

CO1

1) command to display the following msg
such "god! bless us..."

2) Get the manual page of ls command
search for the word 'alphabetical'. find the
next occurrence and then find the previous
occurrence.

ls -l | grep -o 'ls [^]*' | head -n 2

ls -l | grep -o 'ls [^]*' | tail -n 2

ls -l | grep -o 'ls [^]*' | sed -n 2p

③

3) Read your name from the keyword and display it.

```
$ echo "what's your name?";  
read name;  
echo "my name is $name";
```

④ create the directory. Structure dia1/dia4 and dia1/dia2/dia3 with a single command and then change directory to dia3.

\$ ls

\$ mkdir -p dia1/dia4

\$ mkdir -p dia1/dia2/dia3

\$ cd dia1

~~\$~~ \$ ls

\$ cd dia2

\$ ls.

cd dia3.

5) Create some files using vim

↳ vim test.txt

1 Hello

2 world

3

Esc :wq

6) Display current directory?

pwd

7) List files and folders.

a) list the contents of dir1 and all its descendants.

b) list the contents of dir3 in

i) alphabetical order

ii) sorted on time of modification, newest first.

iii) sort on size

- iv) Reverse of all above
 - v) Long listing of files stored on size with smallest first and size.
 - vi) displayed in human readable form.
-

- a) ls -R
- b) i) ls -h
- ii) ls -~~ll~~ lt
- iii) ls -S
- iv) ls -a
- v) ls -l

8) Execute ~~com~~ds and store the output to a first lsoutput.

ls - lr > lsoutput .

cat lsoutput

9) Display the file .

a) starting with the first 10 lines and

b) starting with 10th line with provision for

scrolling up and down .

~~scr~~

a) head -10 filename.txt

b) tail -n +10 fit > ~~out2.txt~~
less out2.txt .

10) execute ls -l and add the output to lsoutput at the end.

ls -l >> lsoutput.txt

cat lsoutput.txt

11) execute ls -l and feed the result to less command to scroll through the directory listing.

ls -l /less

12) a) create a file file1 containing the word "Hello", using cat and off redirection.

b) create another file file2 containing the word "greeting!"

c) display the sentences,

Hello,

yourname

greeting!

10) execute ls -l and add the output to lsoutput at the end.

ls -l >> lsoutput.txt

cat lsoutput.txt

11) execute ls -l and feed the result to less command to scroll through the directory listing.

ls -l | less

12) a) create a file file1 containing the word "HELLO", using cat and off redirection.

b) create another file file2 containing the word "greeting!"

c) display the sentences,

HELLO,

yourname

greeting!

using cat by contrai concatenating file standard o/p and file 2 standard o/p may receive the contents from an echo.

a) cat > file1.txt

Hello

b) cat > file2.txt

Greetings

c) cat - file1 read - P "yourname" n
echo \$n >> file2.

(3) copy the file file1 to new file.

a) if newfile already exist, it should be replaced.

b) if newfile already exist, it should not be replaced

c) if newfile already exist, it should be replaced, but only with the consent of the user.

- d) if newfile already exists it should be replaced only if its content is older than that of file 1.
- e) even if newfile is read only
- f) create a link instead of copying.
- g)

- a) cp file1 newfile
- b) cp -n file2 newfile
- c) cp -i file1 newfile
- d) cp -u file1 newfile
- e)
- f) cp -s file3-fct file4-fct.
- g) cp -Rn dir1 dir2

- Q) create a new directory. dir6 inside dir1
- move all files in dir5 into c't
 - rename the file newfile.cn to oldfile.
 - move the file file1 ~~or~~ q12 to dir6 with name file 3.
 - delete all files whose the names starts with a vowel character, upper or lower case.
 - delete all files whose the names is atleast 3 characters long.
 - delete all hidden folders and files.
-

mkdir -p dir1 / dir6

- mv dir5 dir1 / dir6.
- mv ~~or~~ newfile.txt oldfile.txt.
- mv file1 dir6 / file 3.

- rm [a,e,i,o,u,A,E,I,O,U]*
- rm ?? ? *
- rm -rf *

- 15) create a file testfile1 using vim .
- set line number .
 - type your name and addresses with district and pincode .
—
 - copy paste the contents 10 times
—
 - Replace all occurrence of your designation with a neighbouring district .

16) a) The first 12 users in the system ?

cat /etc /passwd /head -12

b) The last 7 users in the system

cat /etc /passwd /tail -7

c) All but the first 3

cat /etc /passwd /head -3

d) All but last 5

cat /etc /passwd /head -5 -N

e) only the 9th

cat /etc /passwd /head -9 /tail -1

17) use grep to

- a) display all lines in a file that contains the string "abc".

grep abc testfile.

- b) display all lines in a file that does not contain string "abc"

grep -v abc testfile.

18) using expr .

- a) Read two integers x and y . display the sum, difference, product, quotient and remainder of the variables .

read -p "enter a number:" x

read -p "enter another no:" y

echo "sum is" expr \$x+\$y

echo "difference" expr \$x-\$y

echo "product" expr \$x * \$y

echo "quotient :" expr \$x / \$y

echo "remainder :" expr \$x % \$y