



2.FILE PERMISSION AND FILTER COMMANDS

Viswanathan P
2022242001
Msc Integrated IT



Ex. No: 2

Date: 14-08-2023

File Permission and Filter commands in Linux

Aim:

To learn filter commands and file permission commands in Linux.

Commands:

- **sort:**

Description: Sorts lines of text from standard input or a file alphabetically. It is useful for organizing, analysing, and filtering data in files. By default, the sort command sorts file assuming the contents are ASCII.

Syntax: ~\$ sort [options] [param] [filename]

Options:

- r, --reverse Reverse the result of comparisons.
- o, --output=FILE Write result to FILE instead of standard output.
- k, --key=KEYDEF Sort via a key; KEYDEF gives location and type.
- b, --ignore-leading-blanks Ignore leading blank spaces.
- n, --numeric-sort Compare according to string numerical value.

Output:

```
varshini@desktop: ~/2022239013
varshini@desktop:~/2022239013$ cat marks.txt
Name:      Mark1:  Mark2:  Sub1:   Sub2:   Roll_No:
goki       71      86     C      Maths   11
varshini   86      94     Maths  English 13
rohini     79      80     Maths  Tamil   2
ganga      85      98     CA     OS      18
daisy     90      70     CGT    OS      14
varshini@desktop:~/2022239013$ sort marks.txt
daisy     90      70     CGT    OS      14
ganga      85      98     CA     OS      18
goki       71      86     C      Maths   11
Name:      Mark1:  Mark2:  Sub1:   Sub2:   Roll_No:
rohini     79      80     Maths  Tamil   2
varshini   86      94     Maths  English 13
varshini@desktop:~/2022239013$ sort -k 2n marks.txt
Name:      Mark1:  Mark2:  Sub1:   Sub2:   Roll_No:
goki       71      86     C      Maths   11
rohini     79      80     Maths  Tamil   2
ganga      85      98     CA     OS      18
varshini   86      94     Maths  English 13
daisy     90      70     CGT    OS      14
varshini@desktop:~/2022239013$ sort -r marks.txt
varshini   86      94     Maths  English 13
rohini     79      80     Maths  Tamil   2
Name:      Mark1:  Mark2:  Sub1:   Sub2:   Roll_No:
goki       71      86     C      Maths   11
ganga      85      98     CA     OS      18
daisy     90      70     CGT    OS      14
varshini@desktop:~/2022239013$
```

- **uniq:**

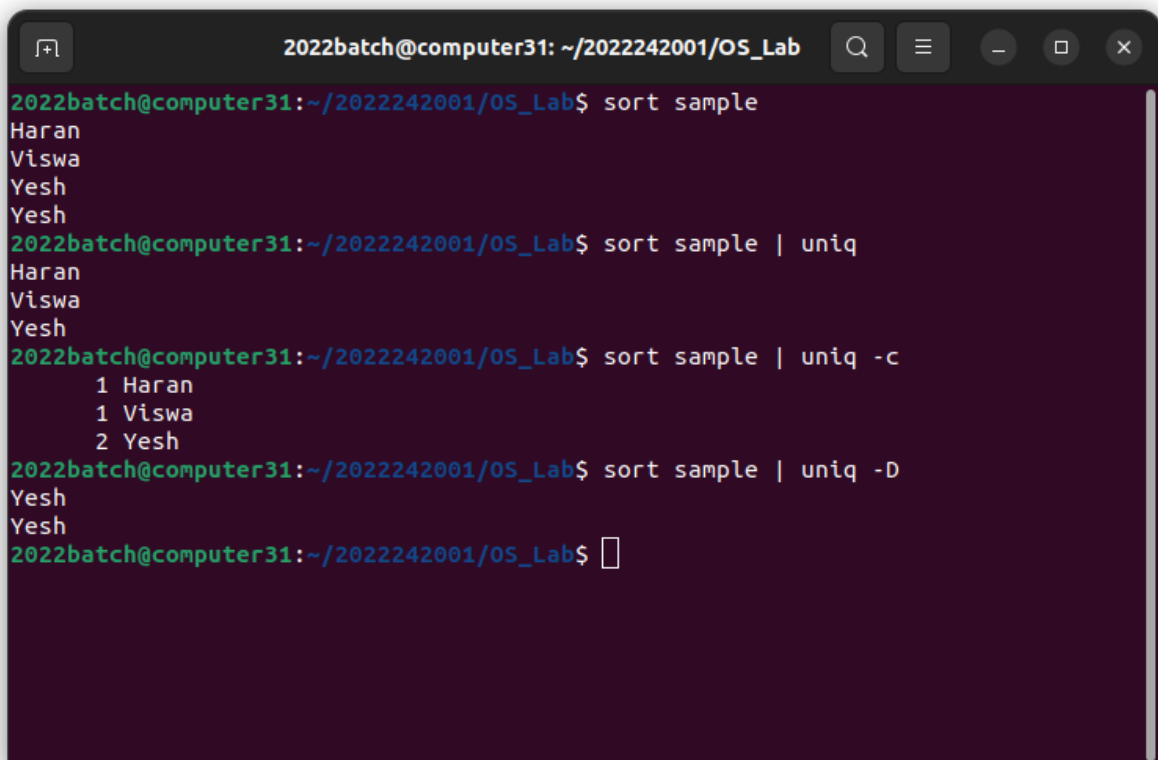
Description: Filter adjacent matching lines from INPUT (or standard input), writing to OUTPUT (or standard output). With no options, matching lines are merged to the first occurrence. Mandatory arguments to long options are mandatory for short options too.

Syntax: `~$ uniq [options] [filename]`

Options:

- c, --count it prefixes the lines by the number of occurrences.
- d, --repeated it is used to print duplicate lines, one for each group.
- D It is used to print all the duplicate lines.
- u, --unique it is used to print unique lines.

Output:

A terminal window titled '2022batch@computer31: ~/2022242001/OS_Lab' showing a series of commands and their outputs. The commands are: 'sort sample' (output: Haran, Viswa, Yesh, Yesh), 'sort sample | uniq' (output: Haran, Viswa, Yesh), 'sort sample | uniq -c' (output: 1 Haran, 1 Viswa, 2 Yesh), and 'sort sample | uniq -D' (output: Yesh, Yesh). The prompt is '~\$' and the cursor is at the end of the last line.

```
2022batch@computer31: ~/2022242001/OS_Lab
2022batch@computer31:~/2022242001/OS_Lab$ sort sample
Haran
Viswa
Yesh
Yesh
2022batch@computer31:~/2022242001/OS_Lab$ sort sample | uniq
Haran
Viswa
Yesh
2022batch@computer31:~/2022242001/OS_Lab$ sort sample | uniq -c
  1 Haran
  1 Viswa
  2 Yesh
2022batch@computer31:~/2022242001/OS_Lab$ sort sample | uniq -D
Yesh
Yesh
2022batch@computer31:~/2022242001/OS_Lab$
```

- **cut:**

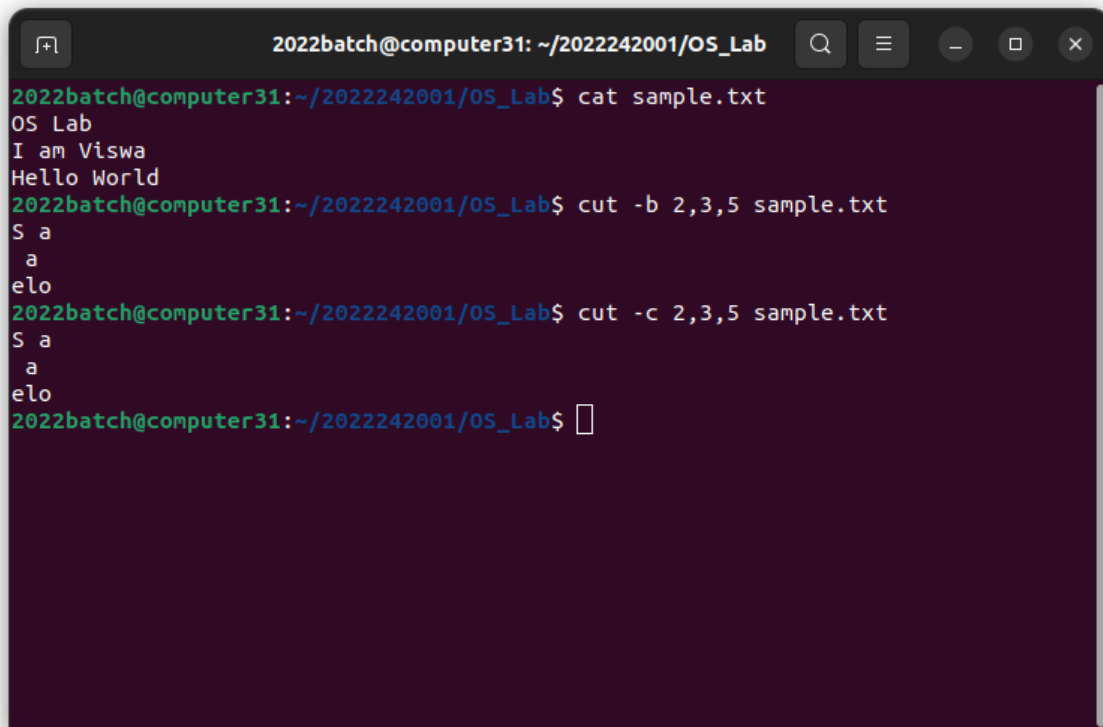
Description: Print selected parts of lines from each FILE to standard output.

Syntax: `~$ cut [options] [param] [filename]`

Options:

- b, --bytes=LIST Select only these bytes
- c, --characters=LIST Select only these characters

Output:



```
2022batch@computer31: ~/2022242001/OS_Lab
2022batch@computer31:~/2022242001/OS_Lab$ cat sample.txt
OS Lab
I am Viswa
Hello World
2022batch@computer31:~/2022242001/OS_Lab$ cut -b 2,3,5 sample.txt
S a
a
elo
2022batch@computer31:~/2022242001/OS_Lab$ cut -c 2,3,5 sample.txt
S a
a
elo
2022batch@computer31:~/2022242001/OS_Lab$
```

- **wc:**

Description: Print newline, word, and byte counts for each FILE, and a total line if more than one FILE is specified. A word is a non-zero-length sequence of characters delimited by white space.

Syntax: ~\$ wc [options] [param] [filename]

Options:

- | | |
|----------------------|----------------------------------|
| -c --bytes | Print the byte count. |
| -m --chars | Print the character count. |
| -l --lines | Print the newline count. |
| -L --max-line-length | Print the maximum display width. |
| -w --words | Print the word counts. |

Output:

```
2022batch@computer31: ~/2022242001/OS_Lab
2022batch@computer31:~/2022242001/OS_Lab$ cat sample.txt
OS Lab
    i am Viswa
Hello World
2022batch@computer31:~/2022242001/OS_Lab$ cat sample.txt | wc
    3      7    31
2022batch@computer31:~/2022242001/OS_Lab$ wc -w sample.txt
7 sample.txt
2022batch@computer31:~/2022242001/OS_Lab$ wc -l sample.txt
3 sample.txt
2022batch@computer31:~/2022242001/OS_Lab$
```

- **nl:**

Description: Write each FILE to standard output, with line numbers added.

Syntax: ~\$ nl [options] [param] [filename]

Options:

- b, --body-numbering=STYLE Use STYLE for numbering body lines
- s, --number-separator=STRING Add STRING after (possible) line number
- v, --starting-line-number=NUMBER First line number for each section.

Output:

```
viswa@desktop: ~/2022242001
viswa@desktop:~/2022242001$ nl test
 1 Welcome
 2 It is Os Lab

 3 Python is the powerful language
 4 It is raining
 5 It is unfair
viswa@desktop:~/2022242001$ nl -b 4 test
nl: invalid body numbering style: '4'
Try 'nl --help' for more information.
viswa@desktop:~/2022242001$ nl -v 4 test
 4 Welcome
 5 It is Os Lab

 6 Python is the powerful language
 7 It is raining
 8 It is unfair
viswa@desktop:~/2022242001$ nl -s "+" test
1+Welcome
2+It is Os Lab

3+Python is the powerful language
4+It is raining
5+It is unfair
viswa@desktop:~/2022242001$
```

- **paste:**

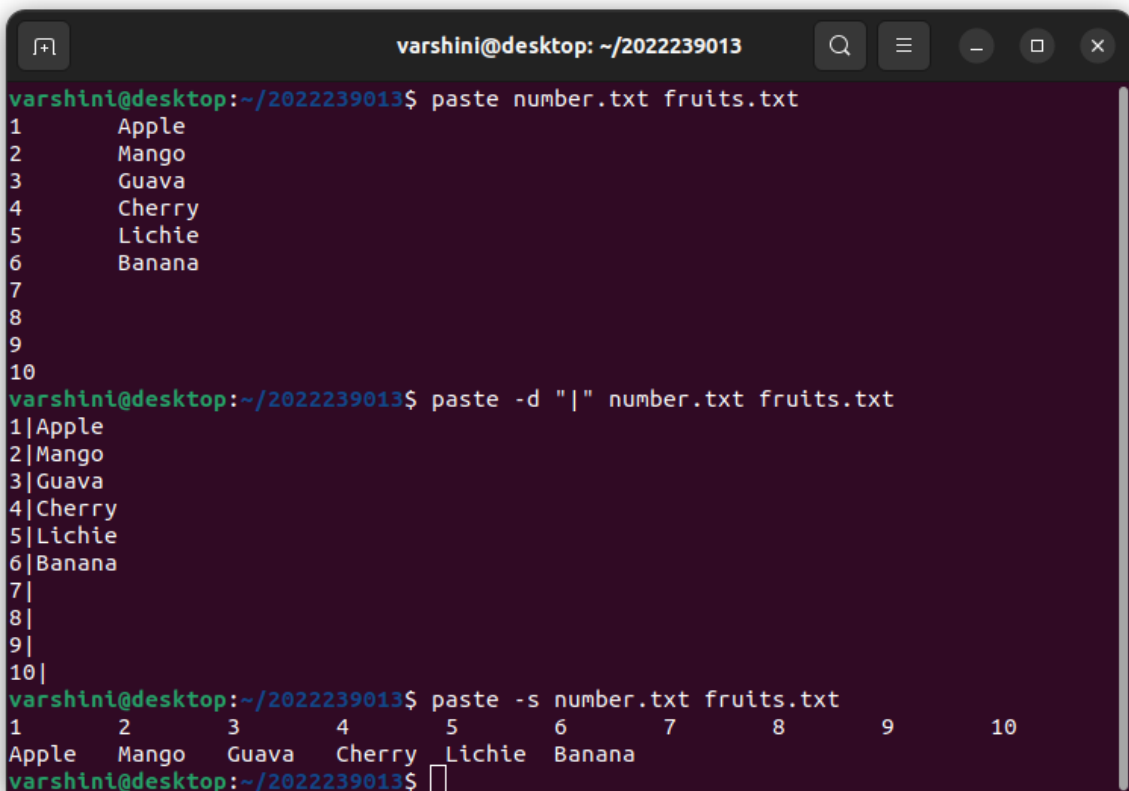
Description: Write lines consisting of the sequentially corresponding lines from each FILE, separated by TABs, to standard output.

Syntax: ~\$ paste [option] [param] [filename] [filename]

Options:

- d, --delimiters=LIST Reuse characters from LIST instead of TABs
- s, --serial Paste one file at a time instead of in parallel

Output:



```
varshini@desktop: ~/2022239013
varshini@desktop:~/2022239013$ paste number.txt fruits.txt
1      Apple
2      Mango
3      Guava
4      Cherry
5      Lichie
6      Banana
7
8
9
10
varshini@desktop:~/2022239013$ paste -d "|" number.txt fruits.txt
1|Apple
2|Mango
3|Guava
4|Cherry
5|Lichie
6|Banana
7|
8|
9|
10|
varshini@desktop:~/2022239013$ paste -s number.txt fruits.txt
1      2      3      4      5      6      7      8      9      10
Apple  Mango  Guava  Cherry Lichie Banana
varshini@desktop:~/2022239013$
```

- **tac:**

Description: Concatenate and print files in reverse.

Syntax: ~\$ tac [filename]

Options:

- h, --help Displays a help message and exits.
- V, --version Displays the version information and exits.

Output:

```
viswa@desktop: ~/2022242001
viswa@desktop:~/2022242001$ tac animals.txt
Dog
Hippo
Rhino
Monkey
Tiger
Lion
viswa@desktop:~/2022242001$ cat animals.txt
Lion
Tiger
Monkey
Rhino
Hippo
Dog
```

- **grep:**

Description: Grep searches for PATTERNS in each FILE. PATTERNS is one or more patterns separated by newline characters, and grep prints each line that matches a pattern. Typically, PATTERNS should be quoted when grep is used in a shell command.

Outputs:

cat student.txt

```
viswa@desktop:~/2022242001$ cat student.txt
Register_no:  Name:      Mark-1: Mark-2: Dept:  Gender:
9013          Peacock    78      90      CS      Female
9021          Turtle     69      80      CS      Male
2003          Koala      85      73      ECE     Female
2001          Tiger      99      98      IT      Male
9001          Lion       68      97      Mech    Male
2002          Elephant   80      70      IT      Female
9008          Snake      40      65      Mech    Male
9006          Fox        98      70      EEE     Male
```

1. grep “male” student.txt

```
viswa@desktop:~/2022242001$ grep "male" student.txt
9013          Peacock    78      90      CS      Female
2003          Koala      85      73      ECE     Female
2002          Elephant   80      70      IT      Female
```

2. grep -n "male" student.txt

```
viswa@desktop:~/2022242001$ grep -n "male" student.txt
2:9013      Peacock      78      90      CS      Female
4:2003      Koala        85      73      ECE      Female
7:2002      Elephant     80      70      IT       Female
```

3. grep -vn "male" student.txt

```
viswa@desktop:~/2022242001$ grep -vn "male" student.txt
1:Register_no: Name: Mark-1: Mark-2: Dept: Gender:
3:9021      Turtle      69      80      CS      Male
5:2001      Tiger       99      98      IT      Male
6:9001      Lion        68      97      Mech    Male
8:9008      Snake       40      65      Mech    Male
9:9006      Fox         98      70      EEE     Male
```

4. grep -c "male" student.txt

```
viswa@desktop:~/2022242001$ grep -c "male" student.txt
3
```

5. grep -l "male" student.txt

```
viswa@desktop:~/2022242001$ grep -l "male" student.txt
student.txt
```

6. grep -x '9006 Fox 98 70 EEE Male' student.txt

```
viswa@desktop:~/2022242001$ grep -x '9006      Fox      98
70      EEE      Male' student.txt
9006      Fox      98      70      EEE      Male
```

7. grep -A3 "2001" student.txt

```
viswa@desktop:~/2022242001$ grep -A3 "2001" student.txt
2001      Tiger      99      98      IT      Male
9001      Lion        68      97      Mech    Male
2002      Elephant     80      70      IT      Female
9008      Snake       40      65      Mech    Male
```

8. grep -l "2001" *

```
viswa@desktop:~/2022242001$ grep -l "2001" *
student.txt
grep: viswa: Is a directory
```

9. grep "e\$" student.txt


```
viswa@desktop:~/2022242001$ grep "e$" student.txt
9013      Peacock      78      90      CS      Female
9021      Turtle      69      80      CS      Male
9001      Lion      68      97      Mech      Male
2002      Elephant      80      70      IT      Female
9008      Snake      40      65      Mech      Male
9006      Fox      98      70      EEE      Male
```

10.grep -E "T" student.txt

```
viswa@desktop:~/2022242001$ grep -E "T" student.txt
9021      Turtle      69      80      CS      Male
2001      Tiger      99      98      IT      Male
2002      Elephant      80      70      IT      Female
```

11.grep -E "ech|T" student.txt

```
viswa@desktop:~/2022242001$ grep -E "ech|T" student.txt
9021      Turtle      69      80      CS      Male
2001      Tiger      99      98      IT      Male
9001      Lion      68      97      Mech      Male
2002      Elephant      80      70      IT      Female
9008      Snake      40      65      Mech      Male
```

12.grep "\:" student.txt

```
viswa@desktop:~/2022242001$ grep "\:" student.txt
Register_no:      Name:      Mark-1: Mark-2: Dept:      Gender:
```

- **chown:**

Description: Chown changes the user and/or group ownership of each given file.

Syntax: ~\$ chown [owner] [filename]

- **chgrp:**

Description: Change the group of each FILE to GROUP. With --reference, change the group each FILE to that of RFILE.

Syntax: ~\$ chgrp

- **chmod:**

Description: Chmod - change file mode bits

Syntax: ~\$ chmod [mode] [filename]

Output:

```
viswa@desktop: ~/2022242001
viswa@desktop:~/2022242001$ chmod 777 animals.txt
viswa@desktop:~/2022242001$ ls -l
total 28
-rwxrwxrwx 1 viswa viswa  34 Sep  9 22:00 animals.txt
-rw-rw-r-- 1 viswa viswa   0 Sep 10 20:35 grass
-rw-rw-r-- 1 viswa viswa  30 Sep 10 20:26 numbers.txt
-rw-rw-r-- 1 viswa viswa 282 Sep 11 06:16 student.txt
-rw-rw-r-- 1 viswa viswa   5 Sep  9 22:13 temp
-rw-rw-r-- 1 viswa viswa  81 Sep 11 06:06 test
-rw-rw-r-- 1 viswa viswa  67 Sep 10 20:39 tree
drwxrwxr-x 2 viswa viswa 4096 Sep  9 22:10 viswa
viswa@desktop:~/2022242001$ chmod 111 animals.txt
viswa@desktop:~/2022242001$ ls -l
total 28
---x--x--x 1 viswa viswa  34 Sep  9 22:00 animals.txt
-rw-rw-r-- 1 viswa viswa   0 Sep 10 20:35 grass
-rw-rw-r-- 1 viswa viswa  30 Sep 10 20:26 numbers.txt
-rw-rw-r-- 1 viswa viswa 282 Sep 11 06:16 student.txt
-rw-rw-r-- 1 viswa viswa   5 Sep  9 22:13 temp
-rw-rw-r-- 1 viswa viswa  81 Sep 11 06:06 test
-rw-rw-r-- 1 viswa viswa  67 Sep 10 20:39 tree
drwxrwxr-x 2 viswa viswa 4096 Sep  9 22:10 viswa
viswa@desktop:~/2022242001$
```

```
2022batch@computer31: ~/2022242001/OS_Lab
2022batch@computer31:~/2022242001/OS_Lab$ chmod +x f1.txt
2022batch@computer31:~/2022242001/OS_Lab$ ls -l
total 56
-rw-r--r-- 1 2022batch domain users 169 Aug 12 17:20 alpha.txt
-rw-r--r-- 1 2022batch domain users   0 Aug 12 17:02 bro.txt
-rwxrwxrwx 1 2022batch domain users  12 Aug 12 15:29 f1
-rwxr-xr-x 1 2022batch domain users  24 Aug 21 13:50 f1.txt
-rw-r--r-- 1 2022batch domain users  24 Aug 21 13:50 f3.txt
-rw-r--r-- 1 2022batch domain users  47 Aug 21 14:03 fruits
-rw-r--r-- 1 2022batch domain users   0 Aug 28 14:32 haran
-rw-r--r-- 1 2022batch domain users 592 Aug 21 13:57 list
-rw-r--r-- 1 2022batch domain users  21 Aug 21 14:02 numbers
-rw-r--r-- 1 2022batch domain users  40 Aug 28 13:31 rajesh.txt
-rw-r--r-- 1 2022batch domain users  71 Aug 12 16:18 rohini
-rw-r--r-- 1 2022batch domain users  22 Aug 28 15:23 sample
-rw-r--r-- 1 2022batch domain users  30 Aug 28 15:26 sample.txt
-rw-r--r-- 1 2022batch domain users 381 Aug 21 14:17 table.txt
-rw-r--r-- 1 2022batch domain users  11 Aug  7 15:33 v
-rw-r--r-- 1 2022batch domain users  11 Aug  7 15:34 va
2022batch@computer31:~/2022242001/OS_Lab$
```

Result:

Thus, the Filter Commands and File permission Commands were learned.