

OUTPUTS

1) \$ date [Enter]

Fri May 19 13:31:24 IST 2023

2) \$ cal 5 2023

Su	Mo	Tu	We	Th	Fr	Sa
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31			

3) \$ ls text1.dir

doc1.txt filelist

\$ ls -l filelist

-rw-r--xrw- 1 student users 5609 Apr 23 09:30 filelist

\$ ls > list

// Saves in 'list' file

4) \$ man ls

// explains about ls command

5) \$ mv file.txt f1.txt

// moves file.txt to f1.txt

6) \$ mkdir test1

// creates subdirectory in home directory student

7) \$ cd Create

8) ~ \$ cd ..

9) \$ uname !

10) \$ rmdir test1

BASIC UNIX COMMANDS

1. date - displays date and time used on the Internet
2. cal - cal command is used to see the calendar of a specific month or year
3. ls - ls command is used to list all files and folders a folder or directory contains.
Options
 - l : displays some attributes as well
 - > : used to save some information
4. man - You can learn about options & arguments to be used in any Linux command
5. mv - used to move or rename files
6. mkdir - used to create subdirectory in working directory
7. cd - used to change the current working directory
8. cd.. - To go back to the home directory
9. uname - to get basic information about operating system
10. rmdir : Remove a directory

OUTPUTS

11) \$ tty

/dev/pts/10

12) \$ cat file.txt

abc

efg

13) \$ ls -l

-rwxrwxrwx 1 student users 5609 Apr 23 18:23:05 IST 2023

\$ ls -lu

-rwxrwxrwx 1 student users 5609 May 18 23:11 05 filelist

\$ ls -i

14) \$ cat -v filelist

\$ cat -n file.txt

2

\$ cat > foo

abcde

[Ctrl + D]

15) \$ cp file1 file2

\$ cp -i chap01/unit1 file1 student@bsen-02 ~/file1

cp: overwrite unit1 (yes/no)? y

\$ cp -R progs newprogs

DATE

EXPTITLE:

EXP. NO.

PAGE NO.

11. `tty` - Helps to know the working terminal/device name

12. `cat` - display file contents

13. `ls` - displays files and folders in a directory

Options

-x : Multicolumnar output

-F : Marks executable files with *

-a : Shows filenames with . and .. beginning

-R : Recursive list

-r : Sorts in reverse order

-l : Shows different attributes

-d dirname : Lists only dirname if dirname is directory

-t : Sorts filenames by last modification time

-u : Sorts filenames by last access time

-i : Displays inode number

14. `cat` - display file contents on the terminal

Options

-v : Display nonprinting characters

-n : Numbering lines

> : used to create file

15. `cp` - It copies a file or a group of files.

Options

-i : Interactive copying which asks user if copy is needed

-R : used to copy directory structures

OUTPUTS

- 16) \$ rm chap01
\$ rm -i chap02
rm: remove chap02 (yes/no)? ?n
- 17) \$ man chap01
-- More -- (18%)
- 18) \$ lp chap01
\$ lp -d laser chap01
\$ lp -t "Chapter" chap01
\$ lp -n3 -m chap01
- 19) \$ file archive.zip
archive.zip : ZIP ARCHIVE
- 20) \$ cat infile
I am the wc command
I count things
I make selective counts
\$ wc infile
3 20 103 infile
\$ wc -w infile
20 infile
- 21) \$ od -b odfile
00 00000 127 150 151 164 145 040...
- 22) \$ cmp chap01 chap02
chap01, chap02 differ: char 9, line 1

DATE
EXP. NO.

EXPT TITLE:

PAGE NO.

16. `rm` - `rm` command deletes one or more files

Options

- i : Interactive deletion which asks user to delete files
- r or -R : Recursive deletion
- f : Forceful removal

17. `more` - Shows contents of a long file one page at a time

18. `lp` - used to print a file line by line

Options

- d : to specify another printer other than default printer
- t : prints title on first page
- m : mail option
- n : print multiple copies
- cancel : cancel a job

19. `file` - determine the type of a file

20. `wc` - displays a word count of the data in the file
use `cat` before `wc`

Options

- l : displays only line count
- w : displays only word count
- c : displays only character count

21. `od` - displays data in octal

22. `cmp` - compares two files byte by byte

OUTPUTS

23) \$ comm file[12]

anil agarwal

barun sengupta

c.k.shukla

chanchal singhvi

lalit chowdury

s.n.das

sumit chakraborty

24) \$ diff file1 file2

0a,2

> anil agarwal

Append after line0 of first file

2c4

< chanchal singhvi

Changing line 2 of first file

> lalit chowdury

Replacing this line

this line

25) \$ ls -ld helpdir progs

drwxr-xr-x 2 kumar metal 512 May 9 10:31 helpdir

26) \$ chmod 666 xstart ; ls -l xstart
-rw-rw-rw- 1 kumar metal 1906 May 10 10:30 xstart
\$ chmod -R 755

// works on hidden files

27) \$ su

Password : *****

- // other shell

ls -l note

-rwxr--x 1 kumar metal 347 May 10 20:30 note

chown sharma note ; ls -l note

-rwxr---x 1 sharma metal 347 May 10 20:30 note

28) \$ chgrp dba dept.1st

-rw-r--r-- 1 kumar dba 139 Jun 8 16:43 dept.1st

- 23) comm : Checks for common entries in given files
- 24) diff : diff command converts one file to another
- 25) -d : lists directory attributes along with ls option
- 26) chmod : Change mode command (chmod) is used to set the permissions of one or more files for all three category of users (user, group and others)
-R : recursive chmod
- 27) chown : Transfers ownership of a file to a user rather than server
We need to use su for change in permission
- 28) chgrp : Change group (chgrp) command changes a file's group owner

OUTPUTS

29) ~ \$ vi filelist

1 This is USP Lab

2 Today is Tuesday

command mode

i

1 This is USP Lab

2 Today is Tuesday

3 We learn commands

Insert mode

Esc key

command mode

:q //quit

30) \$ ls chap*

chap chap01 chap02 chapx chapy chapz chap15

\$ ls chap?

chapx chapy chapz

\$ ls emp*lst

emp.lst empl.lst emp22lst empn.lst

\$ ls chap0[124]

chap01 chap02 chap04

29) The vi Editor

Uses a number of internal commands to navigate to any point in a text file and edit the text there.
Options

1. vi sometext : enters command mode
2. i : insert mode
3. esc key : jump from insert to command mode
4. :w : saving a file
5. :x : save and exit
6. :q : exit
7. :q! : quit without saving

To enter without file name, follow steps

- ~\$vi
- i
- Enter contents
- Esc key
- :w file8.txt
- :q // quits vi editor

30) Metacharacters : Wild cards used for matching patterns
Options

- * : any number of characters except none/including none
- ? : single character

[ijk] : A single character - either i, j, k

[!ijk] : A single character that is not i, j, k

[x-z] : A single character within ASCII range of x to z

[!x-z] : A single character except ASCII of x to z

?pat1, pat2..? : pat1, pat2 etc

OUTPUTS

- 31) \$ cmp fool foo2 > /dev/null
\$ cat /dev/null
\$ -
- 32) \$ who > /dev/tty
/dev/pts/1
- 33) \$ who | tee user.txt
romeo pts/2 Sep 7 08:41 (pc123.heavens.com)
juliet pts/3 Sep 7 17:58 (pc122.heavens.com)
\$ who | cmp user.txt -
\$ - // Standard input is second file
- 34) \$ ln emp.1st employee
\$ ls -li emp.1st employee
29518 -rwxr-xr-x 2 kumar metal 915 May 4 09:58 emp.1st
29518 -rwxr-xr-x 2 kumar metal 915 May 4 09:58 employee
\$ ln -s note note.sym
\$ ls -li note note.sym
9948 -rw-r--r-- 1 kumar group 80 Feb 16 14:52 note
9952 lrwxrwxrwx 1 kumar group 64 Feb 16 15:07 note.sym->note
- 35) \$ pwd
/home/student/test1
- 36) \$ ls -lc emp.1st
-rw-r--r-- 1 kumar metal 811 Mar 16 13:49 emp.1st
\$ ls -lu emp.1st
-rw-r--r-- 1 kumar metal 870 Mar 16 14:30 emp.1st

DATE

EXPT TITLE:

EXP. NO.

PAGE NO.

- 31) /dev/null : To check if a particular file is executed without seeing its output in the terminal
- 32) /dev/tty : Used to access own terminal or other terminal along with size.
- 33) tee : It handles a character stream by duplicating its input. It saves one copy in a file and writes the other to standard output. It can be used along with a pipeline.
- 34) ln : File is linked with ln command, which takes files as arguments. It creates both hard and soft link.

Option

-i : -i to ls option shows that they have the same inode number

-s : helps to create a symbolic link or soft link.

- 35) pwd : indicates present working directory

- 36) Modification and Access Times

ls -l : Time of last file modification

ls -lu : Time of last access

ls -lc : Time of last inode modification

ls -lt : Displays listing in order of modification time

ls -lut : Displays listing in order of access time

ls -la : including hidden files

ls -la.* : only hidden files.

37) **touch**: Changes modification and access times to predefined values

Options

- m : changes only modification time
- a : changes only access time

38) **find**: Recursively examines a directory tree to look for files matching some criteria

Syntax - find path list selection_criteria action

39) **umask**: Used to check permissions assigned to files and directories

- .rw-rw-rw- (octal 666) for regular files
- .rwxrwxrwx (octal 777) for directories

40) **pr**: Used for paginating files and prepares a file for printing by adding suitable headers, footers and formatted text

Options

- k : k → integer, prints in k columns
- t : suppress header and footer
- d : Doublespaces input, reduces clutter
- n : Numbers lines
- o n : Offsets lines by n spaces, increases left margin
- l : set page length

OUTPUTS

- 37) \$ touch 05171715 emp.lst ; ls -l emp.lst
-rw-r--r-- 1 kumar metal 870 May 17 17:15 emp.lst
\$ touch -a 01261650 emp.lst ; ls -l emp.lst
-rw-r--r-- 1 kumar metal 870 Jan 26 16:50 emp.lst
\$ touch -m 02281030 emp.lst ; ls -l emp.lst
-rw-r--r-- 1 kumar metal 870 Feb 28 10:30 emp.lst
- 38) \$ find / -name a.out -print
/home/kumar/scripts/a.out
/home/tiwary/scripts/reports/a.out
- 39) \$ umask
022
- 40) \$ pr dept.lst , May 06 10:38 1997 dept.lst Page 1
01: accounts : 6213
06 : sales : 1006
.. blank lines ..
\$ a.out | pr -t -5

0	4	8	12	16
1	5	8	11	14
2	6	9	12	15
3	7	10	13	16

\$ pr -t -n -d 0 10 dept.lst
| 01:accounts: 6213
| 06 : sales: 1006
\$ pr -l 54 chap01

OUTPUTS

- 41) head emp.1st // shows first ten lines
\$ head -n 3 emp.1st
2233|a.k.shukla|g.m|sales|12/12/52|6000
9876|jai sharma|director|production|12/08/50|7000
5678|sumit|d.g.m|marketing|19/04/43|6000
- 42) \$ tail -n 1 emp.1st
0110|v.k.agrawal|g.m|marketing|31/12/40|9000
\$ tail -f emp.1st
// keeps on writing & displaying
\$ tail -c +512 foo
// copies everything after skipping 511 bytes
- 43) \$ cut -c 6-22, 24-32 shortlist
a.k.shukla | g.m
jai sharma | director
n.k.gupta | chairman
\$ cut -d \| -f 2,3 shortlist
a.k.shukla | g.m.
jai sharma | director
n.k.gupta | chairman
- 44) \$ paste -d "\t" cutlist1 cutlist2
a.k.shukla | g.m. | 2233 | sales | 12/12/52 | 6000
n.k.gupta | chairman | 5423 | admin | 30/08/58 | 5400
- 45) \$ sort -n numfile
2
4
10
27
\$ sort -t "\t" -k 5,7,5.8 shortlist
5678|sumit|dgm|19/04/43|6000
2365|barun|director|11/05/47|7800

41) **heads** : displays the top of the file. When used without option, 10 lines are printed
-n option can be used to indicate number of lines

42) **tail** : displays the end of the file and displays 10 by default

Options

-n : lines to be printed from end

-f : write to files being accessed currently

-c : extract bytes rather than lines.

43) **cut** : Extract columns and fields

Options

-c : To extract a set of columns separated by comma.

-f : cut & extract fields list

-d : extract field delimiter.

44) **paste** : Files cut can be pasted back using paste command but vertically

-d : display delimiter

45) **sort** : Sorts file contents in ascending/descending order

Options

-k : Sort based on a primary key

-r : Reverse order of sorting

-k 5.7 : sort based on columns

-n : numeric sort on numerals

-u : Remove repeated lines

-c : to check if file is sorted

OUTPUTS

46) \$ cut -d "1" -f 3 emp. lst | sort | uniq -u
chairman
\$ cut -d "1" -f 3 emp. lst | sort | uniq -u
d.g.m
director
manager
\$ cut -d "1" -f 3 emp. lst | sort | uniq -u
2 d.g.m
4 director
2 manager
1 chairman

47) \$ tr -s ' ' < emp. lst | head -n 3
2233 | a.k. shukla | g.m. | sales | 12/12/52 | 6000
5678 | sumit | d.g.m | marketing | 19/04/53 | 6000
\$ tr -d '1' < emp. lst | head -n 3
2233 a.k. shukla g.m. sales 1212526000
5678 sumit d.g.m marketing 1904536000

48) \$ grep president emp. lst
\$ -f 3 // Not found
\$ grep -i 'agarwal' emp. lst
3564 | sudhir agarwal | executive | personnel | 06/07/47 | 7500
\$ grep -v 'director' ; \$ wc -l emp. lst
11 emp. lst
\$ grep -n 'marketing' emp. lst
3 : 5678 | sumit | d.g.m | marketing | 19/04/43 | 6000
\$ grep -c 'director' emp. lst
4
\$ grep -l 'manager' *. lst
emp. lst

DATE
EXP. NO.

EXPT TITLE:

PAGE NO.

46) **uniq**: Removes repeated or nonrepeated lines. Often combined with sort

Options

- u : Selects non-repeating lines
- d : Selects the duplicate lines
- c : Counting Frequency of Occurrence

47) **tr**: Takes input from standard input and manipulates individual characters in a file. It translates character using one or two compact expressions.

- d : delete characters
- s : compressing multiple consecutive characters
- c : complement values of expression

48) **grep**: Searches a file or standard input for a pattern
Can be used with set of metacharacters.

Options

- i : Ignoring case
- v : deleting lines
- n : display line numbers
- c : counting lines containing patterns
- l : display filenames
- e : matching multiple patterns
- f : Taking patterns from a file
- x : matches pattern with entire line

OUTPUTS

49) \$ grep "[aA]g[ar][ra]wal" emp.1st
3564 | sudhir agarwal | executive | 07/06/47 | 7500
0110 | v.k. agrawal | g.m. | 12/31/40 | 9000

\$ grep "[aA]gg*[ar][ra]wal" emp.1st
2476 | anil agarwal | manager | 05/01/59 | 5000
0110 | v.k. agrawal | g.m. | 12/31/40 | 9000

\$ grep "T... \$" emp.1st

9876 | jai sharma | director | 03/12/50 | 7000

2365 | barun sengupta | director | 05/11/47 | 7800

3564 | sudhir Agarwal | executive | 07/06/47 | 7500

50) \$ grep -E "[aA]gg?arwal" emp.1st

2476 | anil agarwal | manager | 01/05/59 | 5000

3564 | sudhir agarwall | executive | 06/07/57 | 7500

\$ grep -E "(sen|das) gupta" emp.1st

2365 | barun sengupta | director | 11/05/47 | 7800

1265 | s.n. dasgupta | manager | 12/09/63 | 5600

DATE

EXPT TITLE:

EXP. NO.

PAGE NO.

49) Basic Regular Expressions

To match with metacharacters

Options

* : zero or more occurrences

g* : nothing or g, gg, ggg etc

. : Single character

.* : no. of characters

[pqr] : Single character p, q, r

[^pqr] : except p, q, r

^ (caret) : For matching beginning of line

\$ - matching at end of line

50) Extended regular expressions

Extended regular expressions make it possible to match dissimilar patterns with single expression using -E option

Options

+ : matches one or more occurrence of previous character

? : matches zero or more occurrence of previous character

| : using delimiters