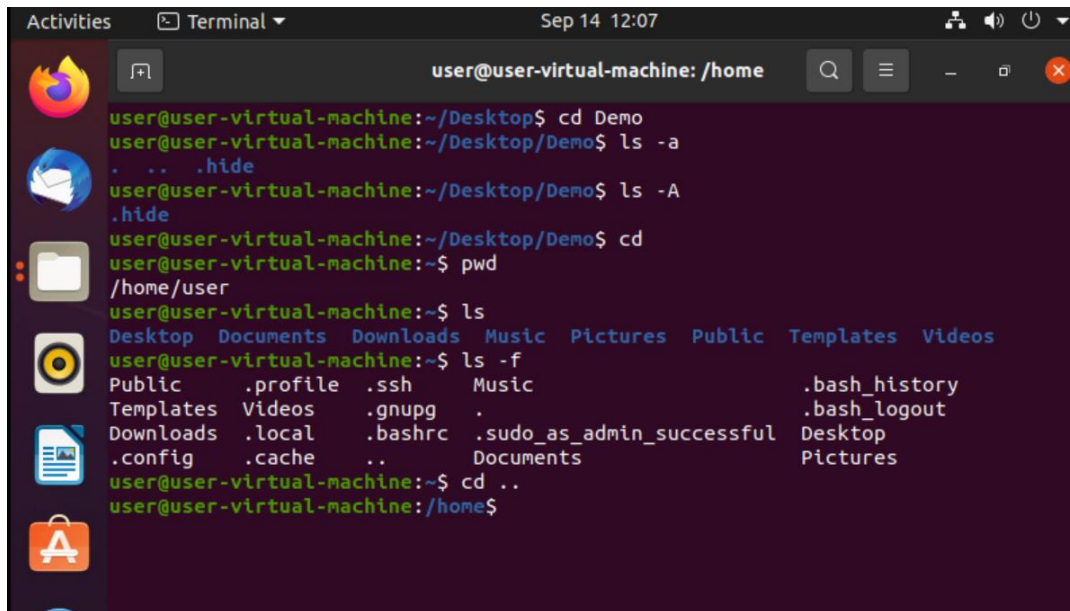
8 ls -f

A terminal window titled 'user@user-virtual-machine: ~' with a date and time of 'Sep 14 12:02'. The terminal shows a series of commands and their outputs. The user navigates to the 'Demo' directory on the Desktop, lists files with 'ls -a' (showing ., .., and .hide), then 'ls -A' (showing .hide), and finally 'cd' to enter the directory. They then use 'pwd' to show the current path as '/home/user'. Finally, they run 'ls' which lists all files and directories in a single line: Desktop, Documents, Downloads, Music, Pictures, Public, Templates, Videos, .profile, .ssh, Music, .bash_history, .bash_logout, .gnupg, ., .local, .bashrc, .sudo_as_admin_successful, Desktop, .config, .cache, .., Documents, and Pictures.

```
user@user-virtual-machine:~/Desktop$ cd Demo
user@user-virtual-machine:~/Desktop/Demo$ ls -a
.  ..  .hide
user@user-virtual-machine:~/Desktop/Demo$ ls -A
.hide
user@user-virtual-machine:~/Desktop/Demo$ cd
user@user-virtual-machine:~$ pwd
/home/user
user@user-virtual-machine:~$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos
user@user-virtual-machine:~$ ls -f
Public      .profile  .ssh      Music
Templates  Videos  .gnupg    .
Downloads  .local   .bashrc   .sudo_as_admin_successful Desktop
.config     .cache   ..        Documents  Pictures
user@user-virtual-machine:~$
```

list all the folder without format

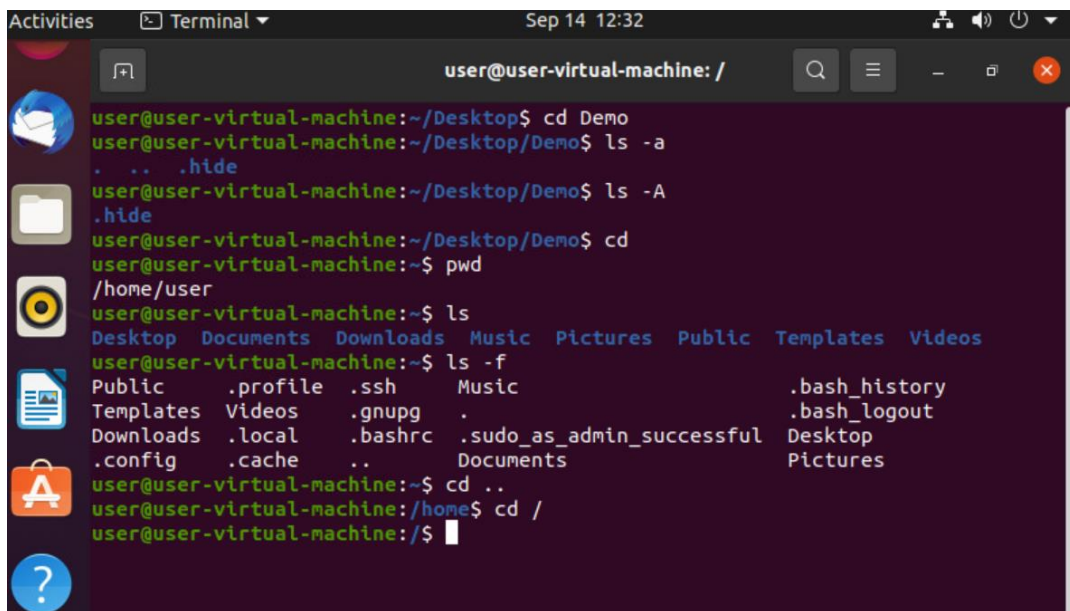
9 cd ..



```
user@user-virtual-machine: /home
user@user-virtual-machine:~/Desktop$ cd Demo
user@user-virtual-machine:~/Desktop/Demo$ ls -a
.  ..  .hide
user@user-virtual-machine:~/Desktop/Demo$ ls -A
.hide
user@user-virtual-machine:~/Desktop/Demo$ cd
user@user-virtual-machine:~$ pwd
/home/user
user@user-virtual-machine:~$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos
user@user-virtual-machine:~$ ls -f
Public      .profile  .ssh      Music      .bash_history
Templates  Videos  .gnupg    .          .bash_logout
Downloads  .local   .bashrc   .sudo_as_admin_successful  Desktop
.config    .cache   ..        Documents  Pictures
user@user-virtual-machine:~$ cd ..
user@user-virtual-machine: /home$
```

this command is used to move to the parent directory of current directory, or the directory one level up from the current directory. “..” represents parent directory.

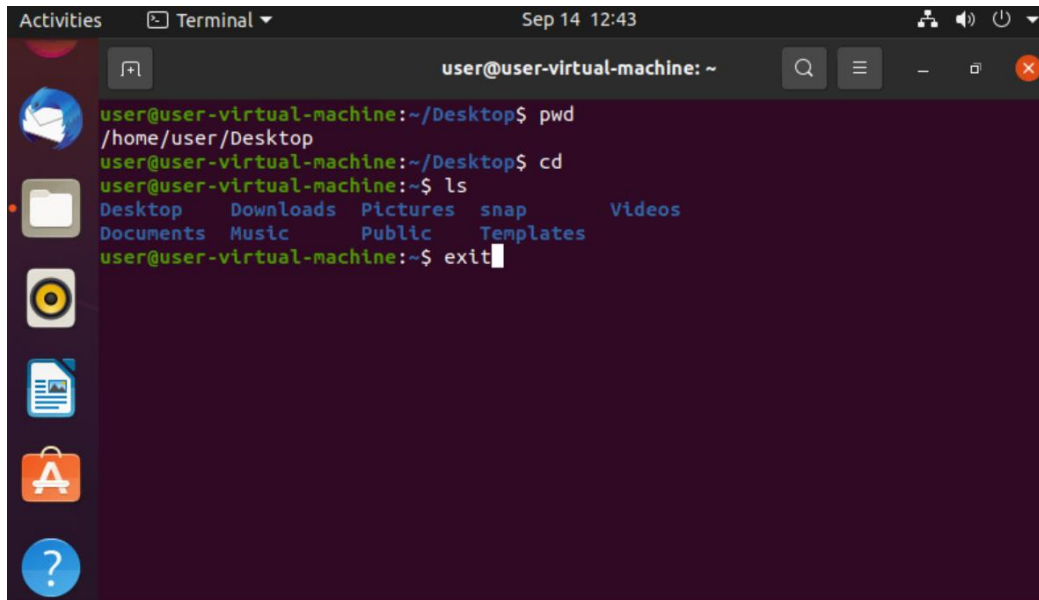
10 cd /



```
user@user-virtual-machine: /
user@user-virtual-machine:~/Desktop$ cd Demo
user@user-virtual-machine:~/Desktop/Demo$ ls -a
.  ..  .hide
user@user-virtual-machine:~/Desktop/Demo$ ls -A
.hide
user@user-virtual-machine:~/Desktop/Demo$ cd
user@user-virtual-machine:~$ pwd
/home/user
user@user-virtual-machine:~$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos
user@user-virtual-machine:~$ ls -f
Public      .profile  .ssh      Music      .bash_history
Templates  Videos  .gnupg    .          .bash_logout
Downloads  .local   .bashrc   .sudo_as_admin_successful  Desktop
.config    .cache   ..        Documents  Pictures
user@user-virtual-machine:~$ cd ..
user@user-virtual-machine: /home$ cd /
user@user-virtual-machine:/$
```

This command is used to change directory to the root directory, The root directory is the first directory in your filesystem hierarchy.

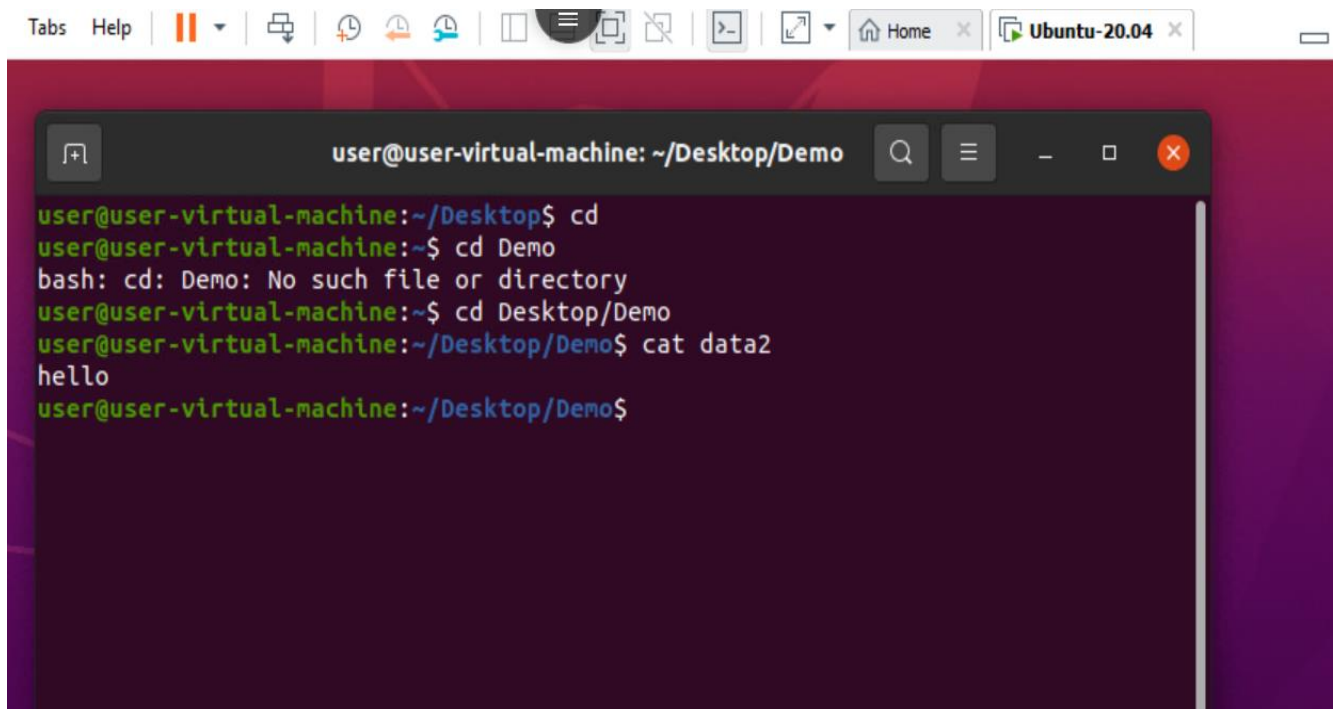
11 exit

A terminal window titled "user@user-virtual-machine: ~" with a search icon, menu icon, and window control buttons. The terminal shows the following commands and output:

```
user@user-virtual-machine:~/Desktop$ pwd
/home/user/Desktop
user@user-virtual-machine:~/Desktop$ cd
user@user-virtual-machine:~$ ls
Desktop  Downloads  Pictures  snap      Videos
Documents Music      Public   Templates
user@user-virtual-machine:~$ exit
```

The exit command lets you quit the shell where it's run.

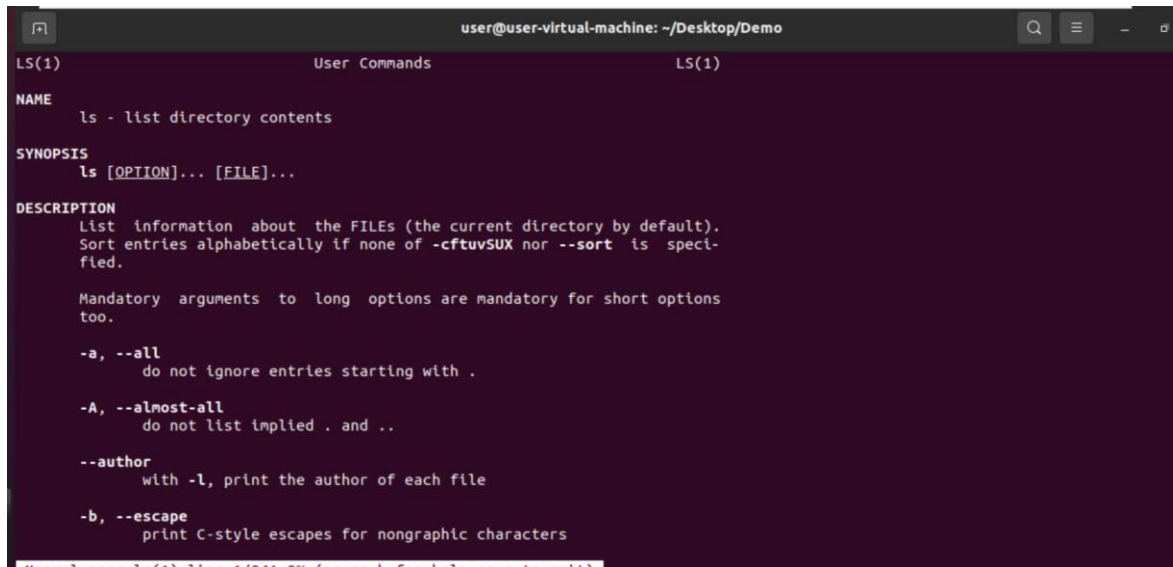
12 cat

A terminal window titled "user@user-virtual-machine: ~/Desktop/Demo" with a search icon, menu icon, and window control buttons. The terminal shows the following commands and output:

```
user@user-virtual-machine:~/Desktop$ cd
user@user-virtual-machine:~$ cd Demo
bash: cd: Demo: No such file or directory
user@user-virtual-machine:~$ cd Desktop/Demo
user@user-virtual-machine:~/Desktop/Demo$ cat data2
hello
user@user-virtual-machine:~/Desktop/Demo$
```

Cat(concatenate) command is very frequently used in Linux. It reads data from the file and gives their content as output. It helps us to create, view, concatenate files.

13 man ls

A screenshot of a terminal window titled "user@user-virtual-machine: ~/Desktop/Demo". The terminal displays the output of the "man ls" command. The output is formatted with sections: "NAME" (ls - list directory contents), "SYNOPSIS" (ls [OPTION]... [FILE]...), and "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.). Under "DESCRIPTION", several options are listed: -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), and -b, --escape (print C-style escapes for nongraphic characters).

man command in Linux is used to **display the user manual of any command** that we can run on the terminal. It provides a detailed view of the command which includes name, description, status.

14 echo


```
user@user-virtual-machine: ~/Desktop/Demo
user@user-virtual-machine:~/Desktop$ cd Demo
user@user-virtual-machine:~/Desktop/Demo$ mv Demo demo
mv: cannot stat 'Demo': No such file or directory
user@user-virtual-machine:~/Desktop/Demo$ echo

user@user-virtual-machine:~/Desktop/Demo$ echo "Torry Harris"
Torry Harris
user@user-virtual-machine:~/Desktop/Demo$
```

echo command in linux is used to display line of text/string that are passed as an argument . This is a built in command that is mostly used in shell scripts and batch files to output status text to the screen or a file.

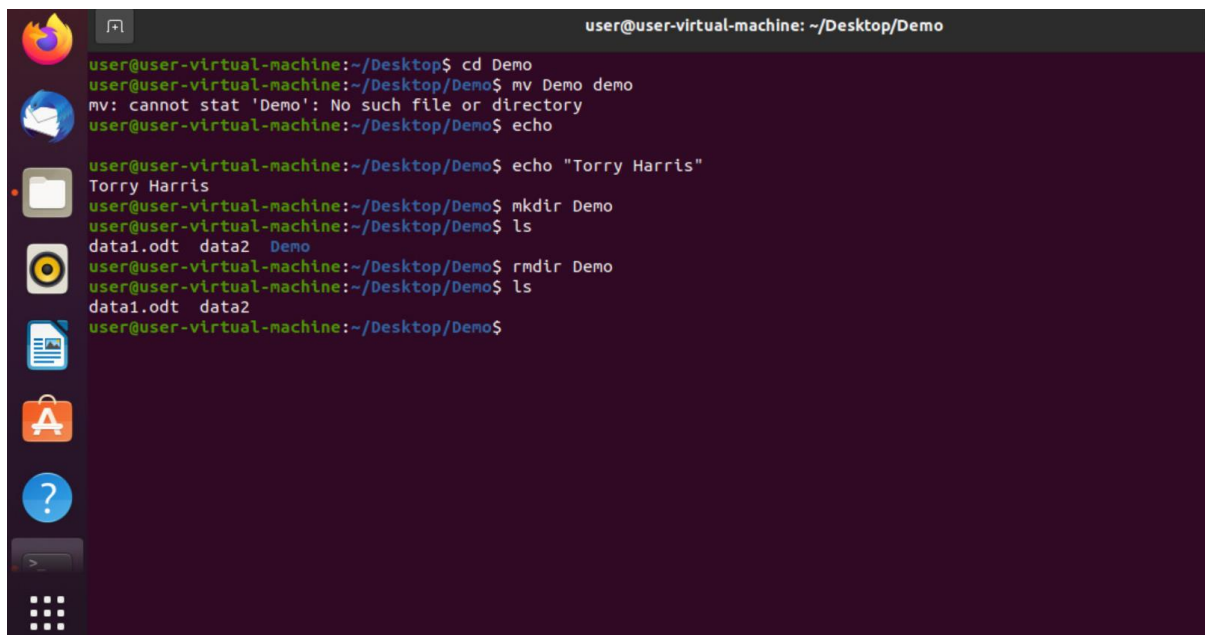
15 mkdir

```
user@user-virtual-machine: ~/Desktop/Demo
user@user-virtual-machine:~/Desktop$ cd Demo
user@user-virtual-machine:~/Desktop/Demo$ mv Demo demo
mv: cannot stat 'Demo': No such file or directory
user@user-virtual-machine:~/Desktop/Demo$ echo

user@user-virtual-machine:~/Desktop/Demo$ echo "Torry Harris"
Torry Harris
user@user-virtual-machine:~/Desktop/Demo$ mkdir Demo
user@user-virtual-machine:~/Desktop/Demo$ ls
data1.odt  data2  Demo
user@user-virtual-machine:~/Desktop/Demo$
```

mkdir command in Linux allows the user to create directories (also referred to as folders in some operating systems). This command can create multiple directories at once as well as set the permissions for the directories. It is important to note that the user executing this command must have enough permissions to create a directory in the parent directory, or he/she may receive a 'permission denied' error.

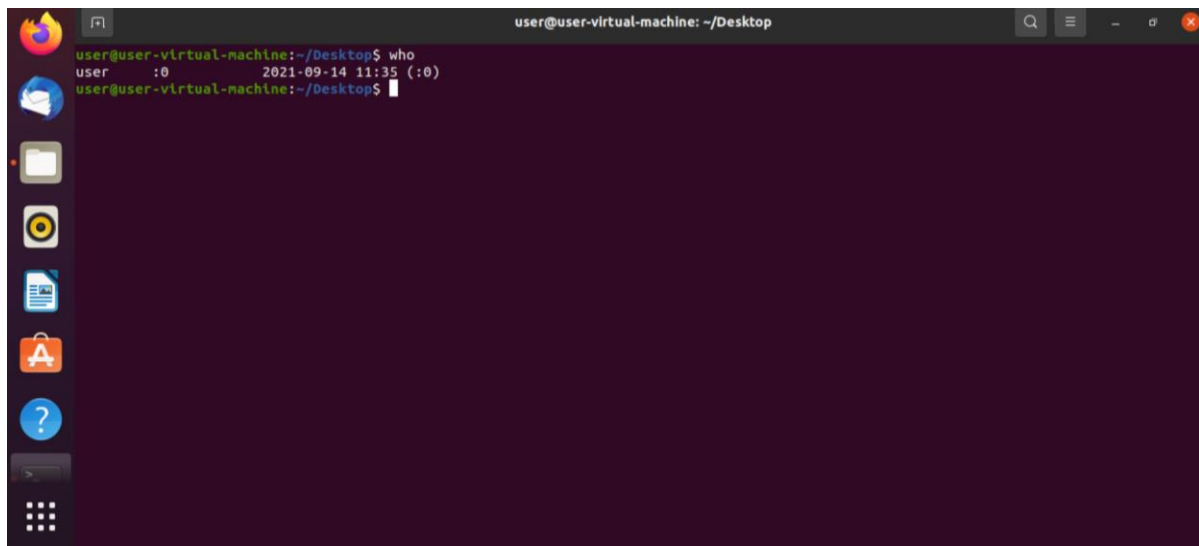
16 rmdir

A terminal window titled 'user@user-virtual-machine: ~/Desktop/Demo' showing a series of commands and their outputs. The user navigates to the 'Demo' directory, attempts to move it to 'demo' (failing with 'No such file or directory'), echoes 'Torry Harris', creates a directory named 'Demo', lists the contents (showing 'data1.odt', 'data2', and 'Demo'), removes the 'Demo' directory, and lists the contents again (showing 'data1.odt' and 'data2').

```
user@user-virtual-machine:~/Desktop$ cd Demo
user@user-virtual-machine:~/Desktop/Demo$ mv Demo demo
mv: cannot stat 'Demo': No such file or directory
user@user-virtual-machine:~/Desktop/Demo$ echo
Torry Harris
user@user-virtual-machine:~/Desktop/Demo$ mkdir Demo
user@user-virtual-machine:~/Desktop/Demo$ ls
data1.odt  data2  Demo
user@user-virtual-machine:~/Desktop/Demo$ rmdir Demo
user@user-virtual-machine:~/Desktop/Demo$ ls
data1.odt  data2
user@user-virtual-machine:~/Desktop/Demo$
```

rmdir command is used to remove empty directories from the filesystem in Linux. The **rmdir** command removes each and every directory specified in the command line only if these directories are empty. So if the specified directory has some sub-directories or files in it then this cannot be removed by the **rmdir** command.

17 who

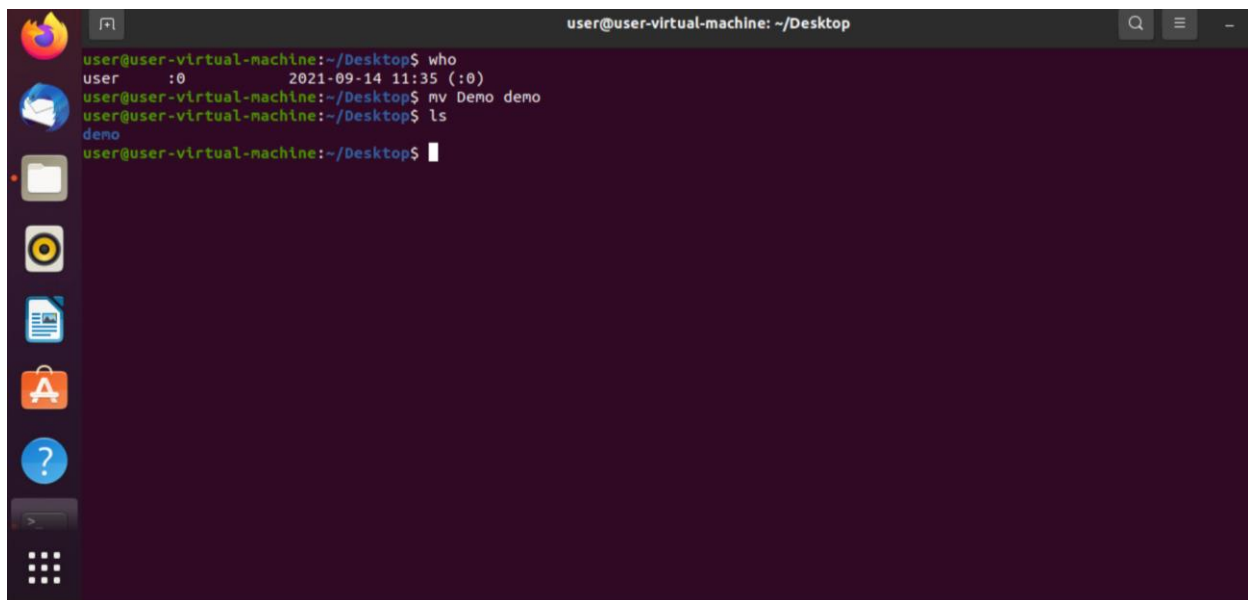
A terminal window titled 'user@user-virtual-machine: ~/Desktop' with a search icon, menu icon, and window control buttons. The terminal shows the command 'who' being executed, resulting in the output: 'user :0 2021-09-14 11:35 (:0)'. The prompt 'user@user-virtual-machine:~/Desktop\$' is visible at the end of the line.

```
user@user-virtual-machine:~/Desktop$ who
user :0 2021-09-14 11:35 (:0)
user@user-virtual-machine:~/Desktop$
```

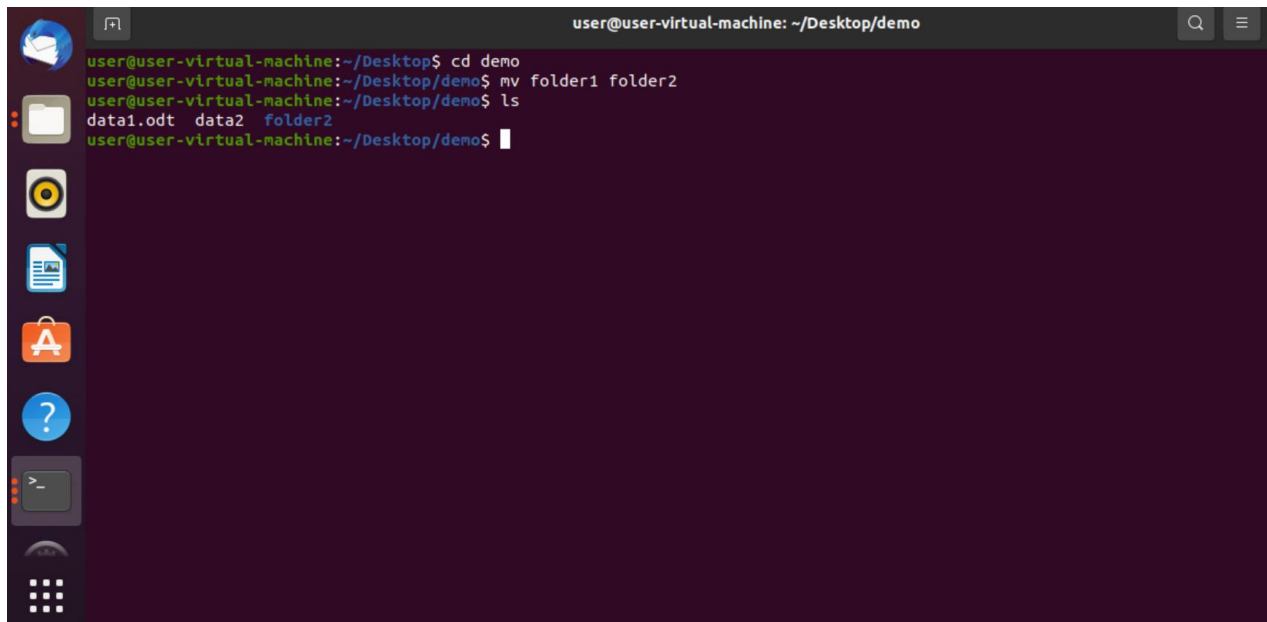
who command is used to find out the following information :

1. Time of last system boot
2. Current run level of the system
3. List of logged in users and more.

18 mv

A terminal window titled 'user@user-virtual-machine: ~/Desktop' with a search icon, menu icon, and window control buttons. The terminal shows a sequence of commands: 'who', 'mv Demo demo', and 'ls'. The output for 'who' is 'user :0 2021-09-14 11:35 (:0)'. The output for 'ls' is 'demo'. The prompt 'user@user-virtual-machine:~/Desktop\$' is visible at the end of the last line.

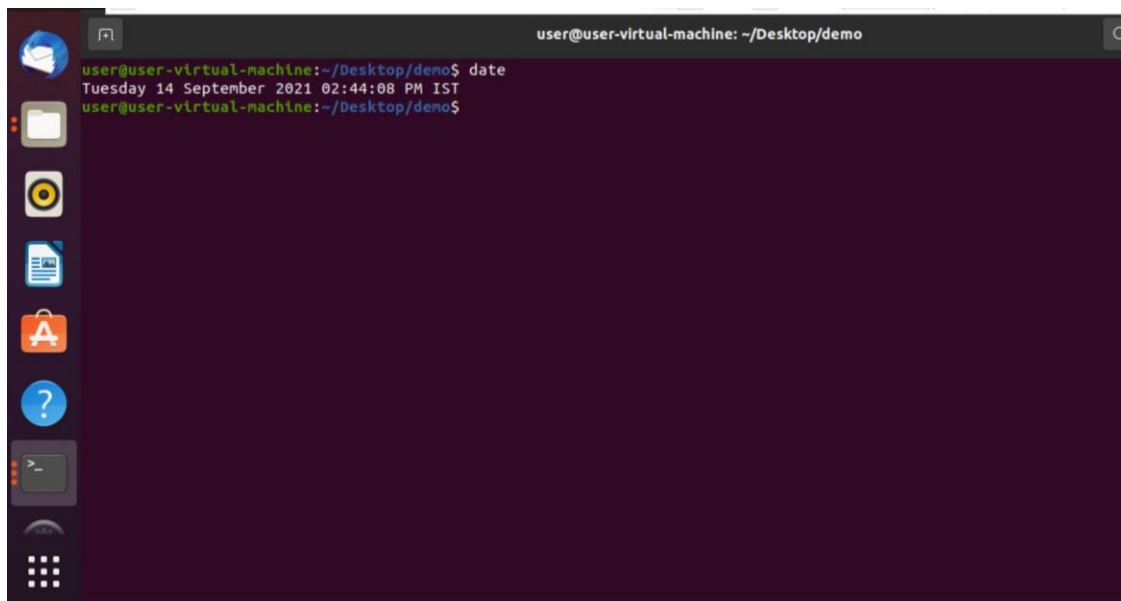
```
user@user-virtual-machine:~/Desktop$ who
user :0 2021-09-14 11:35 (:0)
user@user-virtual-machine:~/Desktop$ mv Demo demo
user@user-virtual-machine:~/Desktop$ ls
demo
user@user-virtual-machine:~/Desktop$
```

A terminal window titled 'user@user-virtual-machine: ~/Desktop/demo'. The terminal shows a sequence of commands: 'cd demo', 'mv folder1 folder2', and 'ls'. The output of 'ls' is 'data1.odt data2 folder2'. The terminal has a dark purple background and a sidebar on the left with various application icons.

```
user@user-virtual-machine:~/Desktop$ cd demo
user@user-virtual-machine:~/Desktop/demo$ mv folder1 folder2
user@user-virtual-machine:~/Desktop/demo$ ls
data1.odt data2 folder2
user@user-virtual-machine:~/Desktop/demo$
```

The mv command in Linux is used to move or rename files and directories.

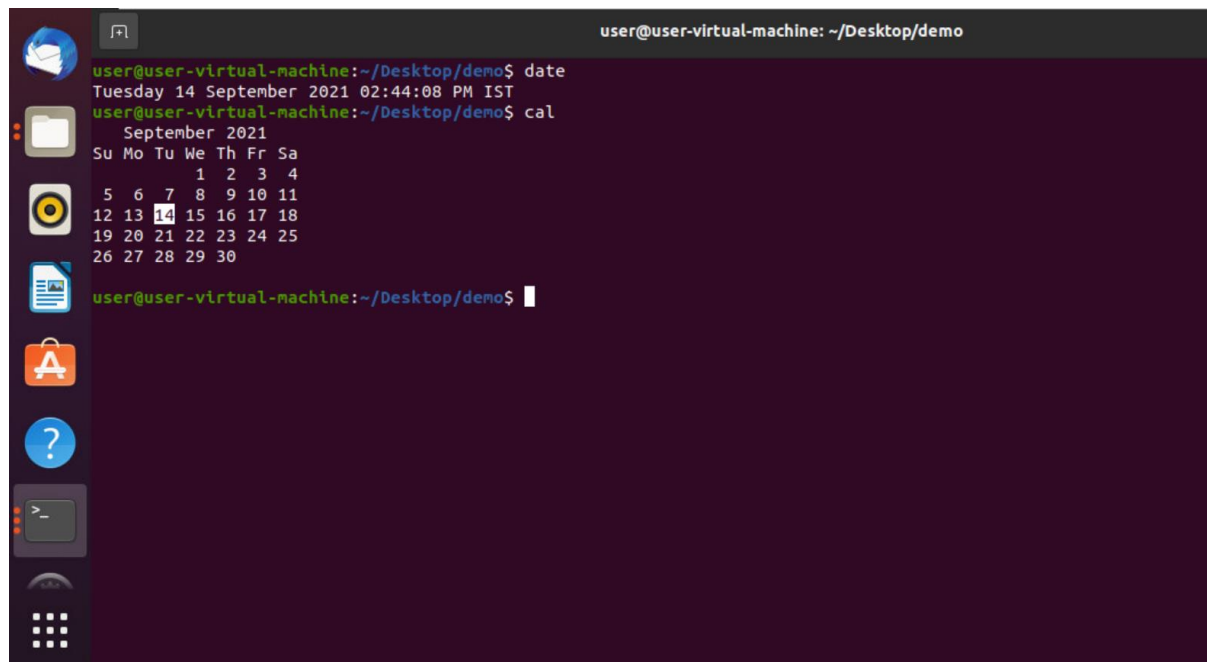
19 date

A terminal window titled 'user@user-virtual-machine: ~/Desktop/demo'. The terminal shows the command 'date' being executed, which outputs 'Tuesday 14 September 2021 02:44:08 PM IST'. The terminal has a dark purple background and a sidebar on the left with various application icons.

```
user@user-virtual-machine:~/Desktop/demo$ date
Tuesday 14 September 2021 02:44:08 PM IST
user@user-virtual-machine:~/Desktop/demo$
```

Display the current date and time in Linux

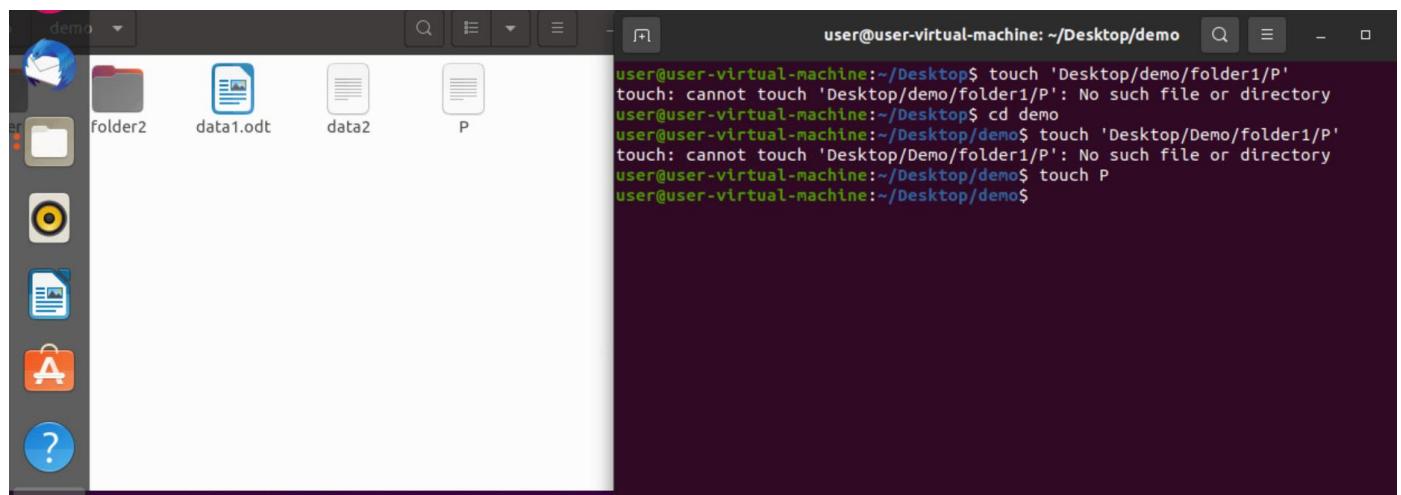
20 cal



```
user@user-virtual-machine: ~/Desktop/demo
user@user-virtual-machine:~/Desktop/demo$ date
Tuesday 14 September 2021 02:44:08 PM IST
user@user-virtual-machine:~/Desktop/demo$ cal
    September 2021
Su Mo Tu We Th Fr Sa
                1  2  3  4
 5  6  7  8  9 10 11
12 13 14 15 16 17 18
19 20 21 22 23 24 25
26 27 28 29 30
user@user-virtual-machine:~/Desktop/demo$
```

By default, the cal command **shows the current month calendar as output**. cal command is a calendar command in Linux which is used to see the calendar of a specific month or a whole year.

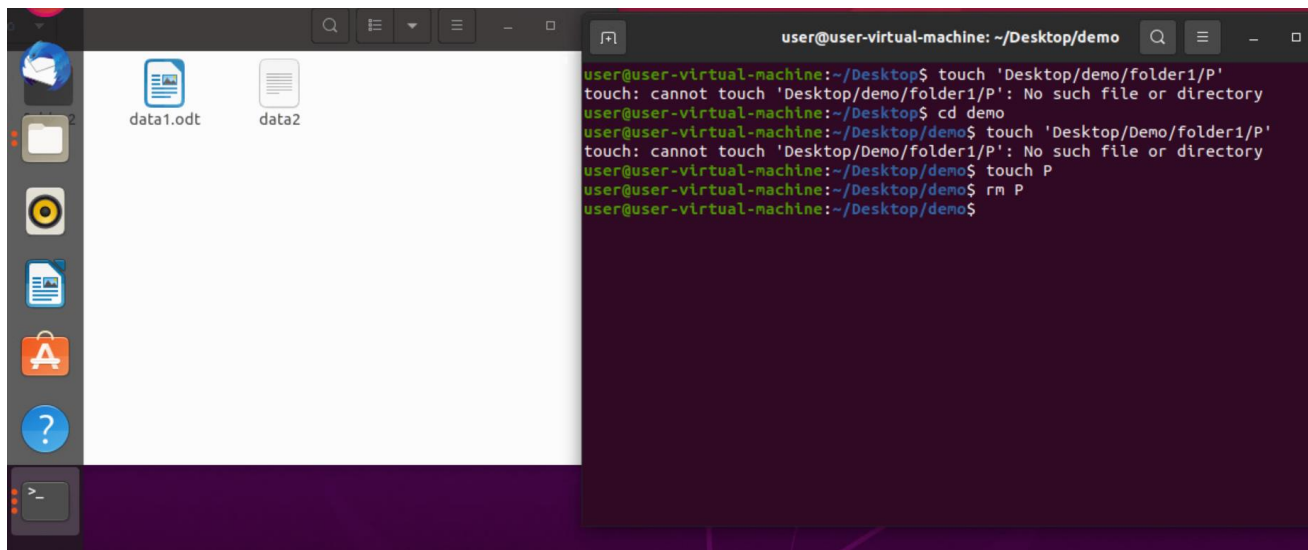
21 touch



```
user@user-virtual-machine: ~/Desktop/demo
user@user-virtual-machine:~/Desktop$ touch 'Desktop/demo/folder1/P'
touch: cannot touch 'Desktop/demo/folder1/P': No such file or directory
user@user-virtual-machine:~/Desktop$ cd demo
user@user-virtual-machine:~/Desktop/demo$ touch 'Desktop/Demo/folder1/P'
touch: cannot touch 'Desktop/Demo/folder1/P': No such file or directory
user@user-virtual-machine:~/Desktop/demo$ touch P
user@user-virtual-machine:~/Desktop/demo$
```

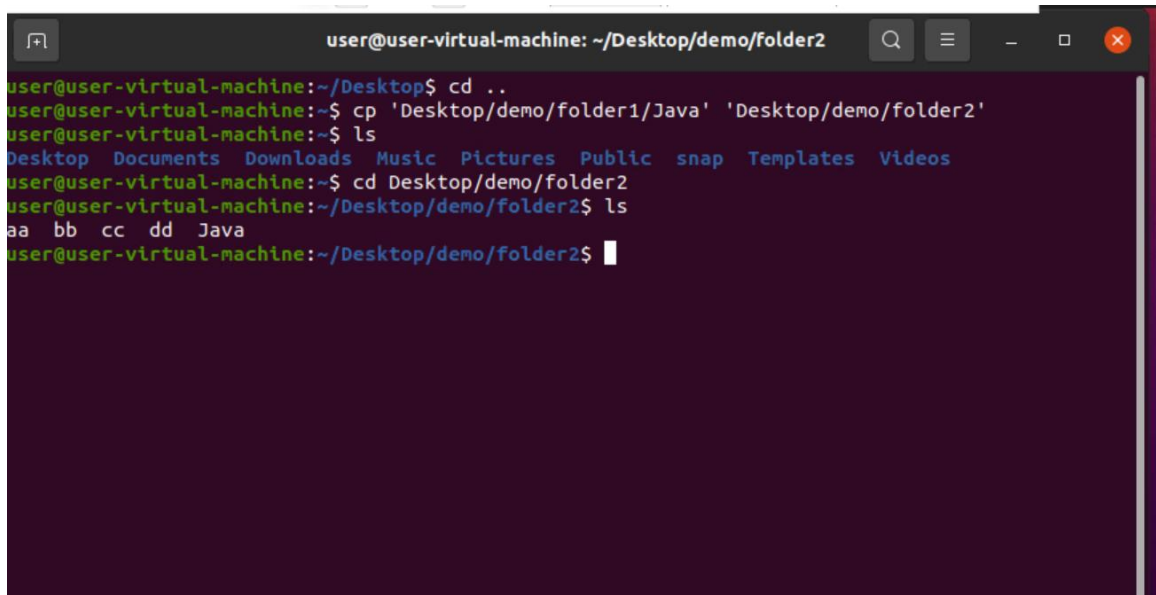
It is used to create a file without any content. The file created using touch command is empty. This command can be used when the user doesn't have data to store at the time of file creation.

22 rm



rm stands for **remove** here. rm command is used to remove objects such as files, directories, symbolic links and so on from the file system like UNIX. To be more precise, rm removes references to objects from the filesystem, where those objects might have had multiple references

23 cp

A terminal window titled 'user@user-virtual-machine: ~/Desktop/demo/folder2'. The terminal shows a sequence of commands: 'cd ..', 'cp 'Desktop/demo/folder1/Java' 'Desktop/demo/folder2'', 'ls', 'cd Desktop/demo/folder2', and 'ls'. The output of the first 'ls' command lists standard Linux desktop directories. The output of the second 'ls' command lists files 'aa', 'bb', 'cc', 'dd', and 'Java'.

```
user@user-virtual-machine: ~/Desktop/demo/folder2
user@user-virtual-machine:~/Desktop$ cd ..
user@user-virtual-machine:~$ cp 'Desktop/demo/folder1/Java' 'Desktop/demo/folder2'
user@user-virtual-machine:~$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  snap  Templates  Videos
user@user-virtual-machine:~$ cd Desktop/demo/folder2
user@user-virtual-machine:~/Desktop/demo/folder2$ ls
aa  bb  cc  dd  Java
user@user-virtual-machine:~/Desktop/demo/folder2$
```

cp stands for **copy**. This command is used to copy files or group of files or directory. It creates an exact image of a file on a disk with different file name. *cp* command require at least two filenames in its arguments.

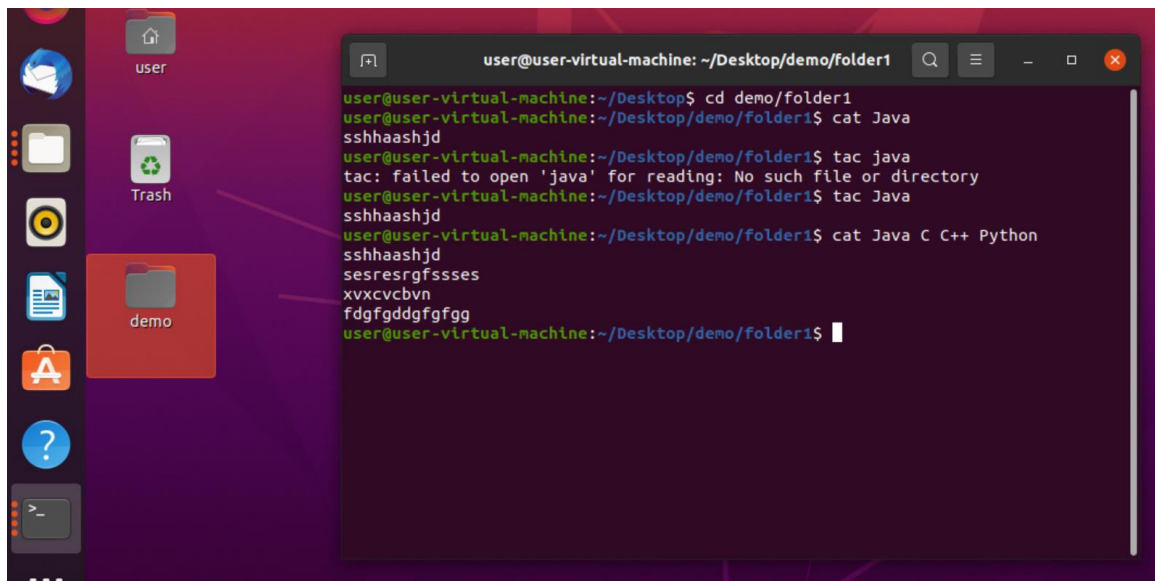
First and second syntax is used to copy Source file to Destination file or Directory. Third syntax is used to copy multiple Sources(files) to Directory.

24 cat z


```
user@user-virtual-machine: ~/Desktop/demo/folder1
user@user-virtual-machine:~/Desktop$ cd demo/folder1
user@user-virtual-machine:~/Desktop/demo/folder1$ cat Java
sshhaashjd
user@user-virtual-machine:~/Desktop/demo/folder1$
```

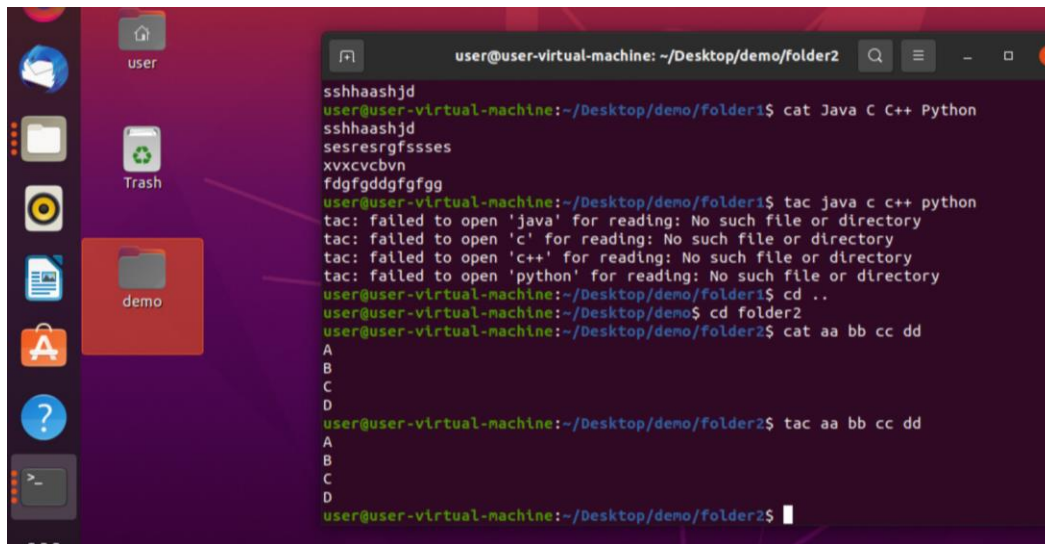
Display the data present in the document in order format.

25 cat a b c d-



```
user@user-virtual-machine: ~/Desktop/demo/folder1
user@user-virtual-machine:~/Desktop$ cd demo/folder1
user@user-virtual-machine:~/Desktop/demo/folder1$ cat Java
sshhaashjd
user@user-virtual-machine:~/Desktop/demo/folder1$ tac java
tac: failed to open 'java' for reading: No such file or directory
user@user-virtual-machine:~/Desktop/demo/folder1$ tac Java
sshhaashjd
user@user-virtual-machine:~/Desktop/demo/folder1$ cat Java C ++ Python
sshhaashjd
sesresrgfsses
xvxcvcbvn
fdgfgddgfgfgg
user@user-virtual-machine:~/Desktop/demo/folder1$
```

26 tac a b c d

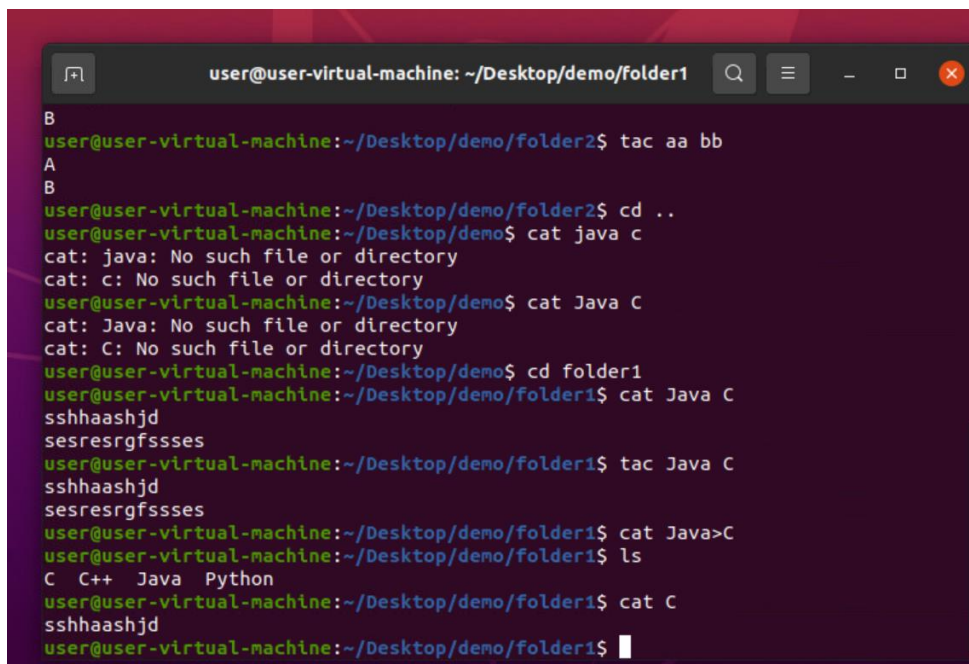


A terminal window titled 'user@user-virtual-machine: ~/Desktop/demo/folder2'. The user has created a directory 'demo' and moved into it. They then create a subdirectory 'folder2' and move into it. The terminal shows the following commands and output:

```
sshhaashjd
user@user-virtual-machine:~/Desktop/demo/folder1$ cat Java C C++ Python
sshhaashjd
sesresrgfssses
xvxcvcbvn
fdgfgddgfgfgg
user@user-virtual-machine:~/Desktop/demo/folder1$ tac java c c++ python
tac: failed to open 'java' for reading: No such file or directory
tac: failed to open 'c' for reading: No such file or directory
tac: failed to open 'c++' for reading: No such file or directory
tac: failed to open 'python' for reading: No such file or directory
user@user-virtual-machine:~/Desktop/demo/folder1$ cd ..
user@user-virtual-machine:~/Desktop/demo$ cd folder2
user@user-virtual-machine:~/Desktop/demo/folder2$ cat aa bb cc dd
A
B
C
D
user@user-virtual-machine:~/Desktop/demo/folder2$ tac aa bb cc dd
A
B
C
D
user@user-virtual-machine:~/Desktop/demo/folder2$
```

tac command in Linux is used to concatenate and print files in reverse. This command will write each FILE to standard output, the last line first. When no file is specified then this command will read the standard input.

27 cat Java>C

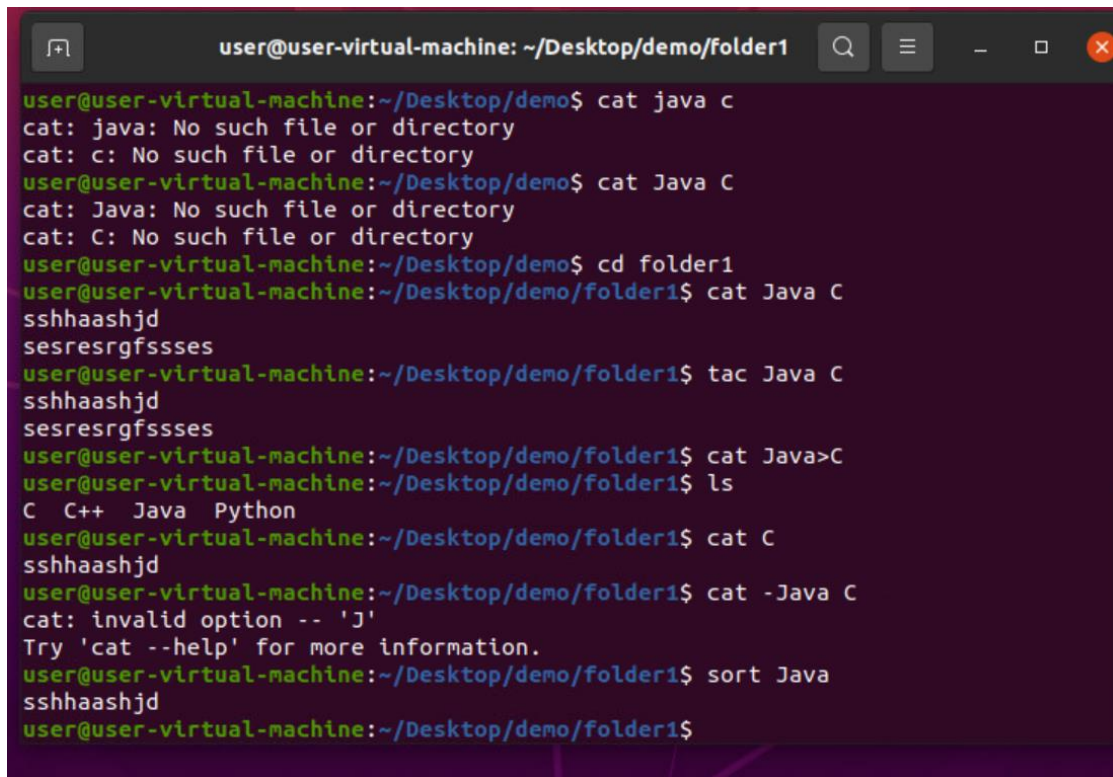


A terminal window titled 'user@user-virtual-machine: ~/Desktop/demo/folder1'. The user has moved into 'folder2' and then back to 'demo'. They attempt to run 'cat java c' but get errors. Then they run 'cat Java C' and get errors. They move into 'folder1' and run 'cat Java C', which outputs the contents of 'Java' and 'C'. Finally, they run 'cat Java>C', which appends the contents of 'Java' to 'C'. The terminal shows the following commands and output:

```
B
user@user-virtual-machine:~/Desktop/demo/folder2$ tac aa bb
A
B
user@user-virtual-machine:~/Desktop/demo/folder2$ cd ..
user@user-virtual-machine:~/Desktop/demo$ cat java c
cat: java: No such file or directory
cat: c: No such file or directory
user@user-virtual-machine:~/Desktop/demo$ cat Java C
cat: Java: No such file or directory
cat: C: No such file or directory
user@user-virtual-machine:~/Desktop/demo$ cd folder1
user@user-virtual-machine:~/Desktop/demo/folder1$ cat Java C
sshhaashjd
sesresrgfssses
sshhaashjd
sesresrgfssses
user@user-virtual-machine:~/Desktop/demo/folder1$ cat Java>C
user@user-virtual-machine:~/Desktop/demo/folder1$ ls
C C++ Java Python
user@user-virtual-machine:~/Desktop/demo/folder1$ cat C
sshhaashjd
user@user-virtual-machine:~/Desktop/demo/folder1$
```

copies the data from source document to destination document.

28 sort



```
user@user-virtual-machine: ~/Desktop/demo/folder1
user@user-virtual-machine:~/Desktop/demo$ cat java c
cat: java: No such file or directory
cat: c: No such file or directory
user@user-virtual-machine:~/Desktop/demo$ cat Java C
cat: Java: No such file or directory
cat: C: No such file or directory
user@user-virtual-machine:~/Desktop/demo$ cd folder1
user@user-virtual-machine:~/Desktop/demo/folder1$ cat Java C
sshhaashjd
sesresrgfssses
user@user-virtual-machine:~/Desktop/demo/folder1$ tac Java C
sshhaashjd
sesresrgfssses
user@user-virtual-machine:~/Desktop/demo/folder1$ cat Java>C
user@user-virtual-machine:~/Desktop/demo/folder1$ ls
C  C++  Java  Python
user@user-virtual-machine:~/Desktop/demo/folder1$ cat C
sshhaashjd
user@user-virtual-machine:~/Desktop/demo/folder1$ cat -Java C
cat: invalid option -- 'J'
Try 'cat --help' for more information.
user@user-virtual-machine:~/Desktop/demo/folder1$ sort Java
sshhaashjd
user@user-virtual-machine:~/Desktop/demo/folder1$
```

SORT command sorts the contents of a text file, line by line.

29 mkdir/2

```
cat: C: No such file or directory
user@user-virtual-machine:~/Desktop/demo$ cd folder1
user@user-virtual-machine:~/Desktop/demo/folder1$ cat Java C
sshhaashjd
sesresrgfssses
user@user-virtual-machine:~/Desktop/demo/folder1$ tac Java C
sshhaashjd
sesresrgfssses
user@user-virtual-machine:~/Desktop/demo/folder1$ cat Java>C
user@user-virtual-machine:~/Desktop/demo/folder1$ ls
C  C++  Java  Python
user@user-virtual-machine:~/Desktop/demo/folder1$ cat C
sshhaashjd
user@user-virtual-machine:~/Desktop/demo/folder1$ cat -Java C
cat: invalid option -- 'J'
Try 'cat --help' for more information.
user@user-virtual-machine:~/Desktop/demo/folder1$ sort Java
sshhaashjd
user@user-virtual-machine:~/Desktop/demo/folder1$ cd
user@user-virtual-machine:~$ mkdir demo/2
mkdir: cannot create directory 'demo/2': No such file or directory
user@user-virtual-machine:~$ cd Desktop
user@user-virtual-machine:~/Desktop$ mkdir demo/2
user@user-virtual-machine:~/Desktop$
```

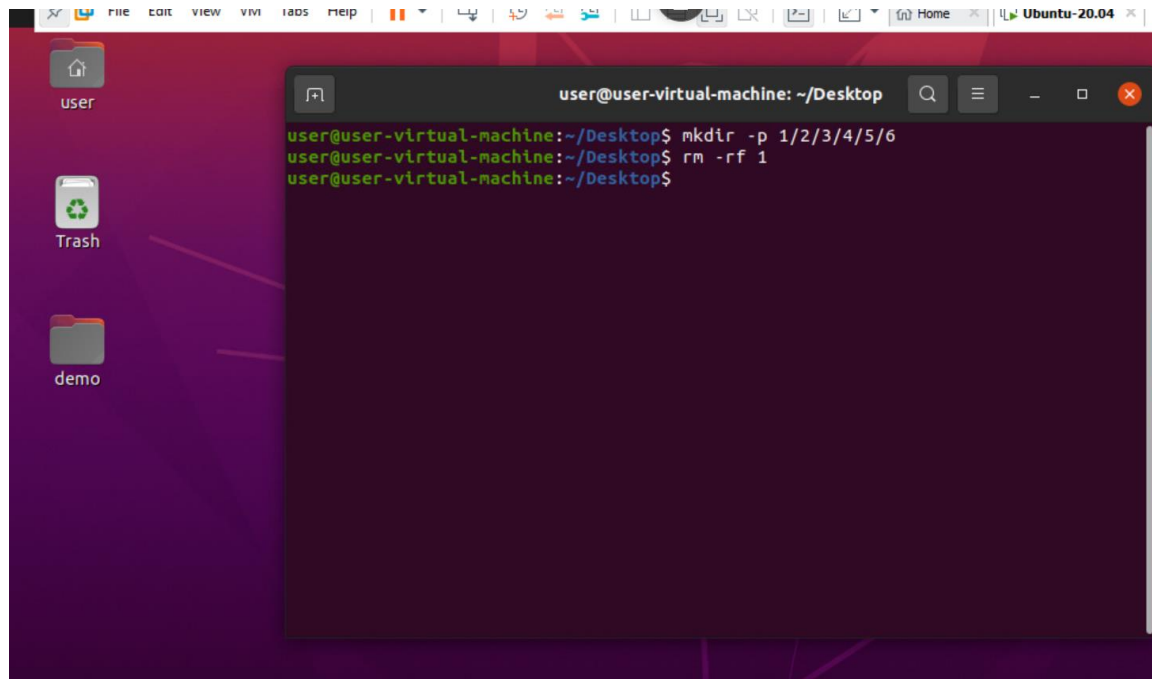
It creates the folder 2 inside demo.

30 mkdir -p 1/ 2/3/4/5/6

```
user@user-virtual-machine: ~/Desktop
user@user-virtual-machine:~/Desktop$ mkdir -p 1/2/3/4/5/6
user@user-virtual-machine:~/Desktop$
```

P helps the to create sub folders inside the parent directory

31 rm -rf 1



It removes entire folder and also its sub folder

32 ls -l

```
user@user-virtual-machine: ~/Desktop/demo/folder1
user@user-virtual-machine:~/Desktop$ mkdir -p 1/2/3/4/5/6
user@user-virtual-machine:~/Desktop$ rm -rf 1
user@user-virtual-machine:~/Desktop$ ls -l
total 4
drwxrwxr-x 6 user user 4096 Sep 14 16:28 demo
user@user-virtual-machine:~/Desktop$ cd demo/folder1
user@user-virtual-machine:~/Desktop/demo/folder1$ ls -l
total 16
-rw-rw-r-- 1 user user 11 Sep 14 16:16 C
-rw-rw-r-- 1 user user 10 Sep 14 14:35 C++
-rw-rw-r-- 1 user user 11 Sep 14 14:31 Java
-rw-rw-r-- 1 user user 14 Sep 14 14:33 Python
user@user-virtual-machine:~/Desktop/demo/folder1$
```

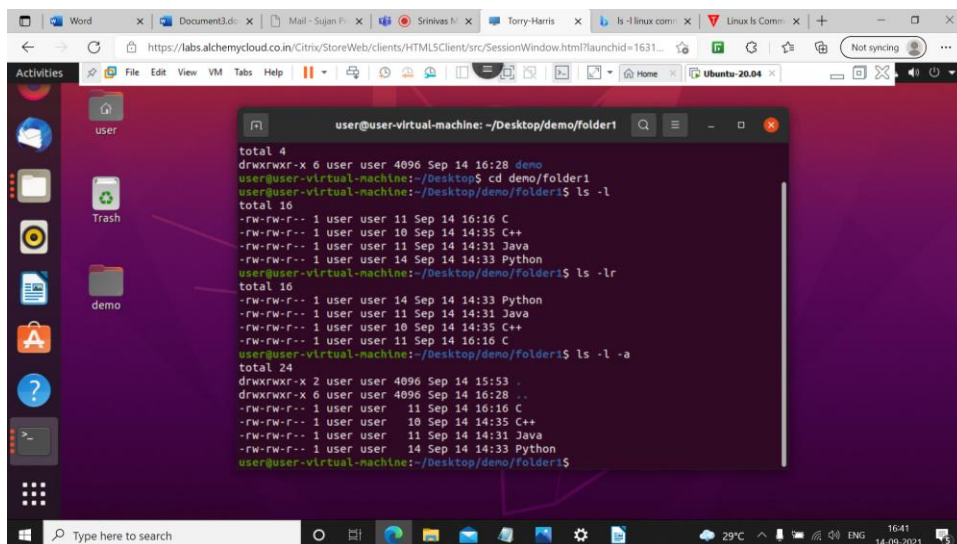
It will show the list in a long list format.

33 ls -lr

```
user@user-virtual-machine: ~/Desktop/demo/folder1
user@user-virtual-machine:~/Desktop$ mkdir -p 1/2/3/4/5/6
user@user-virtual-machine:~/Desktop$ rm -rf 1
user@user-virtual-machine:~/Desktop$ ls -l
total 4
drwxrwxr-x 6 user user 4096 Sep 14 16:28 demo
user@user-virtual-machine:~/Desktop$ cd demo/folder1
user@user-virtual-machine:~/Desktop/demo/folder1$ ls -l
total 16
-rw-rw-r-- 1 user user 11 Sep 14 16:16 C
-rw-rw-r-- 1 user user 10 Sep 14 14:35 C++
-rw-rw-r-- 1 user user 11 Sep 14 14:31 Java
-rw-rw-r-- 1 user user 14 Sep 14 14:33 Python
user@user-virtual-machine:~/Desktop/demo/folder1$ ls -lr
total 16
-rw-rw-r-- 1 user user 14 Sep 14 14:33 Python
-rw-rw-r-- 1 user user 11 Sep 14 14:31 Java
-rw-rw-r-- 1 user user 10 Sep 14 14:35 C++
-rw-rw-r-- 1 user user 11 Sep 14 16:16 C
user@user-virtual-machine:~/Desktop/demo/folder1$
```

It will show the list in a long list format and reverse order.

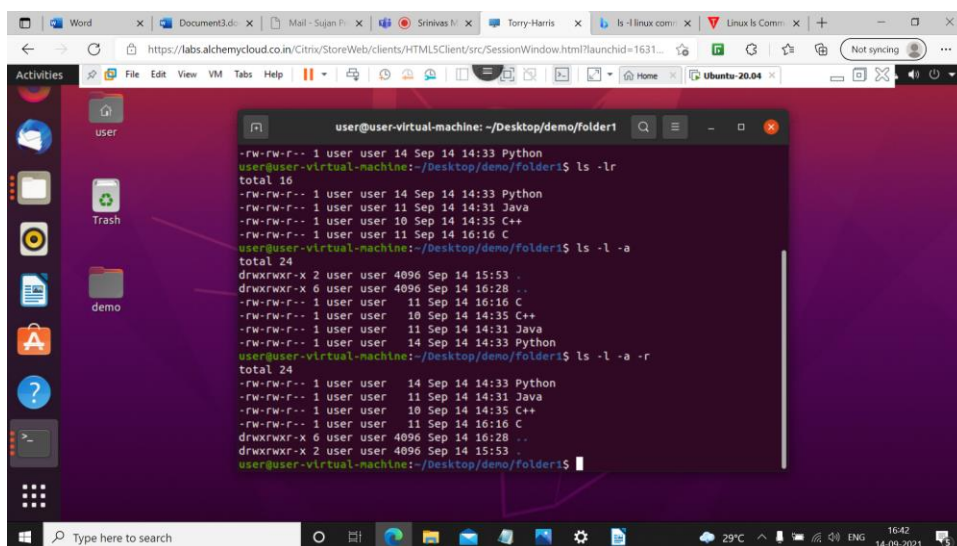
34 ls -l -a



```
total 4
drwxrwxr-x 6 user user 4096 Sep 14 16:28 demo
user@user-virtual-machine: ~/Desktop$ cd demo/folder1
user@user-virtual-machine: ~/Desktop/demo/folder1$ ls -l
total 16
-rw-rw-r-- 1 user user 11 Sep 14 16:16 C
-rw-rw-r-- 1 user user 10 Sep 14 14:35 C++
-rw-rw-r-- 1 user user 11 Sep 14 14:31 Java
-rw-rw-r-- 1 user user 14 Sep 14 14:33 Python
user@user-virtual-machine: ~/Desktop/demo/folder1$ ls -lr
total 16
-rw-rw-r-- 1 user user 14 Sep 14 14:33 Python
-rw-rw-r-- 1 user user 11 Sep 14 14:31 Java
-rw-rw-r-- 1 user user 10 Sep 14 14:35 C++
-rw-rw-r-- 1 user user 11 Sep 14 16:16 C
user@user-virtual-machine: ~/Desktop/demo/folder1$ ls -l -a
total 24
drwxrwxr-x 2 user user 4096 Sep 14 15:53 .
drwxrwxr-x 6 user user 4096 Sep 14 16:28 ..
-rw-rw-r-- 1 user user 11 Sep 14 16:16 C
-rw-rw-r-- 1 user user 10 Sep 14 14:35 C++
-rw-rw-r-- 1 user user 11 Sep 14 14:31 Java
-rw-rw-r-- 1 user user 14 Sep 14 14:33 Python
user@user-virtual-machine: ~/Desktop/demo/folder1$
```

It will show the list in a long list format and enlist the whole list of the current directory including the hidden files.

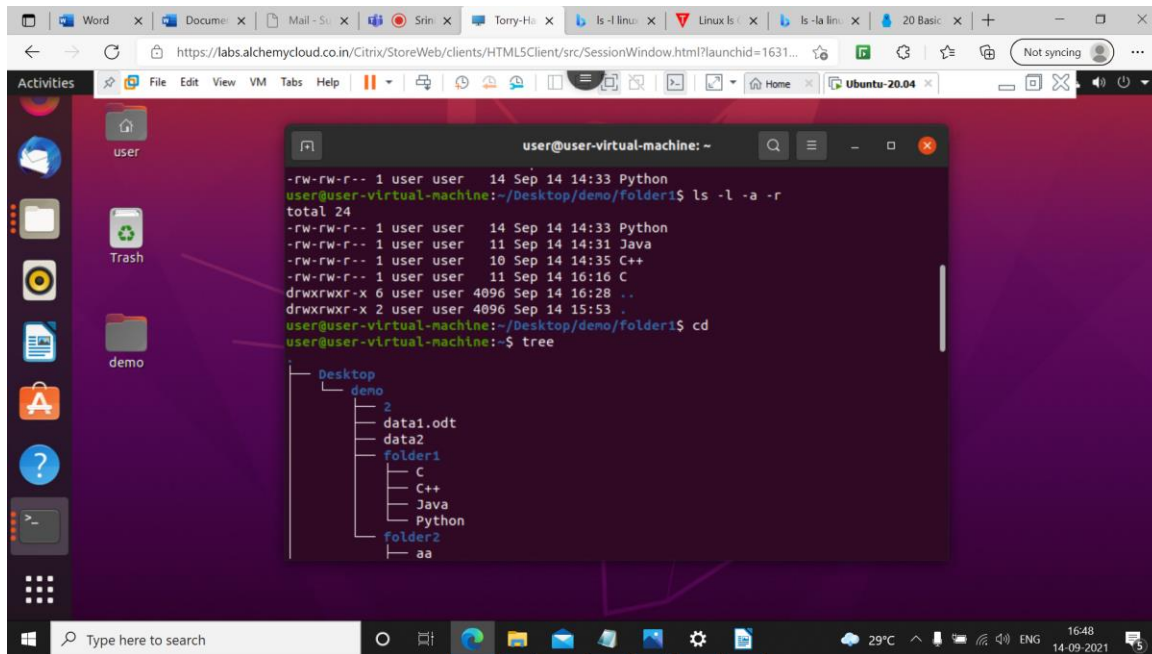
35 ls -l -a -r



```
-rw-rw-r-- 1 user user 14 Sep 14 14:33 Python
user@user-virtual-machine: ~/Desktop/demo/folder1$ ls -lr
total 16
-rw-rw-r-- 1 user user 14 Sep 14 14:33 Python
-rw-rw-r-- 1 user user 11 Sep 14 14:31 Java
-rw-rw-r-- 1 user user 10 Sep 14 14:35 C++
-rw-rw-r-- 1 user user 11 Sep 14 16:16 C
user@user-virtual-machine: ~/Desktop/demo/folder1$ ls -l -a
total 24
drwxrwxr-x 2 user user 4096 Sep 14 15:53 .
drwxrwxr-x 6 user user 4096 Sep 14 16:28 ..
-rw-rw-r-- 1 user user 11 Sep 14 16:16 C
-rw-rw-r-- 1 user user 10 Sep 14 14:35 C++
-rw-rw-r-- 1 user user 11 Sep 14 14:31 Java
-rw-rw-r-- 1 user user 14 Sep 14 14:33 Python
user@user-virtual-machine: ~/Desktop/demo/folder1$ ls -l -a -r
total 24
-rw-rw-r-- 1 user user 14 Sep 14 14:33 Python
-rw-rw-r-- 1 user user 11 Sep 14 14:31 Java
-rw-rw-r-- 1 user user 10 Sep 14 14:35 C++
-rw-rw-r-- 1 user user 11 Sep 14 16:16 C
drwxrwxr-x 6 user user 4096 Sep 14 16:28 ..
drwxrwxr-x 2 user user 4096 Sep 14 15:53 .
user@user-virtual-machine: ~/Desktop/demo/folder1$
```

It is used to reverse the above format.

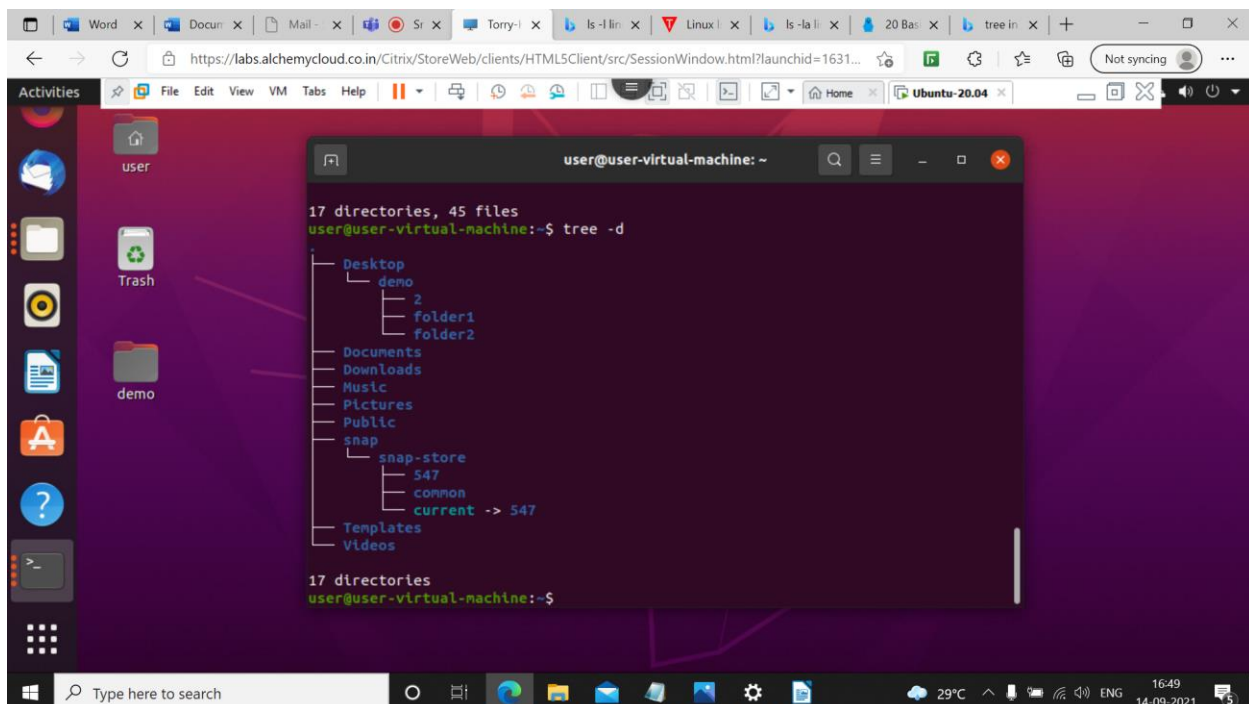
36 tree



```
user@user-virtual-machine: ~  
-rw-rw-r-- 1 user user 14 Sep 14 14:33 Python  
user@user-virtual-machine:~/Desktop/demo/folder1$ ls -l -a -r  
total 24  
-rw-rw-r-- 1 user user 14 Sep 14 14:33 Python  
-rw-rw-r-- 1 user user 11 Sep 14 14:31 Java  
-rw-rw-r-- 1 user user 10 Sep 14 14:35 C++  
-rw-rw-r-- 1 user user 11 Sep 14 16:16 C  
drwxrwxr-x 6 user user 4096 Sep 14 16:28 ..  
drwxrwxr-x 2 user user 4096 Sep 14 15:53 .  
user@user-virtual-machine:~/Desktop/demo/folder1$ cd  
user@user-virtual-machine:~$ tree  
.  
├── Desktop  
│   └── demo  
│       ├── 2  
│       ├── data1.odt  
│       ├── data2  
│       ├── folder1  
│       │   ├── C  
│       │   ├── C++  
│       │   ├── Java  
│       │   ├── Python  
│       │   └── folder2  
│       └── aa
```

The **tree** command is a **Linux** program that lists our directories and files in a more helpful way resembling a **tree** structure.

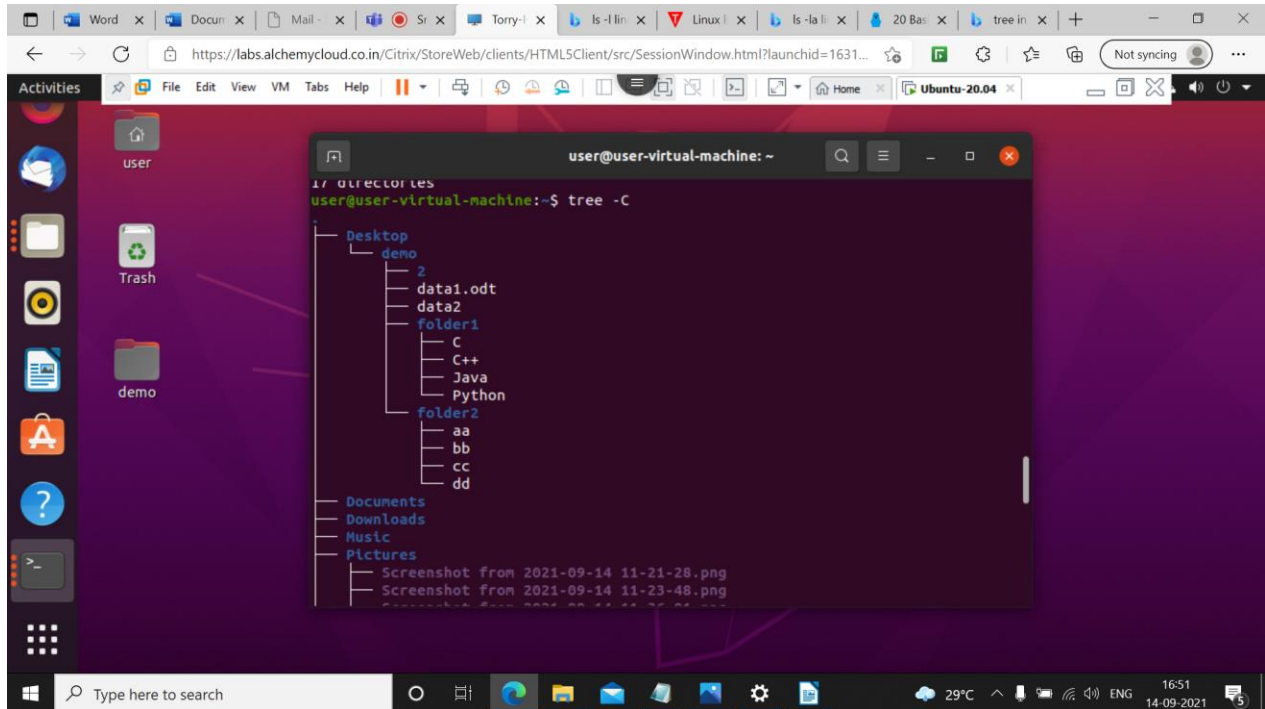
37 tree -d



```
17 directories, 45 files  
user@user-virtual-machine:~$ tree -d  
.  
├── Desktop  
│   └── demo  
│       ├── 2  
│       ├── folder1  
│       └── folder2  
├── Documents  
├── Downloads  
├── Music  
├── Pictures  
├── Public  
├── snap  
│   ├── snap-store  
│   ├── 547  
│   ├── common  
│   └── current -> 547  
├── Templates  
└── Videos  
17 directories  
user@user-virtual-machine:~$
```

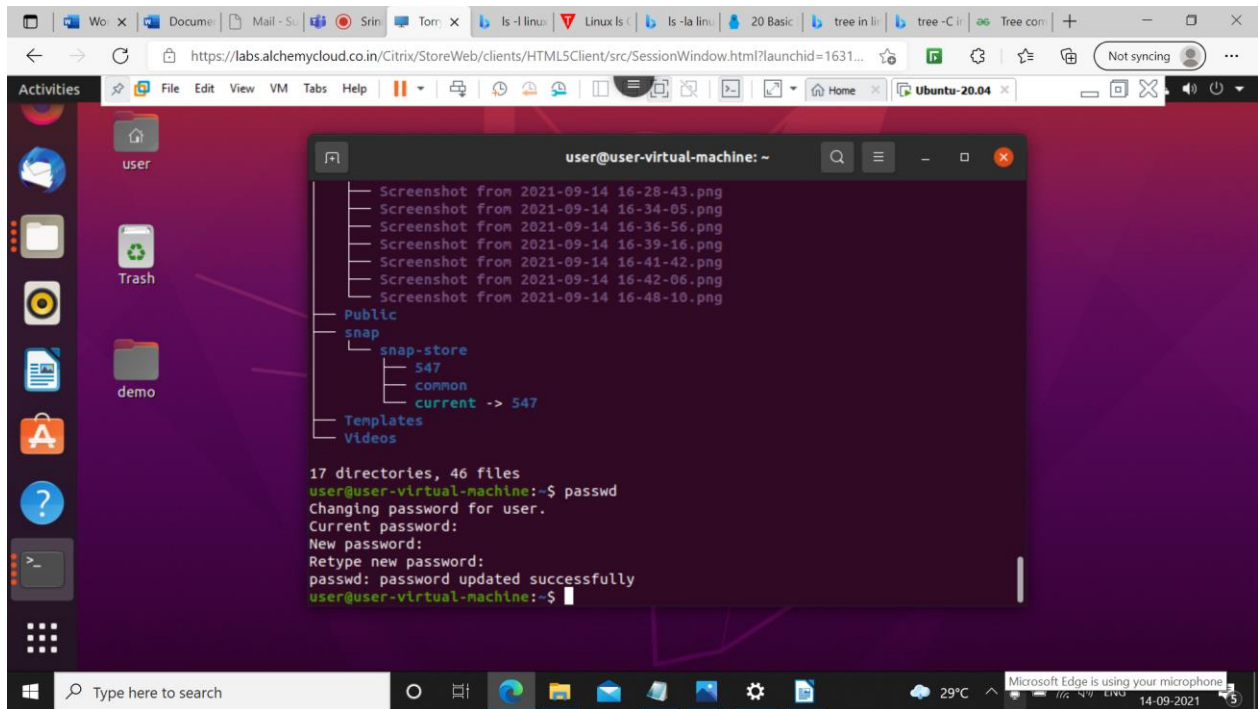
It shows only directories

38 tree-C



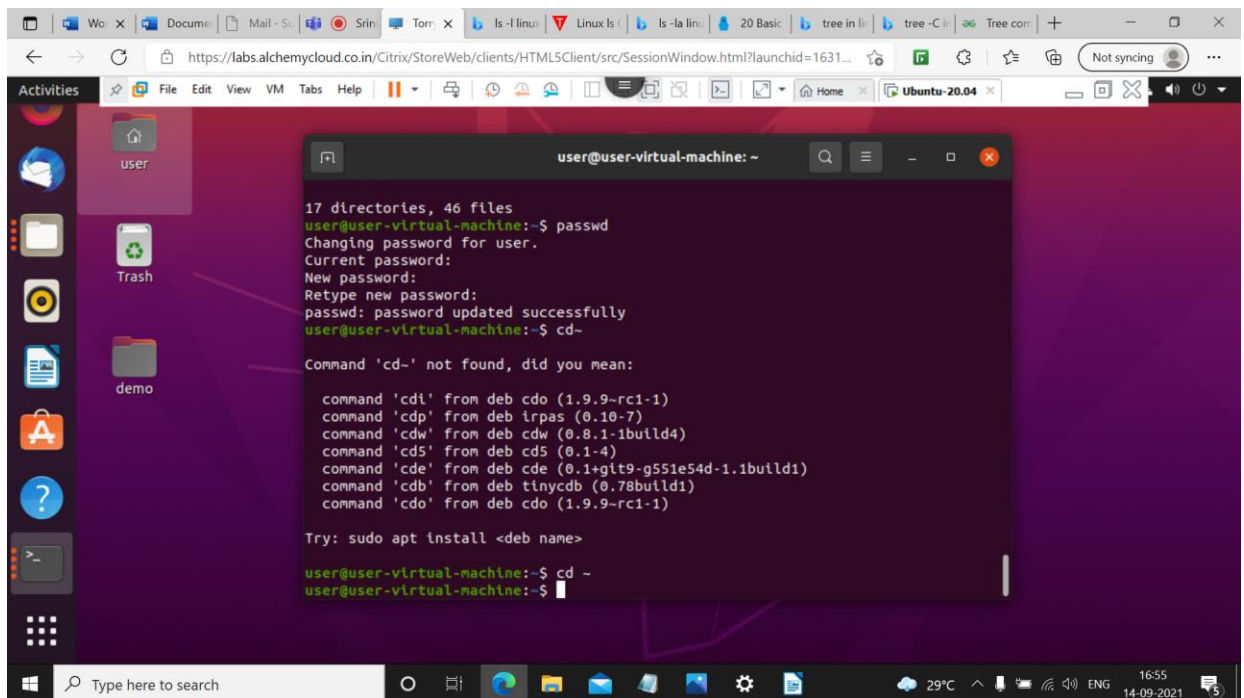
Turn colorization on always, using built-in color defaults if the `LS_COLORS` environment variable is not set. Useful to colorize output to a pipe.

39 passwd



passwd command in Linux is used to change the user account passwords. The root user reserves the privilege to change the password for any user on the system.

40 cd ~



It will get back to the home directory

41 sudo apt -get update

This command is used to download the updates

42 sudo apt -get upgrade

This command is used to install the downloaded updates

43 sudo apt -get update && upgrade

This command is used to do both the things like download & install

44 sudo apt -get remove tree

This command is used to remove the installed tree