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@hostname01 ~]$ vi filename.txt

a. Change the word “Netware” in the second line to “Novell Netware”.

- A network is a group of computers that can communicate with each other, share resources, and access remote hosts or other networks. Novell 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.

~
~
~
~
~
~
:% s/Netware/Novell Netware 1,1
All

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

- A network is a group of computers that can communicate with each other, share resources(such as hard disks and printers),and access remote hosts or other networks. Novell 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.

~

~

~

-- INSERT --

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."

- A network is a group of computers that can communicate with each other, share resources(such as hard disks and printers),and access remote hosts or other networks. Novell 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. "Managing NDS is a fundamental administrator role because NDS provides a single point for accessing and managing most network resources."

~

~

~

-- INSERT -- 1,1 All

Working shell

1. Type some text on the shell separated by space
   - This is an example of text with spaces.

~

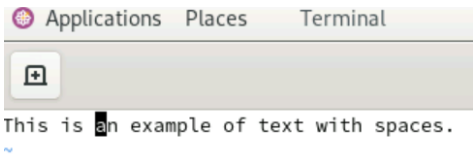
~

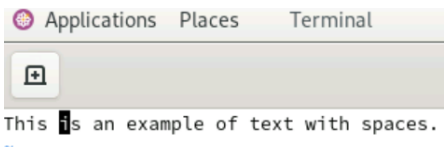
~

-- INSERT -- 1,1 All

1: Move cursor one word back

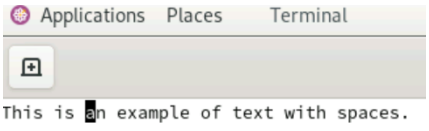
- Press alt b

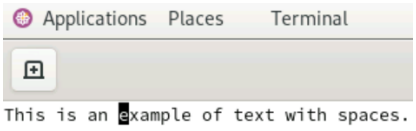




2: Move cursor one word forward

- Alt f





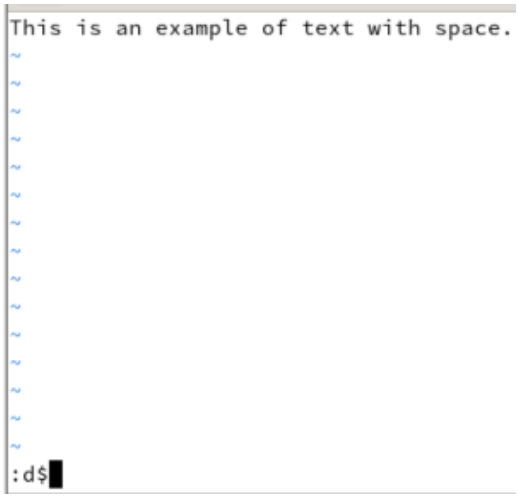
3: Move cursor to the first character

- Ctrl a

4: Move cursor to the end

- Ctrl e

5: Delete test from second word to last character



This

~

~

~

6: Delete the current line

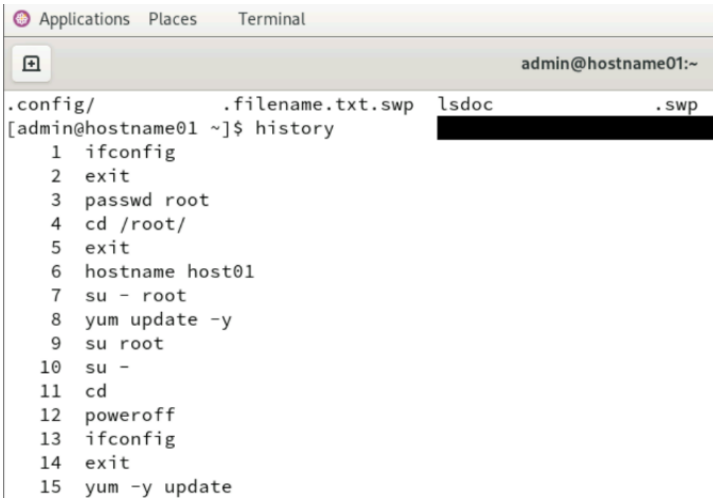
- Press dd

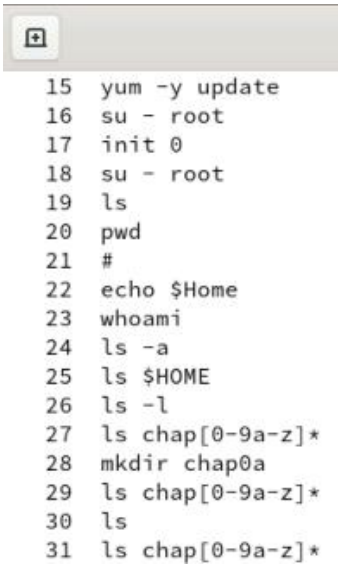
2: In lab 4 we have created a file errorlog.txt. Display it using cat command using

command completion.

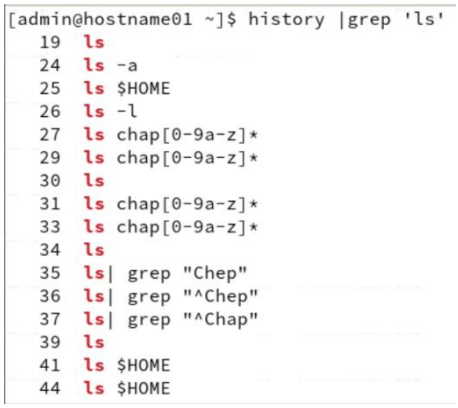
- Enter cat error and <tab>
  [admin@hostname01 ~]$ cat errorlog.txt

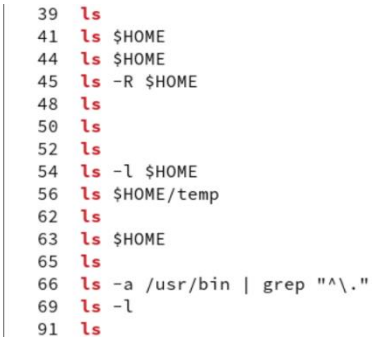
3: Display history of command used so far.

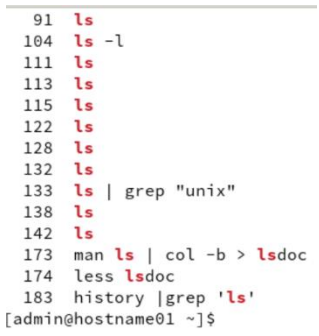




4: Search ls command in history file







5: Repeat the last command rd

- [admin@hostname01 ~]$!!

history |grep 'ls'

```
[admin@hostname01 ~]$ !!
history |grep 'ls'
   19  ls
   24  ls -a
   25  ls $HOME
   26  ls -l
   27  ls chap[0-9a-z]*
   29  ls chap[0-9a-z]*
   30  ls
   31  ls chap[0-9a-z]*
   33  ls chap[0-9a-z]*
   34  ls
   35  ls| grep "Chep"
   36  ls| grep "^Chep"
```

6: Execute 3 command from history file.

```
[admin@hostname01 ~]$ !3
passwd root
passwd: Only root can specify a user name.
```

7: What are the different shells available.

1. Bash shell (Bourne Again Shell)
2. Zsh (Z Shell)
3. Csh (C Shell)
4. Ksh (Korn Shell)
5. Fish (Friendly Interactive Shell)

Understanding access permissions

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

1. Revoke read permission from owner and use cat command.

```
[admin@hostname01 ~]$ touch demofile
[admin@hostname01 ~]$ chmod -r demofile
[admin@hostname01 ~]$ cat demofile
cat: demofile: Permission denied
```

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

```
[admin@hostname01 ~]$ chmod -w demofile
[admin@hostname01 ~]$ vi demofile
```

```
~
~
~
"demofile" [Permission Denied]
```

3. Add read and write permission to owner.

```
[admin@hostname01 ~]$ chmod u+rw demofile
[admin@hostname01 ~]$ cat demofile
[admin@hostname01 ~]$
```

4. Revoke write and execute from other and group
    - [admin@hostname01 ~]$ chmod og-wr demofile
5. Add write permission to group only
    - [admin@hostname01 ~]$ chmod g+w demofile
6. Assign read permission to all
    - [admin@hostname01 ~]$ chmod a+r demofile
7. Revoke read permission from others
    - [admin@hostname01 ~]$ chmod o-r demofile
8. Give the execute permission for the user for a file chap1
    - [admin@hostname01 ~]$ touch chap1
    - [admin@hostname01 ~]$ chmod u+x chap1
9. Give the execute permission for user, group and others for a file add.c
    - [admin@hostname01 ~]$ touch add.c
    - [admin@hostname01 ~]$ 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@hostname01 ~]$ touch aa.c
    - [admin@hostname01 ~]$ chmod u-x aa.c
    - [admin@hostname01 ~]$ chmod go+r aa.c

```
[admin@hostname01 ~]$ chmod og-wr demofile
[admin@hostname01 ~]$ chmod g+w demofile
[admin@hostname01 ~]$ chmod a+r demofile
[admin@hostname01 ~]$ chmod o-r demofile
[admin@hostname01 ~]$ chmod u+x chap1
chmod: cannot access 'chap1': No such file or directory
[admin@hostname01 ~]$ touch chap1
[admin@hostname01 ~]$ chmod u+x chap1
[admin@hostname01 ~]$ chmod a+x add.c
chmod: cannot access 'add.c': No such file or directory
[admin@hostname01 ~]$ touch add.c
[admin@hostname01 ~]$ chmod a+x add.c
[admin@hostname01 ~]$ touch aa.c
[admin@hostname01 ~]$ chmod u-x aa.c
[admin@hostname01 ~]$ chmod go+r aa.c
```

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

```
[admin@hostname01 ~]$ touch a.c kk.c nato myfile
[admin@hostname01 ~]$ chmod u+x a.c kk.c nato myfile
[admin@hostname01 ~]$
```

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

[admin@hostname01 ~]$ mkdir demo

1. Display contents of demo
   [admin@hostname01 ~]$ cp /etc/passwd demo/
   [admin@hostname01 ~]$ ls demo
   Passwd

```
[admin@hostname01 ~]$ cp /etc/passwd demo/
[admin@hostname01 ~]$ ls demo
passwd
```

2. Revoke read permission from demo directory and use ls command on it

```
[admin@hostname01 ~]$ chmod -r demo
[admin@hostname01 ~]$ ls
aa.c    add.c   chap1     Desktop    errorlog.txt  friends  lsdoc   myfile}  new_dir    Public     users
{a.c,   chap0a  demo      Documents  filename.txt  kk.c     Music   nato     newfriend  sec.unix   Videos
a.c     Chap0a  demofile  Downloads  first.unix    kk.c,    myfile  nato,    Pictures   Templates
[admin@hostname01 ~]$
```

3. Revoke write permission from demo directory and try to copy /etc/profile file in it
   - [admin@hostname01 ~]$ chmod -w demo

```
[admin@hostname01 ~]$ chmod -w demo
[admin@hostname01 ~]$ cp /etc/profile demo
cp: cannot stat 'demo/profile': Permission denied
[admin@hostname01 ~]$
```

4. Delete passwd file from demo directory
   - [admin@hostname01 ~]$ rm demo/passwd
     rm: cannot remove 'demo/passwd': Permission denied

```
[admin@hostname01 ~]$ rm demo/passwd
rm: cannot remove 'demo/passwd': Permission denied
```

5. Revoke execute permission from demo directory and try cd command on demo.
    - [admin@hostname01 ~]$ chmod -x demo

```
[admin@hostname01 ~]$ chmod -x demo
[admin@hostname01 ~]$ cd
[admin@hostname01 ~]$ cd demo
bash: cd: demo: Permission denied
[admin@hostname01 ~]$
```

Using Process-Related Commands

1. Find out the PID of the processes that are activated by you
    - [admin@hostname01 ~]$ ps -u 'admin'

```
[admin@hostname01 ~]$ ps -u 'admin'
    PID TTY          TIME CMD
   2257 ?        00:00:01 systemd
   2259 ?        00:00:00 (sd-pam)
   2276 ?        00:00:00 gnome-keyring-d
   2280 tty2     00:00:00 gdm-wayland-ses
   2283 ?        00:00:00 dbus-broker-lau
   2285 ?        00:00:02 dbus-broker
   2289 tty2     00:00:00 gnome-session-b
   2325 ?        00:00:00 gnome-session-c
   2327 ?        00:00:00 gnome-session-b
   2345 ?        00:08:59 gnome-shell
   2361 ?        00:00:00 gvfsd
   2366 ?        00:00:00 gvfsd-fuse
   2373 ?        00:00:00 at-spi-bus-laun
   2379 ?        00:00:00 dbus-broker-lau
   2380 ?        00:00:00 dbus-broker
```

```
   2380 ?        00:00:00 dbus-broker
   2393 ?        00:00:00 gnome-shell-cal
   2394 ?        00:00:00 xdg-permission-
   2412 ?        00:00:02 pipewire
   2413 ?        00:00:03 wireplumber
   2419 ?        00:00:02 pipewire-pulse
   2421 ?        00:00:00 evolution-sourc
   2433 ?        00:00:00 dconf-service
   2445 ?        00:00:00 gvfs-udisks2-vo
   2456 ?        00:00:00 goa-daemon
   2459 ?        00:00:00 gvfs-mtp-volume
   2467 ?        00:00:00 gvfs-gphoto2-vo
   2470 ?        00:00:00 evolution-calen
   2477 ?        00:00:00 gvfs-goa-volume
   2484 ?        00:00:27 goa-identity-se
   2505 ?        00:00:00 evolution-addre
   2524 ?        00:00:00 gjs
```

2. Find out the information about all the processes that are currently active
    - [admin@hostname01 ~]$ ps -ef

```
[admin@hostname01 ~]$ ps -ef
UID          PID    PPID  C STIME TTY          TIME CMD
root           1       0  0 Jan09 ?        00:00:09 /usr/lib/systemd/systemd rhgb -
root           2       0  0 Jan09 ?        00:00:00 [kthreadd]
root           3       2  0 Jan09 ?        00:00:00 [pool_workqueue_]
root           4       2  0 Jan09 ?        00:00:00 [kworker/R-rcu_g]
root           5       2  0 Jan09 ?        00:00:00 [kworker/R-sync_]
root           6       2  0 Jan09 ?        00:00:00 [kworker/R-slub_]
root           7       2  0 Jan09 ?        00:00:00 [kworker/R-netns]
root          10       2  0 Jan09 ?        00:00:00 [kworker/u512:0-events_unbound]
root          11       2  0 Jan09 ?        00:00:00 [kworker/R-mm_pe]
root          12       2  0 Jan09 ?        00:00:00 [kworker/u512:1-netns]
root          13       2  0 Jan09 ?        00:00:00 [rcu_tasks_kthre]
root          14       2  0 Jan09 ?        00:00:00 [rcu_tasks_rude_]
```

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

```
[admin@hostname01 ~]$ ls &
[7] 17299
aa.c   add.c   chap1     Desktop    errorlog.txt  friends  lsdoc
{a.c,  chap0a  demo      Documents  filename.txt  kk.c     Music
a.c    Chap0a  demofile  Downloads  first.unix    kk.c,    myfile
[7]   Done                    ls --color=auto
[admin@hostname01 ~]$ grep &
[7] 17304
Usage: grep [OPTION]... PATTERNS [FILE]...
Try 'grep --help' for more information.
[7]   Exit 2                  grep --color=auto
[admin@hostname01 ~]$ man &
[7] 17310
What manual page do you want?
```

```
What manual page do you want?
For example, try 'man man'.

[7]+  Stopped                 man
[admin@hostname01 ~]$ ps -p 17304
    PID TTY          TIME CMD
[admin@hostname01 ~]$ ps -p 17299
    PID TTY          TIME CMD
[admin@hostname01 ~]$ ps -p 17
    PID TTY          TIME CMD
     17 ?        00:00:04 rcu_preempt
[admin@hostname01 ~]$ ps -p 17310
    PID TTY          TIME CMD
  17310 pts/0    00:00:00 man
[admin@hostname01 ~]$
```

3. Run a job in background

```
[admin@hostname01 ~]$ grep &
[8] 17355
Usage: grep [OPTION]... PATTERNS [FILE]...
Try 'grep --help' for more information.
[8]   Exit 2                  grep --color=auto
```

```
[admin@hostname01 ~]$ bg
[9]+ grep --color=auto "hello" &

[9]+  Stopped                 grep --color=auto "hello"
```

4. Bring a last background job in fore ground

```
[admin@hostname01 ~]$ fg
man
```

5. Run 3 jobs in background and bring first job in foreground

```
[admin@hostname01 ~]$ xlogo & gedit &
[10] 17512
[11] 17513
bash: xlogo: command not found...
[admin@hostname01 ~]$ jobs
[2]   Stopped                 vi filename.txt
[3]   Stopped                 vi filename.txt
[4]   Stopped                 vi file.txt
[5]   Stopped                 vi file.txt
[6]-  Stopped                 vi demofile
[9]+  Stopped                 grep --color=auto "hello"
[10]   Exit 127                xlogo
[11]   Done                    gedit
```

[admin@hostname01 ~]$ fg %1

6. Stop current job
   - Press ctrl +z

```
[admin@hostname01 ~]$ vi

[15]+  Stopped                 vi
```

7. Start stopped job

```
[admin@hostname01 ~]$ fg
vi
```

8. Run a job

   Just enter the command which we want to run.

9. Kill last job
   - [admin@hostname01 ~]$ kill %1
10. Kill your shell using process id
    - [admin@hostname01 ~]$ kill 12339

11. Execute a ls command by setting priority as -10 using nice command

- [admin@hostname01 ~]$ nice -n -10 ls

```
[admin@hostname01 ~]$ sudo nice -n -10 ls

We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:

    #1) Respect the privacy of others.
    #2) Think before you type.
    #3) With great power comes great responsibility.

[sudo] password for admin:
aa.c    add.c   chap1     Desktop    errorlog.txt  friends  lsdoc   myfile}  new_dir    Public     users
{a.c,   chap0a  demo      Documents  filename.txt  kk.c     Music   nato     newfriend  sec.unix   Videos
a.c     Chap0a  demofile  Downloads  first.unix    kk.c,    myfile  nato,    Pictures   Templates
```

12. Display a date on every hour using cron tab

- [admin@hostname01 ~]$ crontab -e
  no crontab for admin - using an empty one
  crontab: installing new crontab
  [admin@hostname01 ~]$ cat /tmp/hourly_dates

```
0 * * * * date > /tmp/hourly_dates
~
~
~
~
~
~
~
~
~
~
~
~
"/tmp/crontab.q7s5dA" 2L, 36B
```