

# Basic Linux Commands

Submitted By:

Akshay Murali

Roll No: 07

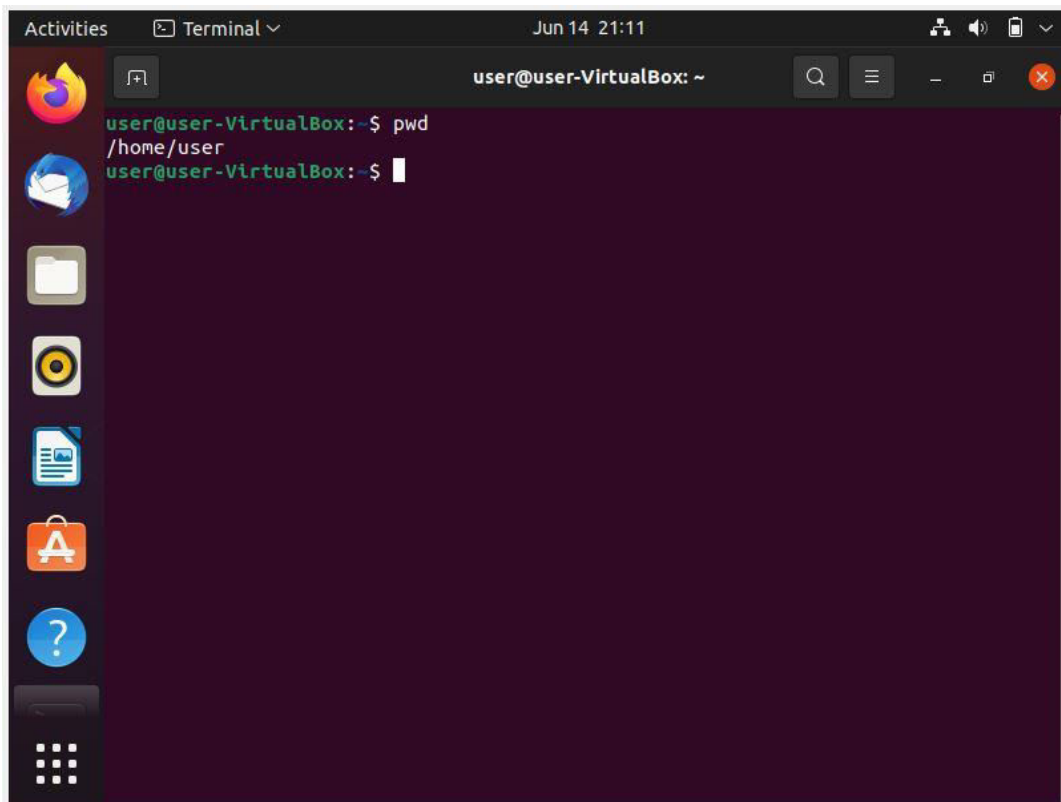
Subject: Network Lab

Batch: MCA A

# BASIC LINUX COMMANDS

## 1. pwd (Print Working Directory)

Use the pwd command to find out the path of the current working directory (folder) you're in.

A screenshot of a Linux terminal window. The window title bar shows 'Activities', 'Terminal', and the date/time 'Jun 14 21:11'. The terminal prompt is 'user@user-VirtualBox: ~'. The user has entered the command 'pwd', and the output is '/home/user'. The terminal window has a dark purple background and a sidebar on the left with various application icons.

```
user@user-VirtualBox: ~  
user@user-VirtualBox: ~$ pwd  
/home/user  
user@user-VirtualBox: ~$
```

2.

## history

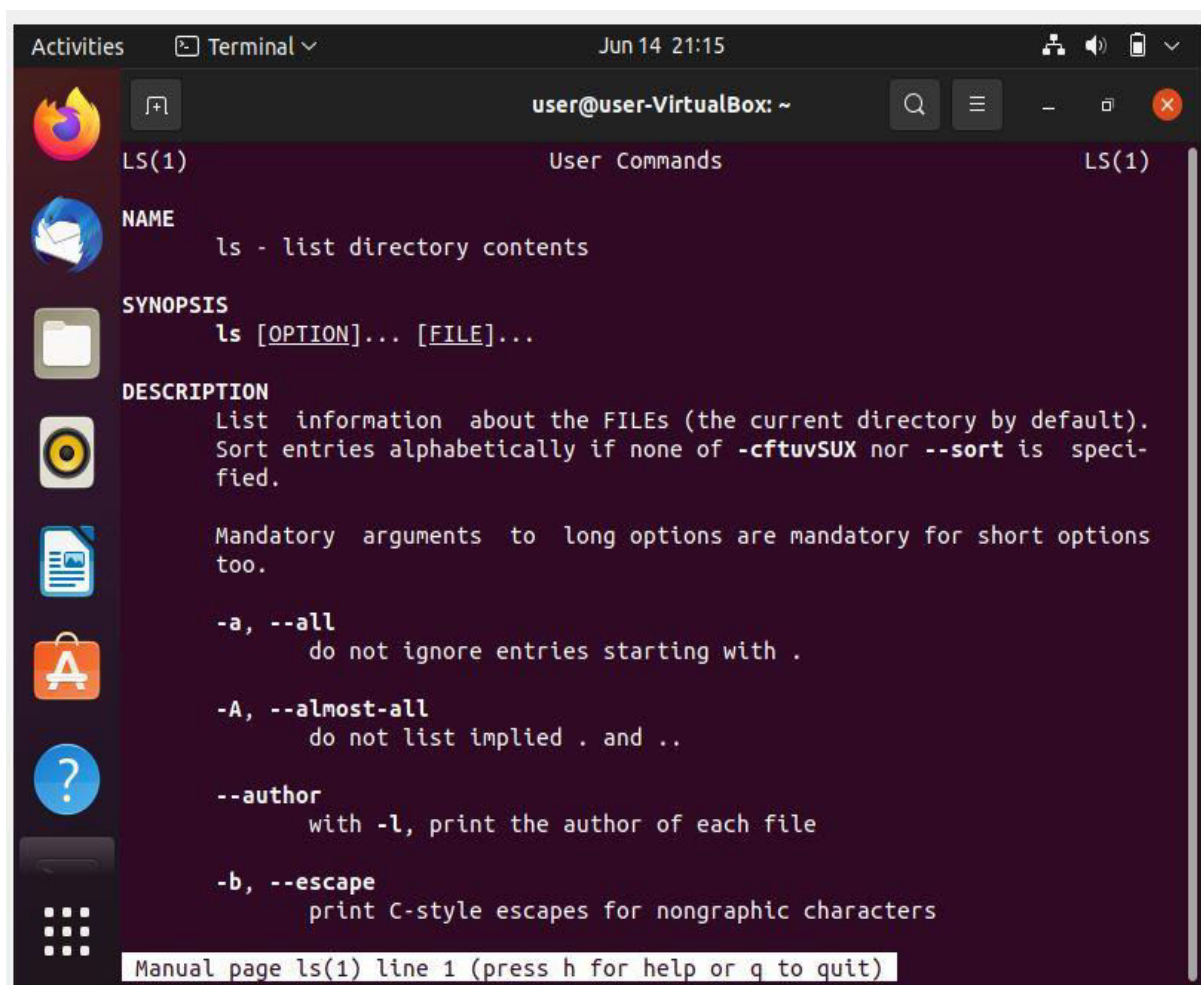
- When you have been using Linux for a certain period of time, you will quickly notice that you can run hundreds of commands everyday. As such, running history command is particularly useful if you want to review the commands you have entered before. ➤ history
- !command number to run a command from history

A screenshot of a Linux terminal window titled 'user@user-VirtualBox: ~'. The terminal shows the execution of the 'pwd' command, which returns '/home/user'. Then, the 'history' command is executed, displaying a list of 14 commands in a numbered format. The commands are: 1 pwd, 2 pwd a, 3 ls, 4 history, 5 clear, 6 ls, 7 cd public, 8 cd Public, 9 cd., 10 cd-, 11 cd, 12 cd Public, 13 ls, and 14 ls!ls!ls!ls!ls. The terminal has a dark purple background and a light blue prompt. On the left side of the terminal window, there is a vertical dock with several application icons: Firefox, a file manager, a terminal, a CD/DVD icon, and a document icon. The window's title bar includes standard Linux window controls (minimize, maximize, close) and a search icon.

3.

## man

If we are confused about the function of certain Linux commands we can easily learn how to use them right from Linux's shell by using the **man** command. For instance, entering **man tail** will show the manual instruction of the **tail** command. **man ls**



The screenshot shows a terminal window titled "user@user-VirtualBox: ~" with a date and time of "Jun 14 21:15". The terminal displays the manual page for the **ls** command. The left sidebar contains icons for various applications. The main content area shows the following text:

```
LS(1)                                User Commands                                LS(1)

NAME
    ls - list directory contents

SYNOPSIS
    ls [OPTION]... [FILE]...

DESCRIPTION
    List information about the FILES (the current directory by default).
    Sort entries alphabetically if none of -cftuvSUX nor --sort is speci-
    fied.

    Mandatory arguments to long options are mandatory for short options
    too.

    -a, --all
        do not ignore entries starting with .

    -A, --almost-all
        do not list implied . and ..

    --author
        with -l, print the author of each file

    -b, --escape
        print C-style escapes for nongraphic characters

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

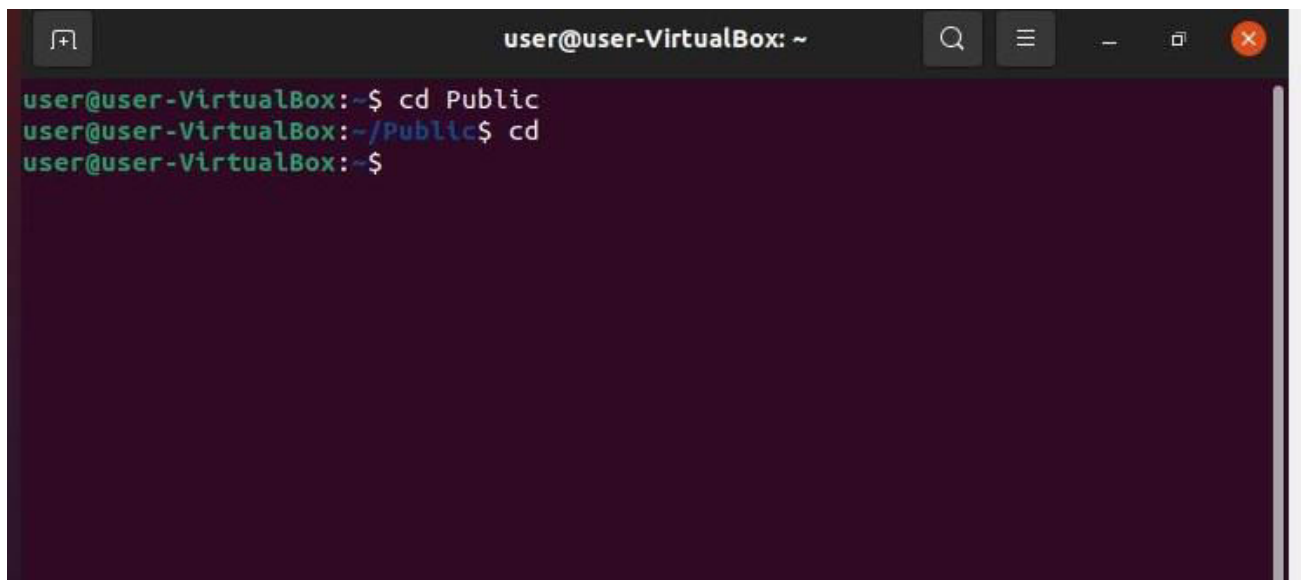
## 4.

### cd

To navigate through the Linux files and directories, use the `cd`. It requires either the full path or the name of the directory, depending on the current working directory that you're in.

Shortcuts to help you navigate quickly:

- `cd ..` (with two dots) to move one directory up
- `cd` to go straight to the home folder
- `cd-` (with a hyphen) to move to your previous directory

A terminal window titled 'user@user-VirtualBox: ~' with search, menu, and window control icons in the title bar. The terminal shows a sequence of commands: 'cd Public' which changes the directory to '/Public', followed by 'cd' which returns to the home directory '~'.

```
user@user-VirtualBox:~$ cd Public
user@user-VirtualBox:~/Public$ cd
user@user-VirtualBox:~$
```

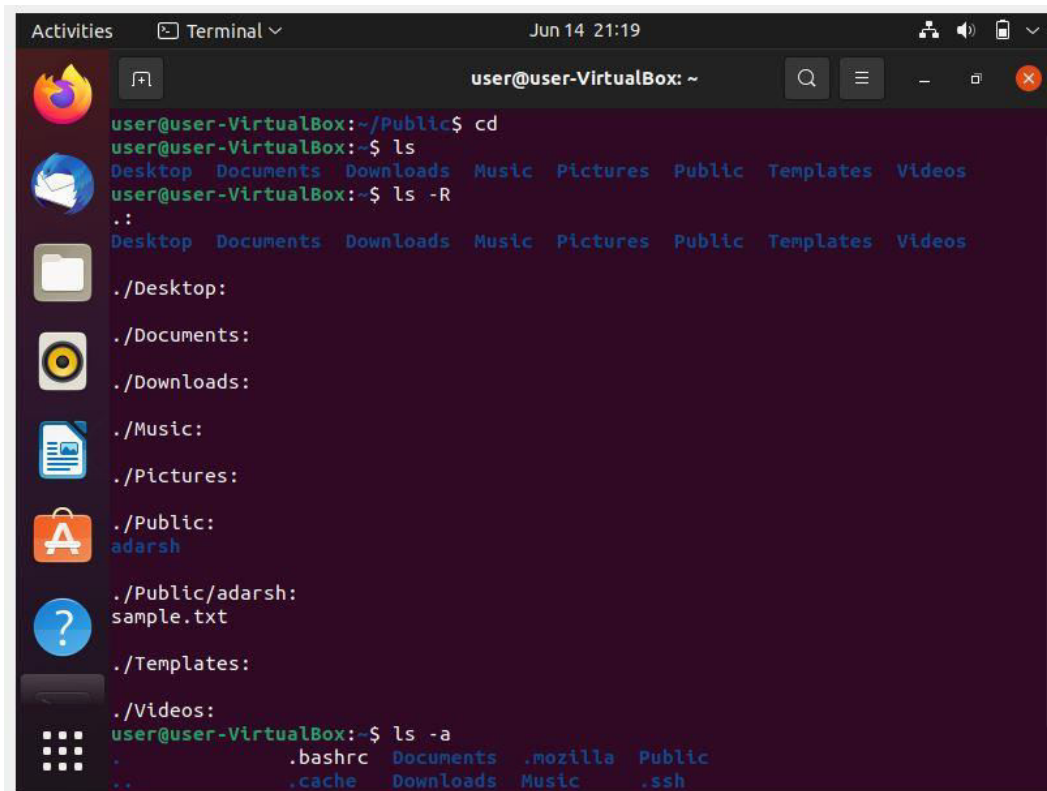
## 5. ls

The `ls` command is used to view the contents of a directory. By default, this command will display the contents of your current working directory.

There are variations you can use with the `ls` command:

- **`ls -R`** will list all the files in the sub-directories as well

- **ls -l** – long listing
- **ls -a** will show the hidden files
- **ls -al** will list the files and directories with detailed information like the permissions, size, owner, etc.
- **ls -t** lists files sorted in the order of “last modified”.
- **ls -r** option will reverse the natural sorting order. Usually used in combination with other switches such as **ls -tr**. This will reverse the time-wise listing.



The screenshot shows a terminal window titled "Terminal" with a date and time of "Jun 14 21:19". The prompt is "user@user-VirtualBox: ~". The user has entered the following commands and received the following output:

```
user@user-VirtualBox:~/Public$ cd
user@user-VirtualBox:~$ ls
Desktop Documents Downloads Music Pictures Public Templates Videos
user@user-VirtualBox:~$ ls -R
.:
Desktop Documents Downloads Music Pictures Public Templates Videos

./Desktop:

./Documents:

./Downloads:

./Music:

./Pictures:

./Public:
adarsh

./Public/adarsh:
sample.txt

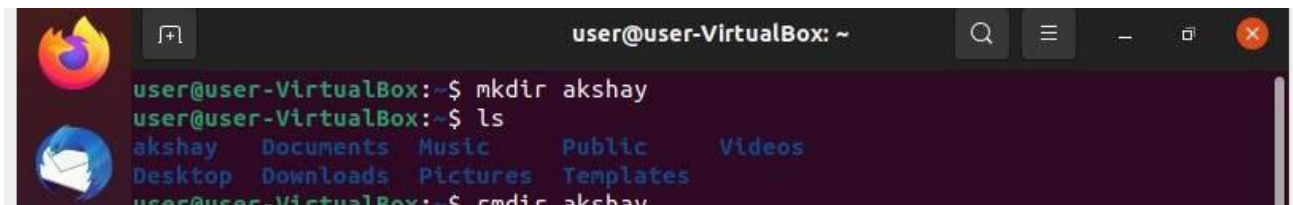
./Templates:

./Videos:
user@user-VirtualBox:~$ ls -a
. .bashrc .cache .Documents .mozilla .Public
.. .cache Downloads Music .ssh
```

## 6. mkdir

Use mkdir command to make a new directory .

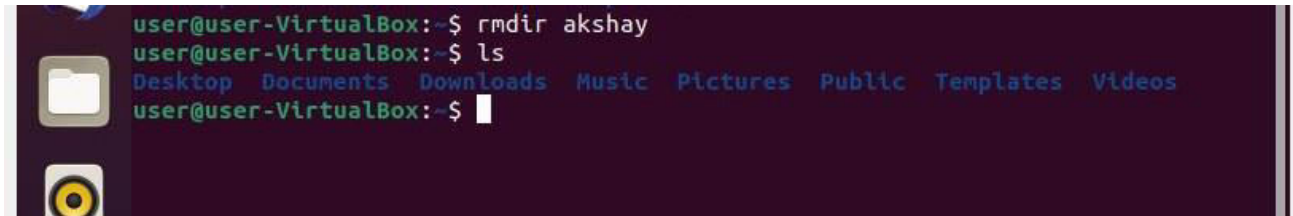
To generate a new directory inside another directory, use this Linux basic command.

A screenshot of a Linux terminal window titled 'user@user-VirtualBox: ~'. The terminal shows the following commands and output: 'mkdir akshay', 'ls', and the output 'akshay Documents Music Public Videos Desktop Downloads Pictures Templates'. The prompt 'user@user-VirtualBox:~\$' is visible at the end of the last line.

```
user@user-VirtualBox:~$ mkdir akshay
user@user-VirtualBox:~$ ls
akshay  Documents  Music      Public     Videos
Desktop Downloads  Pictures   Templates
user@user-VirtualBox:~$
```

## 7. rmdir

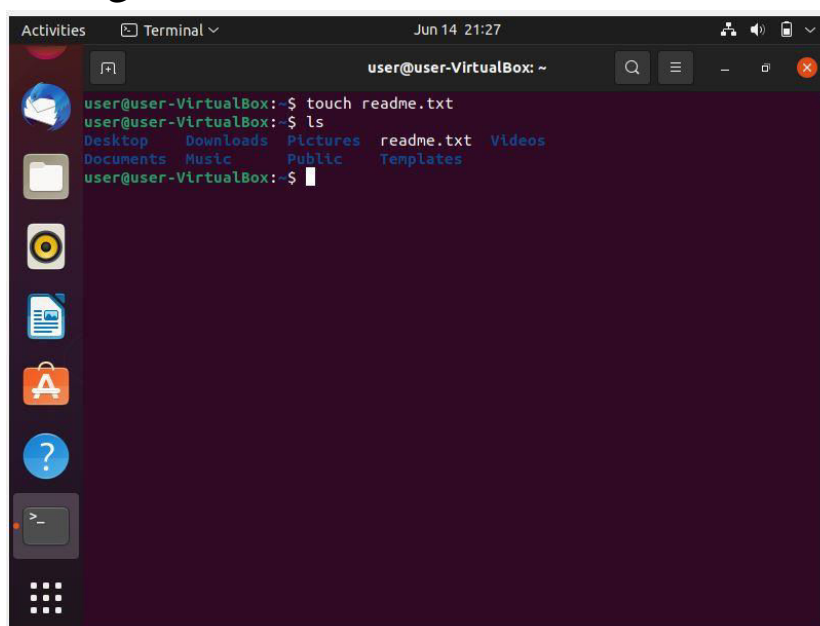
If you need to delete a directory, use the `rmdir` command. However, `rmdir` only allows you to delete empty directories.

A terminal window with a dark purple background. The prompt is 'user@user-VirtualBox:~\$'. The first command is 'rmdir akshay'. The second command is 'ls', which lists the contents of the home directory: Desktop, Documents, Downloads, Music, Pictures, Public, Templates, and Videos. The prompt returns to 'user@user-VirtualBox:~\$' with a cursor.

```
user@user-VirtualBox:~$ rmdir akshay
user@user-VirtualBox:~$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos
user@user-VirtualBox:~$
```

## 8. touch

The `touch` command allows you to create a blank new file through the Linux command line.

A terminal window titled 'Terminal' with a date and time of 'Jun 14 21:27'. The prompt is 'user@user-VirtualBox: ~'. The first command is 'touch readme.txt'. The second command is 'ls', which lists the contents of the home directory: Desktop, Downloads, Pictures, readme.txt, Videos, Documents, Music, Public, and Templates. The prompt returns to 'user@user-VirtualBox:~\$' with a cursor.

```
user@user-VirtualBox:~$ touch readme.txt
user@user-VirtualBox:~$ ls
Desktop  Downloads  Pictures  readme.txt  Videos
Documents Music      Public   Templates
user@user-VirtualBox:~$
```

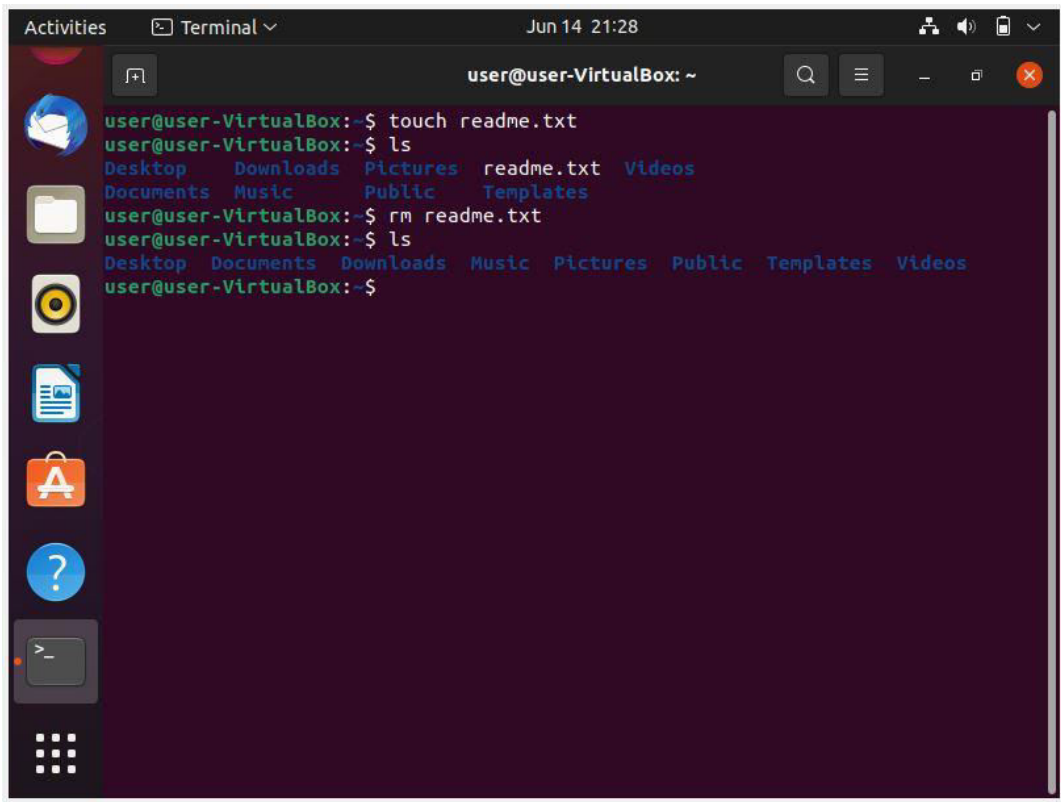
## 9. rm

The `rm` command is used to delete directories and the contents within them.



If you only want to delete the directory — as an alternative to `rmdir` — use `rm -r`.

To remove a file use **`rm filename`**



```
user@user-VirtualBox:~$ touch readme.txt
user@user-VirtualBox:~$ ls
Desktop  Downloads  Pictures  readme.txt  Videos
Documents Music      Public    Templates
user@user-VirtualBox:~$ rm readme.txt
user@user-VirtualBox:~$ ls
Desktop Documents Downloads Music Pictures Public Templates Videos
user@user-VirtualBox:~$
```

## 10. `cat`

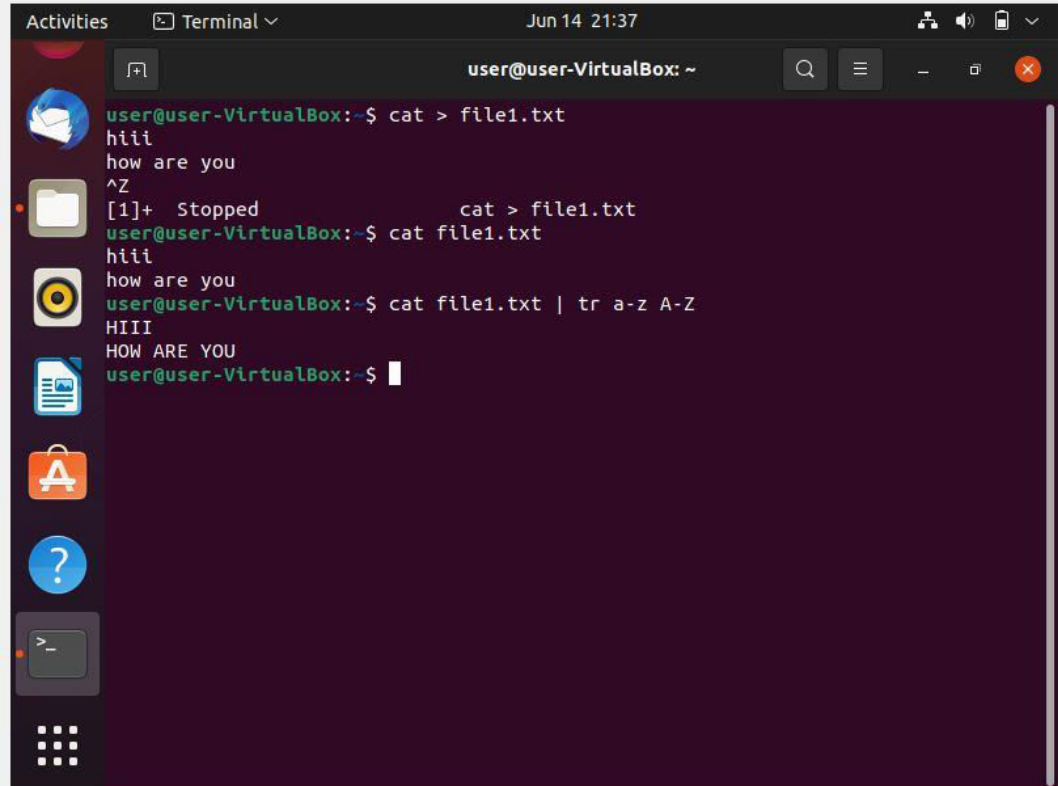
`cat` (short for concatenate) is one of the most frequently used commands in Linux. It is used to list the contents of a file on the standard output `stdout`.

To run this command, type `cat` followed by the file's name and its extension. For instance: `cat file.txt`.

Here are other ways to use the `cat` command:

- **`cat > filename`** creates a new file
- **`cat filename1 filename2 > filename3`** joins two files (1 and 2) and stores the output of them in a new file (3)
- to convert a file to upper or lower case use, **`cat filename`**

- | **tr a-z A-Z >output.txt**
- **cat >>myfile** insert data to a file



A terminal window titled 'user@user-VirtualBox: ~' showing a series of commands and their outputs. The user first creates a file named 'file1.txt' using 'cat > file1.txt' and enters the text 'hihi' and 'how are you'. After pressing Ctrl-Z (^Z), the terminal shows '[1]+ Stopped cat > file1.txt'. The user then runs 'cat file1.txt', which displays the original text. Finally, the user runs 'cat file1.txt | tr a-z A-Z', which displays the transformed text in all caps: 'HIII' and 'HOW ARE YOU'.

```
user@user-VirtualBox:~$ cat > file1.txt
hihi
how are you
^Z
[1]+  Stopped                  cat > file1.txt
user@user-VirtualBox:~$ cat file1.txt
hihi
how are you
user@user-VirtualBox:~$ cat file1.txt | tr a-z A-Z
HIII
HOW ARE YOU
user@user-VirtualBox:~$
```