EXPERIMENT-1

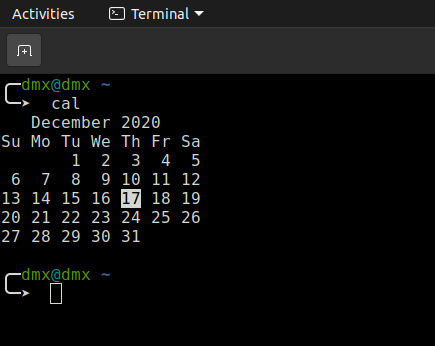
Aim: Study of UNIX commands with all their important options.

Study of following commands

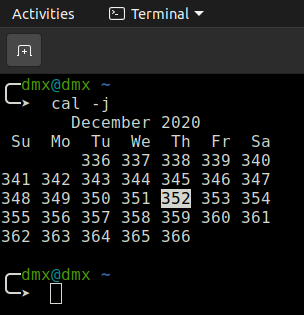
**a) Information Management**

1. **Cal**

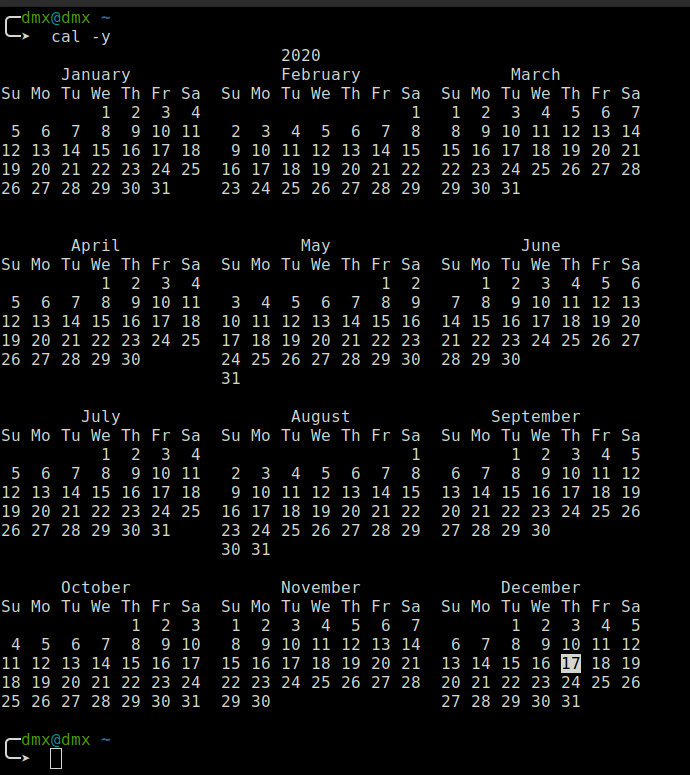
It will show the calendar



**1.1) Cal -j**: - show the Julian year

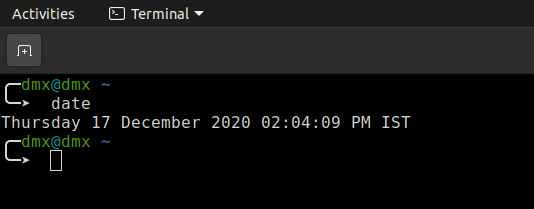


* 1. **) Cal -y**: - show the current year calendar

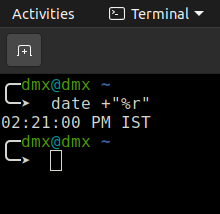


1. **date**

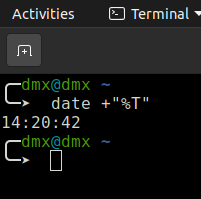
It will show the current day and date with time



**2.1) date +"%r”:** - To display locale’s 12-hour clock time

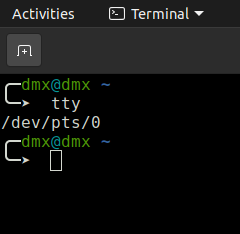


**2.2) date +"%T”:** - Current time

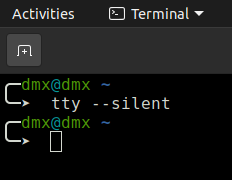


1. **tty**

The tty command of terminal basically prints the file name of the terminal connected to standard input

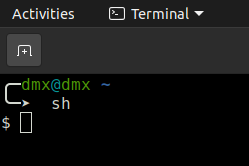


**3.1) --silent, --quiet** print nothing, only return an exit status



1. **sh**

sh is a command language interpreter that executes commands read from a command line string, the standard input, or a specified file



1. **env**

It can be used to print a list of the current environment variables, or to run another program in a custom environment without modifying the current one

**5.1)** **-i, --ignore** - environment start with an empty environment

**5.2) -0, --null -** end each output line with NUL, not newline

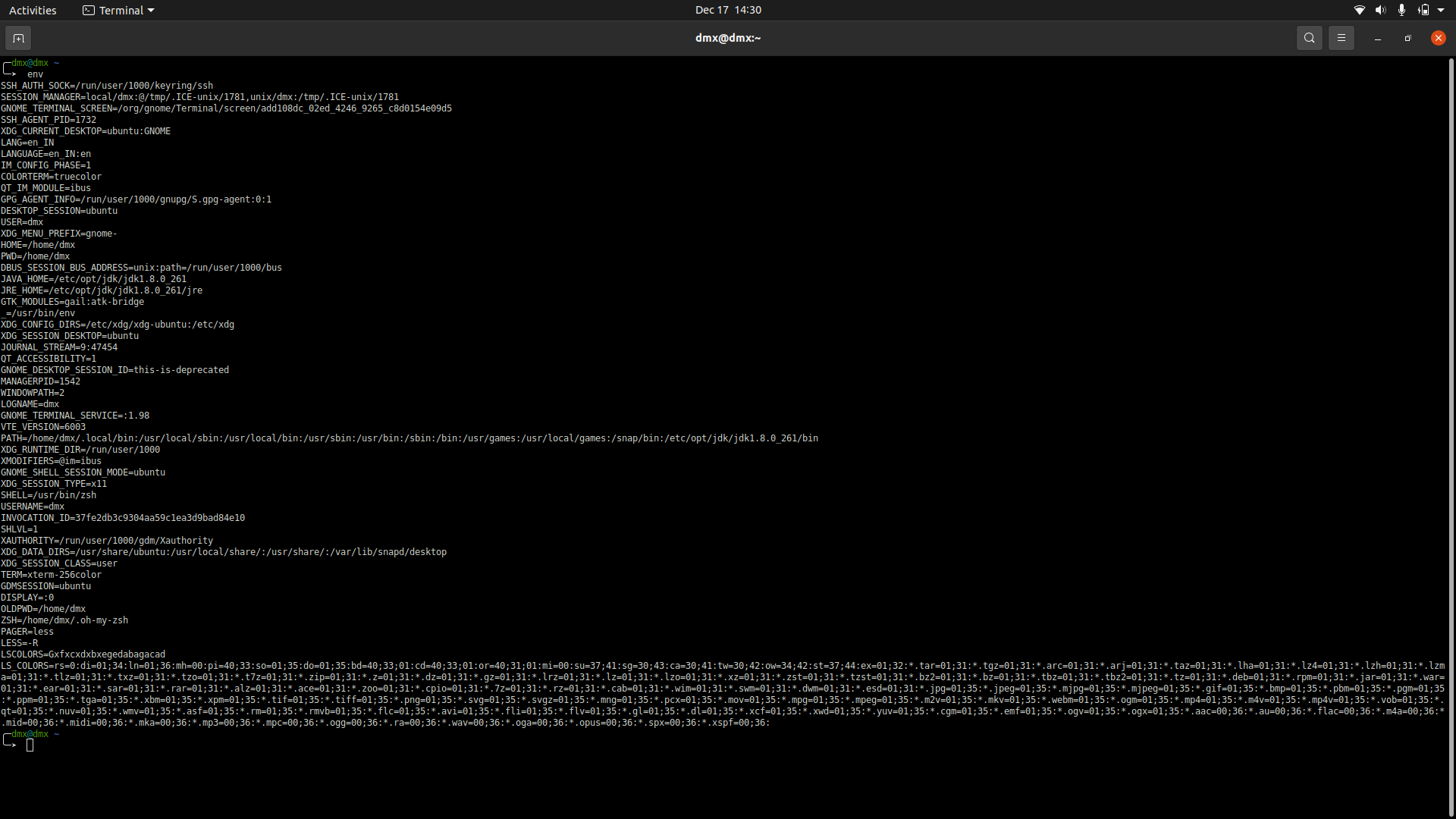
**5.3)** -**u, --unset** = NAME remove variable from the environment

**5.4) -C, --chdir** = DIR change working directory to DIR

**5.5)** **-S, --split-string** = S process and split S into separate arguments;

used to pass multiple arguments on shebang lines

**5.6)** **-v, --debug** - print verbose information for each processing step



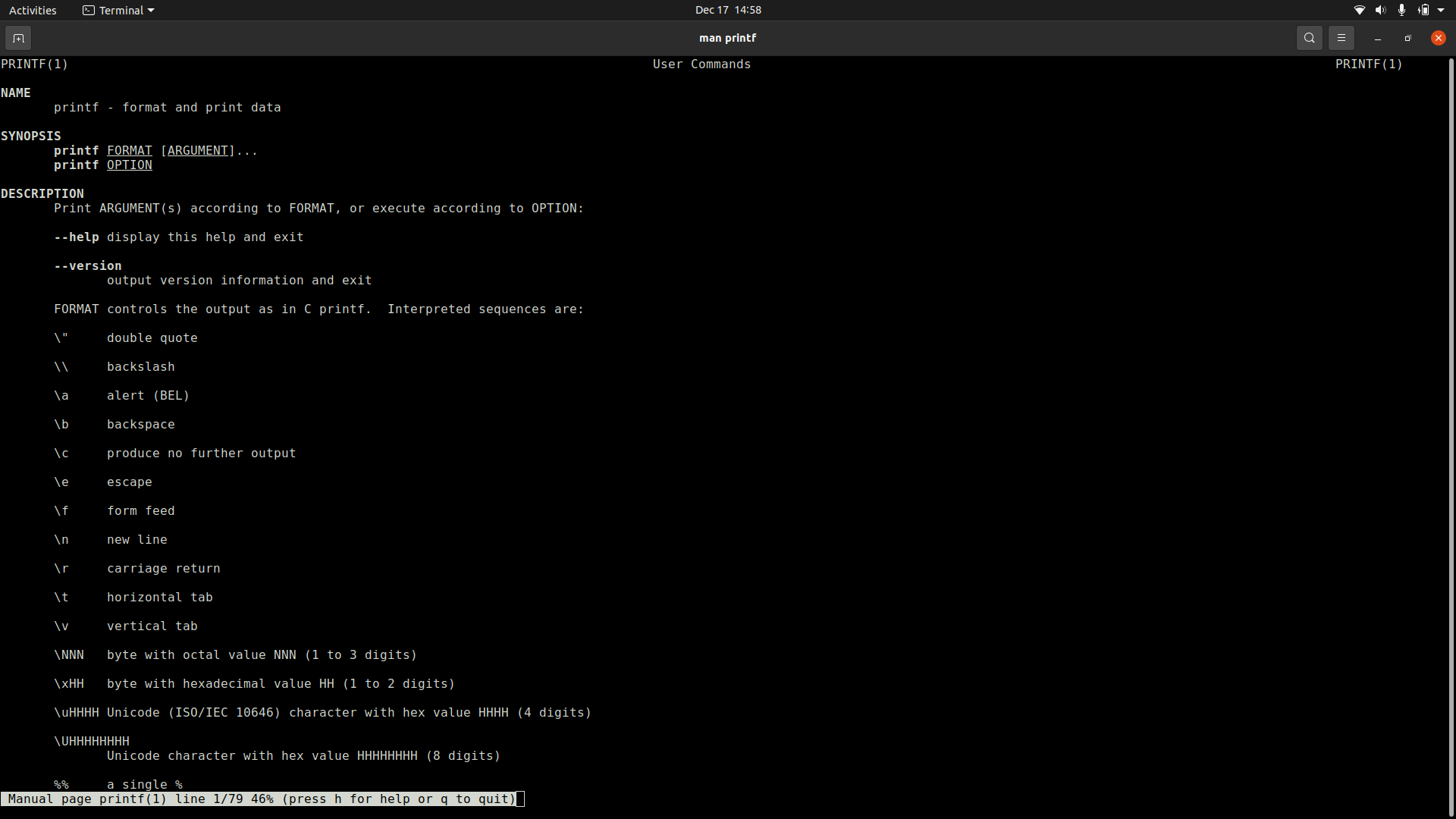
1. **set**

Linux set command is used to set and unset certain flags or settings within the shell environment.

(For Set SS was too long to take)

1. **Man**

man, command in Linux is used to display the user manual of any command that we can run on the terminal



**7.1) -f, --whatis** = equivalent to whatis

**7.2) -k, --apropos** = equivalent to apropos

**7.3) -K, --global-apropos** = search for text in all pages

**7.4) -l, --local-file** = interpret PAGE argument(s) as local filename(s)

**7.5)** -**w, --where, --path, --location** = print physical location of man page(s)

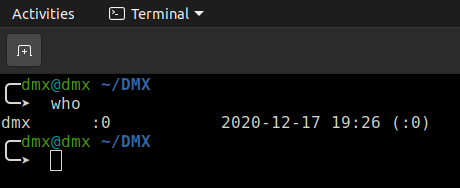
**7.6)** -**W, --where-cat, --location-cat =** print physical location of cat file(s)

**7.7)** -**c, --catman** = used by catman to reformat out of date cat pages

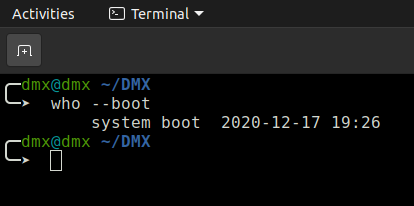
**7.8)** **-R, --recode** = ENCODING output source page encoded in ENCODING

1. **who**

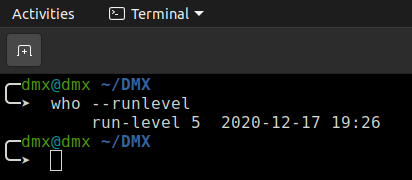
displays a list of users who are currently logged into the computer.



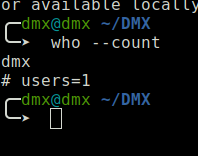
**8.1) who –boot**: - last booting time



**8.2) who –runlevel**: - print current runlevel

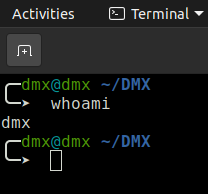


**8.3) who –count:** - count the total user



1. **whoami**

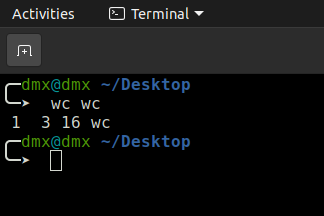
It displays the username of the current user



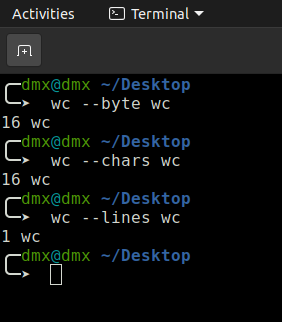
**b) Utility commands**

**1) wc**

It counts the total word in the file.

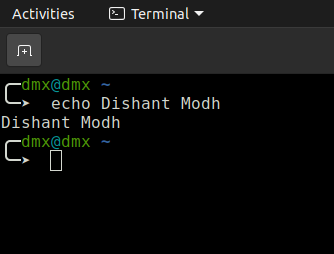


* 1. **--bytes** = print the byte counts
  2. **--chars** = print the character counts
  3. **--lines** = print the newline counts



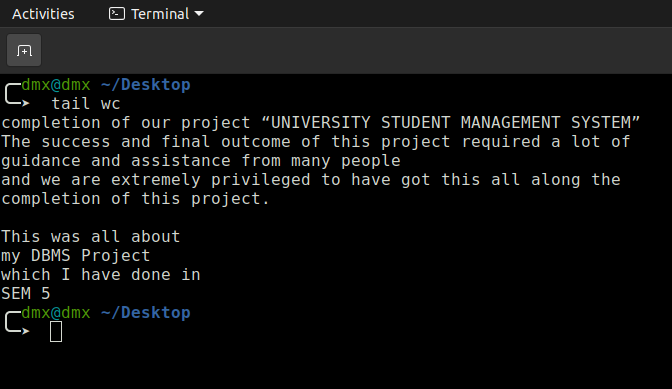
**2) echo**

Is used to display line of text/string that are passed as an argument



**3) tail**

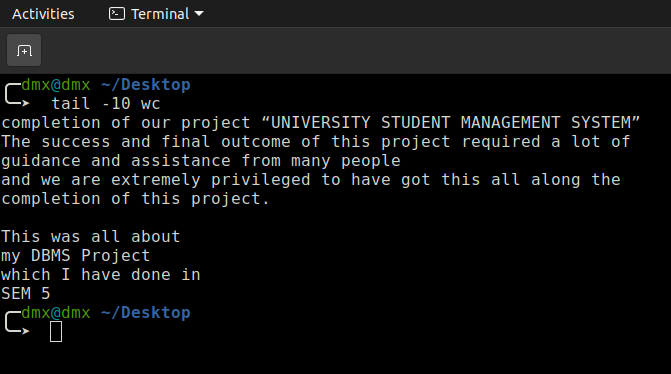
The tail command is a command-line utility for outputting the last part of files given to it via standard input. It writes results to standard output. By default tail returns the last ten lines of each file that it is given.



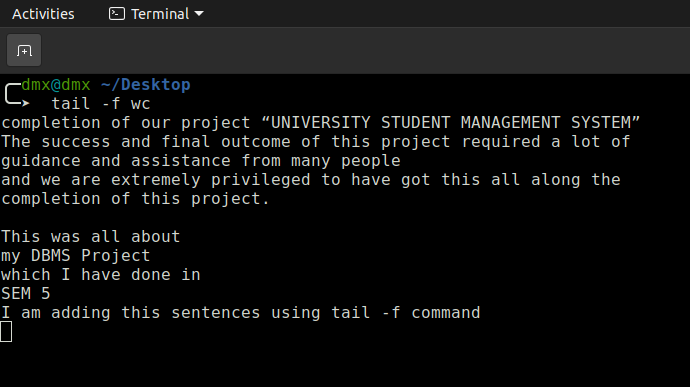
**3.1) tail -n:** - Where n stands for the number of line to be shown.

e.g.

**tail** **-10 wc**

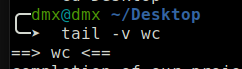


**3.2) tail -f wc:** - To show a real-time, streaming output of a changing file, use the -f



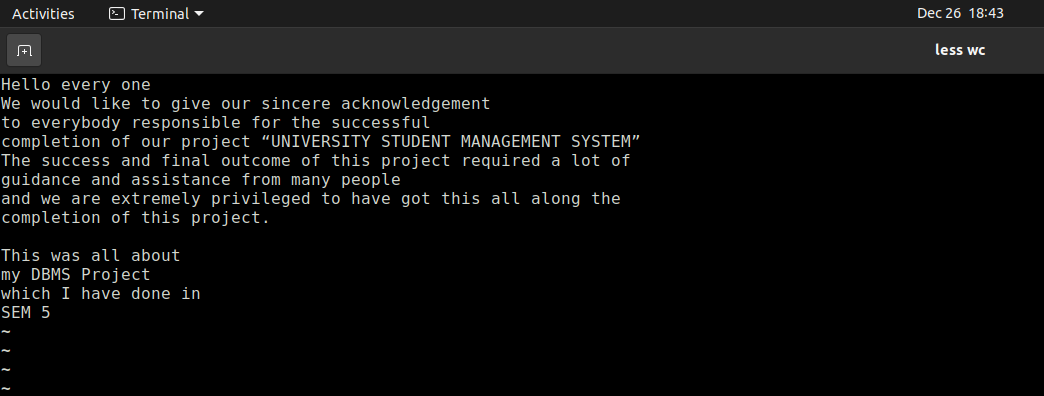
**3.3)** **tail -v wc**: -

**-v, --verbose =** always output headers giving file names



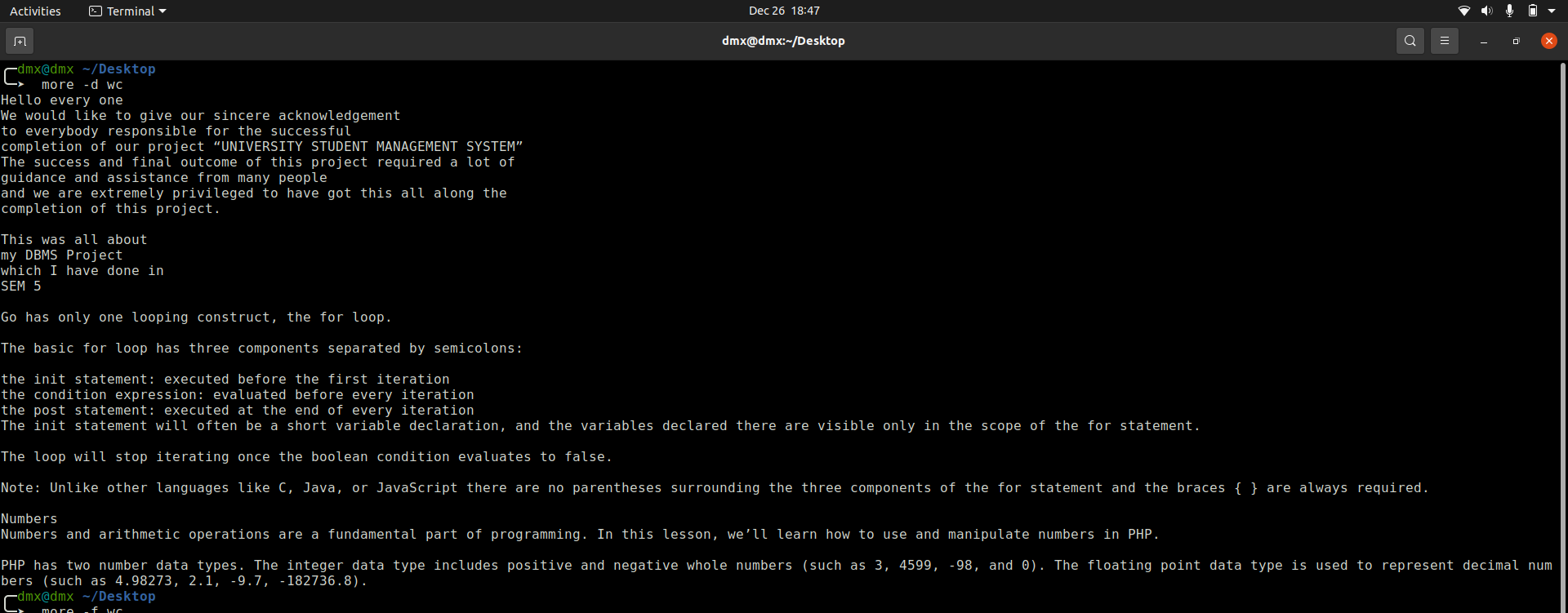
**4) less**

Less is a command line utility that displays the contents of a file or a command output, one page at a time. It is similar to more, but has more advanced features and allows you to navigate both forward and backward through the file.

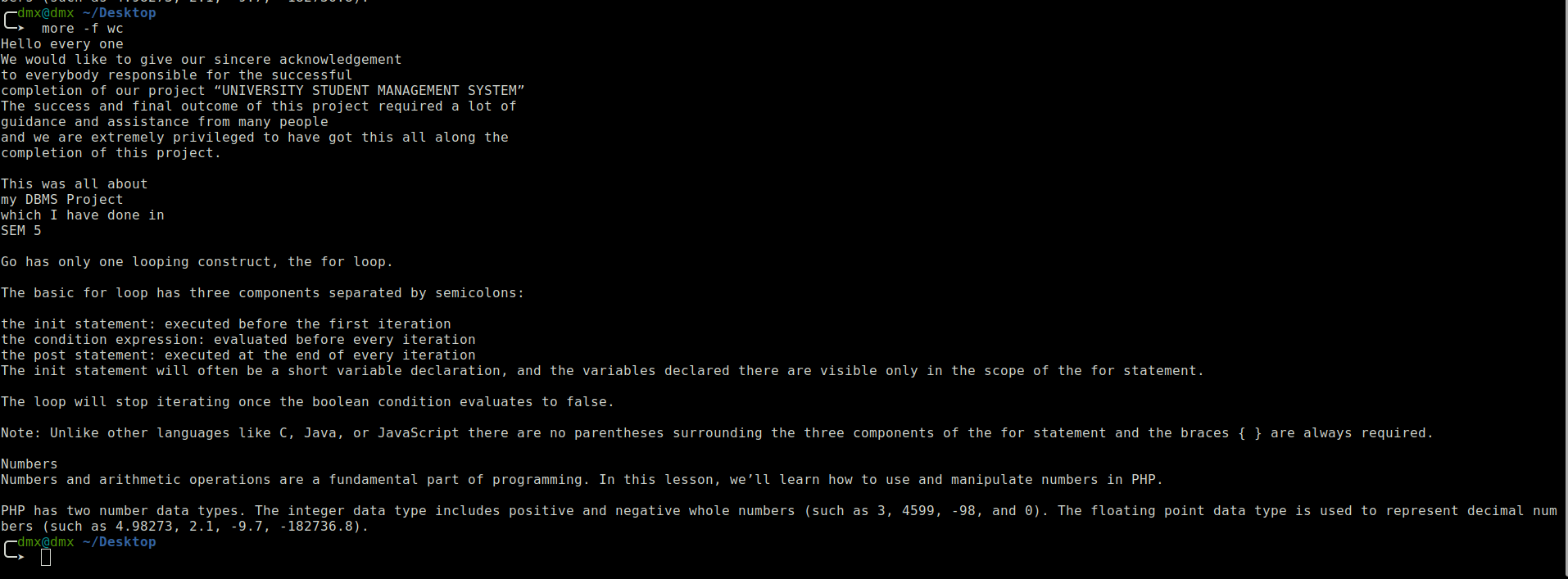


**5) more**

more command is used to view the text files in the command prompt, displaying one screen at a time in case the file is large

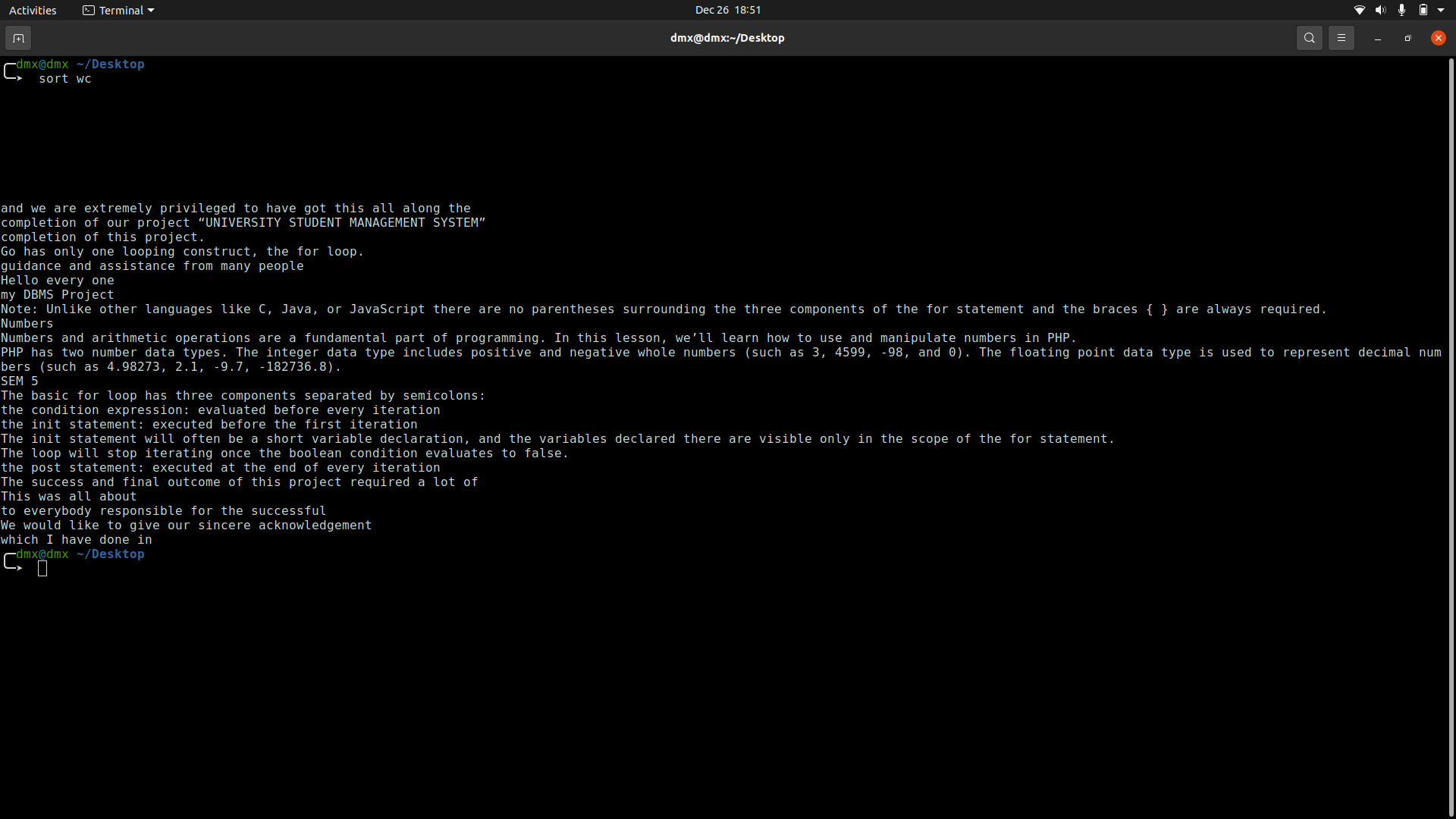
**5.1)** **man -d wc**

**5.2) man -f wc**



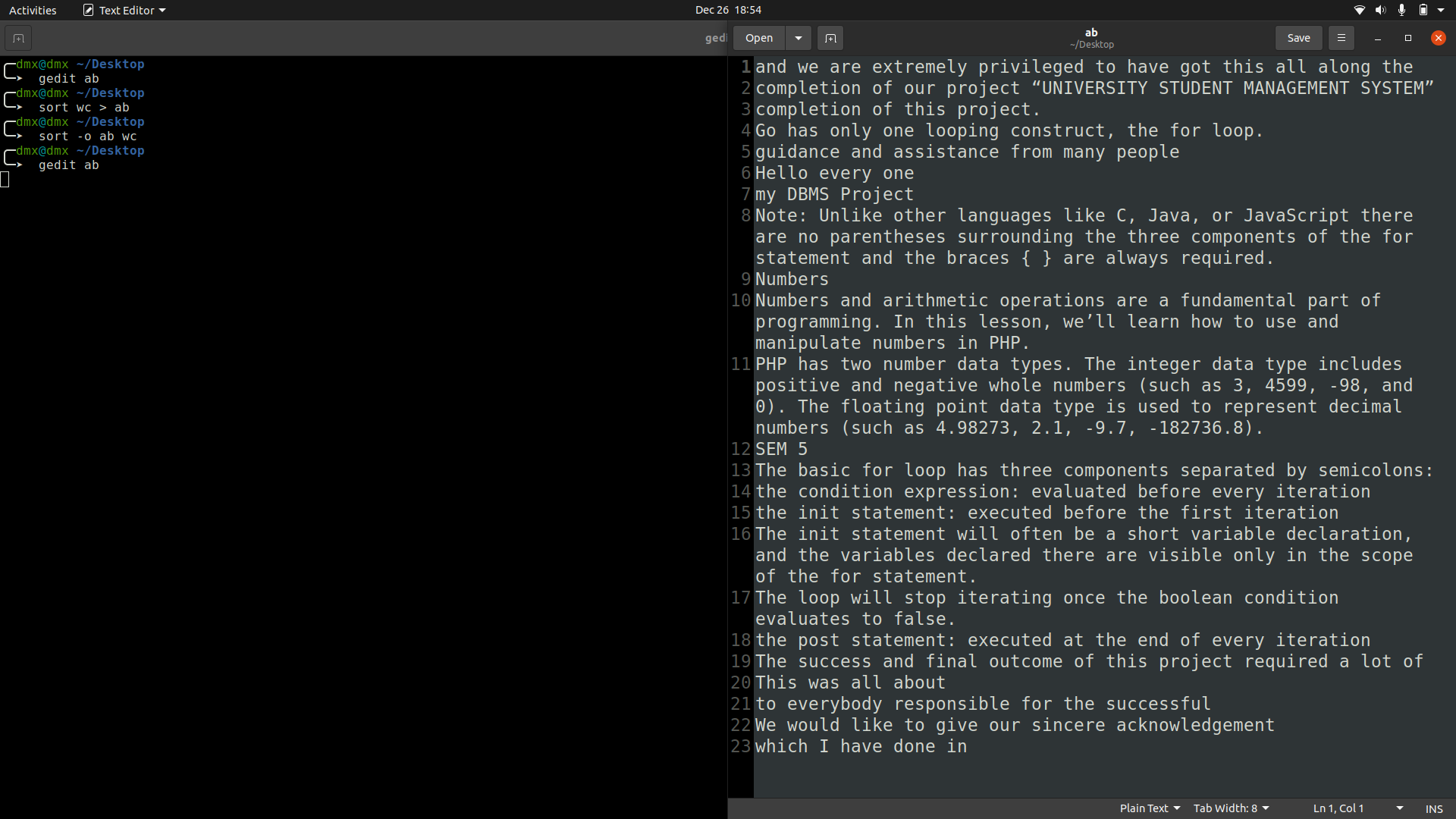
**6) sort**

SORT command is used to sort a file, arranging the records in a particular order.



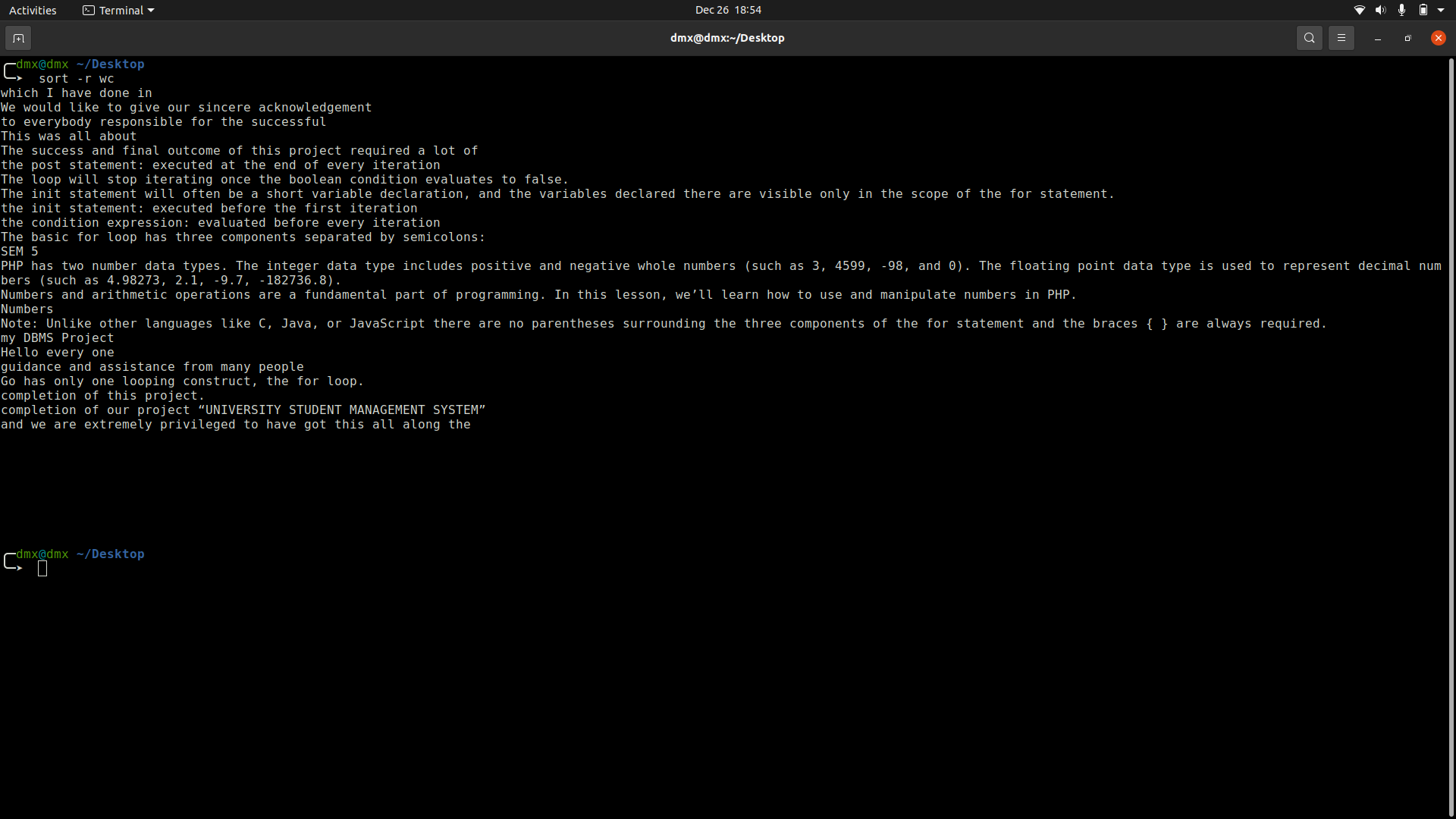
**6.1) sort -o ab wc**

Using the -o option is functionally the same as redirecting the output to a file.



**6.2) sort -r wc**

reverse order



**7) grep**

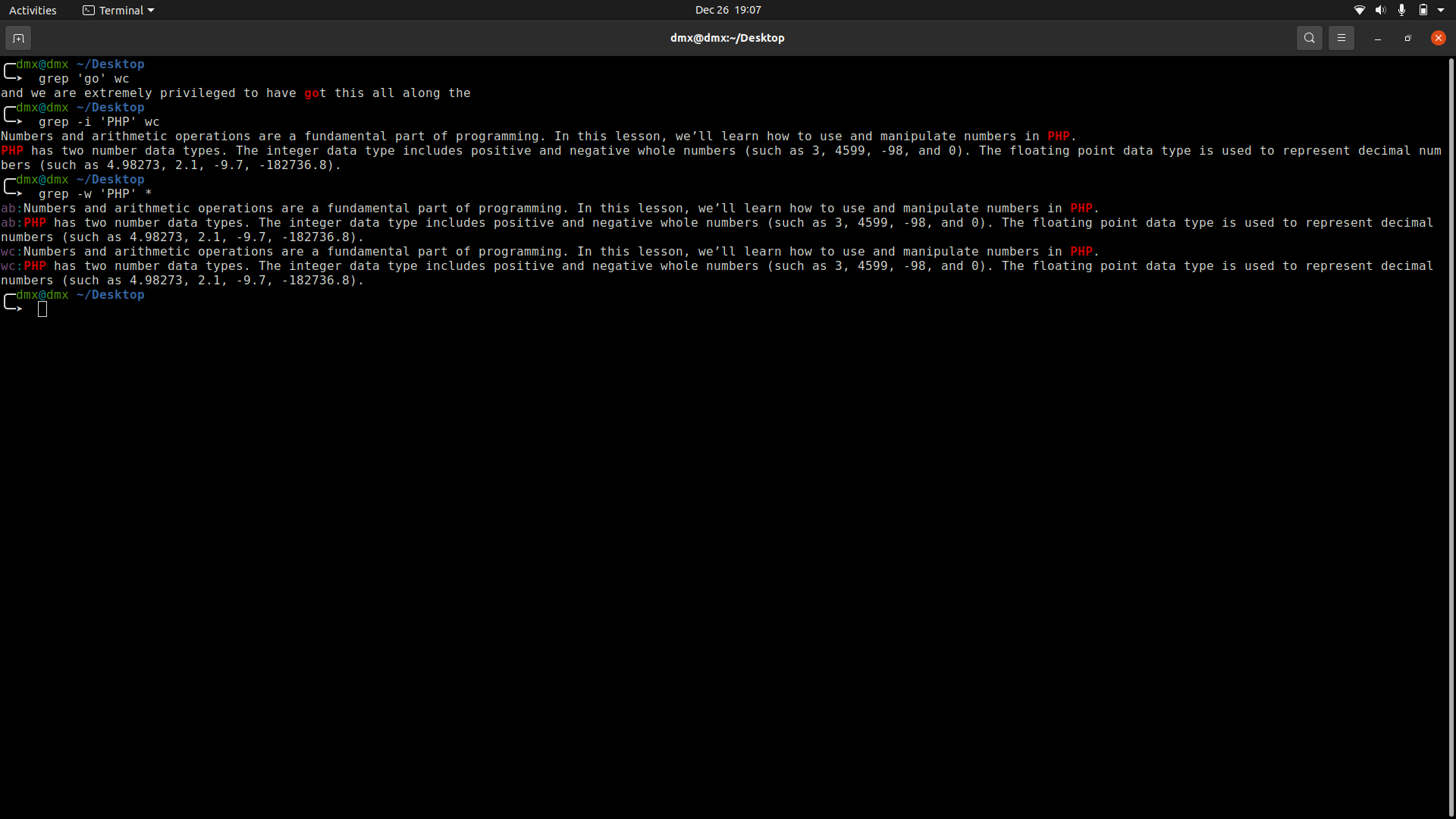
grepis an acronym that stands for Global regular Expression print.

Grep is a Linux / Unix command-line tool used to search for a string of characters in a specified file.

**7.1) grep ‘find’ filename**

**7.2) grep -i ‘find’ filename: - ignore**

**7.3) grep -w ‘find’ \*:** - Find the word in the whole directory

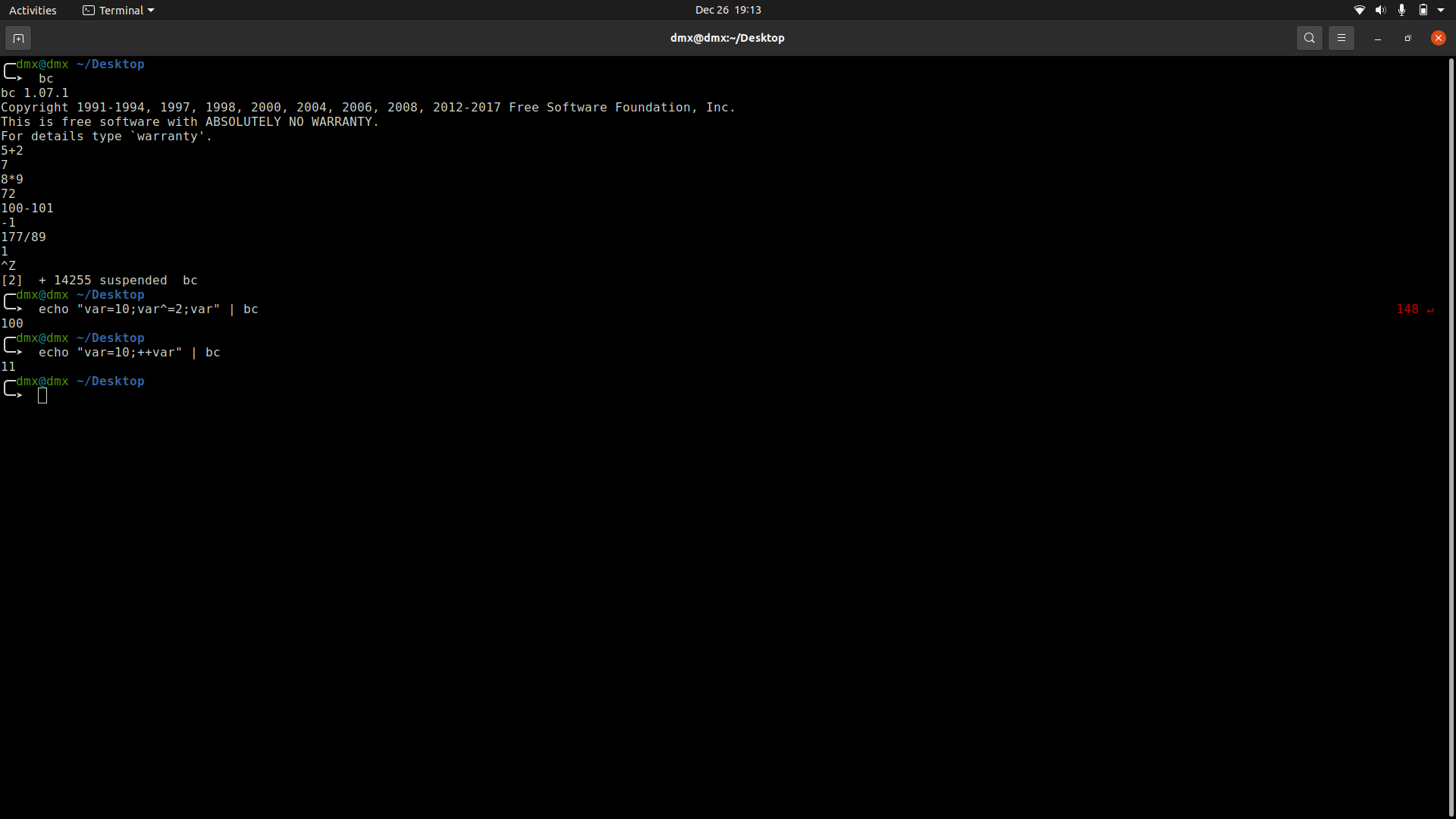


**8) bc**

bc command is used for command line calculator.

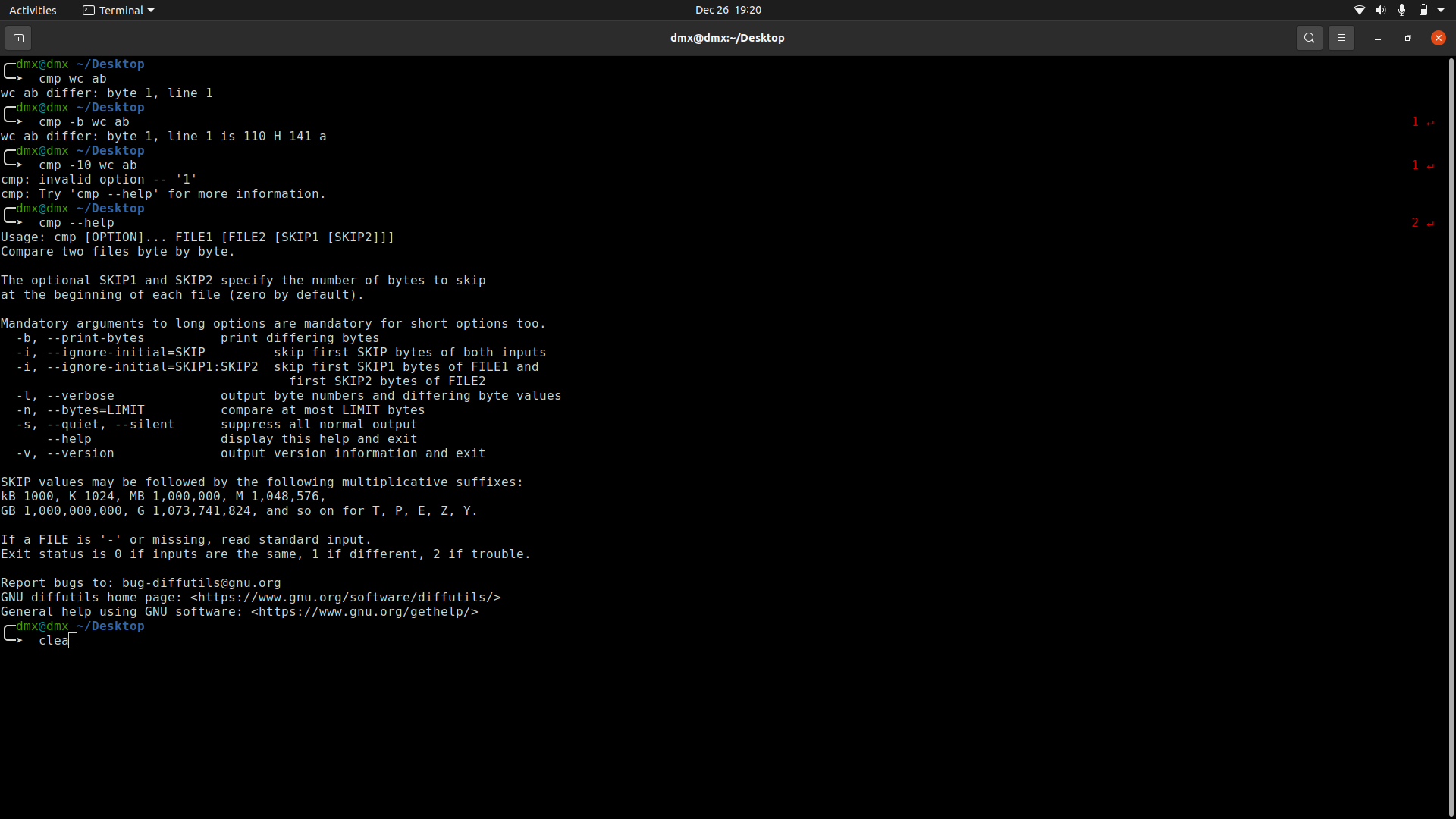
**8.1) echo "var=10; var^=2; var" | bc**

**8.2) echo “var=10; ++var;” | bc**



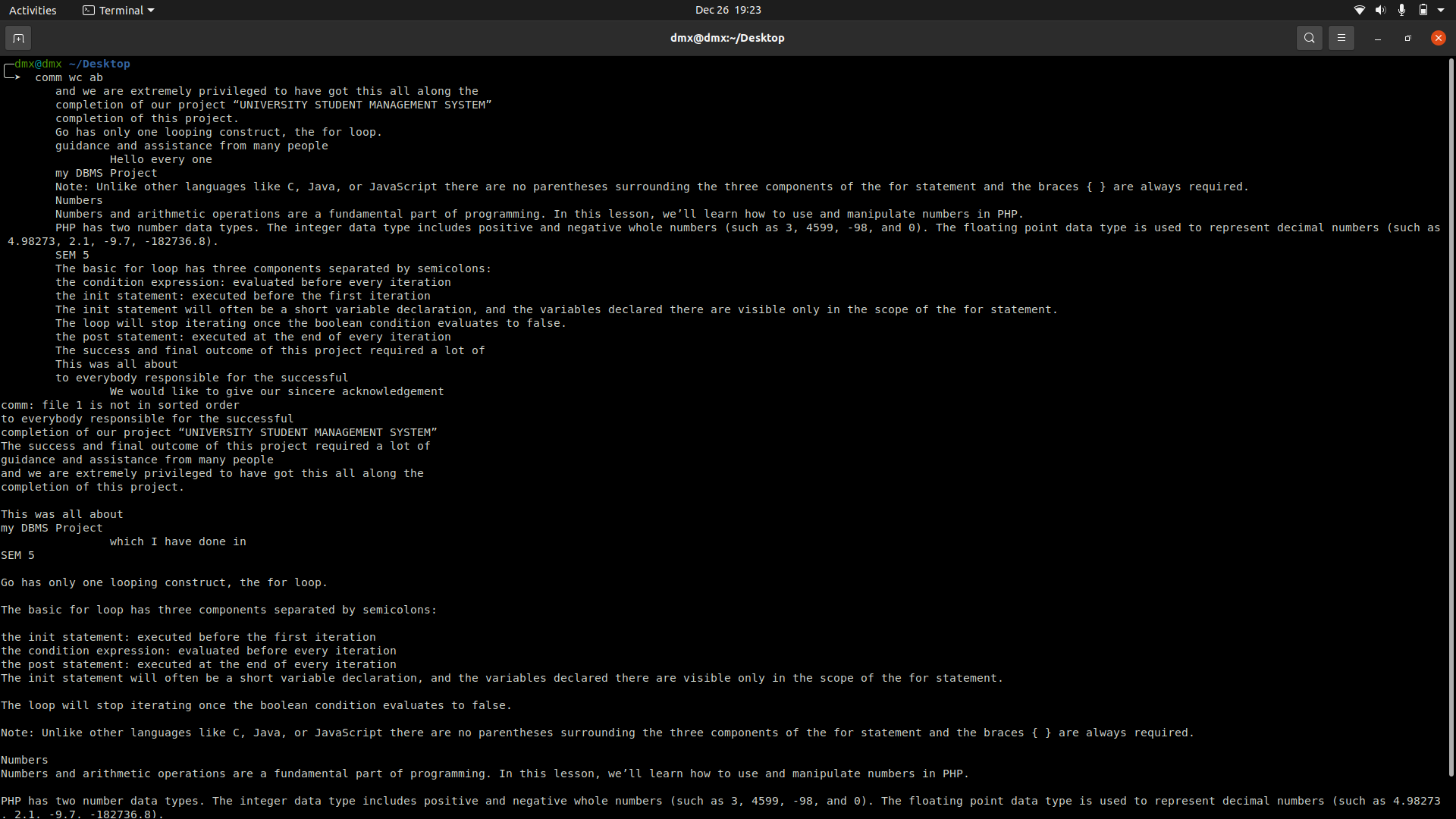
1. **cmp**

cmp command in Linux/UNIX is used to compare the two files byte by byte and helps you to find out whether the two files are identical or not.

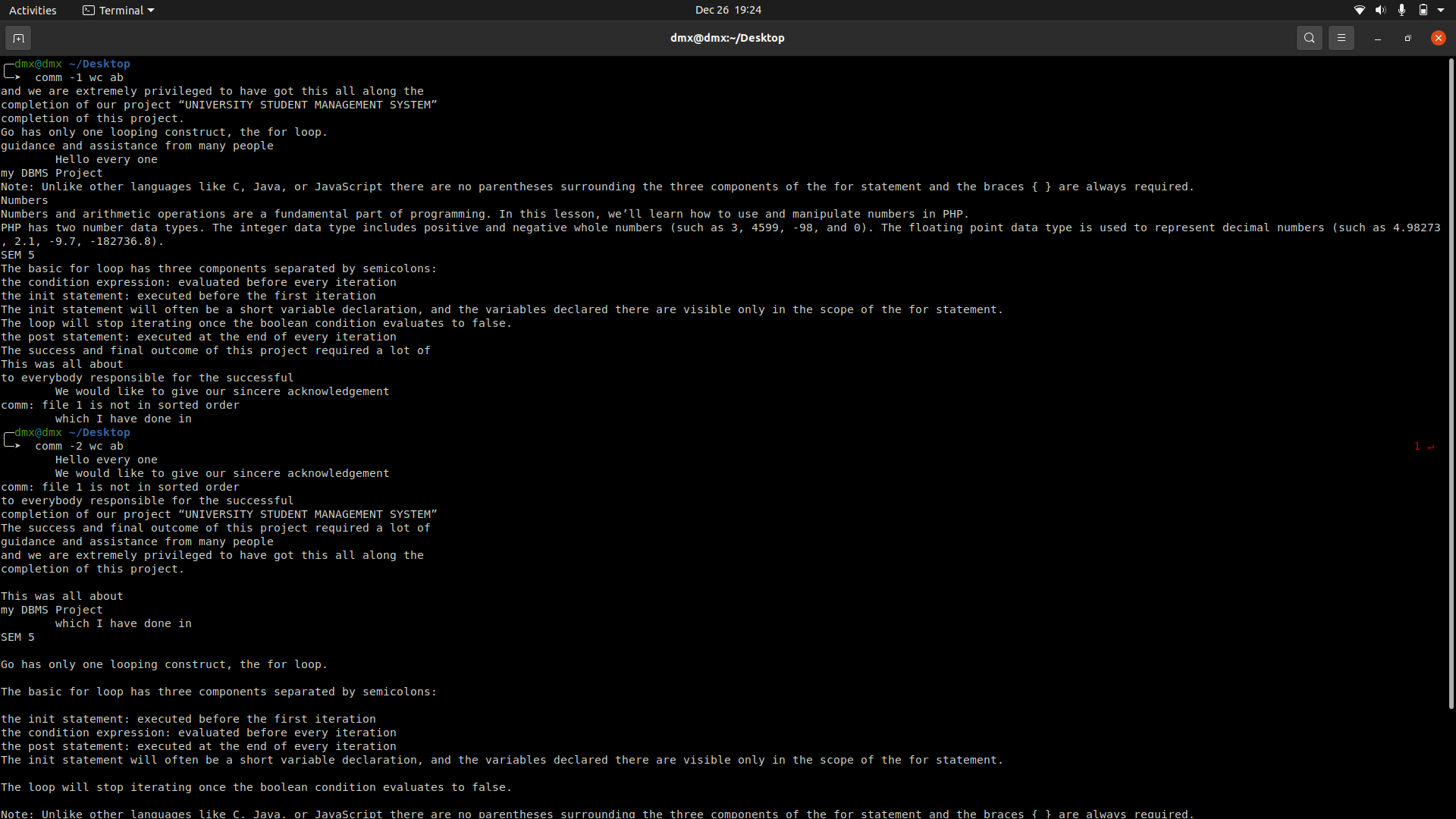


1. **Comm**

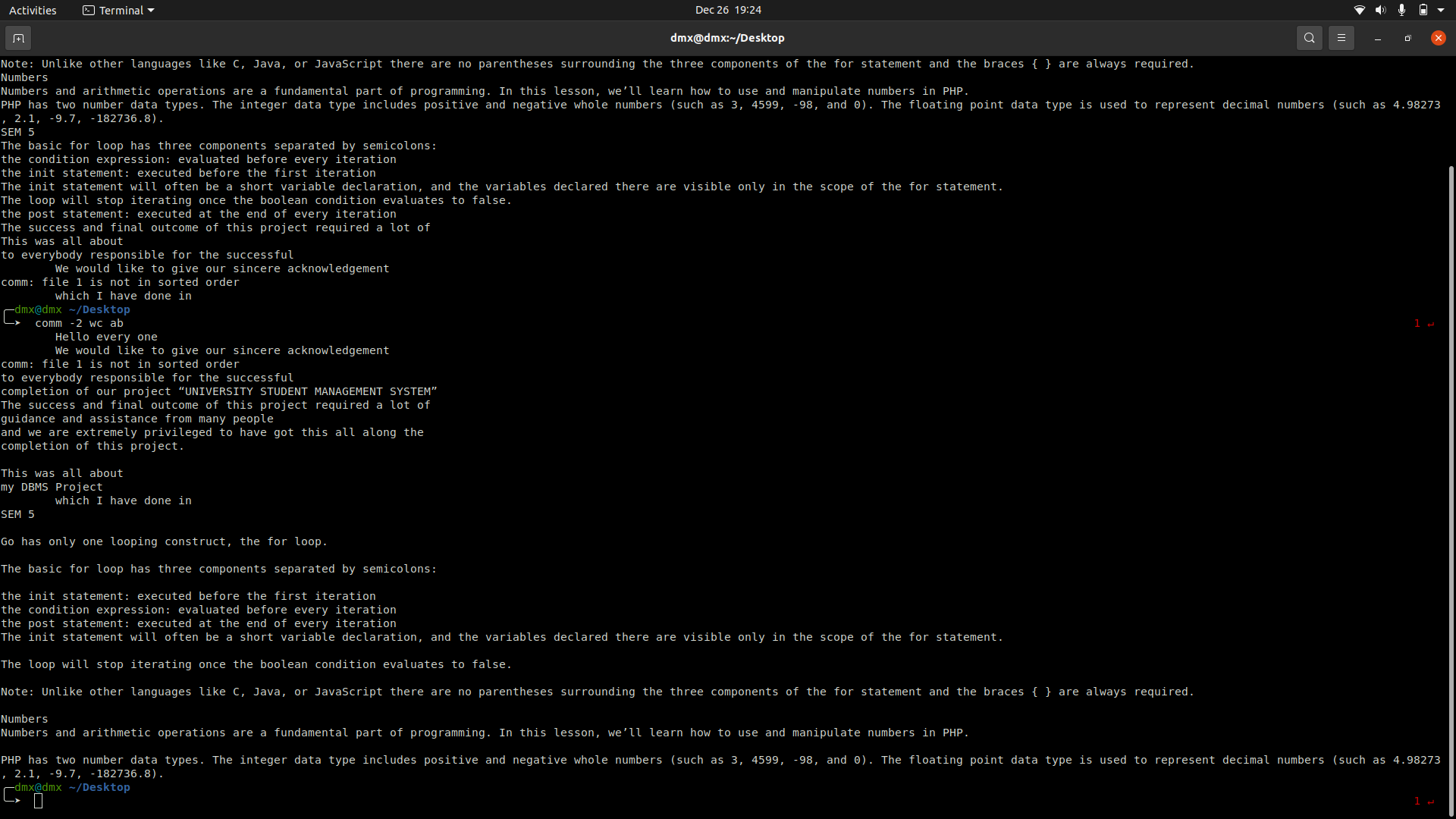
comm compare two sorted files line by line and write to standard output; the lines that are common and the lines that are unique.



**10.1) -1: suppress first column (lines unique to first file).**



**10.2) -2: suppress second column (lines unique to second file)**



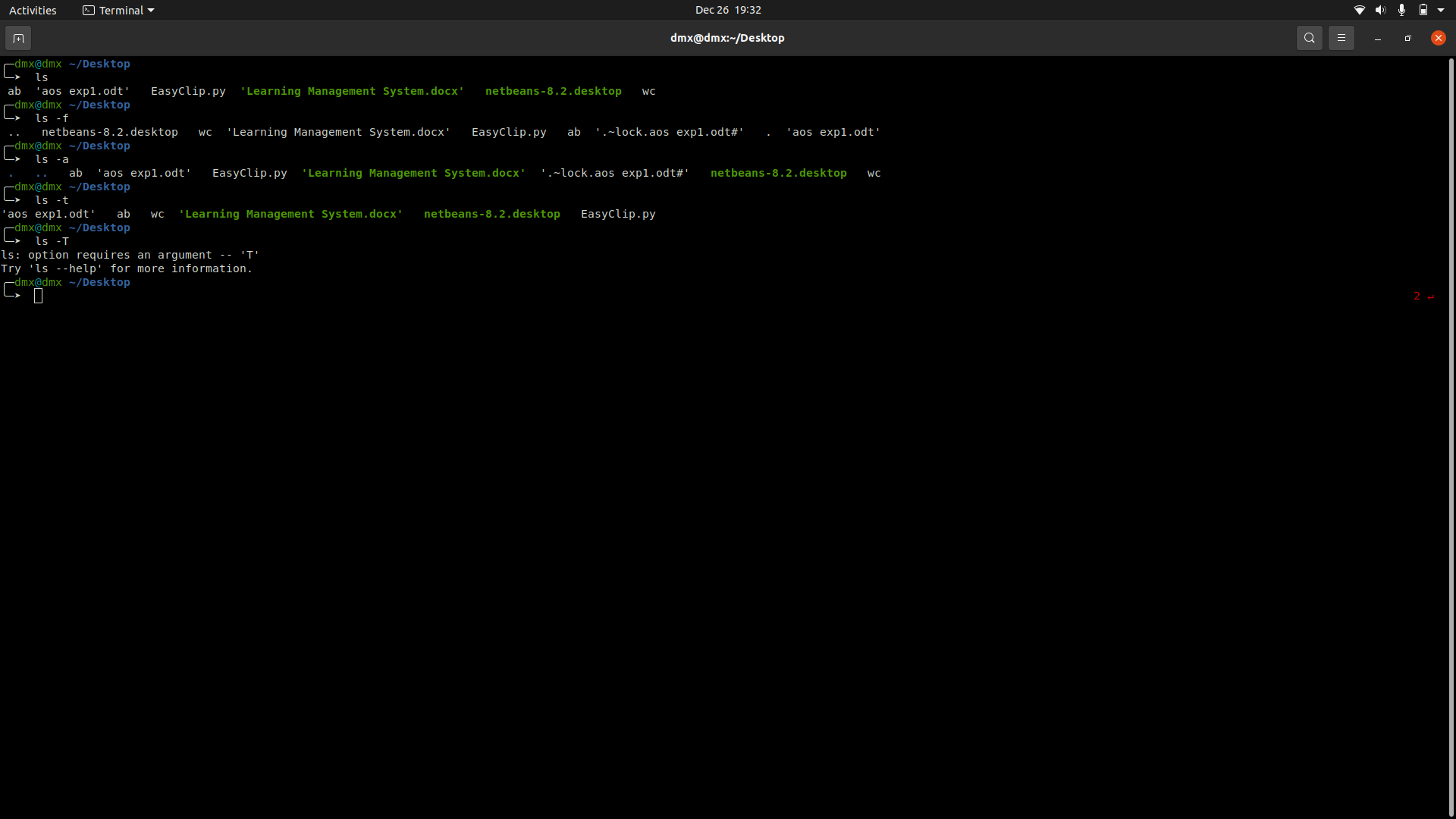
**c) File System Management**

1. ls

List the all-fine name

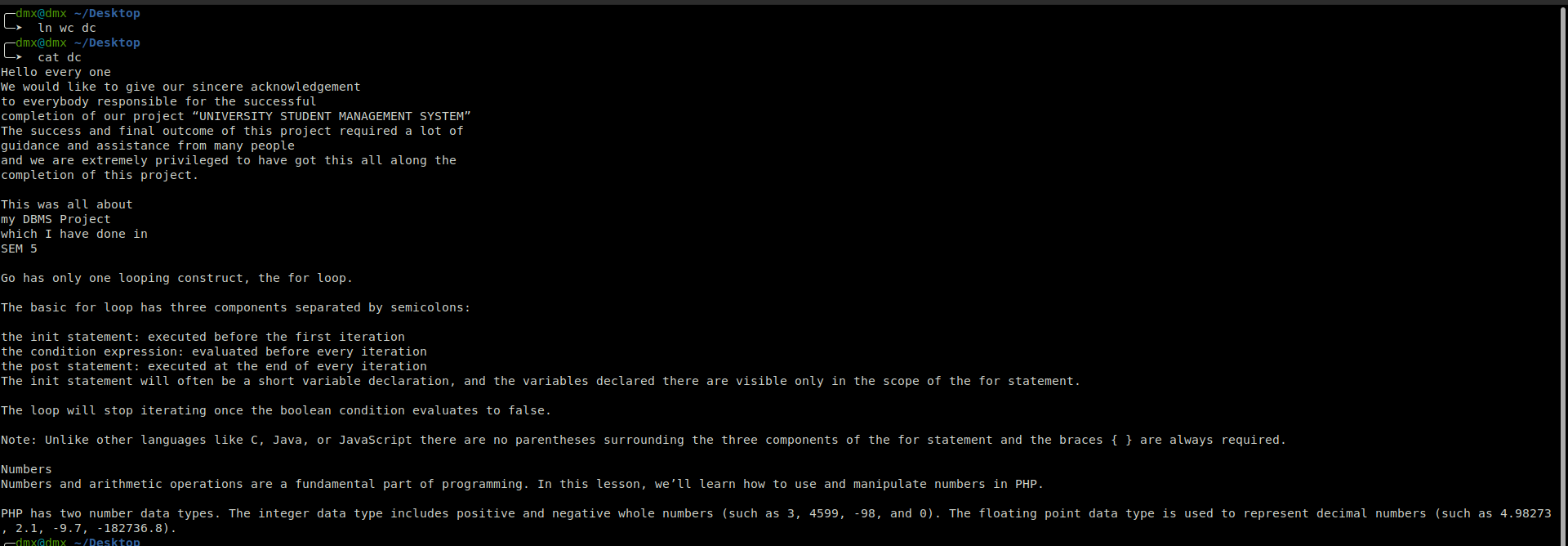
* 1. ls -f: - add one char of \*/=>@| to entries
  2. ls -a: - Show hidden files.

**1.3)** ls -t: - Short by date or time.



1. ln

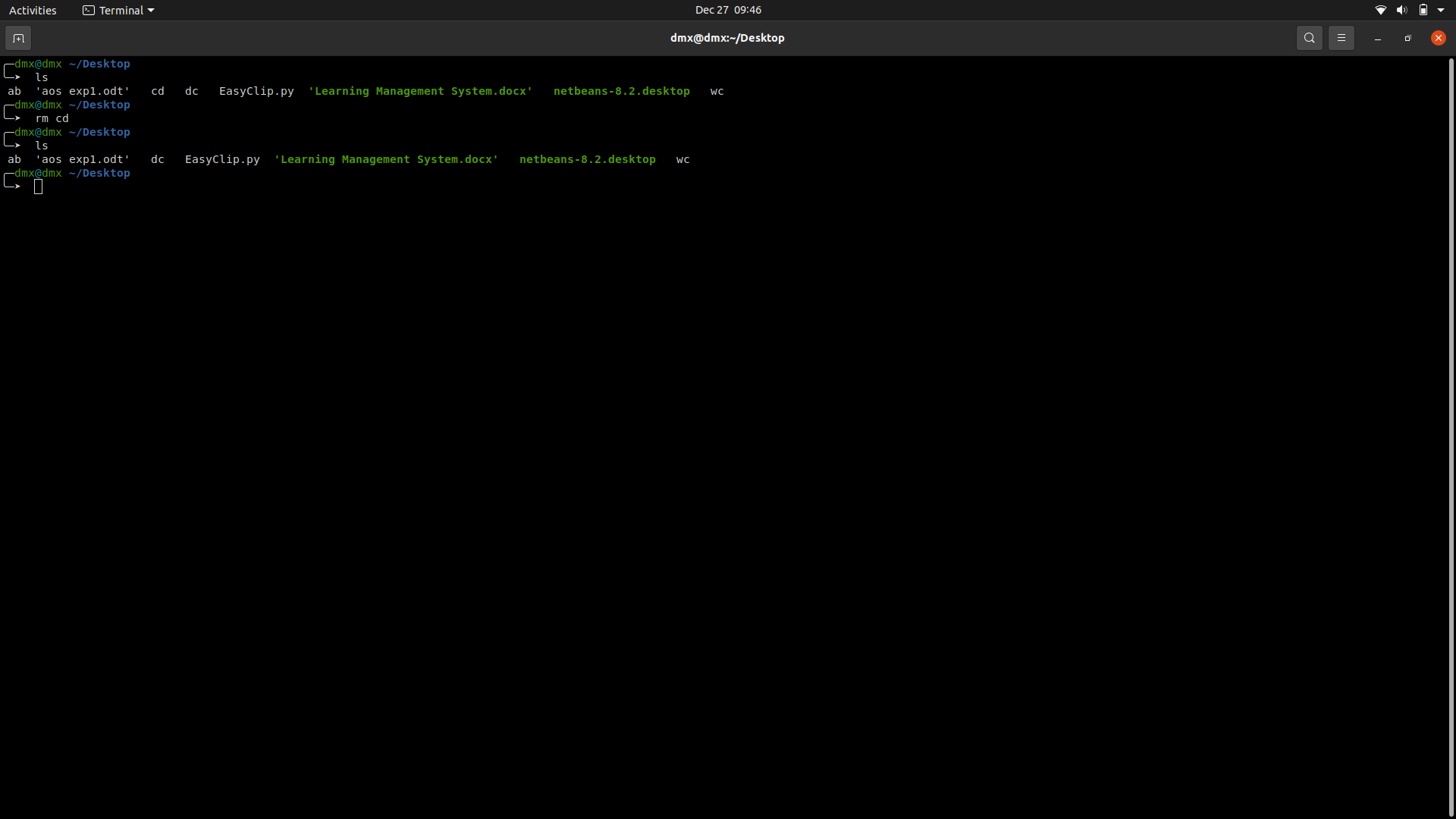
ln - make links between files



1. rm

To remove a file or directory

**3.1)** rm filename



**3.2) rm -d dirname**: - to the Dir.

**3.3) rm filename1 filename2 filename3**: - to remove the multiple file

**3.4) rm \*.png**: - This will remove all the file with extension .png

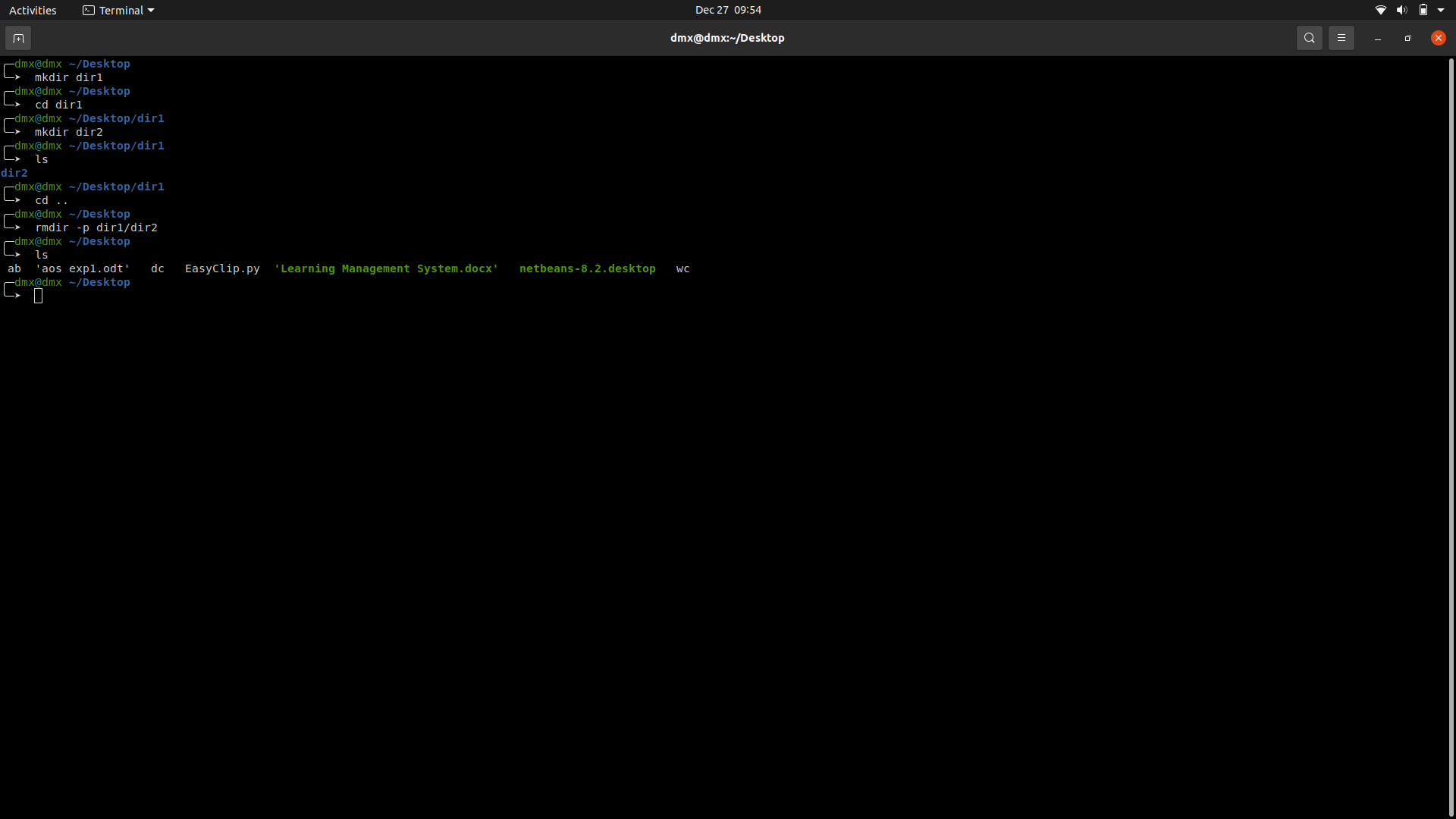
1. **rmdir**

rmdir command is used remove empty directories from the filesystem in Linux.

**4.1)** r**mdir -p**: -In this option each of the directory argument is treated as a pathname of which all components will be removed, if they are already empty, starting from the last component.

**4.2) rmdir -v, –verbose**: -This option displays verbose information for every directory being processed.

**4.3) rmdir –ignore-fail-on-non-empty**: -This option do not report a failure which occurs solely because a directory is non-empty.



1. **Mkdir**

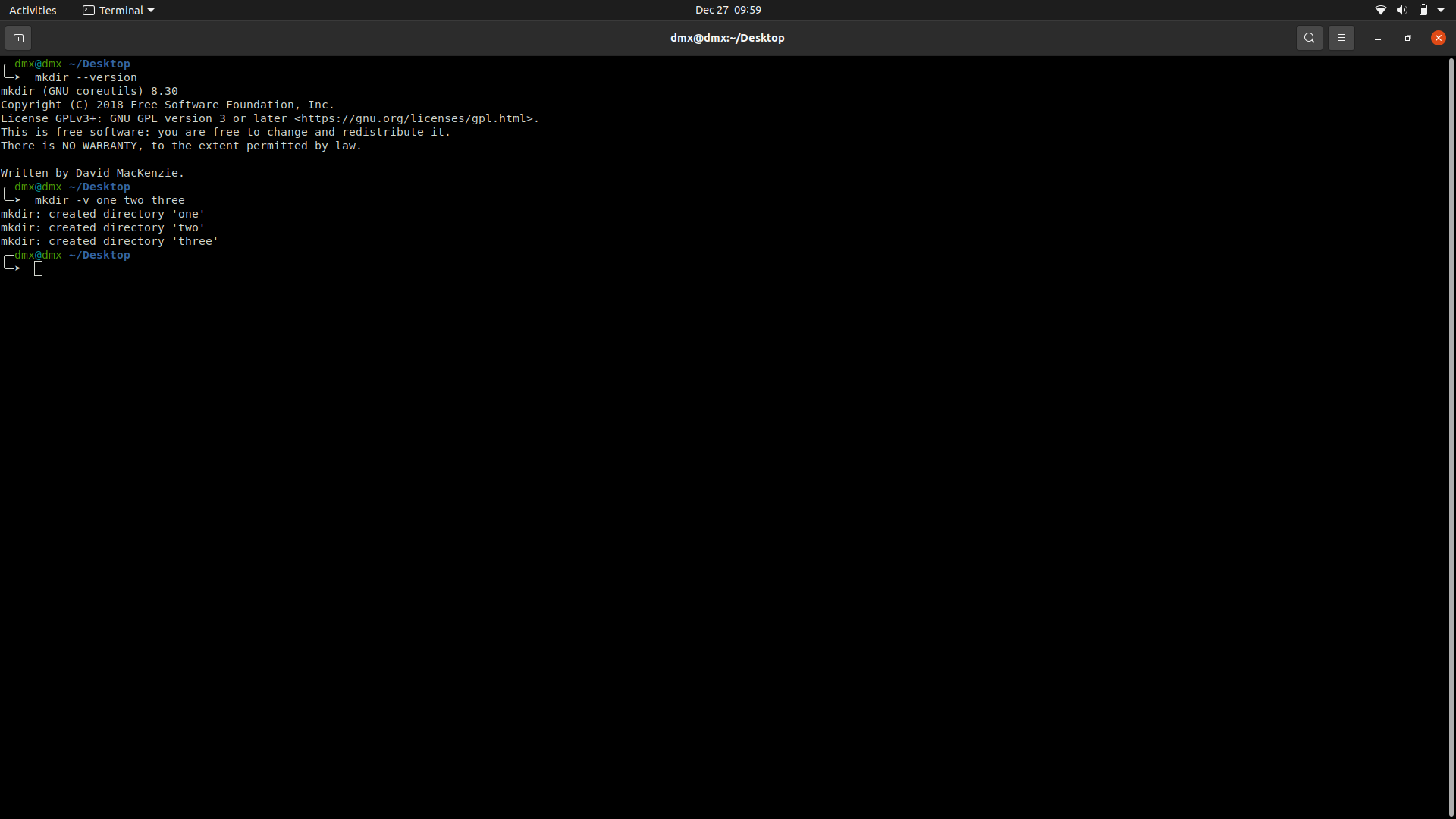
mkdir command in Linux allows the user to create directories.

**5.1) -m, --mode=MODE**: - set file mode (as in chmod), not a=rwx – umask

**5.2) -p, --parents**: -no error if existing, make parent directories as needed

**5.3) -v, --verbose**: -print a message for each created directory

**5.4) -Z: -set SELinux** security context of each created directory

to the default type

1. **file**

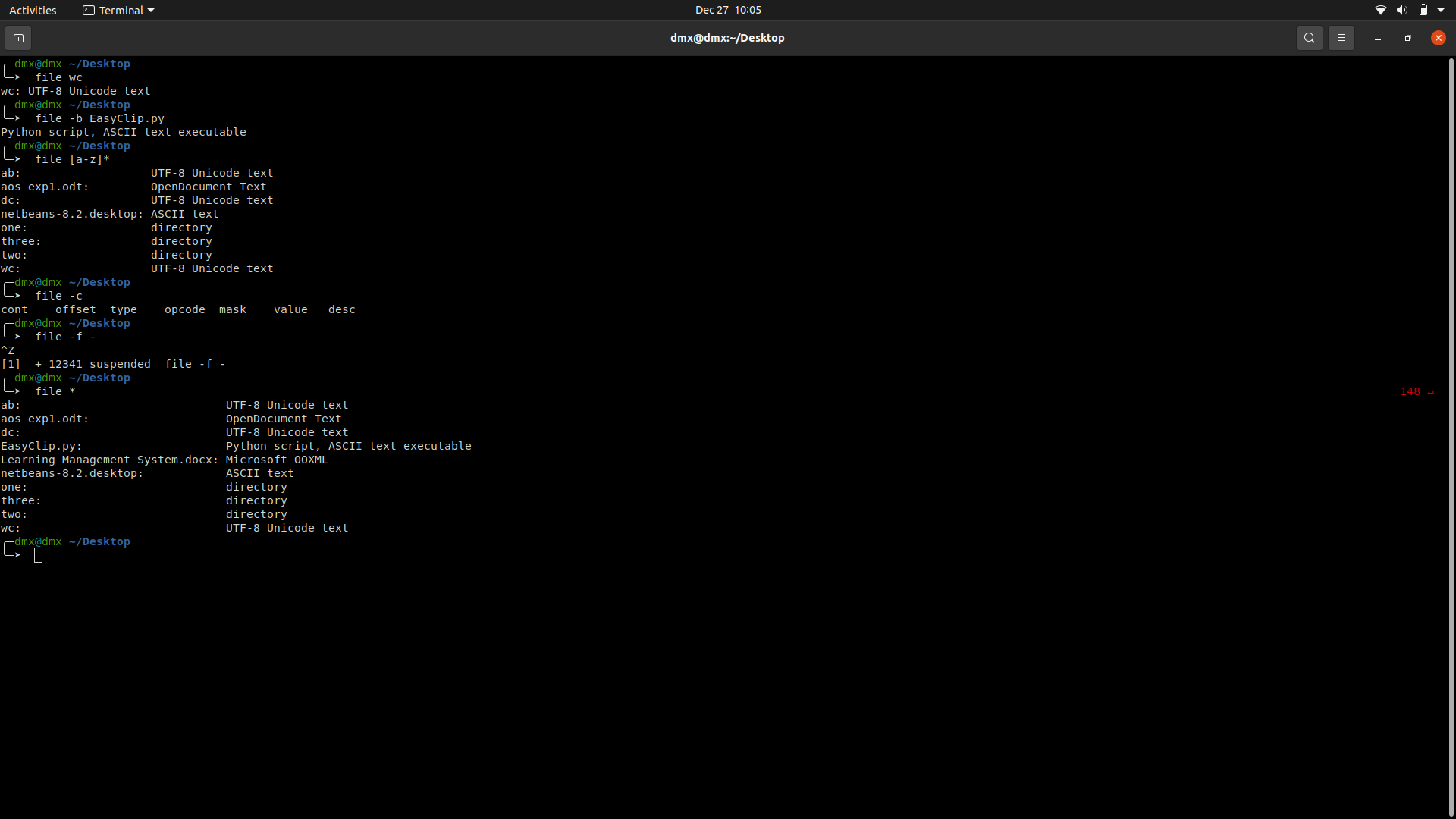
file command is used to determine the type of a file.

**6.1) -b, –brief:** - This is used to display just file type in brief mode

**6.2) [range]\* option**: - To display the file type of files in specific range

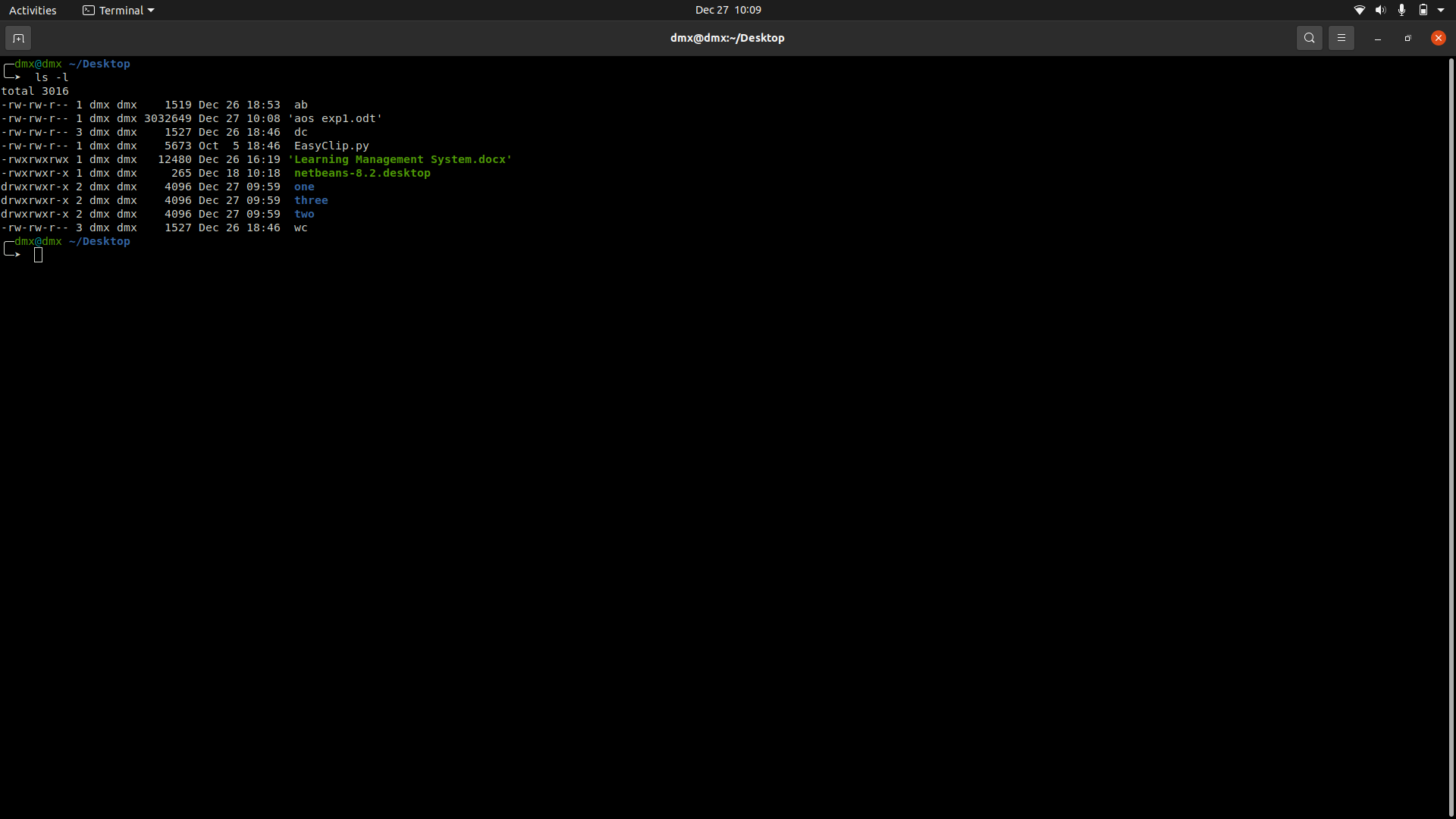
**6.3) -c option**: - Cause a checking printout of the parsed form of the magic file.

**6.4) \* option:** - Command displays the all files’s file type.



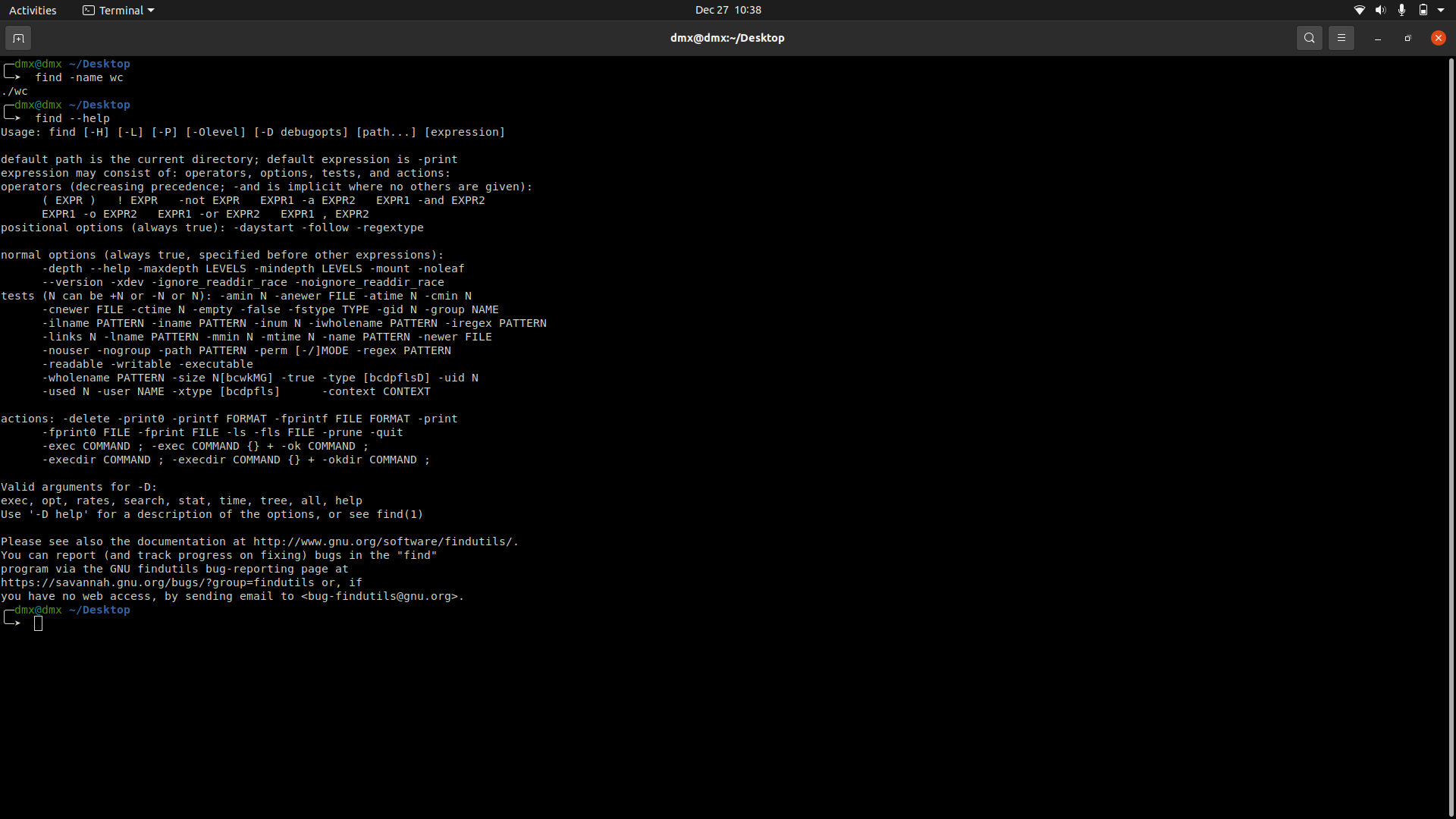
1. **chmod**

The chmod command is used to change the access mode of a file.



1. **find**

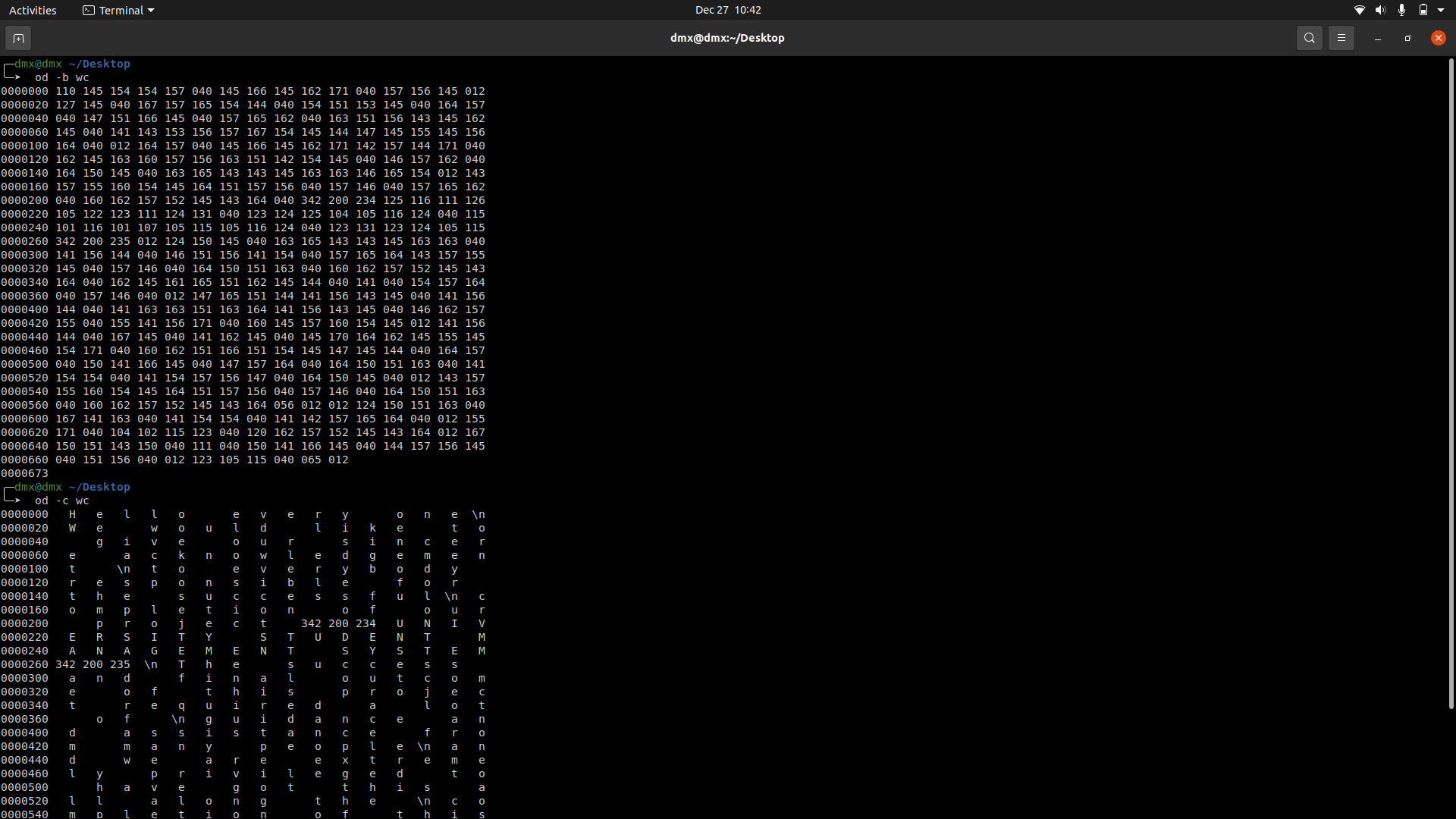
The find command in UNIX is a command line utility for walking a file hierarchy.



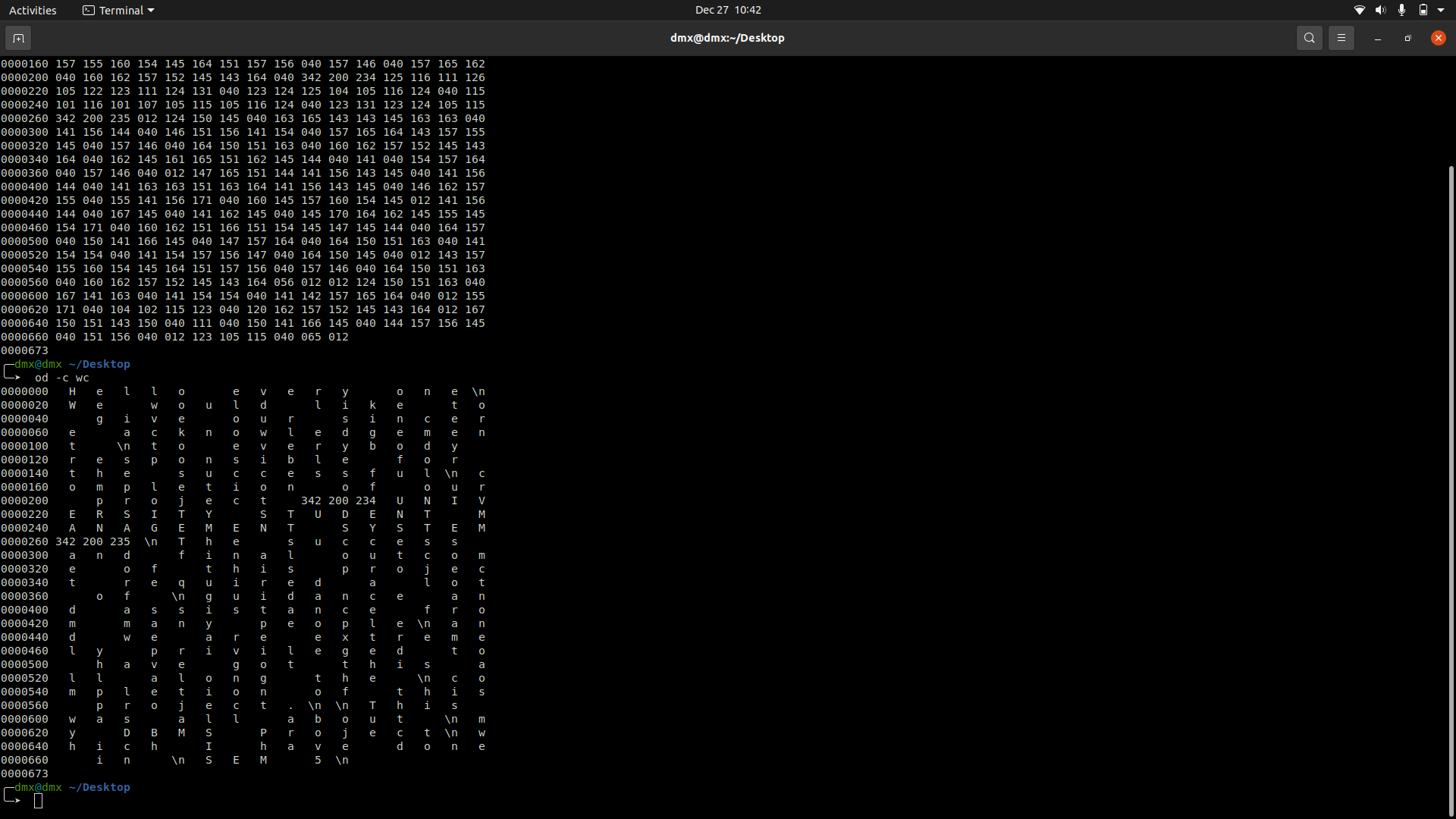
1. **Od**

od command in Linux is used to convert the content of input in different formats with octal format as the default format

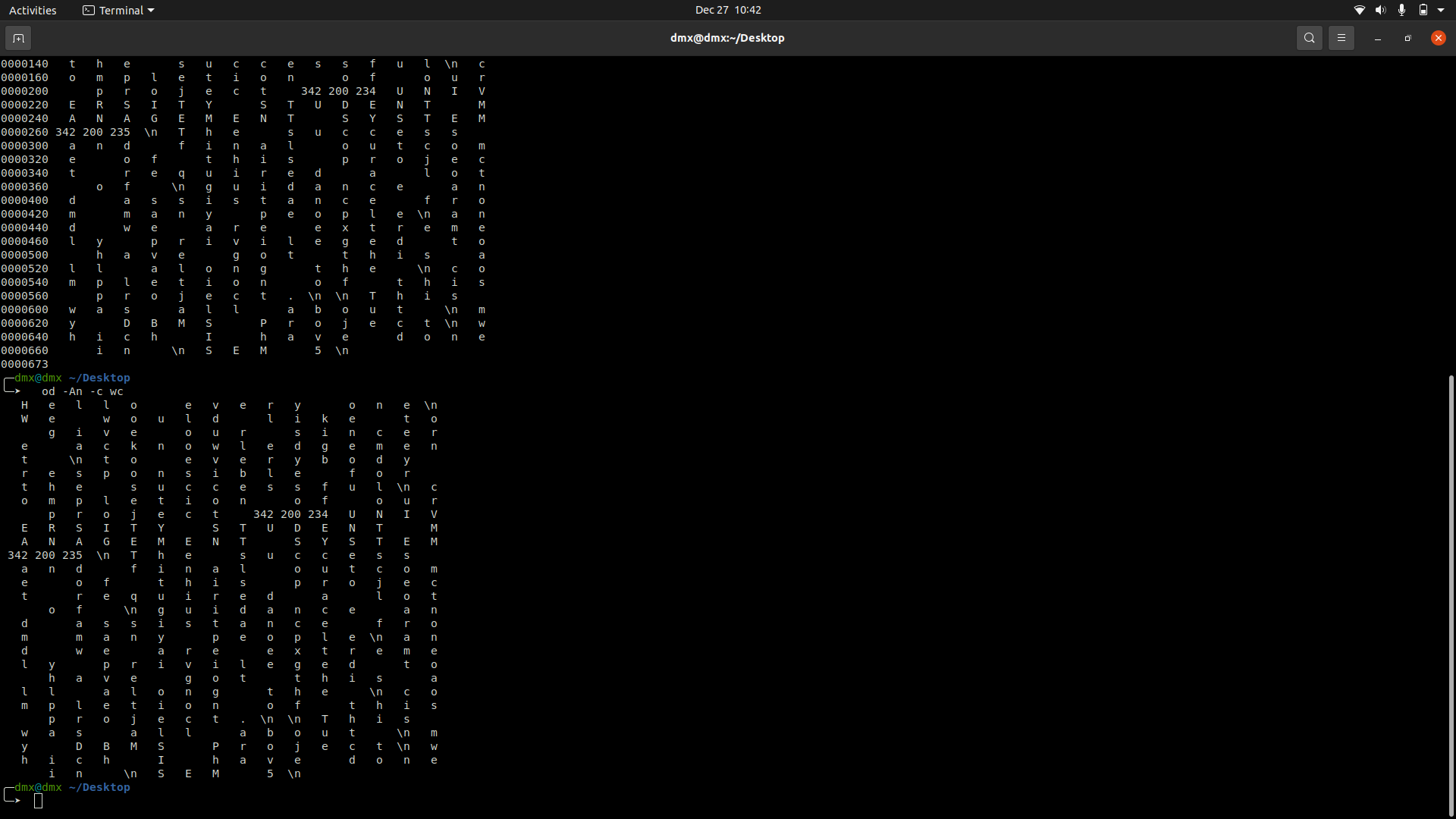
**9.1) -b Option**: - It displays the contents of input in octal format.



**9.2) -c Option**: - It displays the contents of input in character format.

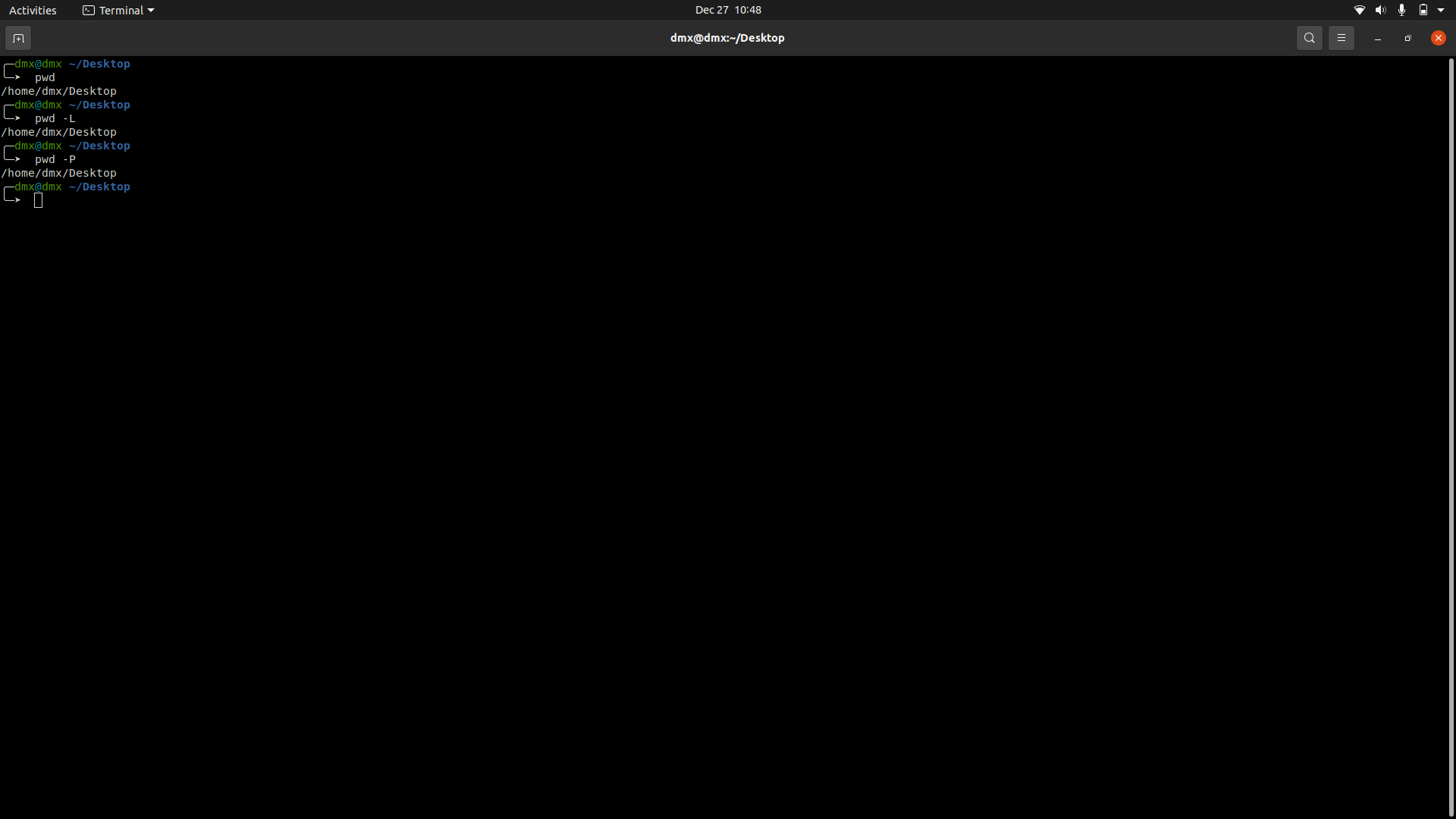


**9.3) -An Option**: - It displays the contents of input in character format but with no offset information.



1. **pwd**

pwd stands for Print Working Directory. It prints the path of the working directory, starting from the root.



**10.1) Pwd -L**: - Prints the symbolic path.

**10.2) Pwd -P**: - Prints the actual path.

1. **Locate**

*locate* command in Linux is used to find the files by name.

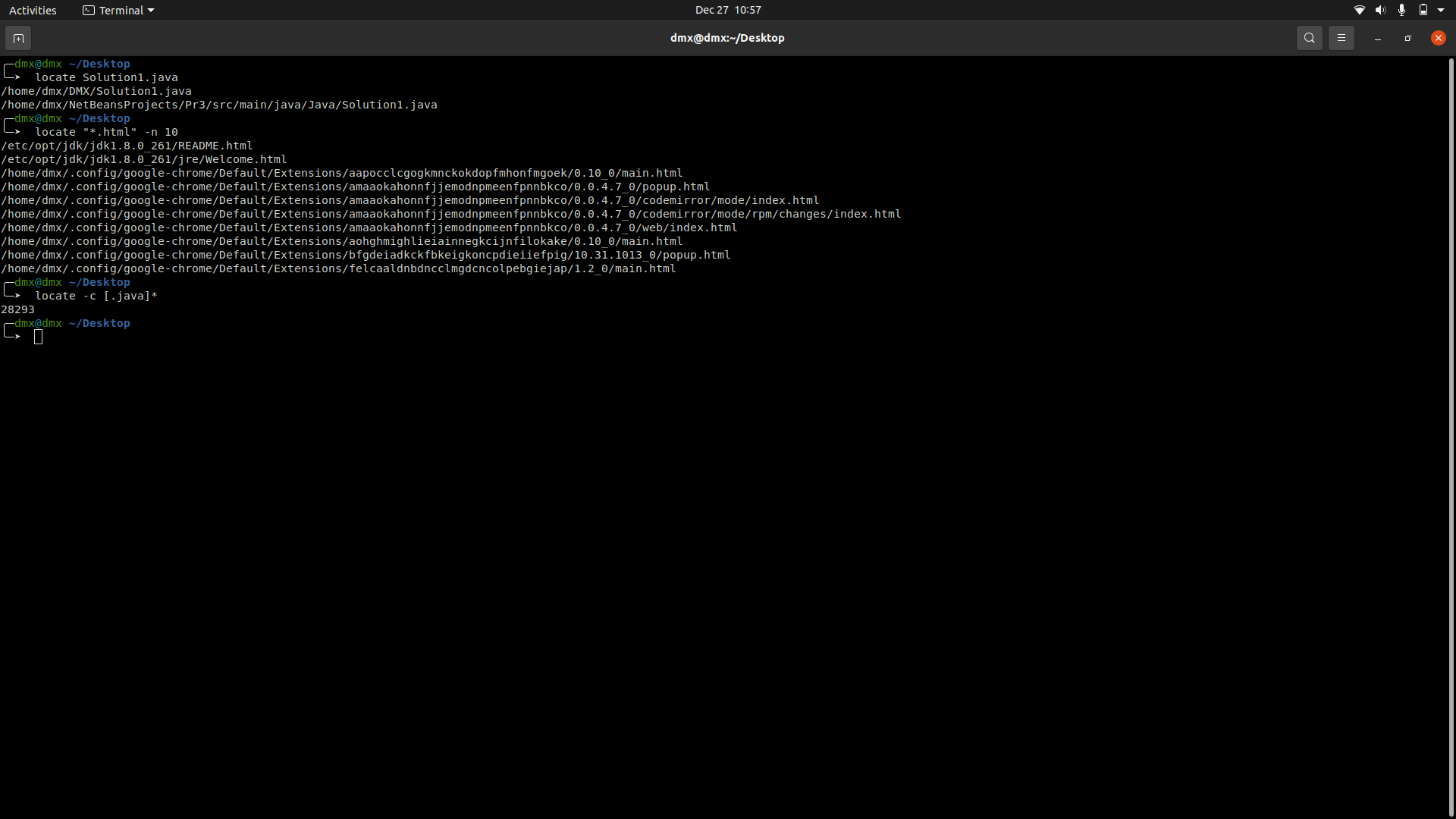
**11.1) locate Solution1.java**

Limit Search Queries to a Specific Number.

**11.2) locate "\*.html" -n 10**

Display The Number of Matching Entries.

**11.3) locate -c [.java]\***



1. **updated**

Tell the update

**sudo apt-get update**

**sudo apt-get upgrade**

1. **mount**

mount command is used to mount the filesystem found on a device to big tree structure

**13.1) -a, --all** = mount all filesystems mentioned in fstab

**13.2) -c, --no-canonicalize** = don't canonicalize paths

**13.3) -f, --fake** = dry run; skip the mount (2) syscall

**13.4) -F, --fork =** fork off for each device (use with -a)

**13.5) -T, --fstab** = <path> alternative file to /etc/fstab

**13.6)** **-i, --internal**-only = don't call the mount. <type> helpers

**13.7) -l, --show-labels =** show also filesystem labels

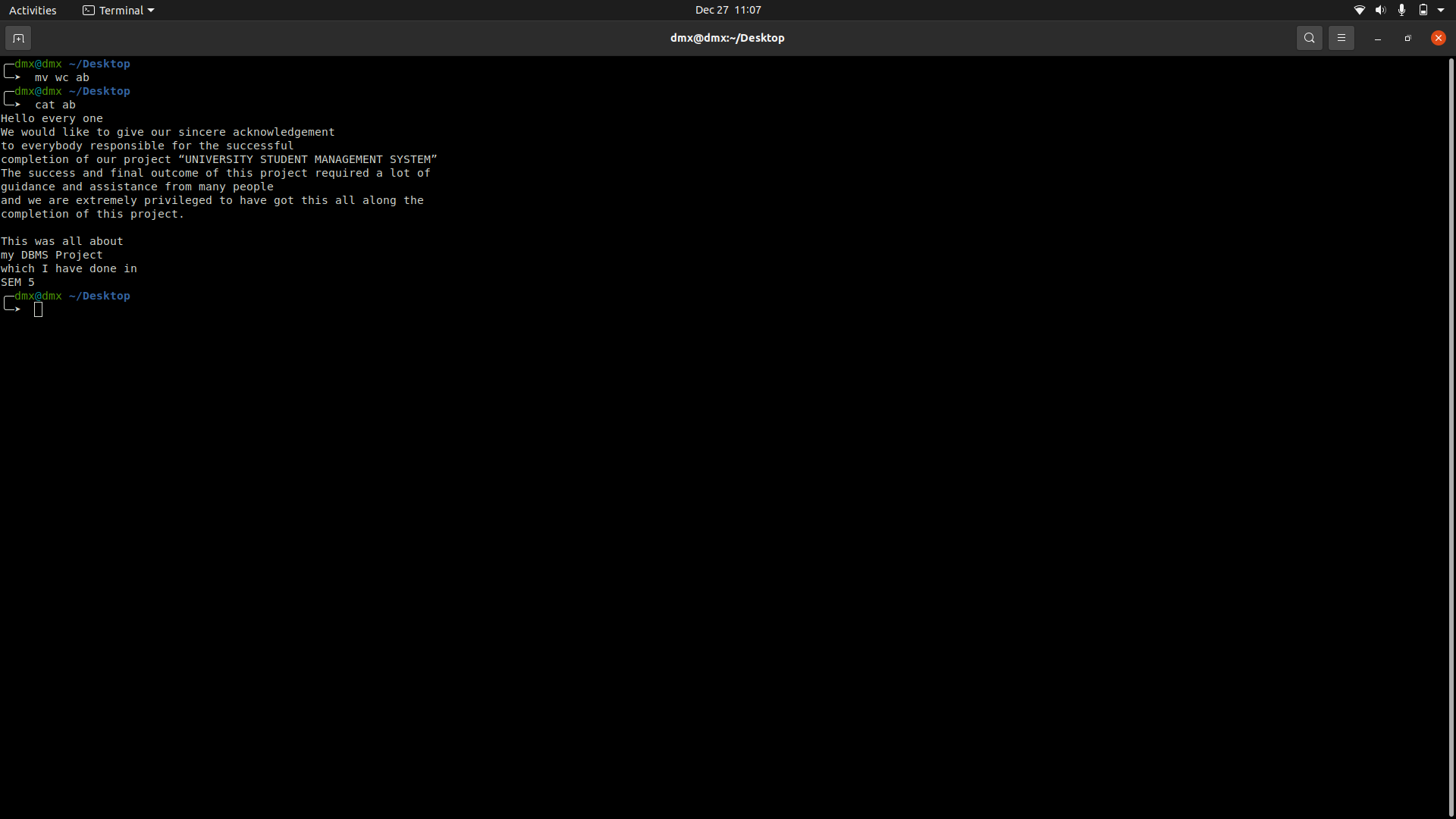
**13.8)** **-n, --no-mtab** = don't write to /etc/mtab

1. **Umount**

To unmount a mounted file system, use the umount command

1. **mv**

mv stands for move. mv is used to move one or more files or directories from one place to another in file system like UNIX.



**15.1) -b: - like** --backup but does not accept an argument

**15.2) -f, --force** = do not prompt before overwriting

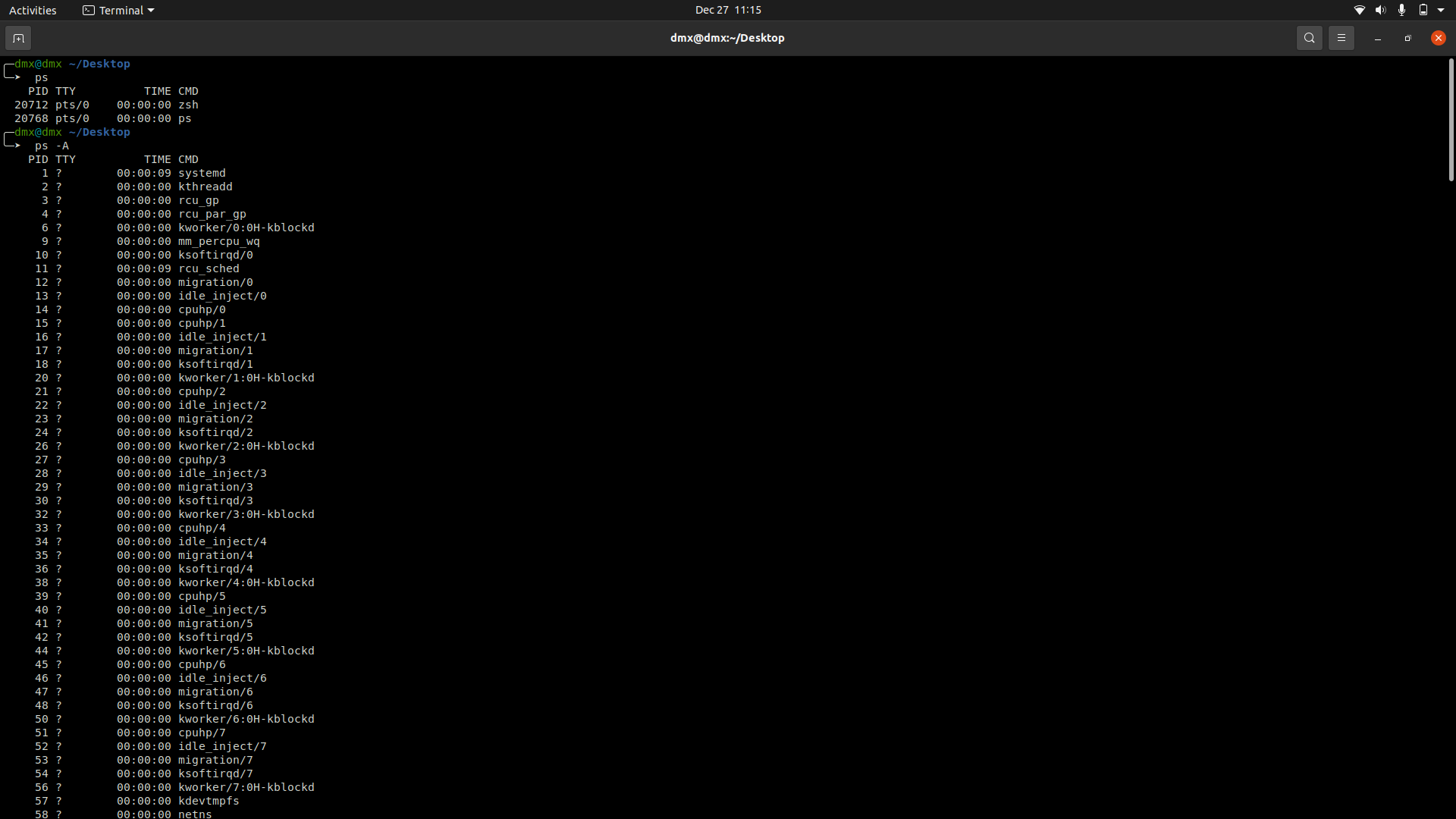
**15.3) -i, --interactive** = prompt before overwrite

**15.4) -n, --no-clobber** = do not overwrite an existing file

**d) Process Management**

**1) ps**

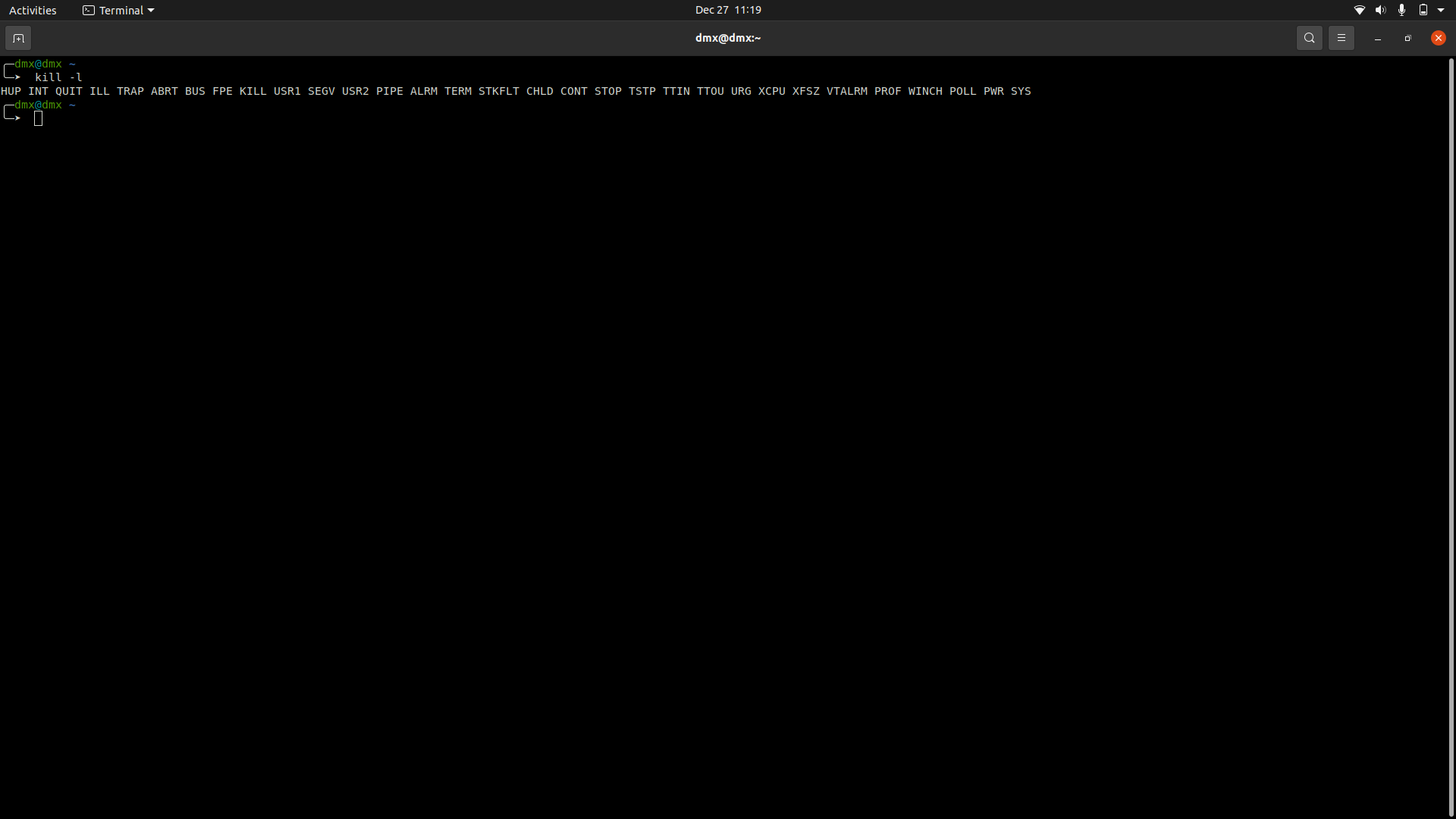
ps for viewing information related with the processes on a system which stands as abbreviation for **“**Process Status”. ps command is used to list the currently running processes and their PIDs along with some other information depends on different options.



**2) kill**

*kill* command in Linux (located in /bin/kill), is a built-in command which is used to terminate processes manually.

**2.1) kill -l**: - To display all the available signals you can use below command option:



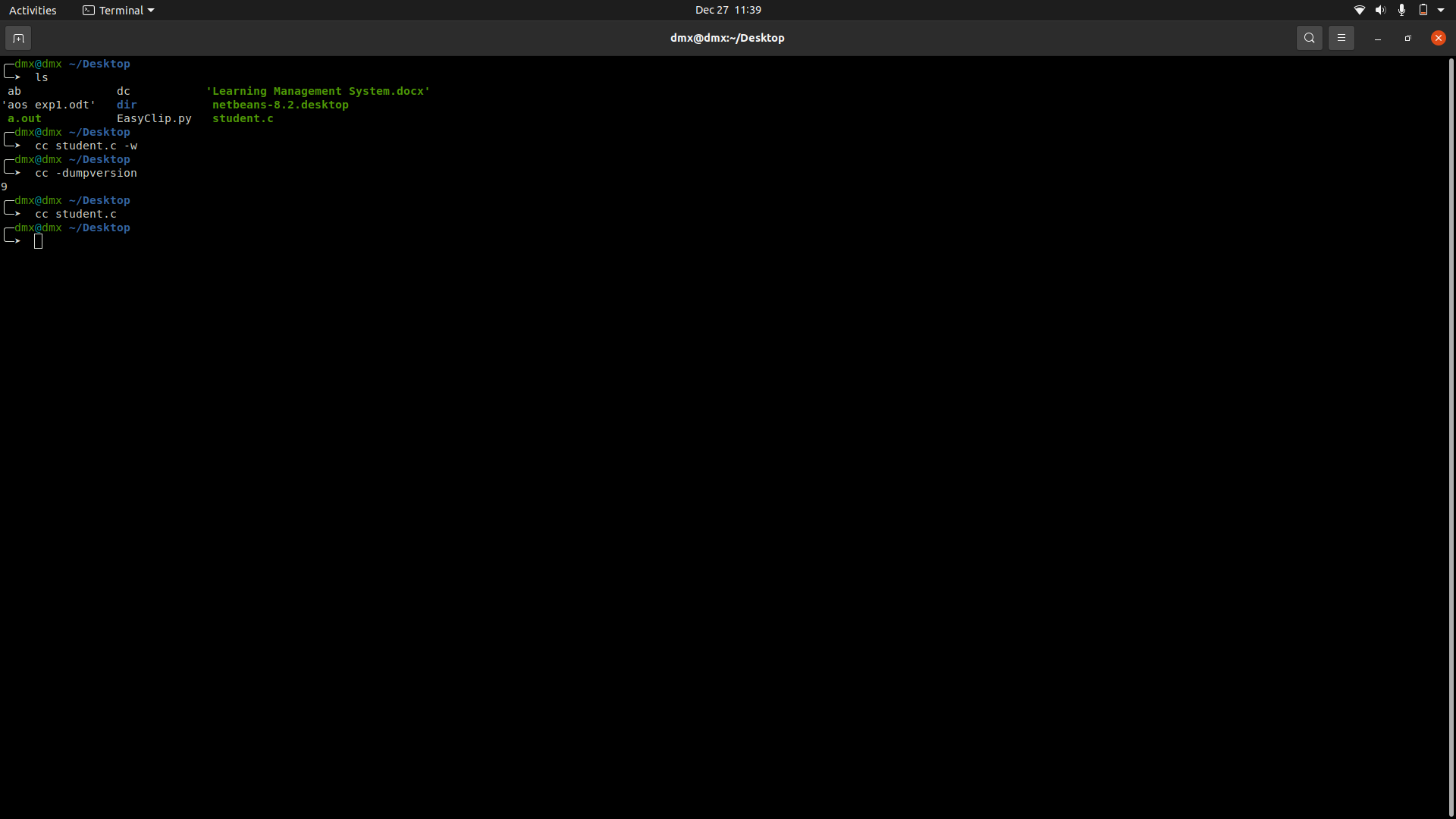
**e) Compilation and debugging**

1. **cc**

cc command is stands for C Compiler, usually an alias command to *gcc* or *clang*.

* 1. Below command will compile the *source\_file.c* file, and create the default executable output file, *a.out*.
  2. cc student.c
  3. cc command with -w option: - This command will compile the *source\_file.c* file, but suppress all the warnings.

**1.4)** cc command with -c option: - This command will compile the *source\_file.c* file, and create an object file *source\_file.o*, which can later be linked to create an executable output file.



1. **gdb**

gdb is the acronym for GNU Debugger. This tool helps to debug the programs written in C, C++, Ada, Fortran, etc.

**f) Editors**

1. **vi**

The VI editor is the most popular and classic text editor in the Linux family. Below, are some reasons which make it a widely used editor –

1. It is available in almost all Linux Distributions

2) It works the same across different platforms and Distributions

1. **Joe**

JOE is a full featured terminal-based screen editor which is distributed under the GNU General Public License (GPL). JOE has been around since 1988 and comes standard with many Linux distributions.

**3)Mcedit**

mcedit is a link to mc, the main GNU Midnight Commander executable. Executing GNU Midnight Commander under this name requests staring the internal editor and opening the *file* specified on the command line. The editor is based on the terminal version of cooledit - standalone editor for X Window System.

**4)emac**

Emacs is one of the oldest and most versatile text editors available for Linux and UNIX-based systems. It’s been around for a long time (more than twenty years for GNU emacs) and is well known for its powerful and rich editing features. Emacs is also more than just a text editor; it can be customized and extended with different “modes”, enabling it to be used like an Integrated Development Environment (IDE) for programming languages like Java, C or Python.

