

# ADVANCED COMPUTER NETWORK ASSIGNMENT

**Topic:** - Take screenshots of basic Linux commands

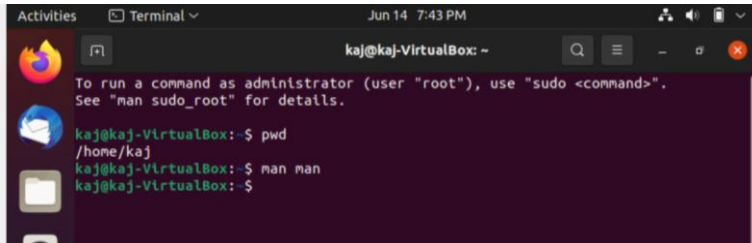
**Submitted By:**  
Anilect Jose  
Roll No: 17  
RMCA A

**Submitted To:**  
Rini Kurian

## 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 terminal window titled 'Terminal' with a date of 'Jun 14 7:43 PM'. The prompt is 'kaj@kaj-VirtualBox: ~'. It shows a message: 'To run a command as administrator (user "root"), use "sudo <command>". See "man sudo\_root" for details.' Below this, the user runs 'pwd' and the output is '/home/kaj'. Then the user runs 'man man' and the prompt returns to '\$'.

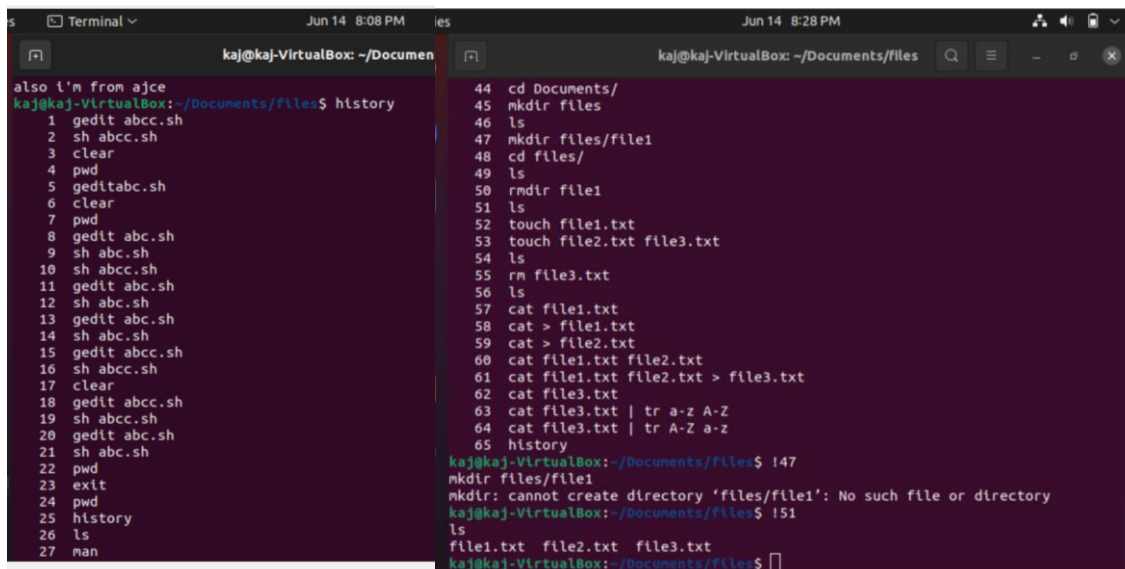
```
kaj@kaj-VirtualBox: ~  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
kaj@kaj-VirtualBox:~$ pwd  
/home/kaj  
kaj@kaj-VirtualBox:~$ man man  
kaj@kaj-VirtualBox:~$
```

### 2. history

Ø 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.

Ø history

Ø !command number to run a command from history

Two terminal windows are shown. The left window, titled 'Terminal', has a date of 'Jun 14 8:08 PM' and shows a list of 27 commands from the history, starting with 'gedit abcc.sh' and ending with 'man'. The right window, titled 'Terminal', has a date of 'Jun 14 8:28 PM' and shows the user navigating to '/Documents/files', creating files, and then using '!147' to re-run a 'mkdir' command from the history. It also shows an error message for a non-existent directory and the use of '!51' to re-run an 'ls' command.

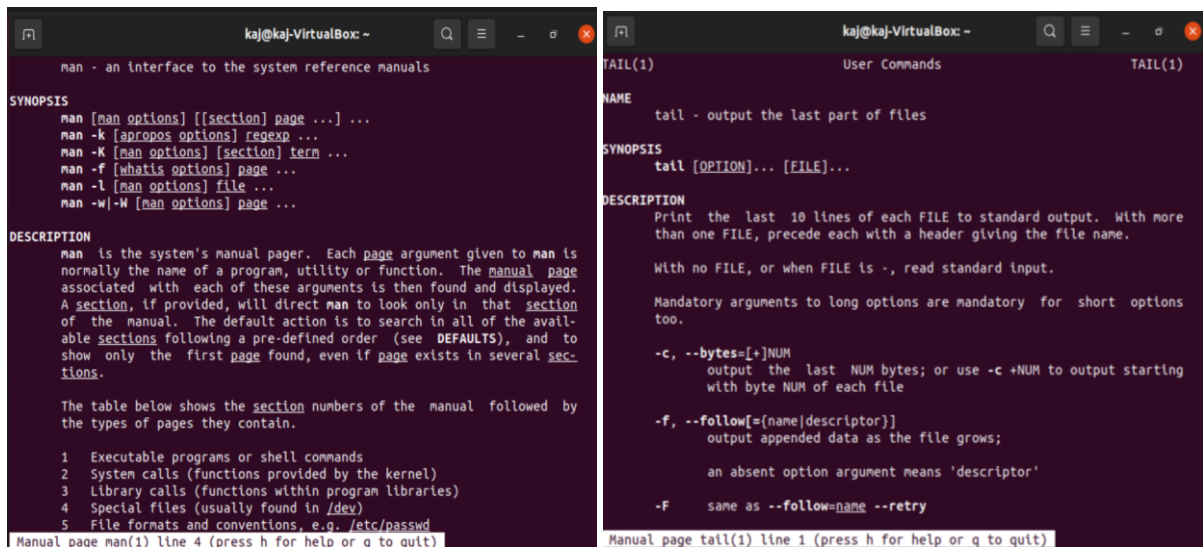
```
kaj@kaj-VirtualBox:~/Documents$ history  
1 gedit abcc.sh  
2 sh abcc.sh  
3 clear  
4 pwd  
5 gedit abc.sh  
6 clear  
7 pwd  
8 gedit abc.sh  
9 sh abc.sh  
10 sh abcc.sh  
11 gedit abc.sh  
12 sh abc.sh  
13 gedit abc.sh  
14 sh abc.sh  
15 gedit abcc.sh  
16 sh abcc.sh  
17 clear  
18 gedit abcc.sh  
19 sh abcc.sh  
20 gedit abc.sh  
21 sh abc.sh  
22 pwd  
23 exit  
24 pwd  
25 history  
26 ls  
27 man  
  
kaj@kaj-VirtualBox:~/Documents/files$  
44 cd Documents/  
45 mkdir files  
46 ls  
47 mkdir files/file1  
48 cd files/  
49 ls  
50 rmdir file1  
51 ls  
52 touch file1.txt  
53 touch file2.txt file3.txt  
54 ls  
55 rm file3.txt  
56 ls  
57 cat file1.txt  
58 cat > file1.txt  
59 cat > file2.txt  
60 cat file1.txt file2.txt  
61 cat file1.txt file2.txt > file3.txt  
62 cat file3.txt  
63 cat file3.txt | tr a-z A-Z  
64 cat file3.txt | tr A-Z a-z  
65 history  
kaj@kaj-VirtualBox:~/Documents/files$ !47  
mkdir files/file1  
mkdir: cannot create directory 'files/fllle1': No such file or directory  
kaj@kaj-VirtualBox:~/Documents/files$ !51  
ls  
file1.txt file2.txt file3.txt  
kaj@kaj-VirtualBox:~/Documents/files$
```

### 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 man

man ls



```
kaj@kaj-VirtualBox: -
man - an interface to the system reference manuals

SYNOPSIS
man [man options] [[section] page ...] ...
man -k [apropos options] regexp ...
man -K [man options] [section] term ...
man -f [whatis options] page ...
man -l [man options] file ...
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 of the available sections following a pre-defined order (see DEFAULTS), and to show only the first page found, even if page 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

Manual page man(1) line 4 (press h for help or q to quit)
```

```
kaj@kaj-VirtualBox: -
TAIL(1) User Commands TAIL(1)

NAME
tail - output the last part of files

SYNOPSIS
tail [OPTION]... [FILE]...

DESCRIPTION
Print the last 10 lines of each FILE to standard output. With more than one FILE, precede each with a header giving the file name.

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

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

-c, --bytes=[+]NUM
output the last NUM bytes; or use -c +NUM to output starting with byte NUM of each file

-f, --follow[=(name|descriptor)]
output appended data as the file grows;
an absent option argument means 'descriptor'

-F same as --follow=name --retry

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

## 4. cd

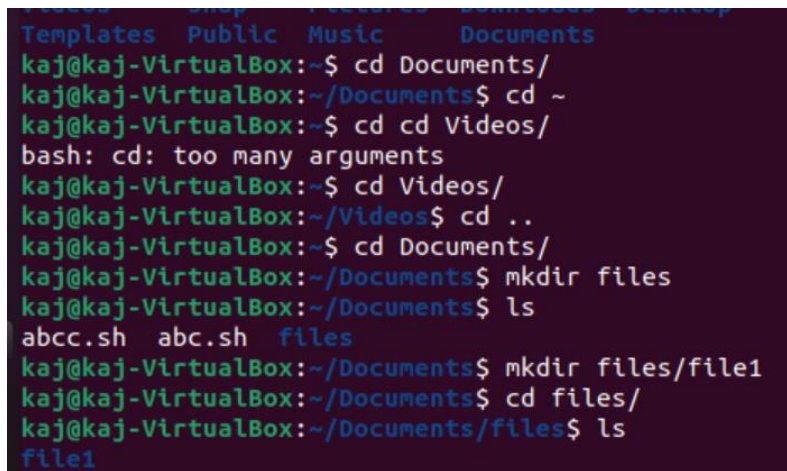
To navigate through the Linux files and directories, use the `cd` It requires either the full path or the name of the directory, depending on the current working directory that you're in.

Shortcuts to help you navigate quickly:

Ø `cd ..` (with two dots) to move one directory up

Ø `cd` to go straight to the home folder

Ø `cd-` (with a hyphen) to move to your previous directory



```
kaj@kaj-VirtualBox: ~$ cd Documents/
kaj@kaj-VirtualBox: ~/Documents$ cd ~
kaj@kaj-VirtualBox: ~$ cd cd Videos/
bash: cd: too many arguments
kaj@kaj-VirtualBox: ~$ cd Videos/
kaj@kaj-VirtualBox: ~/Videos$ cd ..
kaj@kaj-VirtualBox: ~$ cd Documents/
kaj@kaj-VirtualBox: ~/Documents$ mkdir files
kaj@kaj-VirtualBox: ~/Documents$ ls
abcc.sh abc.sh files
kaj@kaj-VirtualBox: ~/Documents$ mkdir files/file1
kaj@kaj-VirtualBox: ~/Documents$ cd files/
kaj@kaj-VirtualBox: ~/Documents/files$ ls
file1
```

## 5. ls

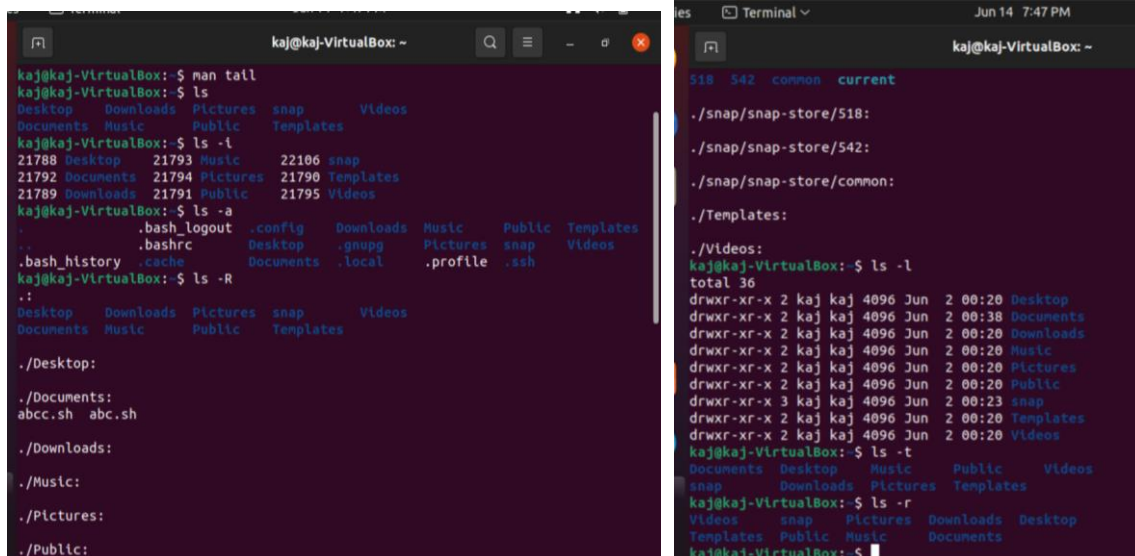
The `ls` command is used to view the contents of a directory.

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.

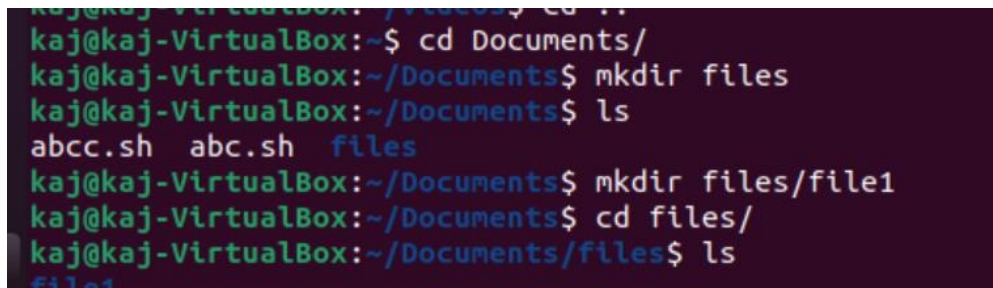


The first terminal window shows the output of `man ls` and `ls -l` in a `kaj@kaj-VirtualBox: ~` shell. The `ls -l` output lists files with permissions, owner, group, size, date, and filename. The second terminal window shows the output of `ls -t` and `ls -r` in the same shell, demonstrating how these options sort files by last modified time and reverse the order, respectively.

## 6. mkdir

Use `mkdir` command to make a new directory .

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



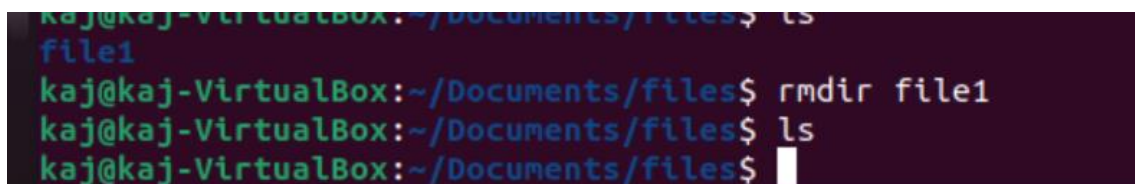
The terminal shows the following commands and output in a `kaj@kaj-VirtualBox: ~` shell:

```
kaj@kaj-VirtualBox: ~$ cd Documents/
kaj@kaj-VirtualBox: ~/Documents$ mkdir files
kaj@kaj-VirtualBox: ~/Documents$ ls
abcc.sh  abc.sh  files
kaj@kaj-VirtualBox: ~/Documents$ mkdir files/file1
kaj@kaj-VirtualBox: ~/Documents$ cd files/
kaj@kaj-VirtualBox: ~/Documents/files$ ls
file1
```

## 7. rmdir

If you need to delete a directory, use the `rmdir` command.

However, `rmdir` only allows you to delete empty directories.



The terminal shows the following commands and output in a `kaj@kaj-VirtualBox: ~/Documents/files` shell:

```
kaj@kaj-VirtualBox: ~/Documents/files$ ls
file1
kaj@kaj-VirtualBox: ~/Documents/files$ rmdir file1
kaj@kaj-VirtualBox: ~/Documents/files$ ls
```

## 8. touch

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

```
kaj@kaj-VirtualBox:~/Documents/files$ ls
kaj@kaj-VirtualBox:~/Documents/files$ touch file1.txt
kaj@kaj-VirtualBox:~/Documents/files$ touch file2.txt file3.txt
kaj@kaj-VirtualBox:~/Documents/files$ ls
file1.txt  file2.txt  file3.txt
kaj@kaj-VirtualBox:~/Documents/files$
```

## 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**

```
kaj@kaj-VirtualBox:~/Documents/files$ ls
file1.txt  file2.txt  file3.txt
kaj@kaj-VirtualBox:~/Documents/files$ rm file3.txt
kaj@kaj-VirtualBox:~/Documents/files$ ls
file1.txt  file2.txt
kaj@kaj-VirtualBox:~/Documents/files$
```

## 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:

Ø **cat > filename** creates a new file

Ø **cat filename1 filename2>filename3** joins two files (1 and 2) and stores the output of them in a new file (3)

Ø to convert a file to upper or lower case use, **cat filename | tr a-z A-Z >output.txt**

Ø **cat >>myfile** insert data to a file

```
file1.txt  file2.txt
kaj@kaj-VirtualBox:~/Documents/files$ cat file1.txt
kaj@kaj-VirtualBox:~/Documents/files$ cat > file1.txt
Hello World
I'm Anilect Jose
kaj@kaj-VirtualBox:~/Documents/files$ cat > file2.txt
Also I'm from AJCE
kaj@kaj-VirtualBox:~/Documents/files$ cat file1.txt file2.txt
Hello World
I'm Anilect Jose
Also I'm from AJCE
kaj@kaj-VirtualBox:~/Documents/files$ cat file1.txt file2.txt > file3.txt
kaj@kaj-VirtualBox:~/Documents/files$ cat file3.txt
Hello World
I'm Anilect Jose
Also I'm from AJCE
kaj@kaj-VirtualBox:~/Documents/files$ cat file3.txt | tr a-z A-Z
HELLO WORLD
I'M ANILECT JOSE
ALSO I'M FROM AJCE
kaj@kaj-VirtualBox:~/Documents/files$ cat file3.txt | tr A-Z a-z
hello world
i'm anilect jose
also i'm from ajce
kaj@kaj-VirtualBox:~/Documents/files$
```