Working with a Vi Editor:

1: Create a file using vi. Enter the following text:

A network is a group of computers that can communicate with each other, share resources, and access remote hosts or other networks. Netware is a computer network operating system designed to connect, manage, and maintain a network and its services. Some of the network services are Netware Directory Services (NDS), file system, printing and security.

- [admin@sushil Desktop]$ vi network.txt

a. Change the word "Netware" in the second line to "Novell Netware".

- :%s/Netware/Novell

b. Insert the text "(such as hard disks and printers)" after "share resources" in the first line.

- /share resources
- a to append text and type "such as hard disks and printers"

c. Append the following text to the file:

"Managing NDS is a fundamental administrator role because NDS provides a single point for accessing and managing most network resources."

- Vi.network
- G and type text and save

Working shell

1. Type some text on the shell separated by space
   - [admin@sushil Desktop]$ Hello this is test sentence.

1: Move cursor one word back

- Alt + b

2: Move cursor one word forward

- Alt + f

3: Move cursor to the first character

- Ctrl + a

4: Move cursor to the end

- Ctrl + e

5: Delete test from second word to last character

- Ctrl + a, Alt + f, Ctrl + k

6: Delete the current line

- Ctrl + u

2: In lab 4 we have created a file errorlog.txt. Display it using cat command using command completion.

- [admin@sushil Desktop]$ cat errorlog.txt

3: Display history of command used so far.

- [admin@sushil Desktop]$ history

```
  1  ifconfig
  2  exit
  3  passwd root
  4  cd /root/
  5  exit
  6  hostname host01
  7  su - root
  8  yum update -y
  9  su root
 10  su -
 11  cd
 12  poweroff
 13  ifconfig
```

4: Search ls command in history file

- [admin@sushil Desktop]$ history | grep ls

```
 19  ls
 28  echio ls
 29  echo ls
 30  echo ls -l
253  ls -d /etc/*[ab]*
255  ls
```

257 ls
259 ls
265 history | grep ls

5: Repeat the last command rd

- [admin@sushil Desktop]$ cat errorlog.txt
- cat: data.txt: No such file or directory
- [admin@sushil Desktop]$!!
- cat errorlog.txt
- cat: data.txt: No such file or directory

6: Execute 3 command from history file.

267 touch myfile
268 rm mtfile
269 rm myfile
270 history

- [admin@sushil Desktop]$!267; !268; !269
  touch myfile; rm mtfile; rm myfile
  rm: cannot remove 'mtfile': No such file or directory

7: What are the different shells available.

- [admin@sushil Desktop]$ cat /etc/shells
  /bin/sh
  /bin/bash
  /usr/bin/sh
  /usr/bin/bash

Understanding access permissions

7.1: Create an empty file “demofile” and perform following instruction

- [admin@sushil Desktop]$ touch demofile
- [admin@sushil Desktop]$ ls
- command_substitution.sh errorlog.txt lsdoc newfriends users
- demofile friends network.txt test_python variables.sh

1. Revoke read permission from owner and use cat command.
   - [admin@sushil Desktop]$ chmod u-r demofile
   - [admin@sushil Desktop]$ cat demofile
   - cat: demofile: Permission denied

2. Revoke write permission from owner and open using vi editor and add some contain in it.

- [admin@sushil Desktop]$ chmod u-w demofile
- [admin@sushil Desktop]$ vi demofile
- Permission denied (to edit when opened)

3. Add read and write permission to owner.

- [admin@sushil Desktop]$ chmod u+rw demofile

4. Revoke write and execute from other and group

- [admin@sushil Desktop]$ chmod go-rw demofile

5. Add write permission to group only

- [admin@sushil Desktop]$ chmod g+w demofile

6. Assign read permission to all
   - [admin@sushil Desktop]$ chmod a+r demofile
7. Revoke read permission from others
   - [admin@sushil Desktop]$ chmod o-r demofile
8. Give the execute permission for the user for a file chap1
   - [admin@sushil Desktop]$ chmod u+x chap1
9. Give the execute permission for user, group and others for a file add.c
   - [admin@sushil Desktop]$ touch add.c
   - [admin@sushil Desktop]$ chmod a+x add.c

10. Remove the execute permission from user, give read permission to group and others for a file aa.c

- [admin@sushil Desktop]$ chmod u-x,g+r,o+r aa.c

11. Give execute permission for users for a.c, kk.c, nato and myfile using single command

- [admin@sushil Desktop]$ chmod u+x a.c kk.c nato myfile

7.2: Create an directory “demo” and copy /etc/passwd file in it

- [admin@sushil ~]$ sudo mkdir demo

  [sudo] password for admin:

  [admin@sushil ~]$ ls

  Demo

  [admin@sushil ~]$ sudo cp /etc/passwd demo/

1. Display contents of demo
   - [admin@sushil ~]$ ls demo
2. Revoke read permission from demo directory and use ls command on it
   - [admin@sushil ~]$ sudo chmod go-rx demo

   [admin@sushil ~]$ ls demo

   ls: cannot open directory 'demo': Permission denied

3. Revoke write permission from demo directory and try to copy /etc/profile file in it
   - [admin@sushil ~]$ sudo chmod u-w demo

   [admin@sushil ~]$ sudo cp /etc/profile demo/

4. Delete passwd file from demo directory

   →[admin@sushil ~]$ sudo rm demo/passwd

5. Revoke execute permission from demo directory and try cd command on demo.

   →[admin@sushil ~]$ sudo chmod u-x demo

   [admin@sushil ~]$ cd demo

   bash: cd: demo: Permission denied

Using Process-Related Commands

1. Find out the PID of the processes that are activated by you
   - [admin@sushil ~]$ ps -u $USER
   - PID TTY TIME CMD
   - 2049 ? 00:00:02 systemd
   - 2051 ? 00:00:00 (sd-pam)
   - 2067 ? 00:00:00 gnome-keyring-d
   - 2071 tty2 00:00:00 gdm-wayland-ses
   - 2075 ? 00:00:00 dbus-broker-lau
   - 2077 ? 00:00:03 dbus-broker
2. Find out the information about all the processes that are currently active

   - [admin@sushil ~]$ ps -ef
   - UID PID PPID C STIME TTY TIME CMD
   - root 1 0 0 Jan27 ? 00:00:12 /usr/lib/systemd/systemd r
   - root 2 0 0 Jan27 ? 00:00:00 [kthreadd]
   - root 3 2 0 Jan27 ? 00:00:00 [pool_workqueue_]
   - root 4 2 0 Jan27 ? 00:00:00 [kworker/R-rcu_g]

3. Start a different process in the background. Find out the status of the background process using the PID of the same.

   - [admin@sushil ~]$ sleep 100 &
   - [2] 83286
   - [admin@sushil ~]$ ps -p $!
   - PID TTY TIME CMD
   - 83286 pts/0 00:00:00 sleep

4. Run a job in background
   - [admin@sushil ~]$ sleep 100 &
   - [2] 83286

5. Bring a last background job in fore ground
   - [admin@sushil ~]$ fg

   bash: fg: job has terminated
   [1]+ Done sleep 100

6. Run 3 jobs in background and bring first job in foreground
   - [admin@sushil ~]$ sleep 100 &
   - [3] 83306
   - [2] Done
   - [admin@sushil ~]$ sleep 200 &

- [4] 83311
- [admin@sushil ~]$ sleep 300 &
- [5] 83316
- [admin@sushil ~]$ fg %1
- sleep 100

7. Stop current job
   - [admin@sushil ~]$ kill -SIGSTOP %1
   - [1]+ Stopped vim greet.sh (wd: ~/programs)
   - (wd now: ~)
   - [4] Done sleep 200

8. Start stopped job
   - [admin@sushil ~]$ jobs
   - [1]+ Stopped vim greet.sh (wd: ~/programs)
   - [2]- Stopped sleep 100

9. Run a job
   - [admin@sushil ~]$ sleep 100
   - [2] 83286
10. Kill last job
    - [admin@sushil ~]$ kill %1

    [1]+ Stopped vim greet.sh (wd: ~/programs)
    (wd now: ~)
11. Kill your shell using process id
    - [admin@sushil ~]$ kill -9 $$
12. Execute a ls command by setting priority as -10 using nice command
    - [admin@sushil ~]$ sudo nice -n -10 ls
13. Display a date on every hour using cron tab
    - [admin@sushil ~]$ crontab -e
      crontab: installing new crontab
      [admin@sushil ~]$ crontab -l
      0 * * * * date >> /home/hourly_date.log