Lecture _6

LINUX essentials

Dr .Sara Mohamed

Copying , moving and removing Files

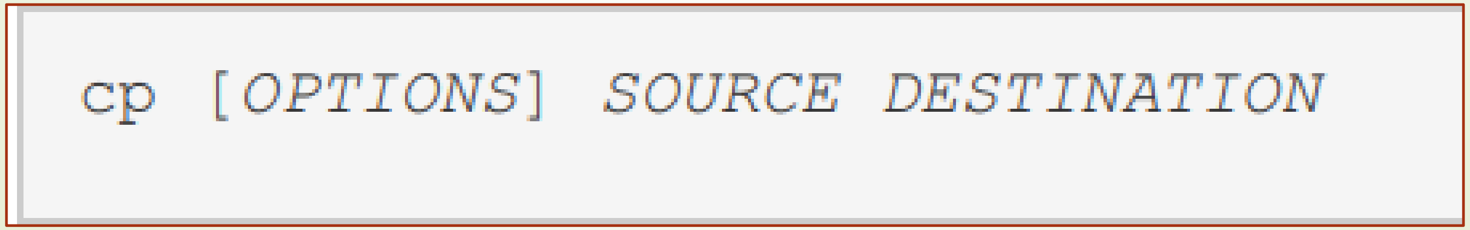
Copying Files

- Creating copies of files can be useful for numerous reasons:

  - If a copy of a file is created before changes are made, then it is possible to revert back to the original.
  - A copy of a file can be used to transfer a file to removable media devices.
  - A copy of an existing document can be used as a template for a new document.

How to Copy Files and Directories in Linux | cp Command

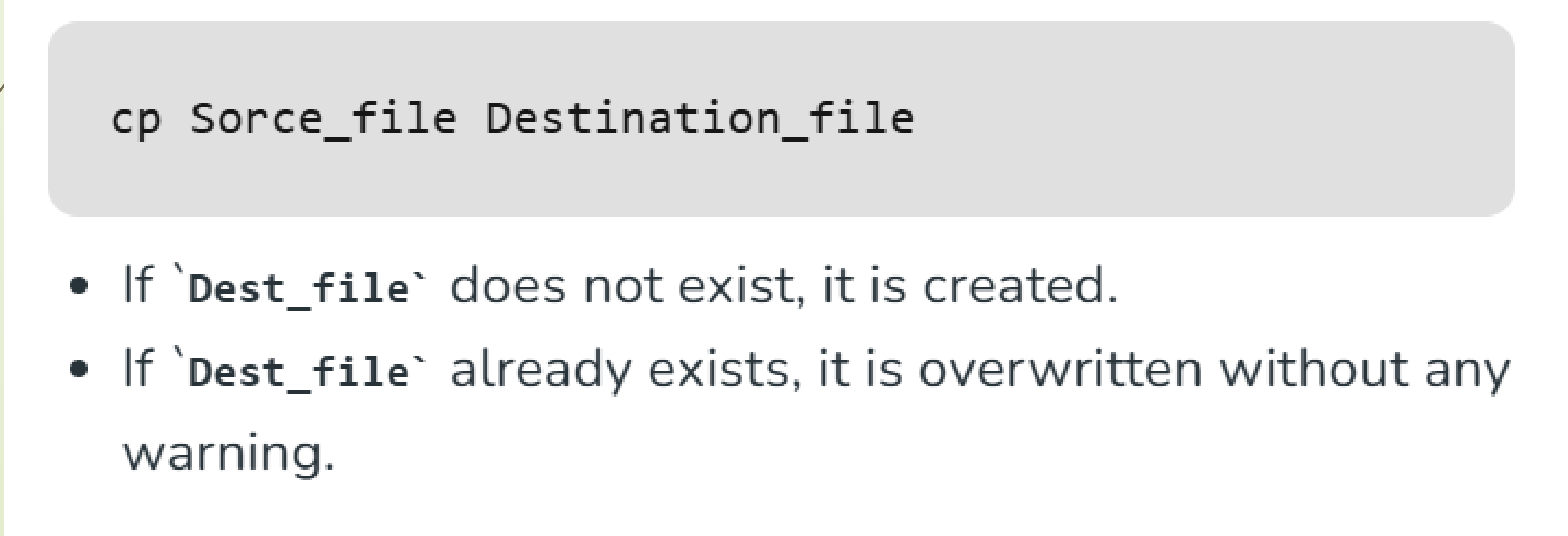
- The cp (copy) command in Linux is used to duplicate files or directories from one location to another within the file system.
  - It duplicates files or directories while preserving their contents.
  - If the destination file already exists, it is overwritten without warning.
  - The -r or -R option allows copying entire directories recursively.
  - File permissions and structure are retained during the copy process.



```
cp [OPTIONS] SOURCE DESTINATION
```

Copying Between Two Files in Linux

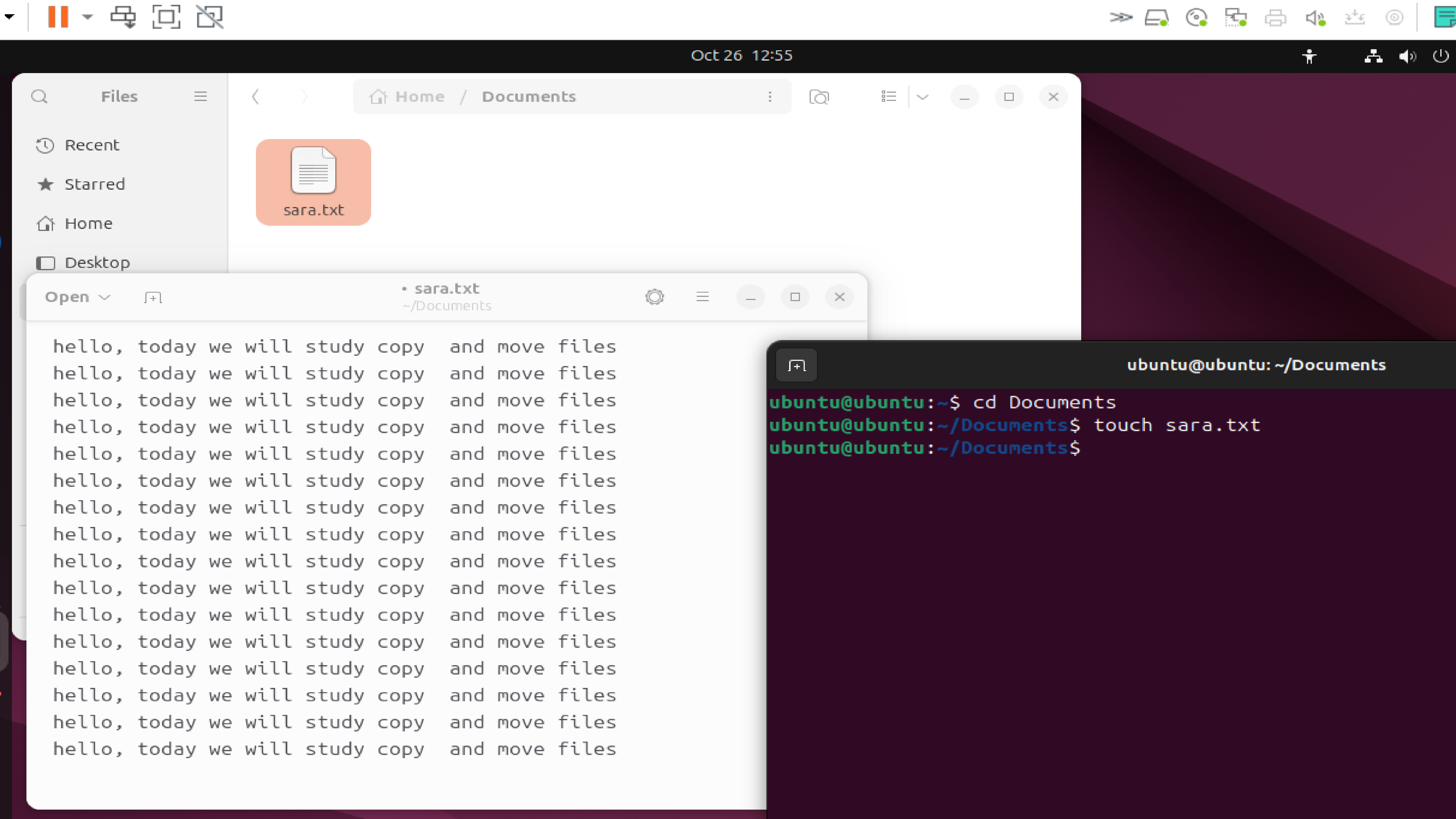
- If the `cp` command contains two file names, it copies the contents of the first file to the second file. If the second file doesn't exist, it is created, and the content is copied into it. However, if the second file already exists, it is overwritten without warning.



```
cp Sorce_file Destination_file
```

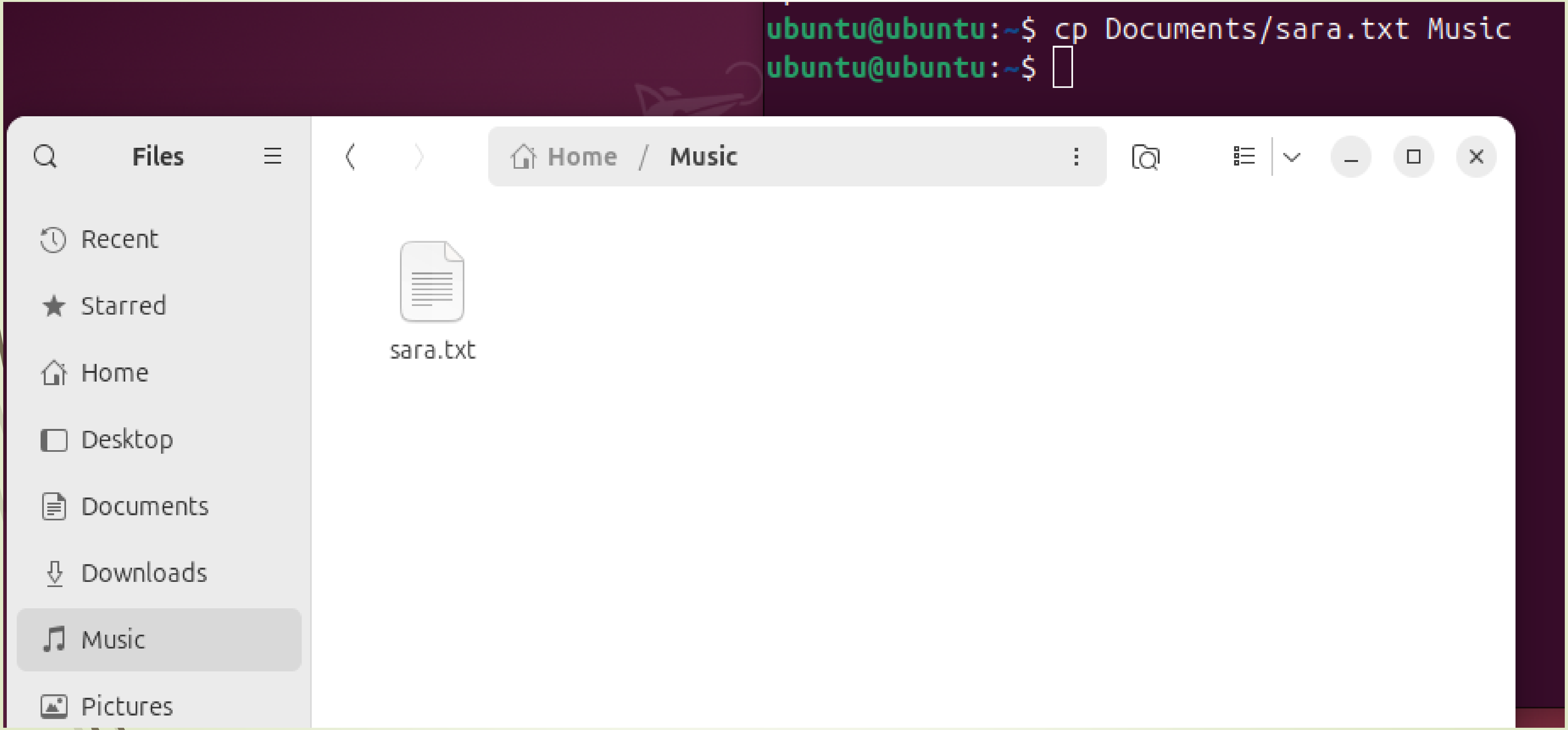
- If `Dest_file` does not exist, it is created.
- If `Dest_file` already exists, it is overwritten without any warning.

Example_1: copy the sara.txt file to Music Directory



```
hello, today we will study copy  and move files
hello, today we will study copy  and move files
hello, today we will study copy  and move files
hello, today we will study copy  and move files
hello, today we will study copy  and move files
hello, today we will study copy  and move files
hello, today we will study copy  and move files
hello, today we will study copy  and move files
hello, today we will study copy  and move files
hello, today we will study copy  and move files
hello, today we will study copy  and move files
hello, today we will study copy  and move files
hello, today we will study copy  and move files
hello, today we will study copy  and move files
hello, today we will study copy  and move files
hello, today we will study copy  and move files
```

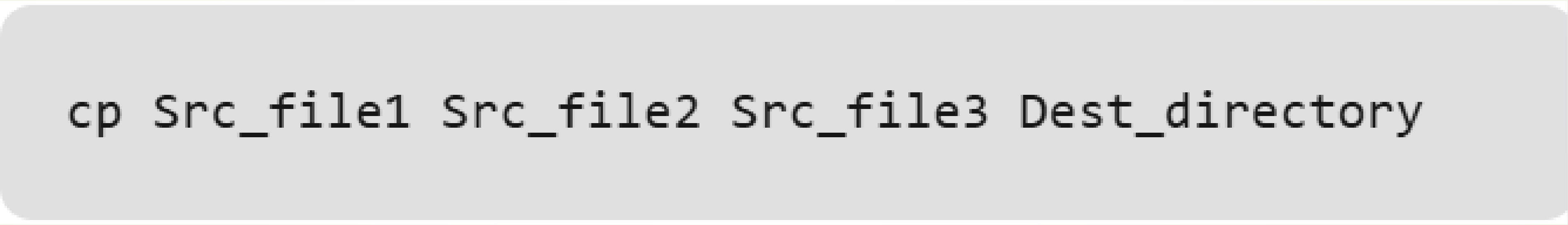
```
ubuntu@ubuntu:~$ cd Documents
ubuntu@ubuntu:~/Documents$ touch sara.txt
ubuntu@ubuntu:~/Documents$
```



```
ubuntu@ubuntu:~$ cp Documents/sara.txt Music
ubuntu@ubuntu:~$ 
```

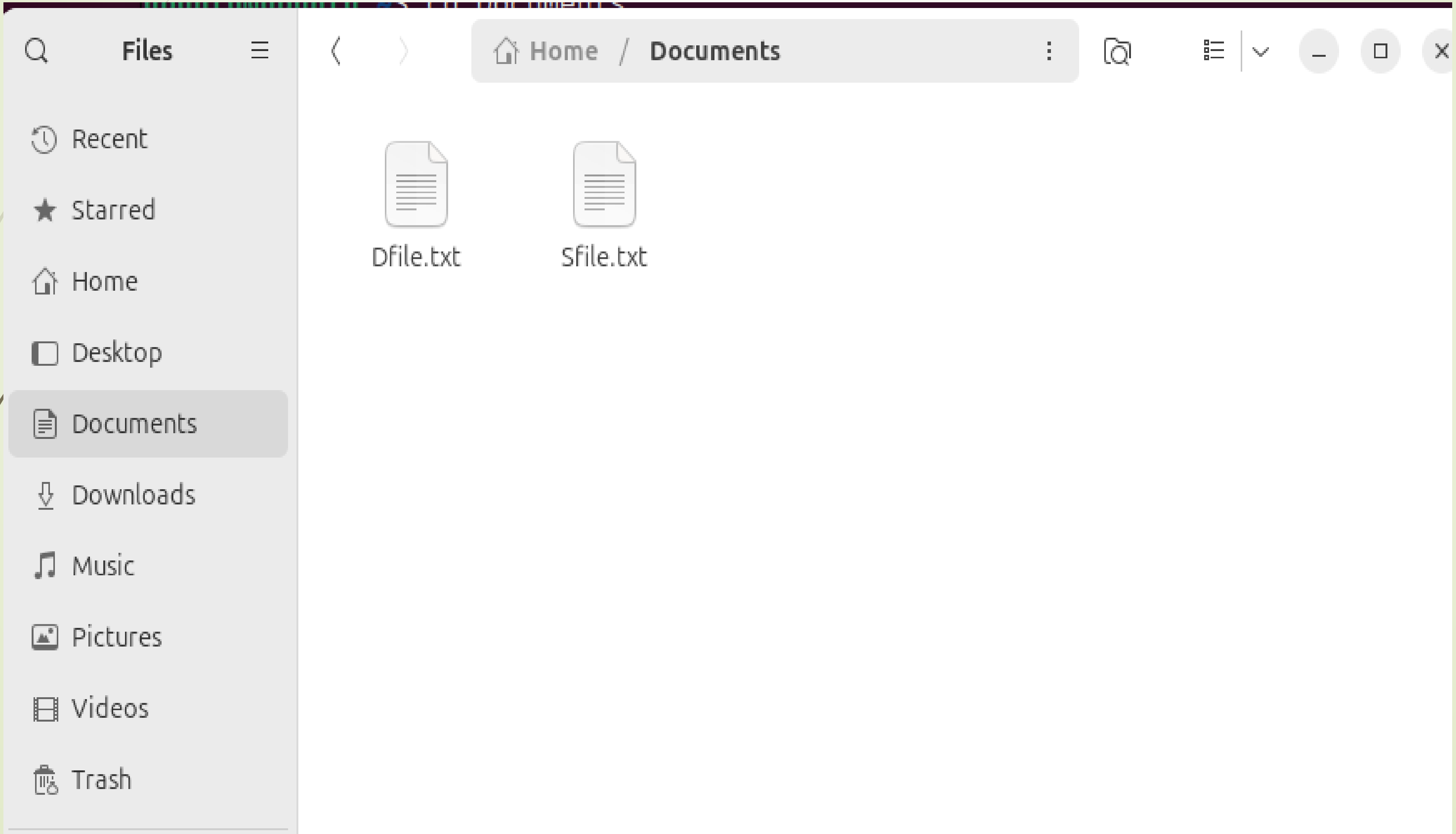
Copy files to a Directory in Linux

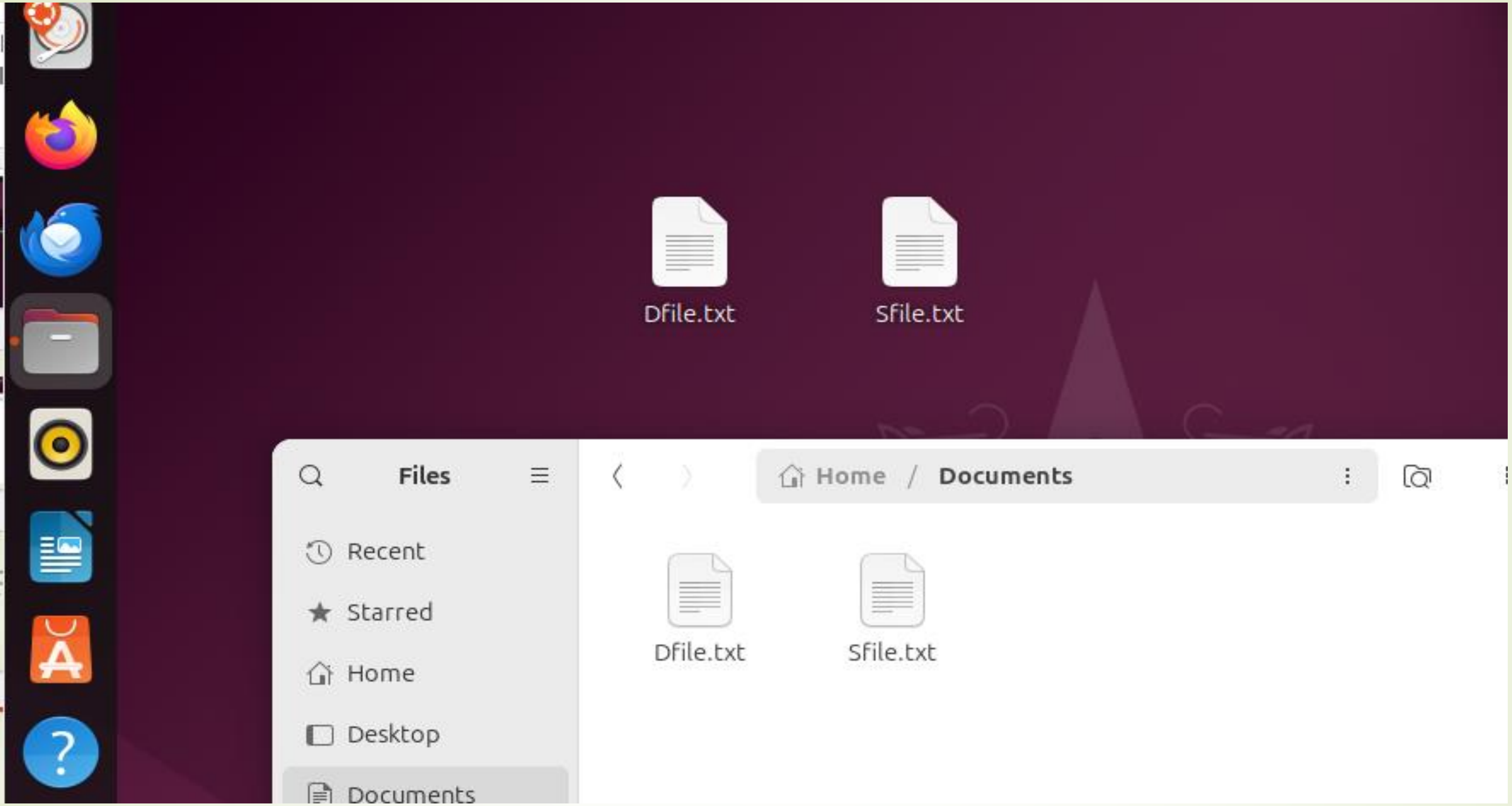
- When the cp command has one or more source file arguments and is followed by a destination directory argument, it copies each source file to the destination directory with the same name. If the destination directory does not exist, it is created. If it already exists, the files are overwritten without warning.

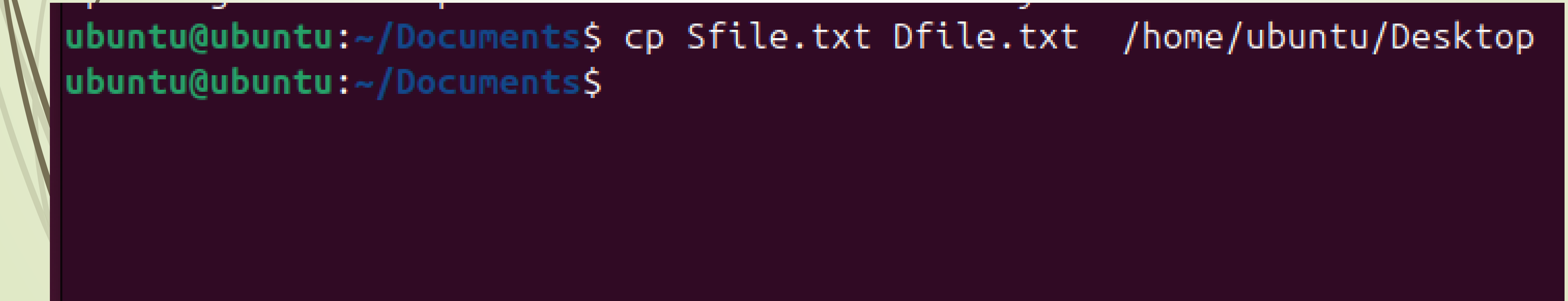


```
cp Src_file1 Src_file2 Src_file3 Dest_directory
```

Example 1: Copy two files : Sfile.txt and Dfile.txt from Documents directory to Desktop

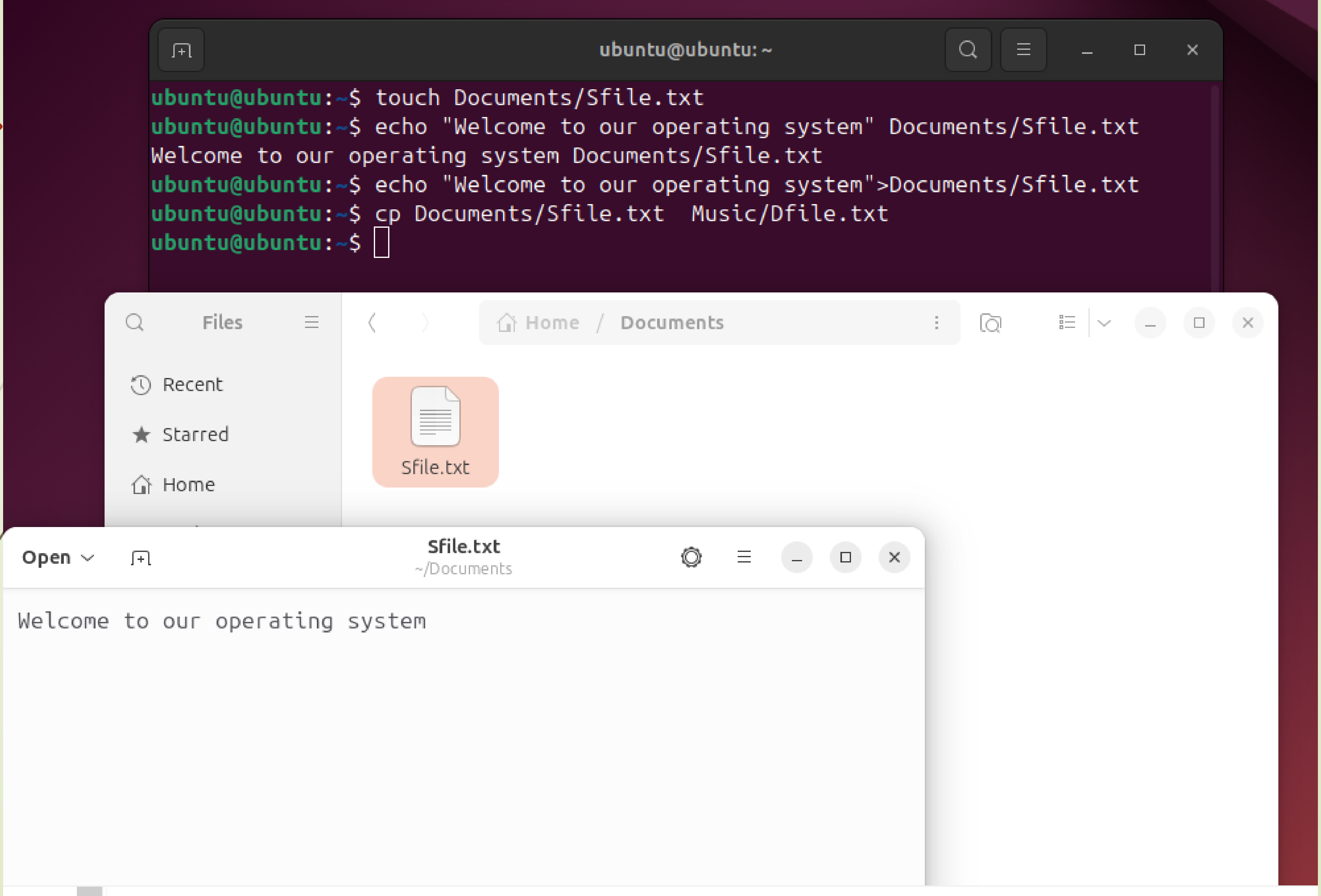






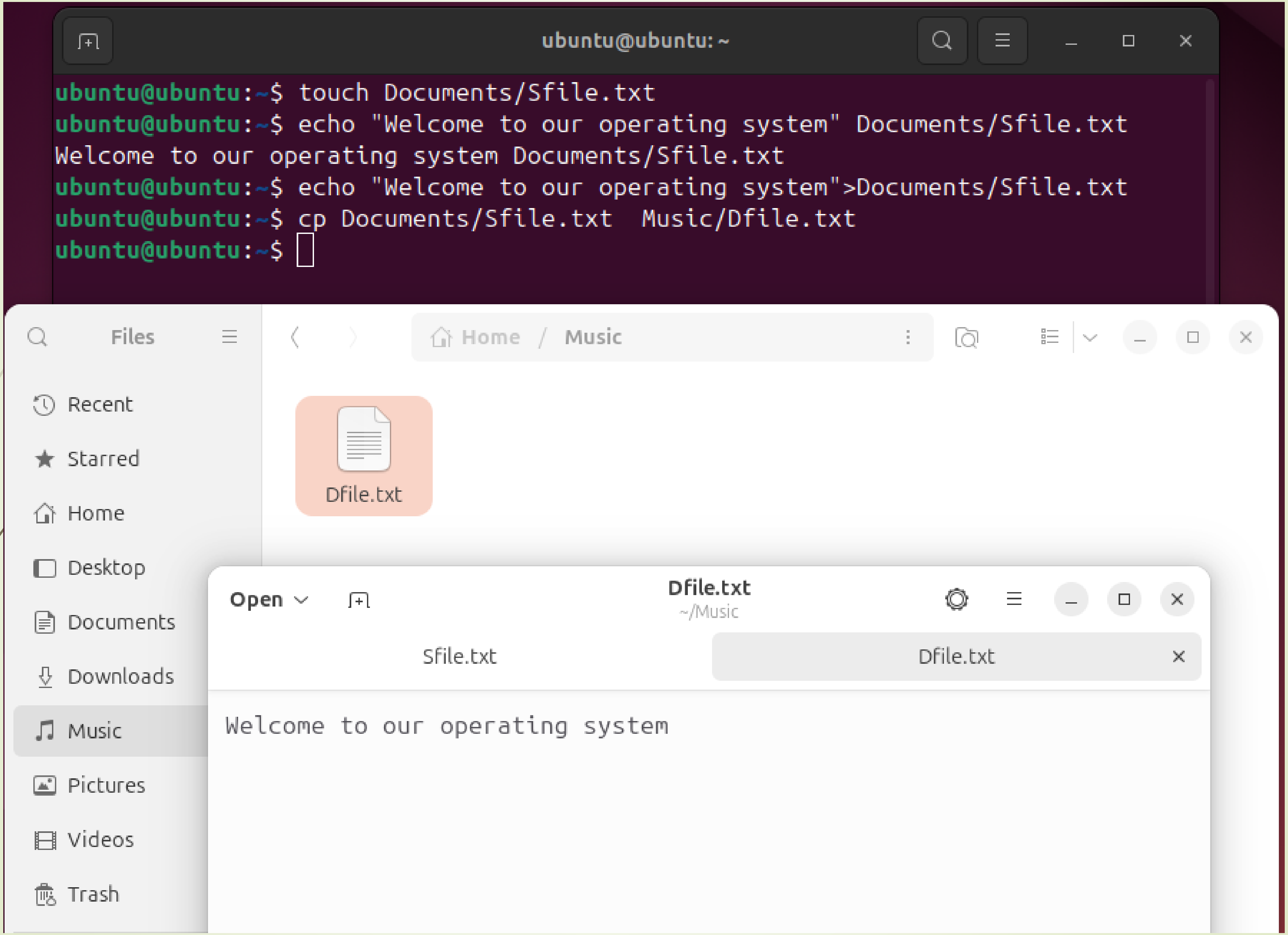
```
ubuntu@ubuntu:~/Documents$ cp Sfile.txt Dfile.txt  /home/ubuntu/Desktop
ubuntu@ubuntu:~/Documents$
```

Example_2: Copy the content of Sfile.txt (in Documents Directory) to a new file called Dfile.txt in Music Directory.



```
ubuntu@ubuntu:~$ touch Documents/Sfile.txt
ubuntu@ubuntu:~$ echo "Welcome to our operating system" Documents/Sfile.txt
Welcome to our operating system Documents/Sfile.txt
ubuntu@ubuntu:~$ echo "Welcome to our operating system">Documents/Sfile.txt
ubuntu@ubuntu:~$ cp Documents/Sfile.txt  Music/Dfile.txt
ubuntu@ubuntu:~$ 
```

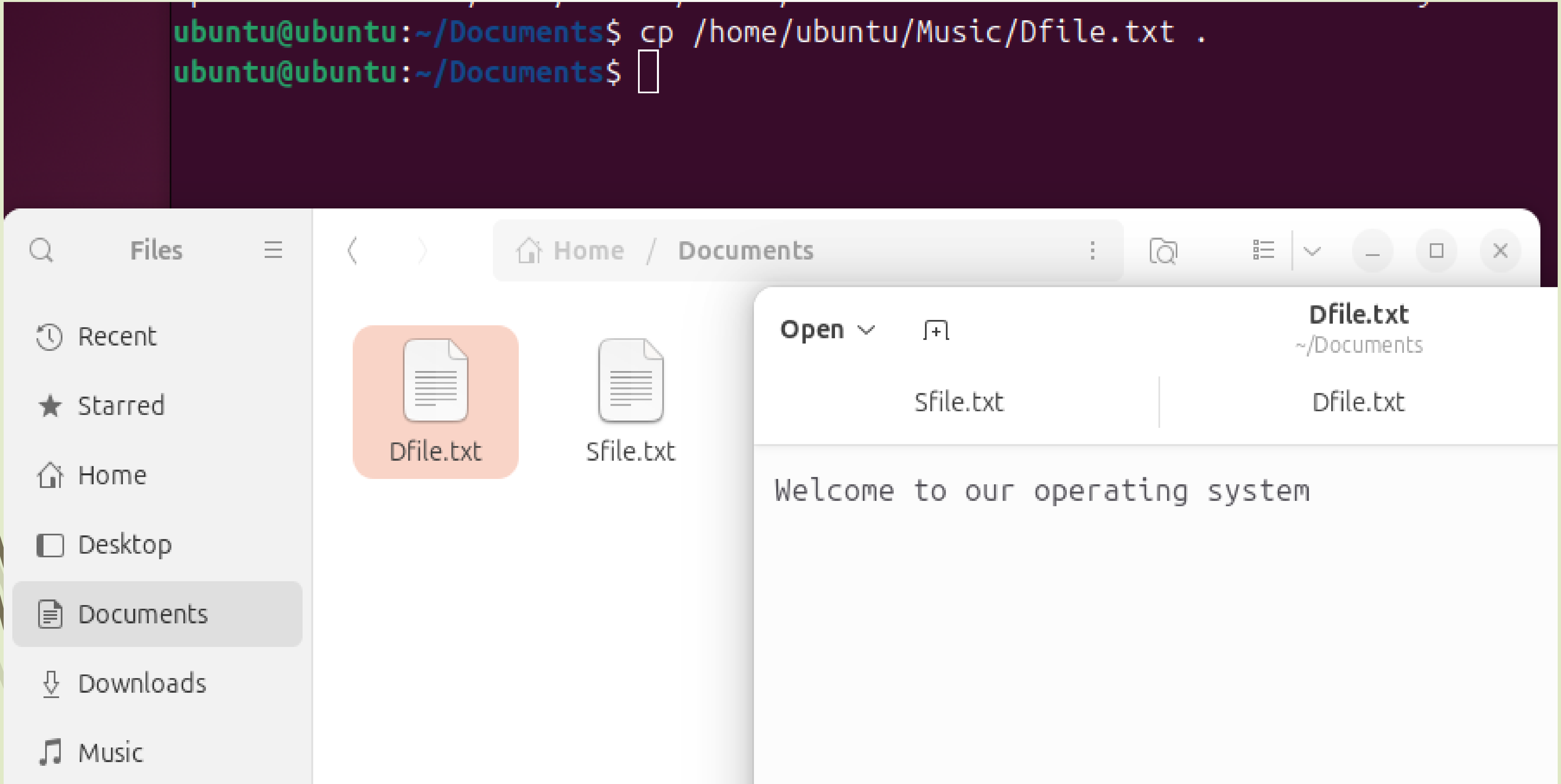
```
Welcome to our operating system
```



```
ubuntu@ubuntu:~$ touch Documents/Sfile.txt
ubuntu@ubuntu:~$ echo "Welcome to our operating system" Documents/Sfile.txt
Welcome to our operating system Documents/Sfile.txt
ubuntu@ubuntu:~$ echo "Welcome to our operating system">Documents/Sfile.txt
ubuntu@ubuntu:~$ cp Documents/Sfile.txt  Music/Dfile.txt
ubuntu@ubuntu:~$
```

```
Welcome to our operating system
```

Example_3: Copy a file from a source location to the current directory

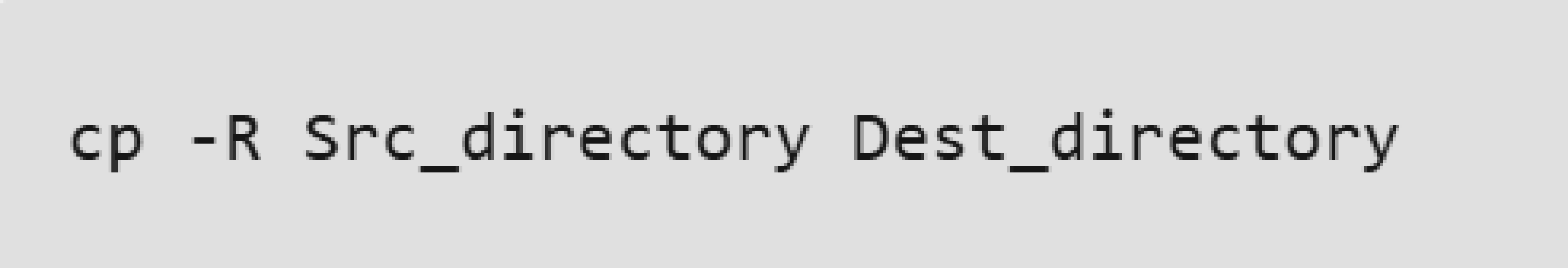


```
ubuntu@ubuntu:~/Documents$ cp /home/ubuntu/Music/Dfile.txt .
ubuntu@ubuntu:~/Documents$ 
```

```
Welcome to our operating system
```

How to Copy Directories in Linux

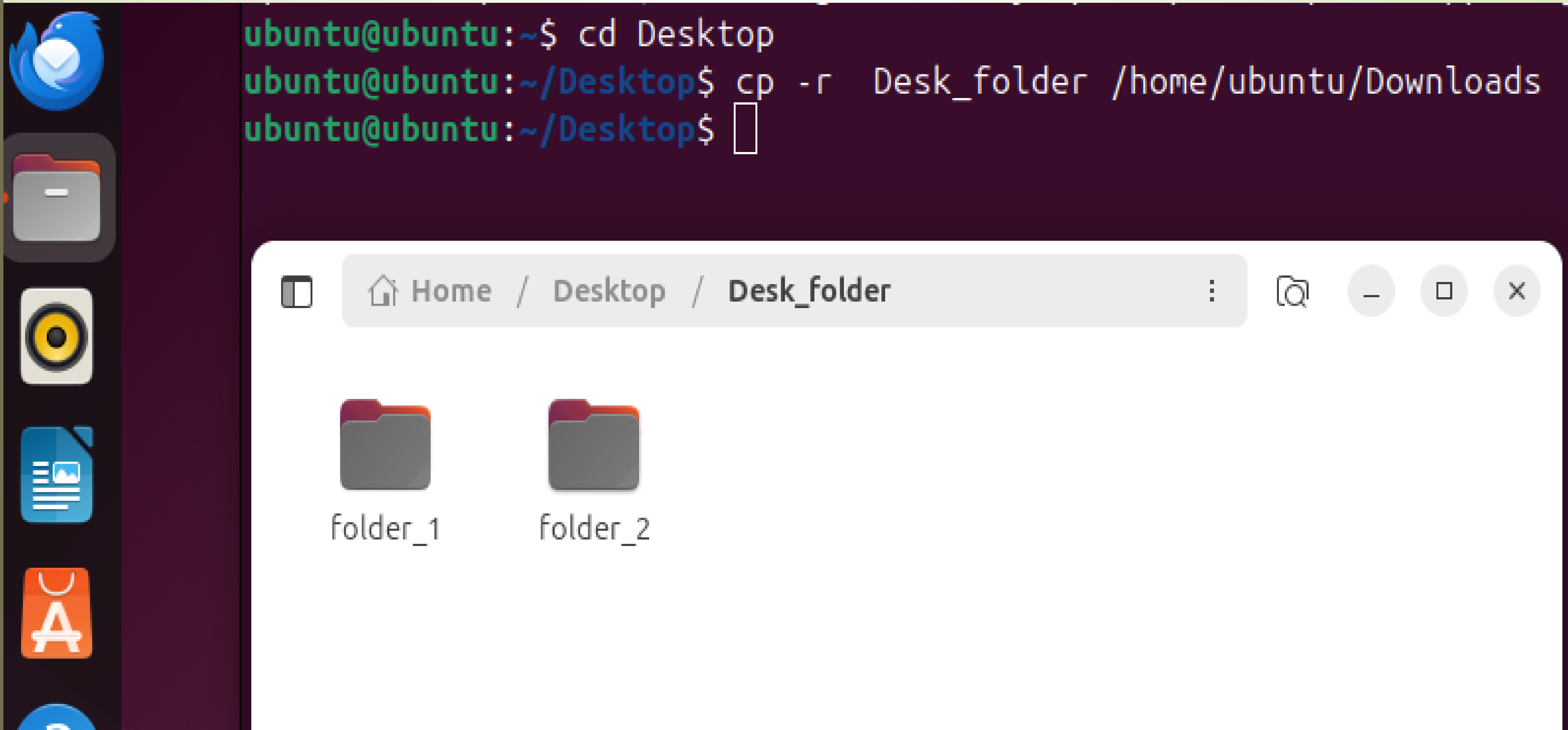
- In this mode, if the cp command contains two directory names, it copies all files from the source directory to the destination directory. The `-R` option is typically used to indicate recursive copying for directories.



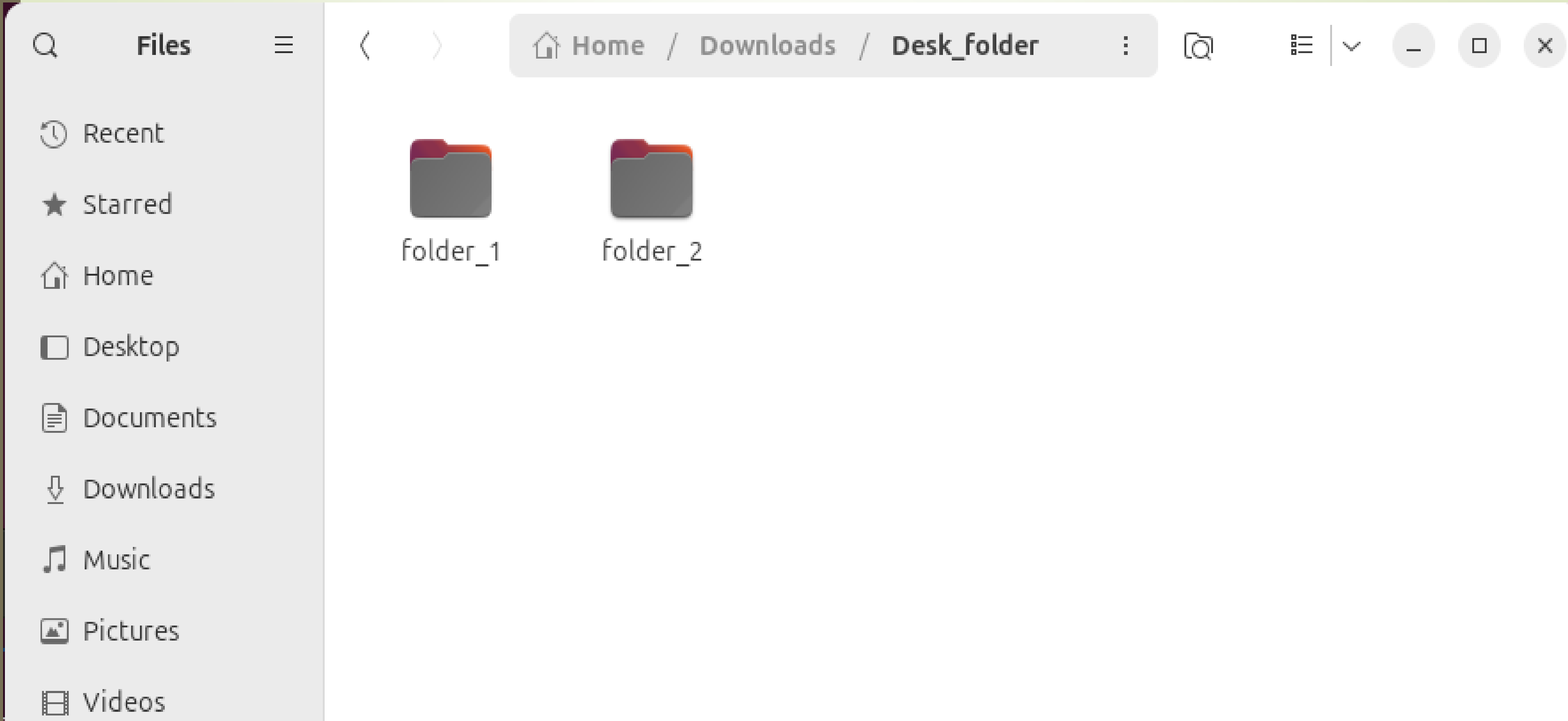
```
cp -R Src_directory Dest_directory
```

- The behaviour depends on whether `Dest_directory` exists or not. If it doesn't exist, `cp` creates it and copies the content of `Src_directory` recursively. If `Dest_directory` exists, the copy of `Src_directory` becomes a sub-directory under `Dest_directory`

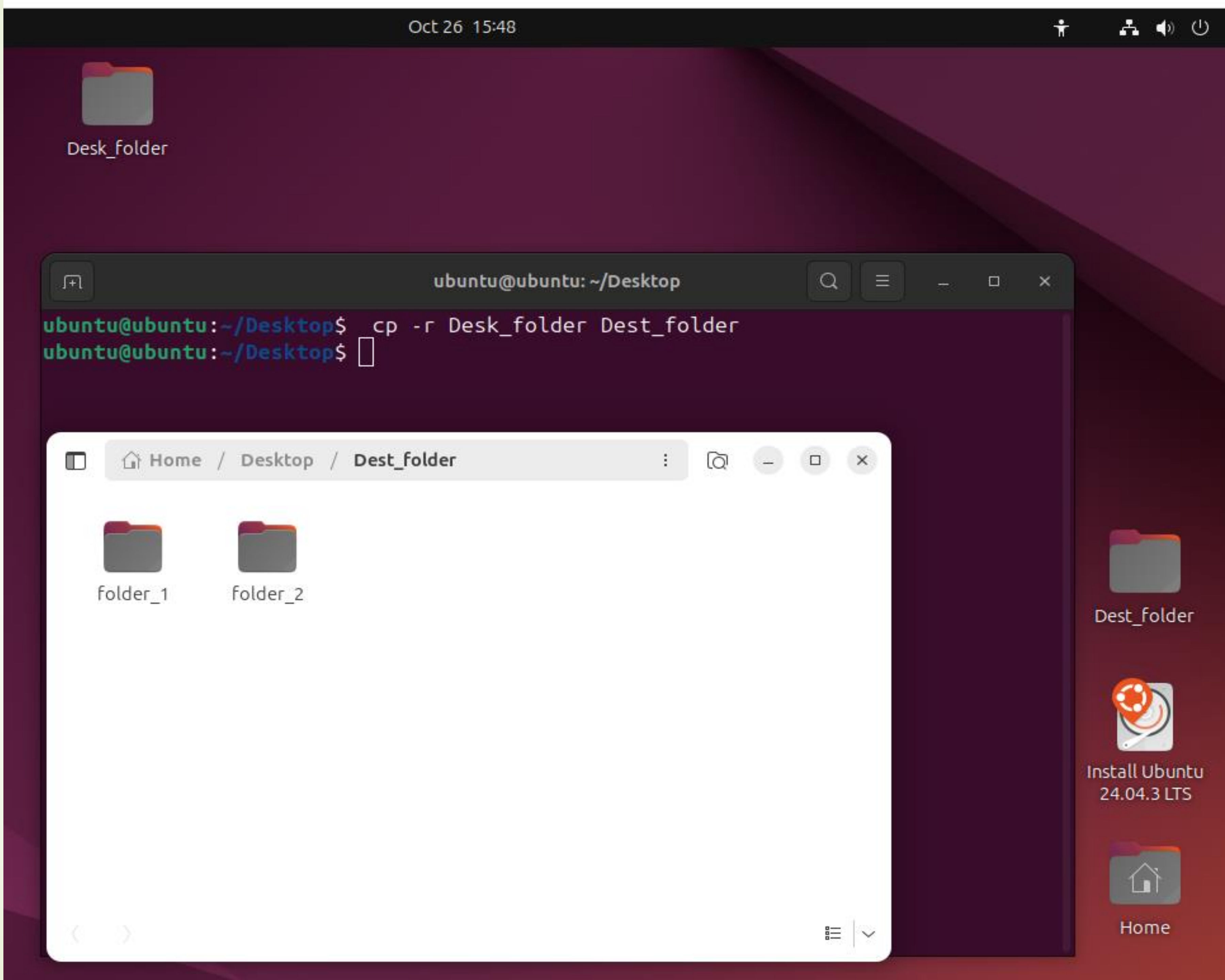
Example: Copy the Desk_folder directory to Downloads directory



```
ubuntu@ubuntu:~$ cd Desktop
ubuntu@ubuntu:~/Desktop$ cp -r  Desk_folder /home/ubuntu/Downloads
ubuntu@ubuntu:~/Desktop$ 
```


Files
Home / Downloads / Desk_folder
Recent
Starred
Home
Desktop
Documents
Downloads
Music
Pictures
Videos
folder_1
folder_2

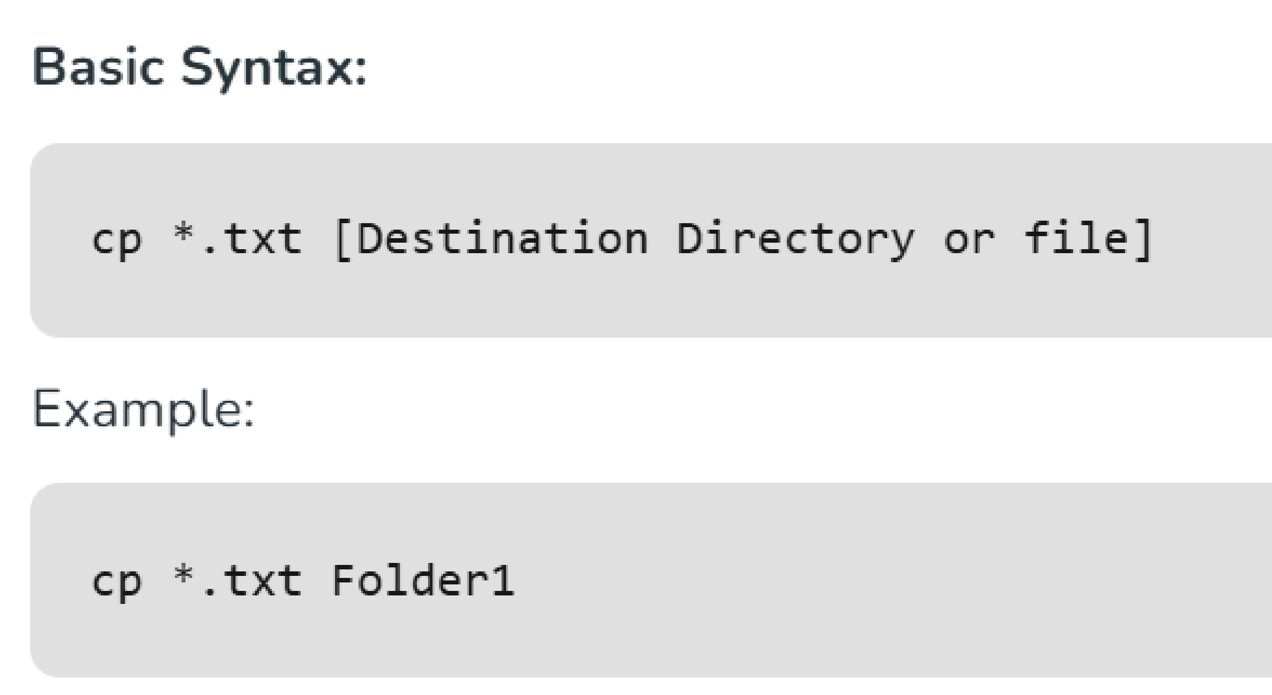
Example2:Copy the Desk_folder directory contents to the Dest_folder Directory located in Desktop Directory



```
ubuntu@ubuntu:~/Desktop$  cp -r Desk_folder Dest_folder
ubuntu@ubuntu:~/Desktop$ 
```

Copy a File in Linux Using `*` Option

- **Copying using * wildcard:** The star wildcard represents anything i.e., all files and directories. Suppose we have many texts documents in a directory and want to copy it to another directory, it takes lots of time if we copy files 1 by 1 or command becomes too long if specify all these file names as the argument, but by using * wildcard it becomes simple.

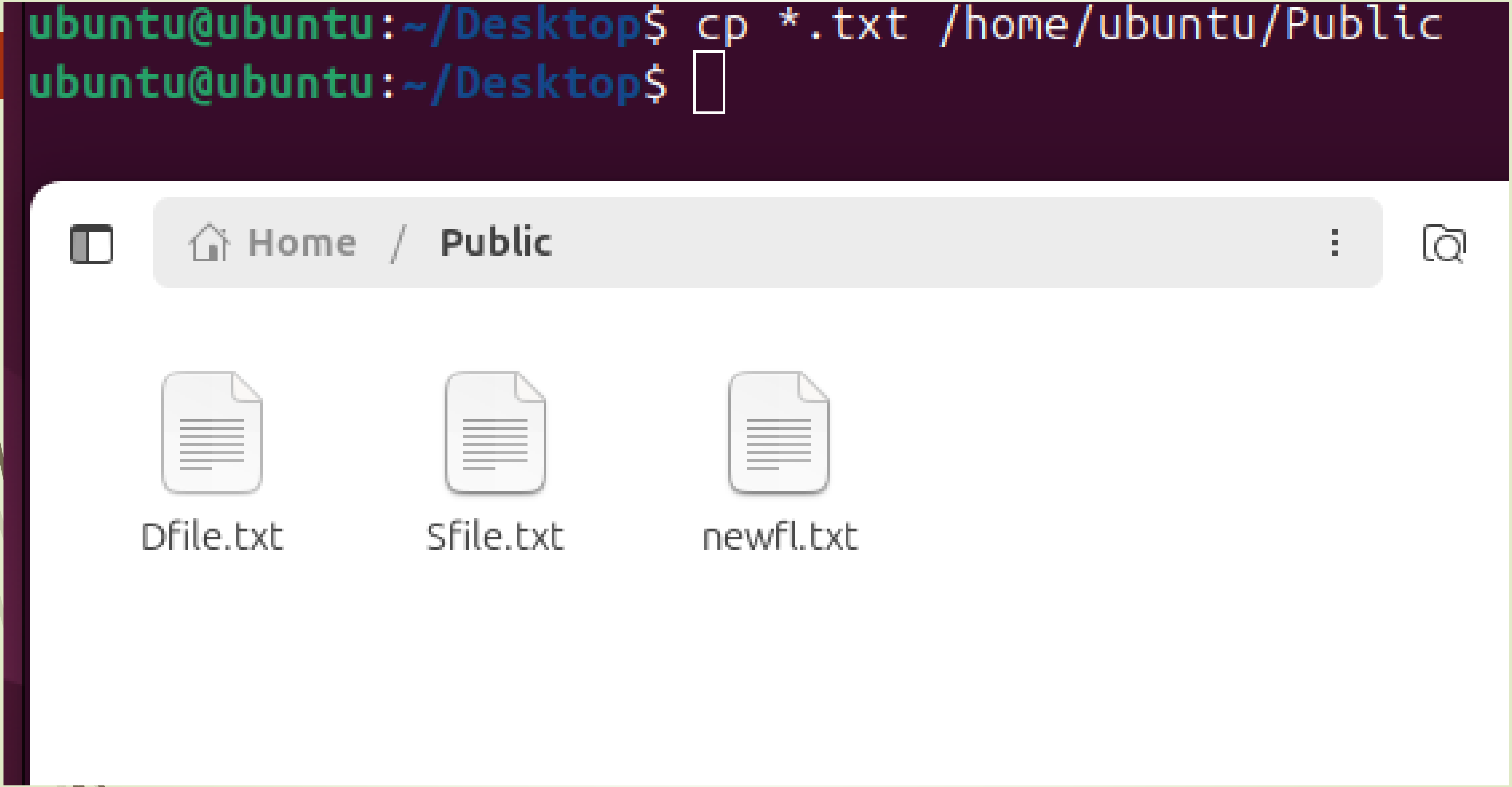


Basic Syntax:

```
cp *.txt [Destination Directory or file]
```

Example:

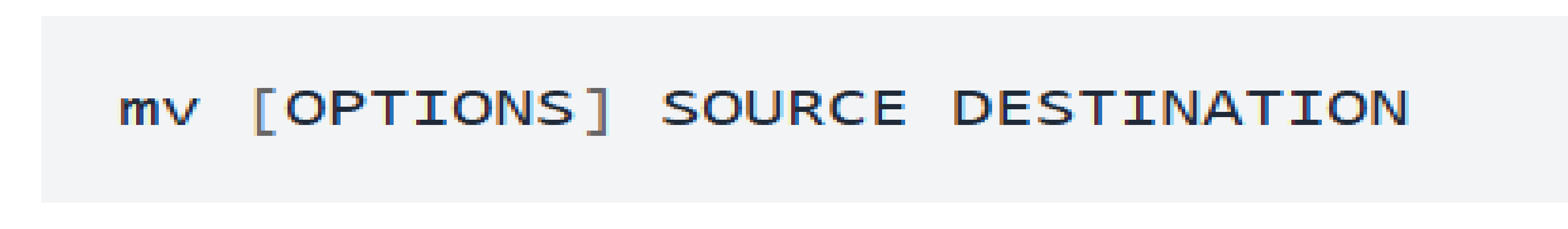
```
cp *.txt Folder1
```



```
ubuntu@ubuntu:~/Desktop$ cp *.txt /home/ubuntu/Public
ubuntu@ubuntu:~/Desktop$ 
```

Moving Files

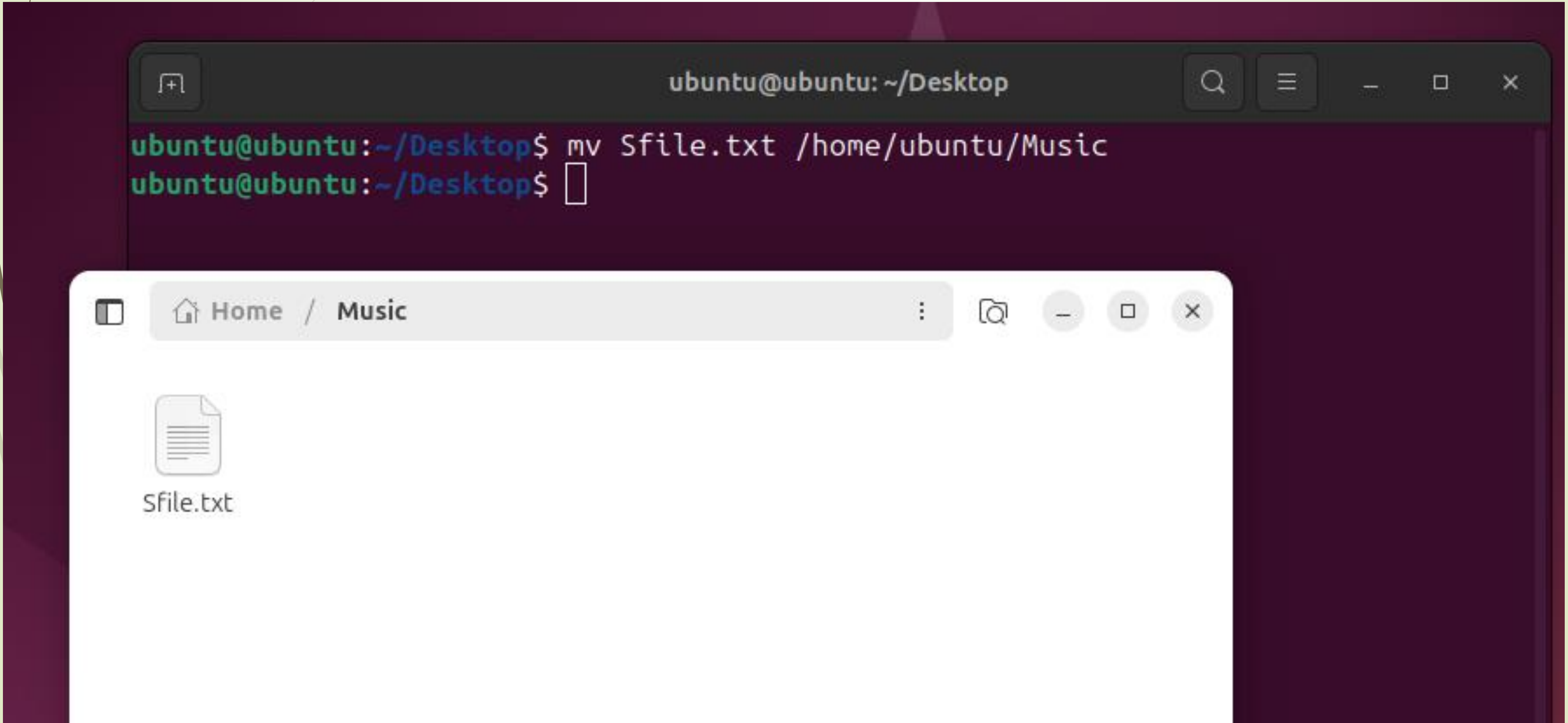
- The mv command (short from move) is used to rename and move and files and directories from one location to another. The syntax for the mv command is as follows:



```
mv [OPTIONS] SOURCE DESTINATION
```

- The mv command requires at least two arguments. The first argument is the source, a path to the file to be moved. The second argument is the destination, a path to where the file will be moved to. The files to be moved are sometimes referred to as the source, and the place where the files are to be placed is called the destination.

For example, to move the Sfile.txt from the current working directory to Music Directory

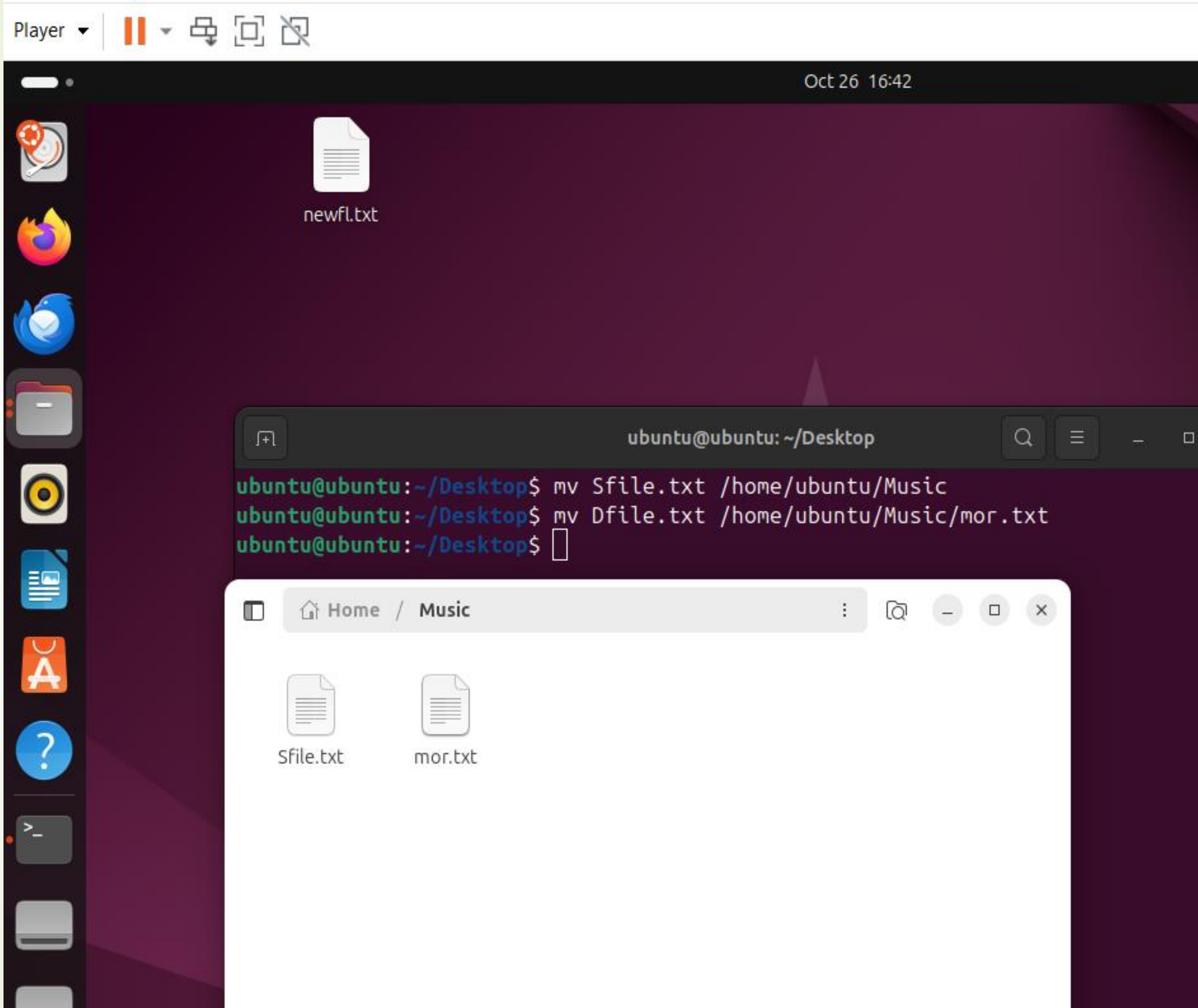


```
ubuntu@ubuntu:~/Desktop$ mv Sfile.txt /home/ubuntu/Music
ubuntu@ubuntu:~/Desktop$ 
```

To rename a file you need to specify the destination file name

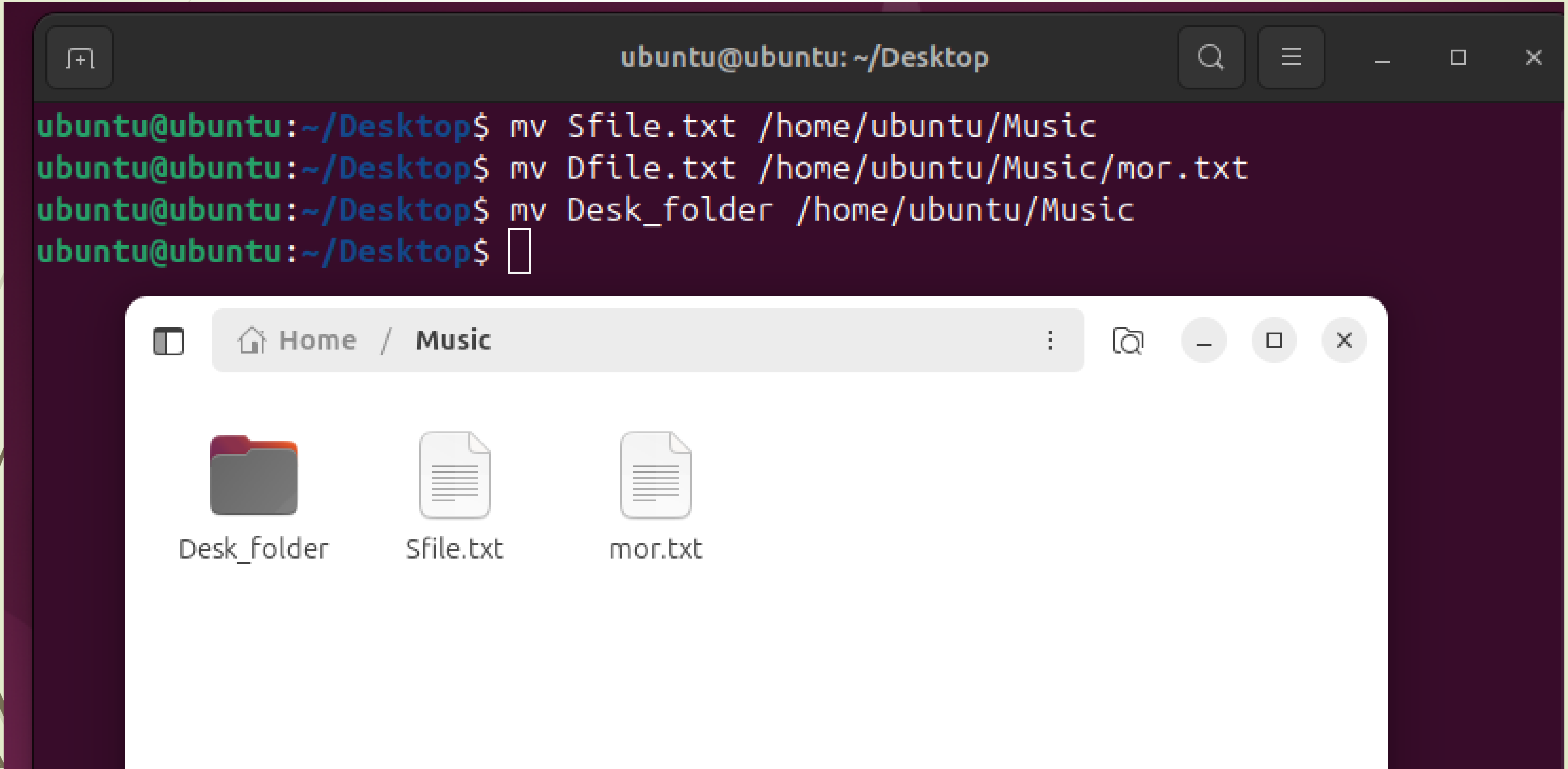
Example2:

Move the file Dfile.txt from Desktop directory to Music directory with anew name Mor.txt



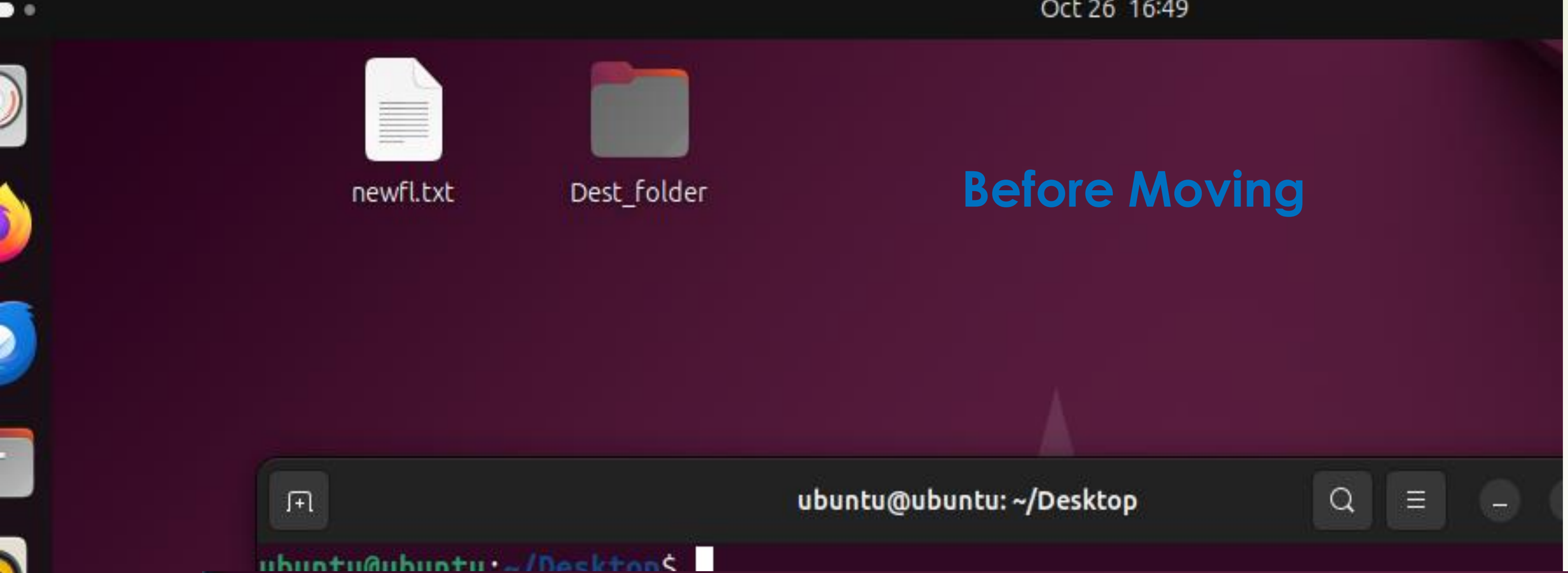
```
ubuntu@ubuntu:~/Desktop$ mv Sfile.txt /home/ubuntu/Music
ubuntu@ubuntu:~/Desktop$ mv Dfile.txt /home/ubuntu/Music/mor.txt
ubuntu@ubuntu:~/Desktop$ 
```

The syntax for moving directories is the same as when moving files for example move the Desk_folder to Music directory:



```
ubuntu@ubuntu:~/Desktop$ mv Sfile.txt /home/ubuntu/Music
ubuntu@ubuntu:~/Desktop$ mv Dfile.txt /home/ubuntu/Music/mor.txt
ubuntu@ubuntu:~/Desktop$ mv Desk_folder /home/ubuntu/Music
ubuntu@ubuntu:~/Desktop$ 
```

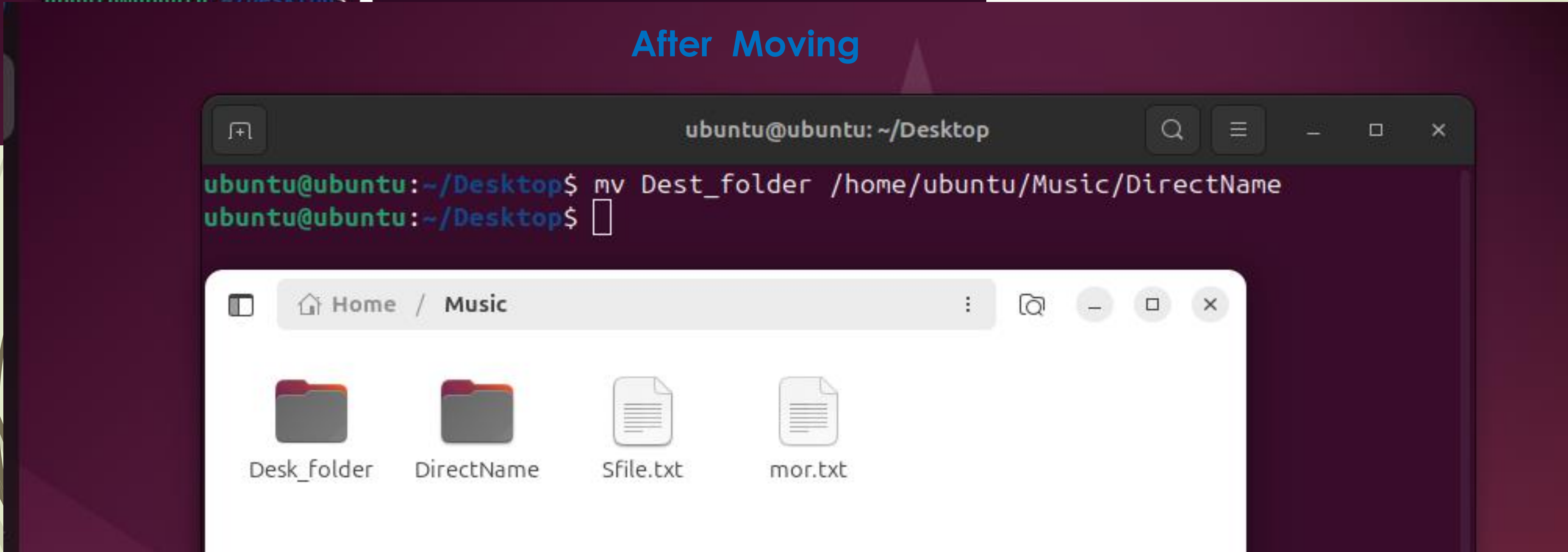
To move a directory to another one doesn't exist then it will be renamed.



Before Moving

ubuntu@ubuntu: ~/Desktop

```
ubuntu@ubuntu:~/Desktop$
```



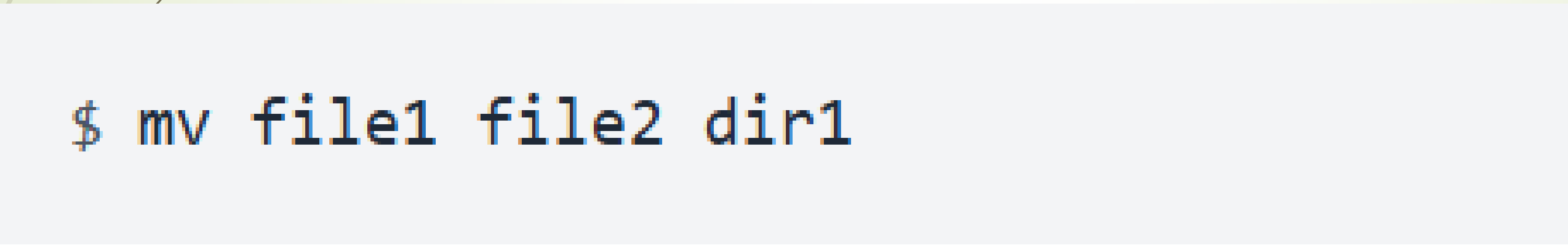
After Moving

ubuntu@ubuntu: ~/Desktop

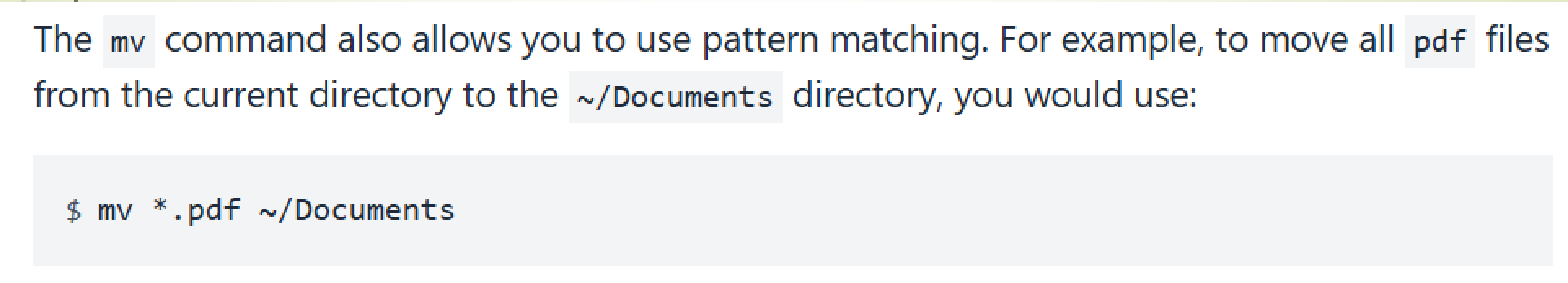
```
ubuntu@ubuntu:~/Desktop$ mv Dest_folder /home/ubuntu/Music/DirectName
ubuntu@ubuntu:~/Desktop$
```

Moving Multiple Files and Directories

- To move multiple files and directories, specify the files you want to move as the source. For example, to move the files file1 and file2 to the dir1 directory you would type:



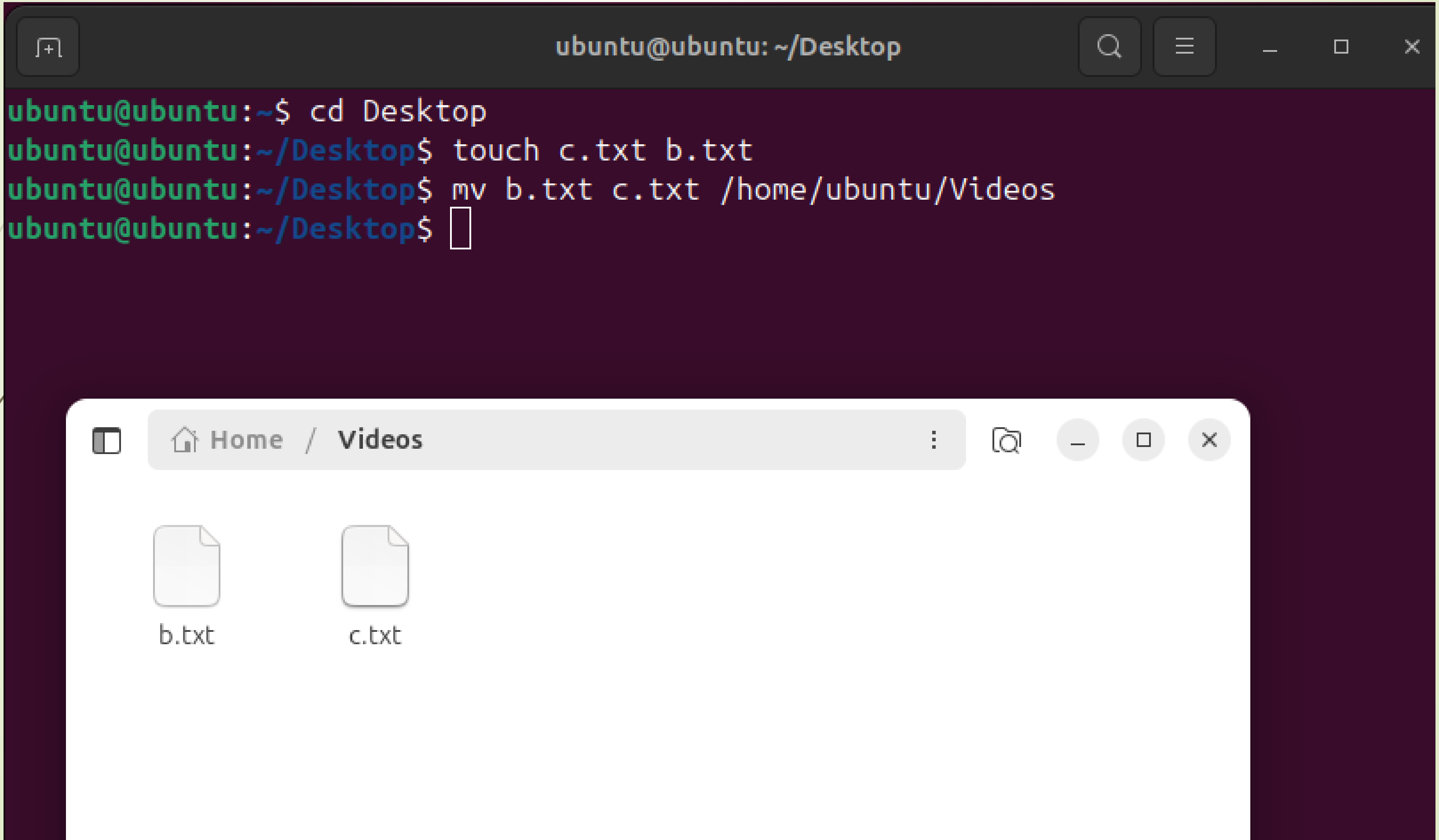
```
$ mv file1 file2 dir1
```



The `mv` command also allows you to use pattern matching. For example, to move all `pdf` files from the current directory to the `~/Documents` directory, you would use:

```
$ mv *.pdf ~/Documents
```

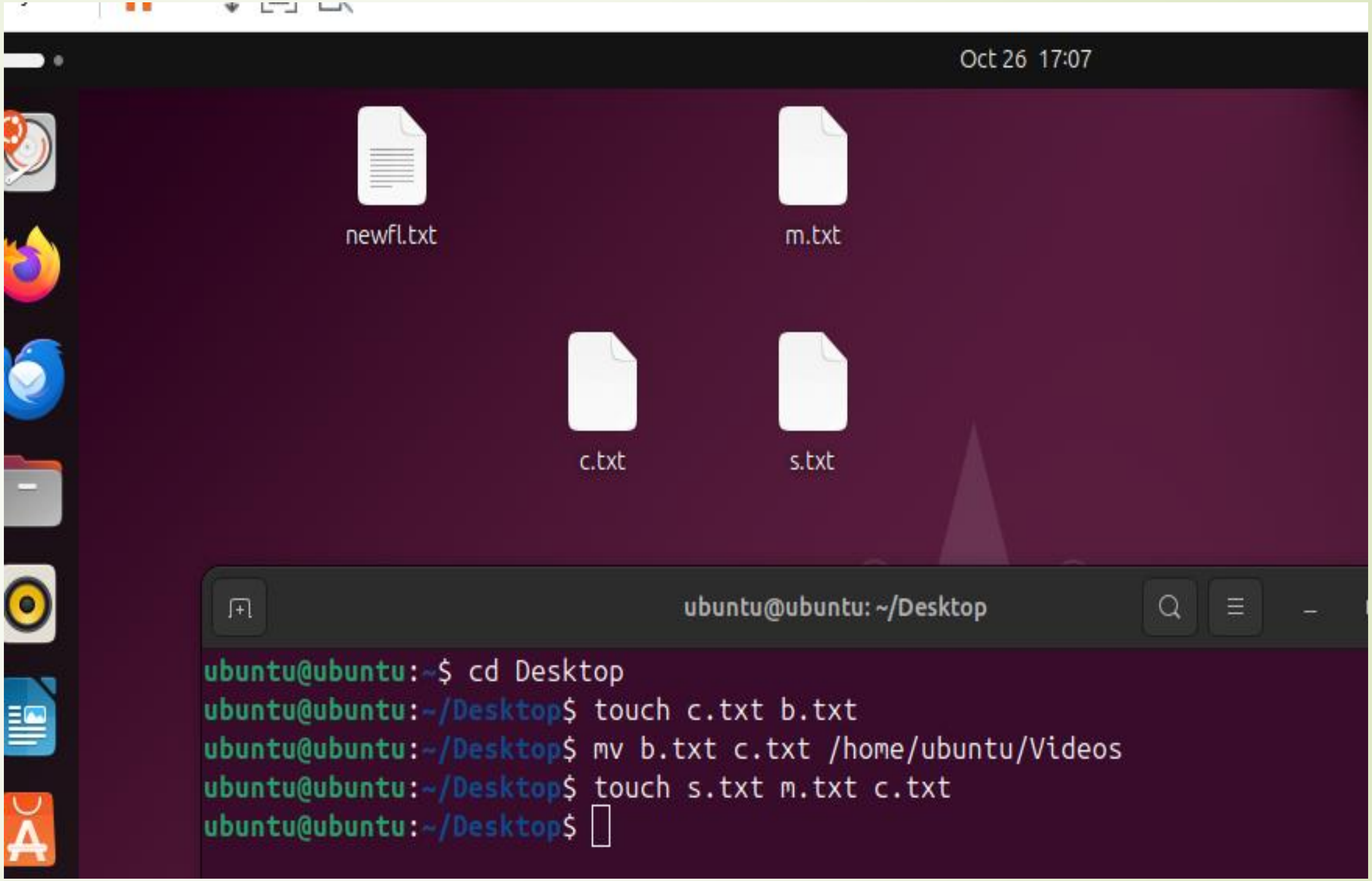
Example : move b.txt and c.txt from Desktop to Videos



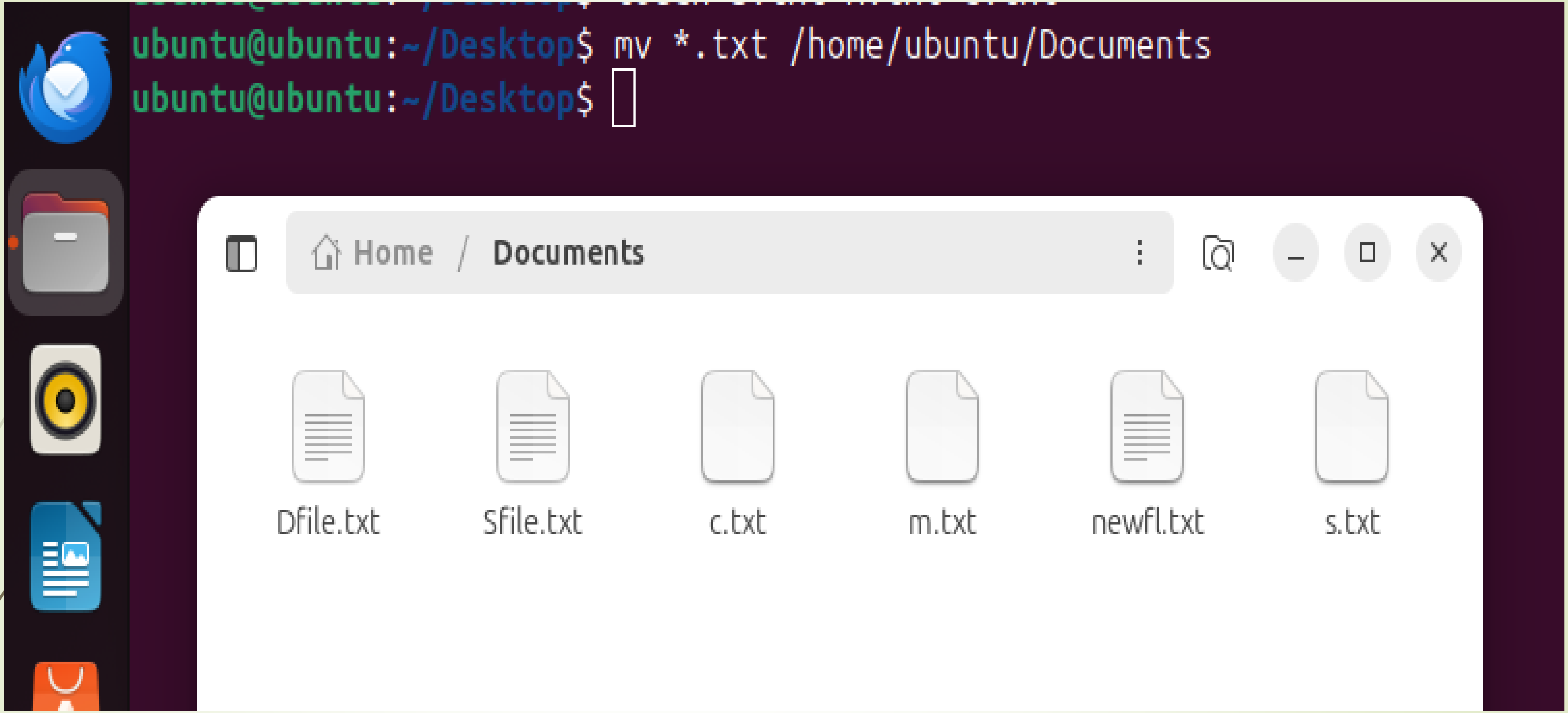
```
ubuntu@ubuntu:~$ cd Desktop
ubuntu@ubuntu:~/Desktop$ touch c.txt b.txt
ubuntu@ubuntu:~/Desktop$ mv b.txt c.txt /home/ubuntu/Videos
ubuntu@ubuntu:~/Desktop$ 
```

Example: move more than one file

Before Moving



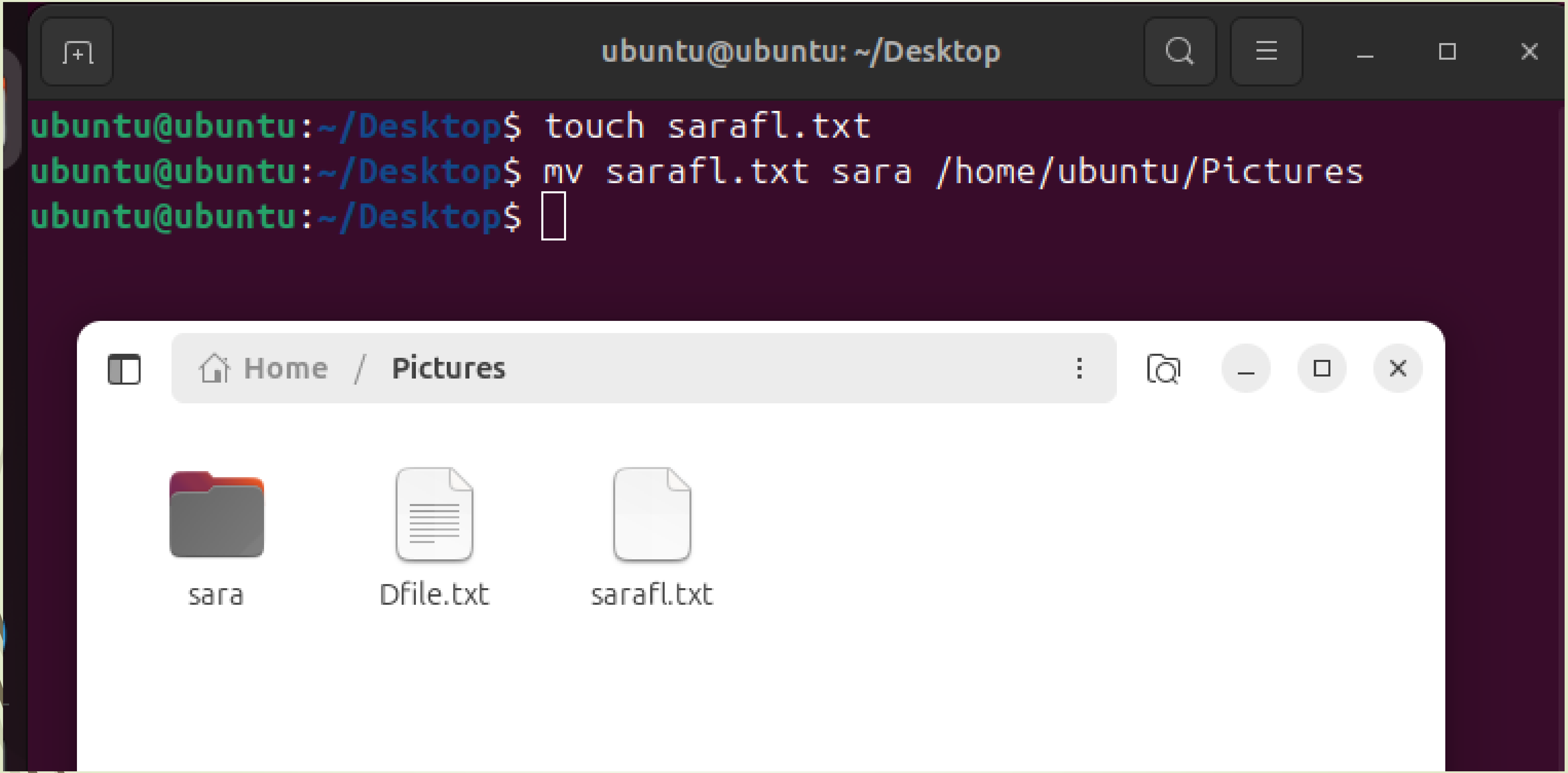
```
ubuntu@ubuntu:~$ cd Desktop
ubuntu@ubuntu:~/Desktop$ touch c.txt b.txt
ubuntu@ubuntu:~/Desktop$ mv b.txt c.txt /home/ubuntu/Videos
ubuntu@ubuntu:~/Desktop$ touch s.txt m.txt c.txt
ubuntu@ubuntu:~/Desktop$ 
```



```
ubuntu@ubuntu:~/Desktop$ mv *.txt /home/ubuntu/Documents
ubuntu@ubuntu:~/Desktop$ 
```

After Moving

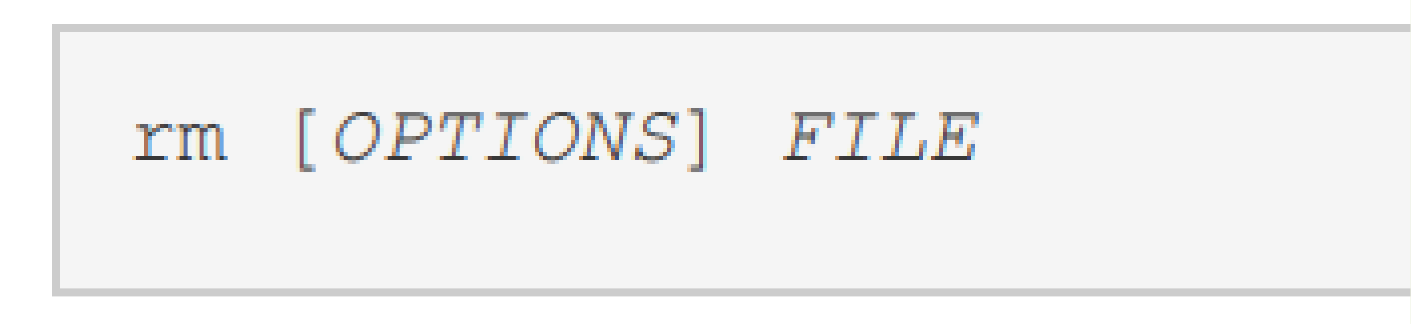
Example: Move files and directories



```
ubuntu@ubuntu:~/Desktop$ touch sarafl.txt
ubuntu@ubuntu:~/Desktop$ mv sarafl.txt sara /home/ubuntu/Pictures
ubuntu@ubuntu:~/Desktop$ 
```

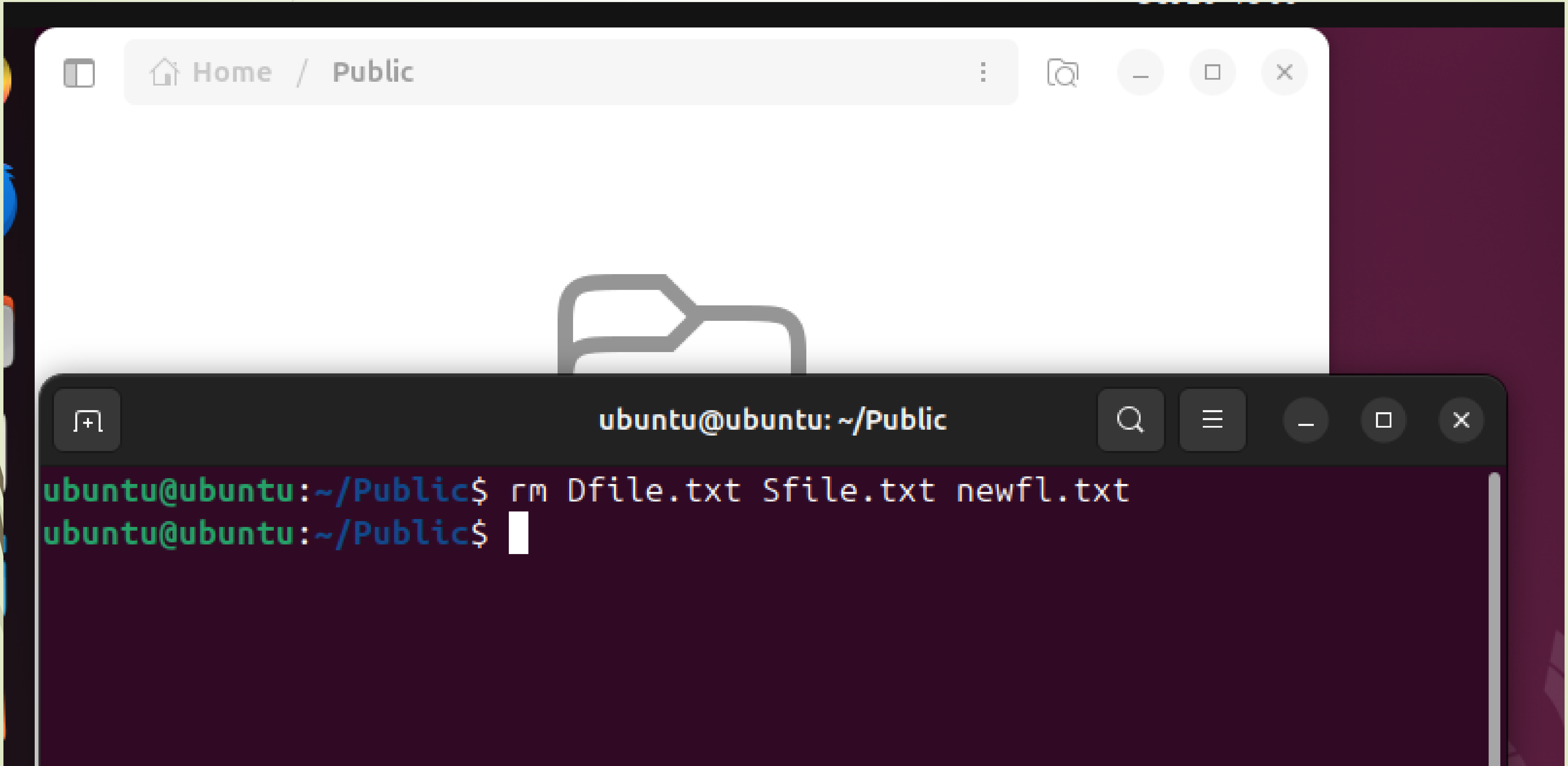
Removing Files

- The rm command is used to delete files and directories. It is important to keep in mind that deleted files and directories do not go into a "trash can" as with desktop-oriented operating systems. When a file is deleted with the rm command, it is almost always permanently gone:



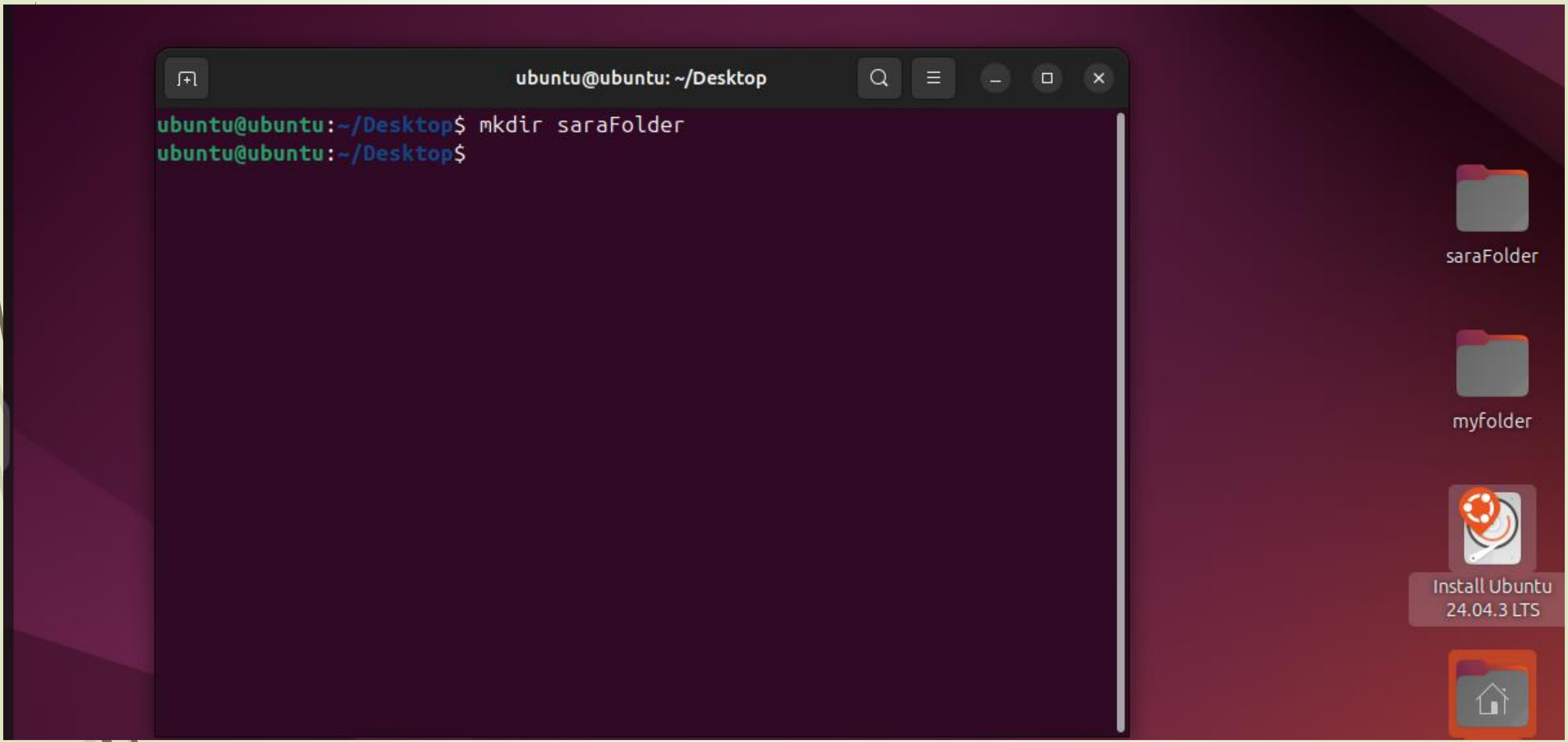
```
rm [OPTIONS] FILE
```

Example for removing the files from Public directory



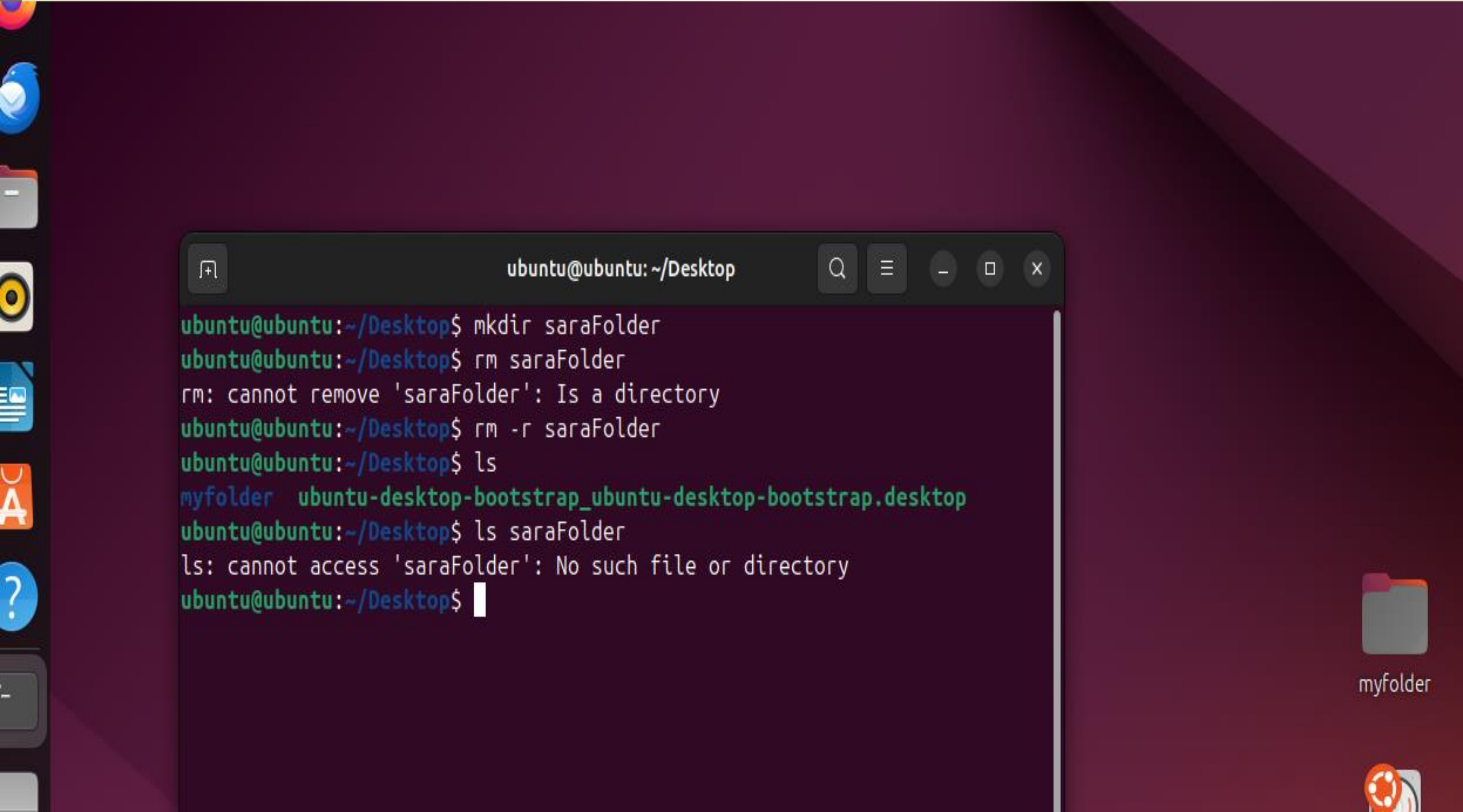
```
ubuntu@ubuntu:~/Public$ rm Dfile.txt Sfile.txt newfl.txt
ubuntu@ubuntu:~/Public$ 
```

To create directory, use the command mkdir



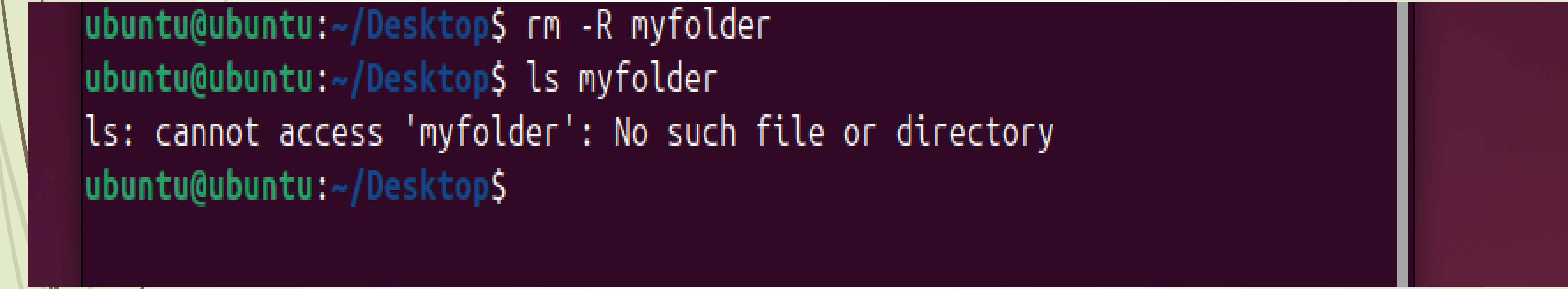
```
ubuntu@ubuntu:~/Desktop$ mkdir saraFolder
ubuntu@ubuntu:~/Desktop$
```

The rm command will ignore directories that it's asked to remove; to delete a directory, use a recursive option, either the -r or -R options. Just be careful since these options are "recursive", this will delete all files and all subdirectories.

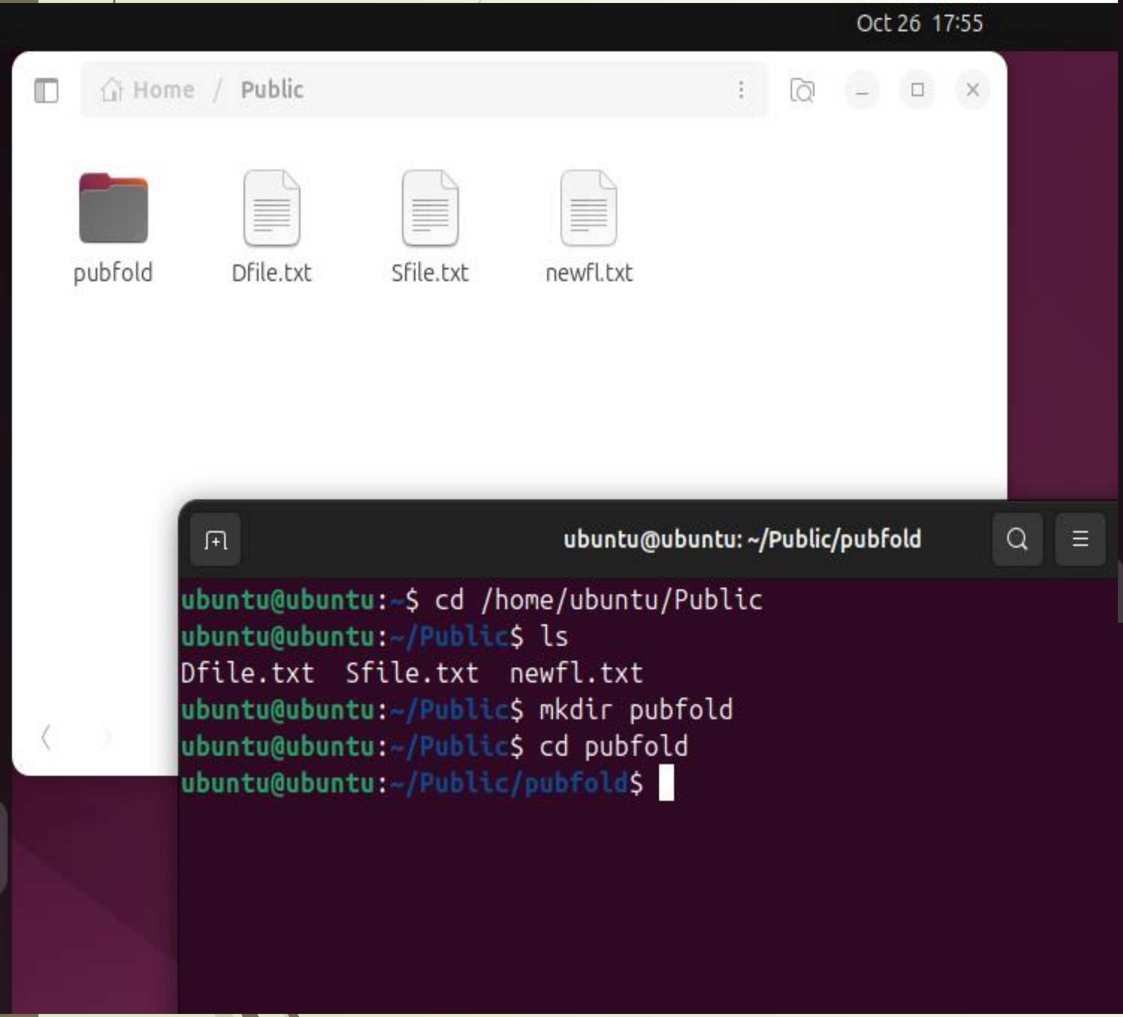


```
ubuntu@ubuntu:~/Desktop$ mkdir saraFolder
ubuntu@ubuntu:~/Desktop$ rm saraFolder
rm: cannot remove 'saraFolder': Is a directory
ubuntu@ubuntu:~/Desktop$ rm -r saraFolder
ubuntu@ubuntu:~/Desktop$ ls
myfolder  ubuntu-desktop-bootstrap_ubuntu-desktop-bootstrap.desktop
ubuntu@ubuntu:~/Desktop$ ls saraFolder
ls: cannot access 'saraFolder': No such file or directory
ubuntu@ubuntu:~/Desktop$ 
```

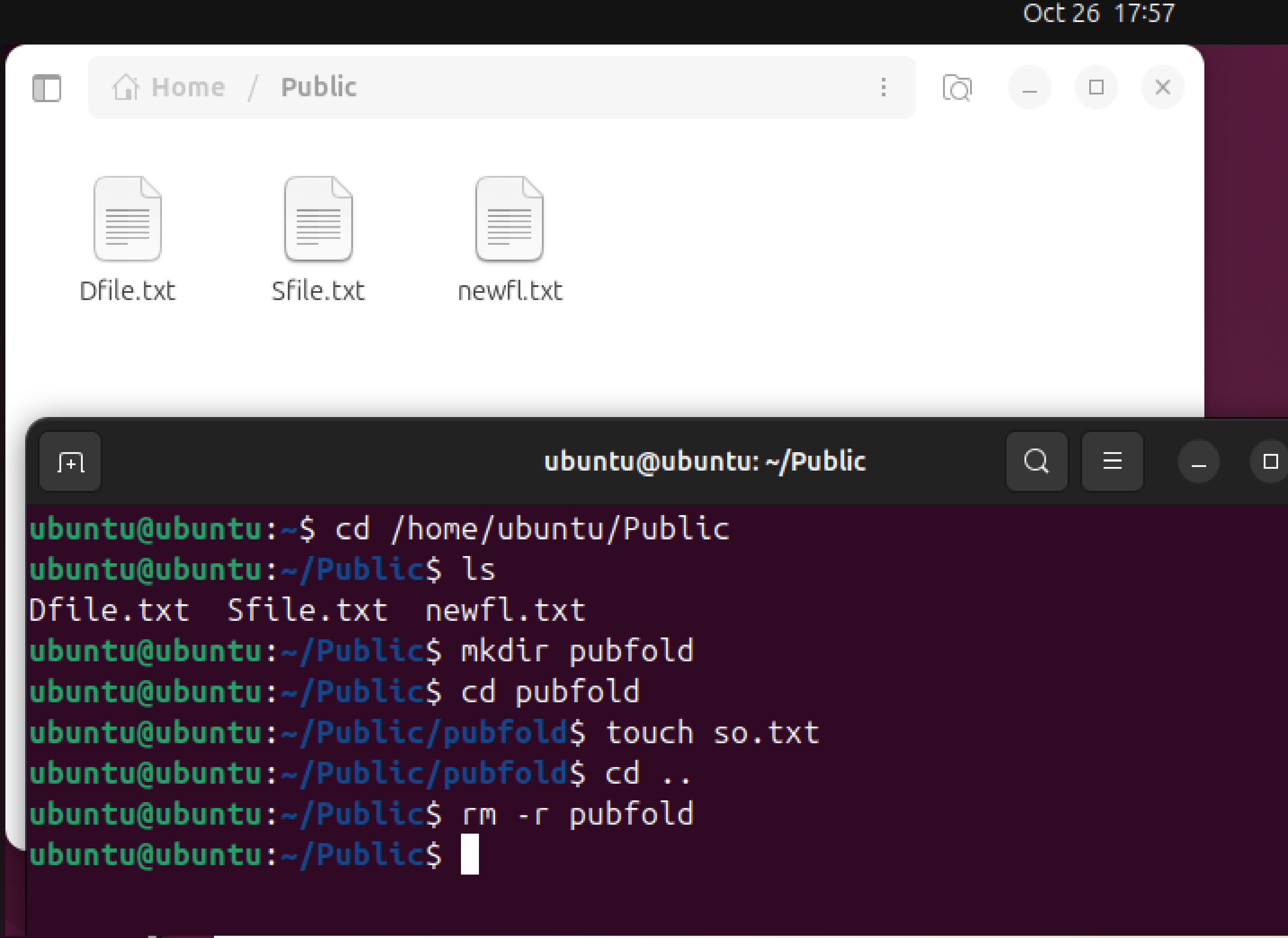
Example for using –R option with the rm command



```
ubuntu@ubuntu:~/Desktop$ rm -R myfolder
ubuntu@ubuntu:~/Desktop$ ls myfolder
ls: cannot access 'myfolder': No such file or directory
ubuntu@ubuntu:~/Desktop$
```



```
ubuntu@ubuntu:~$ cd /home/ubuntu/Public
ubuntu@ubuntu:~/Public$ ls
Dfile.txt  Sfile.txt  newfl.txt
ubuntu@ubuntu:~/Public$ mkdir pubfold
ubuntu@ubuntu:~/Public$ cd pubfold
ubuntu@ubuntu:~/Public/pubfold$ 
```



```
ubuntu@ubuntu:~$ cd /home/ubuntu/Public
ubuntu@ubuntu:~/Public$ ls
Dfile.txt  Sfile.txt  newfl.txt
ubuntu@ubuntu:~/Public$ mkdir pubfold
ubuntu@ubuntu:~/Public$ cd pubfold
ubuntu@ubuntu:~/Public/pubfold$ touch so.txt
ubuntu@ubuntu:~/Public/pubfold$ cd ..
ubuntu@ubuntu:~/Public$ rm -r pubfold
ubuntu@ubuntu:~/Public$ 
```

Example: remove the pubfold which contains so.txt

References

- https://www.geeksforgeeks.org/linux-unix/cp-command-linux-examples/
- https://linuxize.com/post/how-to-move-files-in-linux-with-mv-command/
- Ramses van Zon," Securing File Access Permissions on Linux ", SciNet HPC, University of Toronto ,27 October 2022.
- https://www.geeksforgeeks.org/linux-unix/uname-command-in-linux-with-examples/