**--------Creation of files -----------**

Creation of Files:

In Linux, we can create files in the following ways:

By using touch command (to create empty file)

By using cat command

By using editors like gedit, vi, nano etc

**9. cat :-**

cat > file1.txt

Eg:

$ cat > file1.txt

Hello Friends

Listen Carefully

Otherwise Linux will give Left and Right ctrl+d To save and exit

If file1.txt is not already available, then file1.txt will be created with our provided data.

If file1.txt is already available with some content, then old data will be over written with

our provided new data.

Instead of overwriting, if we want append operation then we should use >> with cat command.

cat >> file1.txt extra content ctrl+d

**>> for appending**

) If we are using Touch Comamnd, but the File is already available then what will happend?

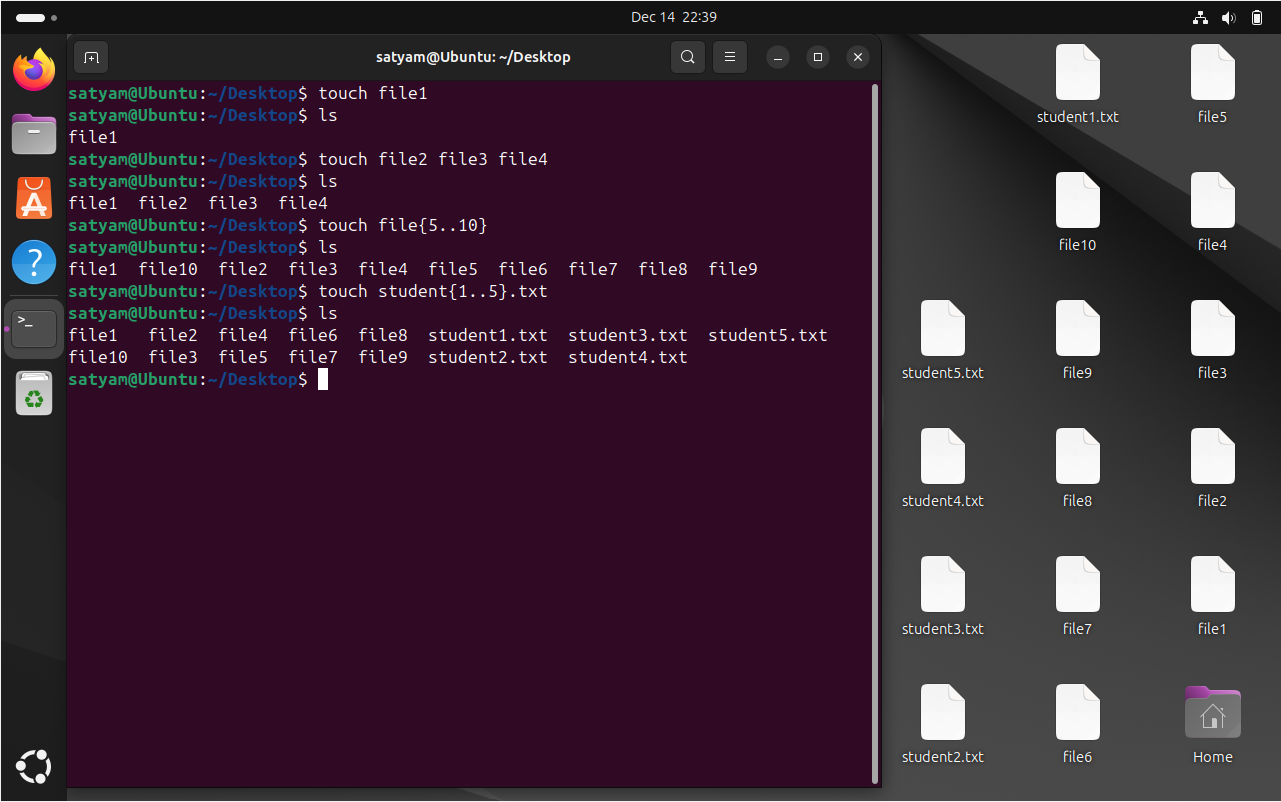
The content of the file won't be changed. But last modified date and time (i.e., timestamp) will be updated

**10. Touch :-**

touch command is a way to create empty files (there are some other mehtods also).

You can update the modification and access time of each file with the help of touch command.

 creating files using touch



touch file1   ------> change timestamp ( both access and modify time)

touch -a file1    -----> change access file of file1

touch -m file1 ------> change modify time of file1

touch -r file1 file2 -----> use file1's timestamp as reference and change timestamp of file2

                                          ( now file2 timestamp will change and become same as file1 )

try :- touch -r file2 -a file1   and observe using stat command what happens

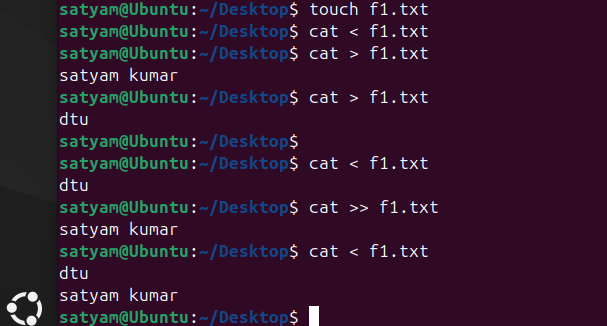
**---------View Content of the Files -------------**

We can view content of the file by using the following commands

* cat
* tac
* rev
* head
* tail
* less
* more

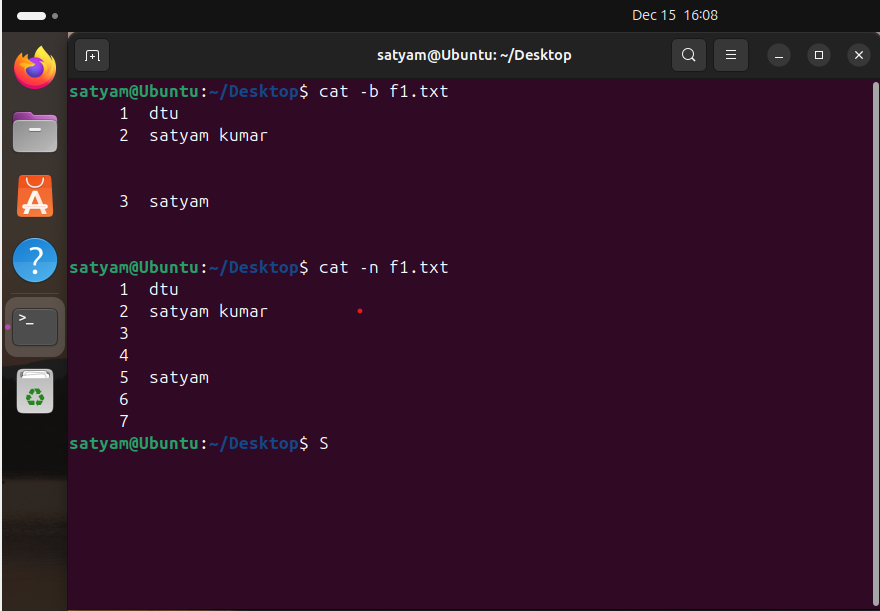
**11. cat :-**

$ cat < file1.txt



-n  option to give line numbering to file content

-b to give numbering to all lines apart from blank lines



We can view multiple files content at a time by using cat command.

$ cat file1.txt file2.txt file3.txt

Various utilities of cat Command:

To create new file with some content

$ cat > filename data ctrl+d

To append some extra data to existing file

$ cat >> filename extra data ctrl+d

To view content of file

$ cat < filename or $ cat filename

Copy content of one file to another file

$ cat input.txt > output.txt

To copy content of multiple files to a single file

$ cat file1.txt file2.txt file3.txt > file4.txt

Merging/appending of one file content to another file

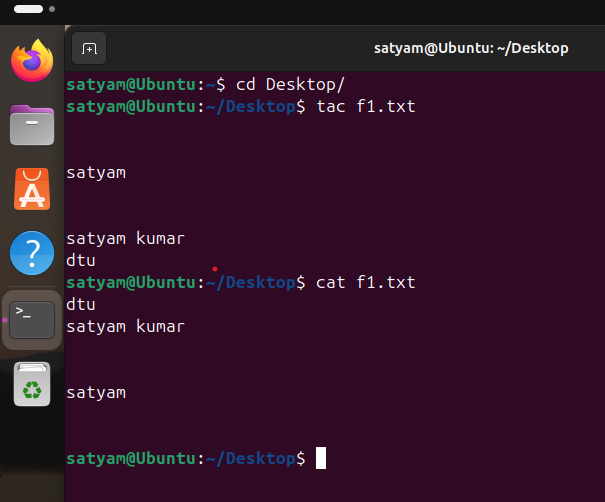
$ cat file1.txt >> file2.txt

**12. tac :-**

It is the reverse of cat.

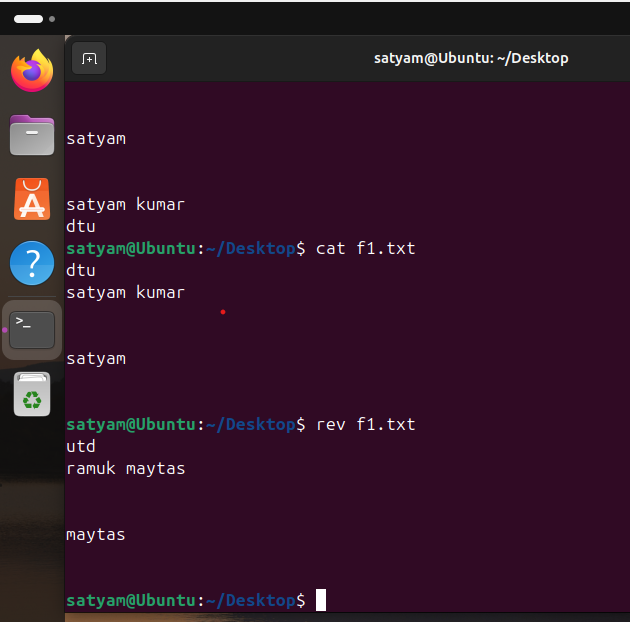
It will display file content in reverse order of lines. i.e first line will become last line and last line will become first line.

This is vertical reversal.



**rev :**

rev means reverse. Here each line content will be reversed. It is horizontal reversal.



cat command will display total file content at a time. It is best suitable for small files. If the file contains huge lines then it is not recommended to use cat command. We should go for head, tail, less and more commands.

**head :**

We can use head command to view top few lines of content.

✽ head file1.txt

It will display top 10 lines of file1.txt.

10 is the default value of number of lines.

✽ head -n 30 file1.txt OR head -30 file1.txt To display top 30 lines of the file.

Instead of 30 we can specify any number.

✽ head -n -20 file1.txt

To display all lines of file1.txt except last 20 lines.

✽ head -c 100 file1.txt

To display first 100 bytes of file content

**tail :**

We can use tail command to view few lines from bottom of the file. It is opposite to head command.

✽ tail file1.txt

Last 10 lines will be displayed.

✽ tail -n 30 file1.txt OR tail -30 file1.txt OR tail -n -30 file1.txt It will display last 30 lines.

✽ tail -n +4 file1.txt

It will display from 4th line to last line

✽ tail -c 200 file1.txt

It will display 200 bytes of content from bottom of the file.

**more :**

We can use more command to view file content page by page.

✽ more file1.txt

It will display first page.

Enter     To view next line

 Space Bar     To view next page

 q     To quit/exit

✽ more -d file1.txt

-d option meant for providing details like --More--(5%)[Press space to continue, 'q' to quit.]

**less :**

By using more command, we can view file content page by page only in forward direction.

If we want to move either in forward direction or in backward direction then we should go for less command.

**d**  To go to next page.(d means down)

**b** To go to previous page. (b means backward)

**Creation of Hidden Files and Directories:-**

If any file starts with '.' , such type of file is called hidden file.

If we don't want to display the files then we have to go for hidden files.

Hidden files meant for hiding data. All system files which are internally required by kernal are hidden files.

We can create hidden files just like normal files, only difference is file name should starts with dot.

touch .securefile1.txt

 cat > .securefile1.txt

Even by using editors also we can create hidden files.

We can create hidden directories also just like normal directories. mkdir .db\_info

Note: By using hidden files and directories we may not get full security. To make more secure we have to use proper permissions. For this we should use 'chmod' command.

**Interconversion of Normal Files and Hidden Files:**

Based on our requirement, we can convert normal file as hidden file and viceversa.

mv a.txt .a.txt

We are converting normal file a.txt as hidden file.

 mv .a.txt a.txt

Similarly directories also

mv dir1 .dir1

mv .dir1 dir1