**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.

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

A computer screen with a black screen

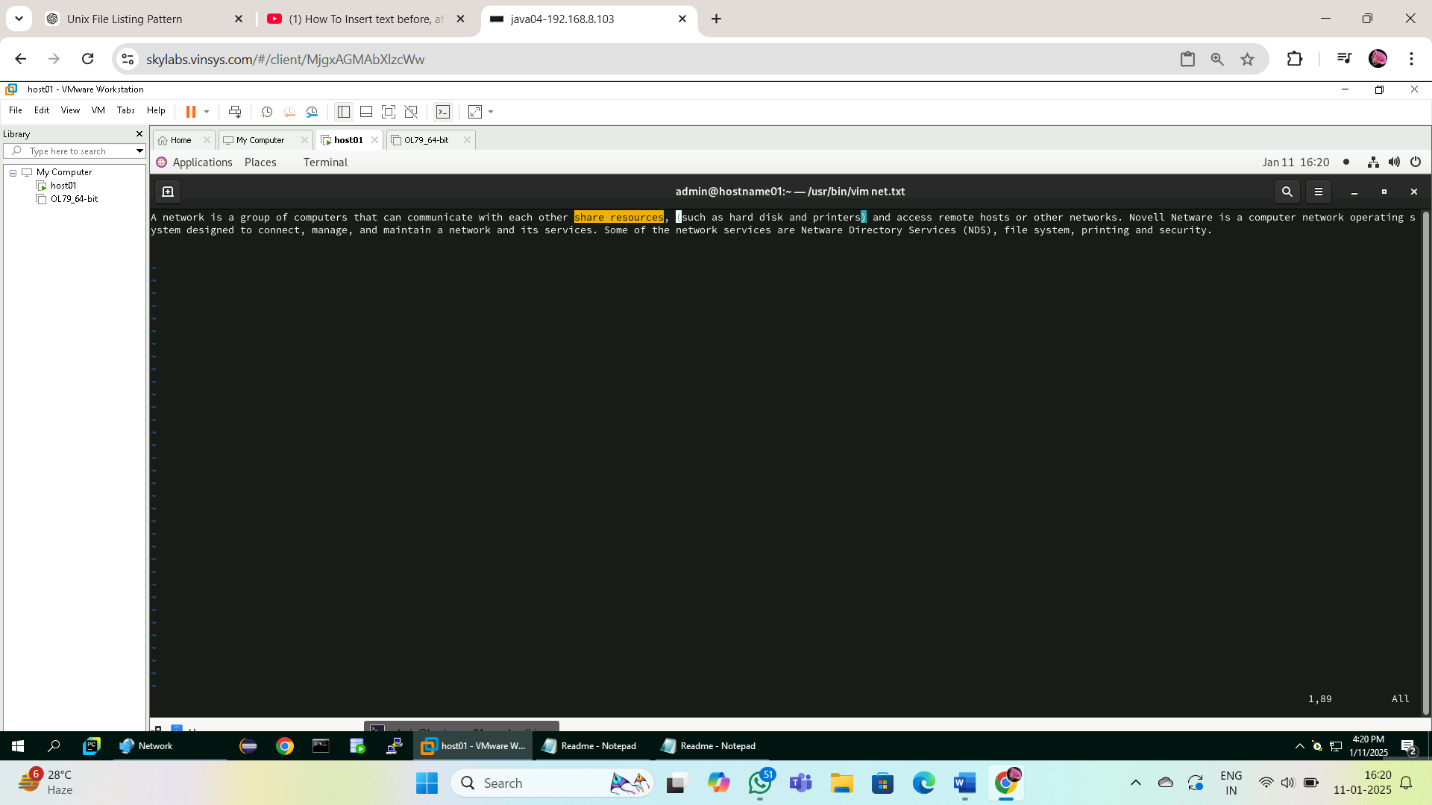
Description automatically generated

A computer screen shot of a black screen

Description automatically generated

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

first line.



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 computer screen shot of a black screen

Description automatically generated

Working shell

1. Type some text on the shell separated by space

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

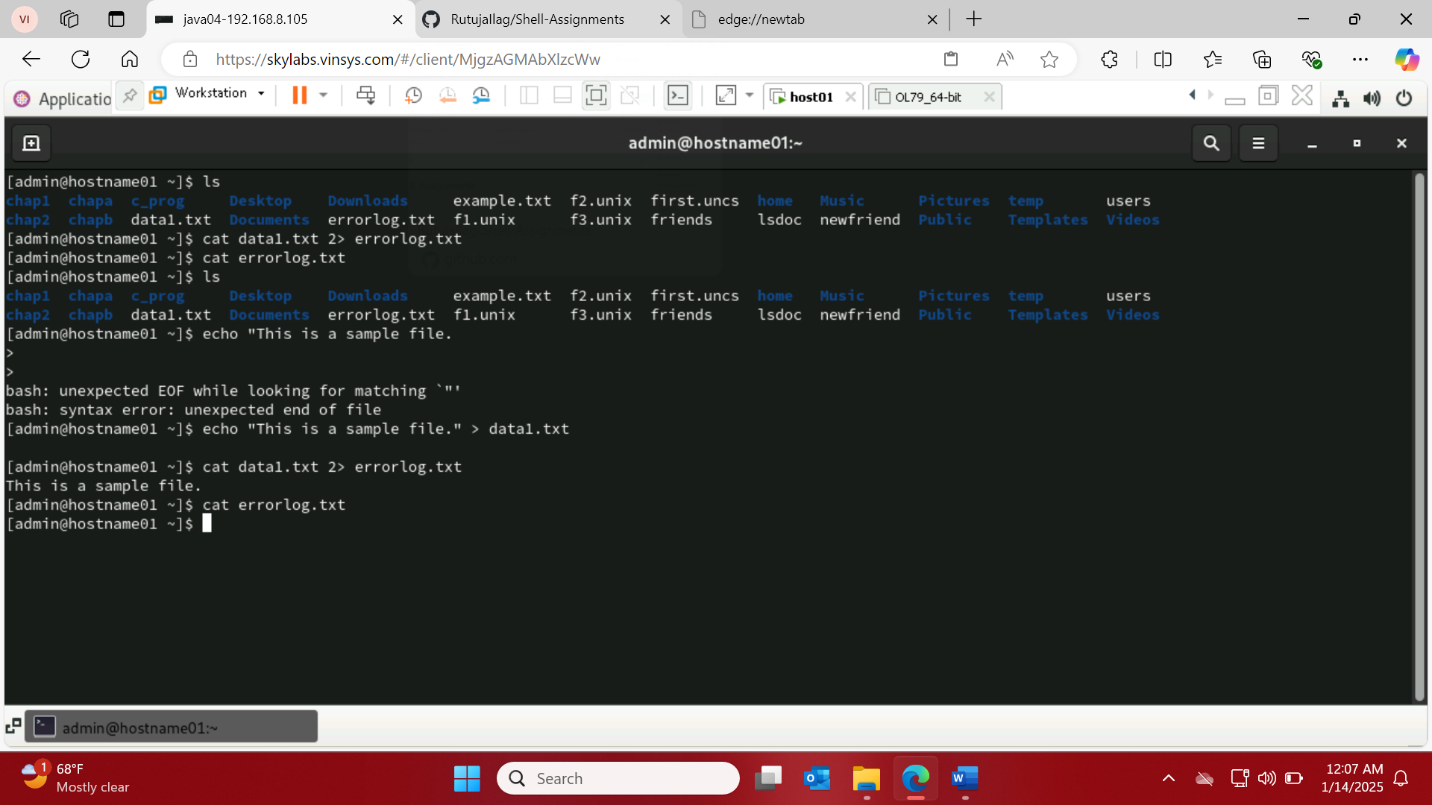
4: Move cursor to the end : Ctrl+E

5: Delete test from second word to last character: Alt+F then Ctrl+K

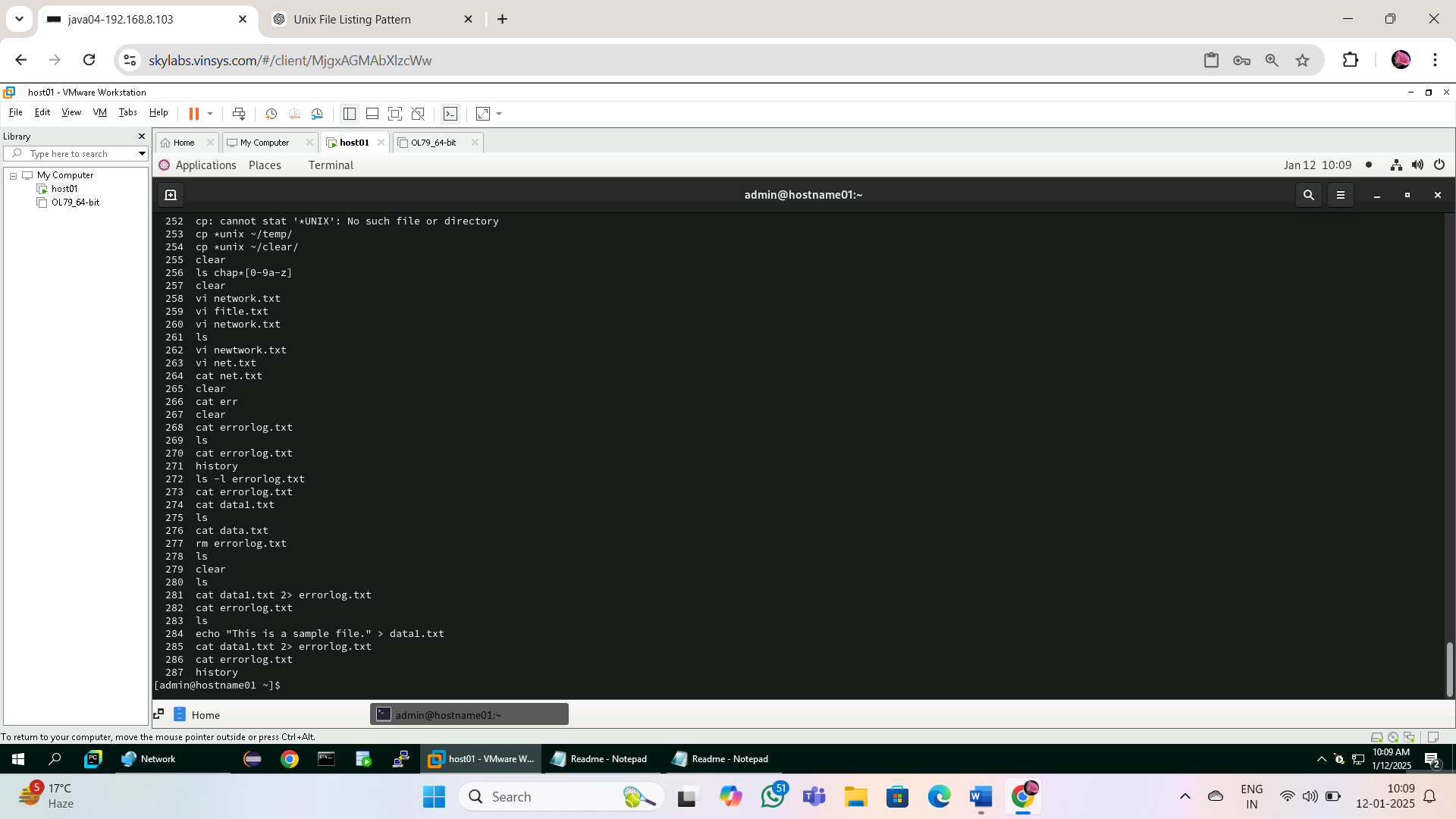
6: Delete the current line : Ctrl+U

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

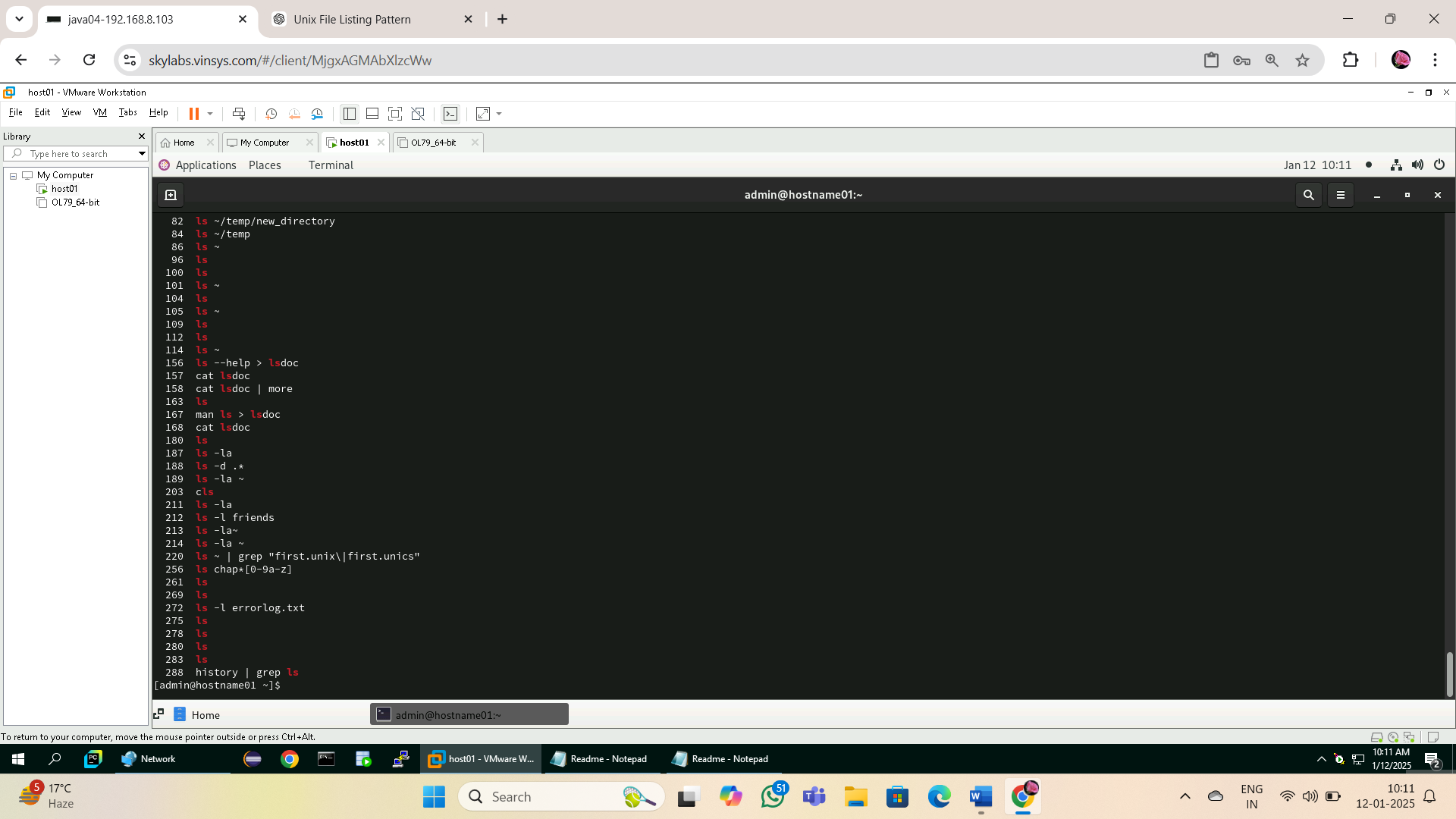
command completion.



3: Display history of command used so far.



4: Search ls command in history file



5: Repeat the last command rd

!!

6: Execute 3 command from history file.

A computer screen shot of a black screen

Description automatically generated

7: What are the different shells available.

cat /etc/shells

Understanding access permissions

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

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

chmod u-r demofile

2. Revoke write permission from owner and open using vi

editor and add some contain in it.

chmod u-w demofile

1. Add read and write permission to owner.

Chmod u+rw demofile

1. Revoke write and execute from other and group

ls -l demofile

chmod go-wx demofile

1. Add write permission to group only

chmod g+w demofile

1. Assign read permission to all

chmod a+r demofile

1. Revoke read permission from others

Chmod o-r demofile

1. Give the execute permission for the user for a file chap1

chmod u+x chap1

1. Give the execute permission for user, group and others for a file add.c

touch add.c

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 ~]$ chmod u-x aa.c**

**[admin@hostname01 ~]$ chmod og+r aa.c**

**[admin@hostname01 ~]$ ls -l aa.c**

**-rw-r--r--. 1 admin admin 0 Jan 12 11:03 aa.**

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

single command

**chmod u+x a.c kk.c nato myfile**

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

**[admin@hostname01 ~]$ mkdir demo**

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

1. Display contents of demo

**ls demo**

2. Revoke read permission from demo directory and use ls

command on it

**chmod -r demo**

3. Revoke write permission from demo directory and try to copy

/etc/profile file in it

**chmod -w demo**

**cp /etc/profile demo/**

4. Delete passwd file from demo directory

**rm demo/passwd**

5. Revoke execute permission from demo directory and try cd

command on demo. **chmod -x demo**

**cd demo**

**Using Process-Related Commands**

1. Find out the PID of the processes that are activated by you : ps -u $USER

2. Find out the information about all the processes that are currently active : ps aux

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

process using the PID of the same. : sleep 60s

jobs

jobs -l

ps -p 13077

4. Run a job in background

bg

5. Bring a last background job in fore ground

fg %1

6. Run 3 jobs in background and bring first job in foreground : jobs

fg %1

7. Stop current job : jobs

Sleep 20s

8. Start stopped job :fg %3

9. Run a job :nano myfile.txt

10. Kill last job : kill %%

11. Kill your shell using process id :ps -ef | grep sleep

Kill

12. Execute a ls command by setting priority as -10 using nice command : nice -n -10 ls

13. Display a date on every hour using cron tab: crontab -e

crontab -l

cat ~/hourly\_date.log