Amr Hassan Abd EL-Ghaffar
01027179463
Monday & Thursday 7 : 10 pm

Lab 02

1. a. cd
2. b. pwd
3. d. cd ..
4. c. cd ../..
5. d. ls -al
6. b. touch ~/helloworld.py
7. b. cd ..
8. c. cd ../../tmp
9. b. *b
10. a. b*
11. b. /etc/passwd
12. a. Home directory
13. c. Login shell

---

1. Log in to your machine as an user. In your home directory, create three subdirectories: " Music, Pictures, and Videos "

```
vagrant@amr:~$ cd
vagrant@amr:~$ mkdir Music Pictures Videos
```

2. Use the touch command to create sets of empty practice files to use during this lab. In each set, replace X with the numbers 1 through 6.
• Create six files with names of the form songX.mp3.
• Create six files with names of the form snapX.jpg.
• Create six files with names of the form filmX.avi.

```
vagrant@amr:~$ touch song{1..6}.mp3 snap{1..6}.jpg film{1..6}.avi
```

3. Move the song files (.mp3 extension) to the Music directory, the snapshot files (.jpg extension) to the Pictures directory, and the movie files (.avi extension) to the Videos directory.

```
vagrant@amr:~$ mv *.mp3 Music/
vagrant@amr:~$ mv *.jpg Pictures/
vagrant@amr:~$ mv *.avi Videos/
```

4. Create three subdirectories for organizing your files, and name the subdirectories friends, family, and work. Use a single command to create all three subdirectories at the same time.

```
vagrant@amr:~$ mkdir friend family work
```

5. Copy files that contain numbers 1 and 2 to the friends directory and files that contain numbers 3 and 4 to the family directory. Keep in mind that you are making copies; therefore, the original files must remain in their original locations after you complete the step.

```
vagrant@amr:~$ cp Music/song{1..2}.mp3 Pictures/snap{1..2}.jpg Videos/film{1..2}.avi friend/
vagrant@amr:~$ cp Music/song{3..4}.mp3 Pictures/snap{3..4}.jpg Videos/film{3..4}.avi family/
```

6. Copy the family and friends directories and their contents to the work directory .

```
vagrant@amr:~$ cp -r friend/ family/ work/
vagrant@amr:~$ ll work/
total 16
drwxrwxr-x  4 vagrant vagrant 4096 Mar  7 14:59 ./
drwxr-xr-x 10 vagrant vagrant 4096 Mar  7 14:47 ../
drwxrwxr-x  2 vagrant vagrant 4096 Mar  7 14:59 family/
drwxrwxr-x  2 vagrant vagrant 4096 Mar  7 14:59 friend/
```

7. It is time to clean up the directories. Use a command to recursively delete the family, friends, and work directories and their contents

```
vagrant@amr:~$ rm -rf friend/ family/ work/
```

8. In your home directory, create three subdirectories: " Links, and Files " :
Change your directory to files directory and create file named " target.file " write a text in this file " any text from your mind text must be more than 10 line only 2 lines starts with " devops " word and to lines contain " devops " word in the middle

```
vagrant@amr:~$ mkdir Links Files
vagrant@amr:~$ cd Files/
vagrant@amr:~/Files$ touch target.file
vagrant@amr:~/Files$ vim target.file
```

1. list the file content using 2 ways

```
vagrant@amr:~/Files$ cat target.file
DevOps is the bridge between development and operations, ensuring smooth collaboration.
It brings automation, monitoring, and CI/CD pipelines to streamline workflows.
By integrating DevOps practices, teams achieve faster and more reliable deployments.
With a strong focus on automation, DevOps minimizes human errors and improves efficiency.
Many organizations see DevOps as a game-changer for scaling their infrastructure.
Through continuous integration and delivery, DevOps accelerates software releases.
Collaboration between teams improves significantly when DevOps principles are embraced.
DevOps culture fosters a mindset of agility, innovation, and shared responsibility.
Companies investing in DevOps experience reduced downtime and improved system reliability.
Ultimately, DevOps transforms software development into a seamless and adaptive process.
vagrant@amr:~/Files$
vagrant@amr:~/Files$ less target.file
```

2. list only first 5 lines of the file

```
vagrant@amr:~/Files$ head -n 5 target.file
DevOps is the bridge between development and operations, ensuring smooth collaboration.
It brings automation, monitoring, and CI/CD pipelines to streamline workflows.
By integrating DevOps practices, teams achieve faster and more reliable deployments.
With a strong focus on automation, DevOps minimizes human errors and improves efficiency.
Many organizations see DevOps as a game-changer for scaling their infrastructure.
```

3.list only 2 lines on the file

```
vagrant@amr:~/Files$ head -n 2 target.file
DevOps is the bridge between development and operations, ensuring smooth collaboration.
It brings automation, monitoring, and CI/CD pipelines to streamline workflows.
```

4. list only the lines that contain " devops " word

```
vagrant@amr:~/Files$ cat target.file | grep DevOps
DevOps is the bridge between development and operations, ensuring smooth collaboration.
By integrating DevOps practices, teams achieve faster and more reliable deployments.
With a strong focus on automation, DevOps minimizes human errors and improves efficiency.
Many organizations see DevOps as a game-changer for scaling their infrastructure.
Through continuous integration and delivery, DevOps accelerates software releases.
Collaboration between teams improves significantly when DevOps principles are embraced.
DevOps culture fosters a mindset of agility, innovation, and shared responsibility.
Companies investing in DevOps experience reduced downtime and improved system reliability.
Ultimately, DevOps transforms software development into a seamless and adaptive process.
```

5. list only the lines that contain " devops " word in the beginning of the line .

```
vagrant@amr:~/Files$ cat target.file  | grep ^DevOps*
DevOps is the bridge between development and operations, ensuring smooth collaboration.
DevOps culture fosters a mindset of agility, innovation, and shared responsibility.
```

6. Create a hard link called /home/home name/links/file.hardlink for the /home/ home name/files/target.file file. Verify the link count for the original file and the new linked file.

```
vagrant@amr:~/Files$ ll
total 12
drwxrwxr-x 2 vagrant vagrant 4096 Mar  7 15:17 ./
drwxr-xr-x 9 vagrant vagrant 4096 Mar  7 15:53 ../
-rw-rw-r-- 1 vagrant vagrant  859 Mar  7 15:17 target.file
vagrant@amr:~/Files$ ln target.file ../Links/file.hardlink
vagrant@amr:~/Files$ ll ../Links/
total 12
drwxrwxr-x 2 vagrant vagrant 4096 Mar  7 15:54 ./
drwxr-xr-x 9 vagrant vagrant 4096 Mar  7 15:53 ../
-rw-rw-r-- 2 vagrant vagrant  859 Mar  7 15:17 file.hardlink
```

7. Create a symbolic link called /home/home name/tempdir that points to the /tmp directory on your machine. Verify the newly created symbolic link.

```
vagrant@amr:~$ ln -s /tmp/ ./tempdir
vagrant@amr:~$ ll
total 56
drwxr-xr-x 9 vagrant vagrant 4096 Mar  7 20:25 ./
drwxr-xr-x 4 root    root    4096 Feb 24 19:49 ../
-rw------- 1 vagrant vagrant  165 Mar  7 14:46 .bash_history
-rw-r--r-- 1 vagrant vagrant  220 Apr  9  2014 .bash_logout
-rw-r--r-- 1 vagrant vagrant 3637 Apr  9  2014 .bashrc
drwx------ 2 vagrant vagrant 4096 Feb 24 19:49 .cache/
drwxrwxr-x 2 vagrant vagrant 4096 Mar  7 15:17 Files/
drwxrwxr-x 2 vagrant vagrant 4096 Mar  7 15:54 Links/
drwxrwxr-x 2 vagrant vagrant 4096 Mar  5 21:15 Music/
drwxrwxr-x 2 vagrant vagrant 4096 Mar  5 21:15 Pictures/
-rw-r--r-- 1 vagrant vagrant  675 Apr  9  2014 .profile
drwx------ 2 vagrant vagrant 4096 Feb 24 19:49 .ssh/
lrwxrwxrwx 1 vagrant vagrant    5 Mar  7 20:25 tempdir -> /tmp/
drwxrwxr-x 2 vagrant vagrant 4096 Mar  5 21:16 Videos/
-rw------- 1 vagrant vagrant  781 Mar  7 15:17 .viminfo
```

8. Assign a value to a local shell variable. Variable names can contain uppercase or lowercase letters, digits, and the underscore character. Retrieve the variable value
Ex : variable course name value devops diploma .

```
vagrant@amr:~/tempdir$ Course_Name="DevOps Diploma"
vagrant@amr:~/tempdir$ echo $Course_Name
DevOps Diploma
```

9.
Explore the shell environment of your user & current user and group information

```
vagrant@amr:~$ echo $SHELL
/bin/bash
vagrant@amr:~$ cat /etc/passwd | grep $USER
vagrant:x:1000:1000::/home/vagrant:/bin/bash
```

Current working directory

```
vagrant@amr:~$ pwd
/home/vagrant
```

Environment variables that specify the user's home directory and the locations of the user's executable files

```
vagrant@amr:~$ echo $HOME
/home/vagrant
vagrant@amr:~$ echo $PATH
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games
```

10. Switch to the root user in a non-login shell and explore the new shell environment then exit .

```
vagrant@amr:~$ su
Password:
root@amr:/home/vagrant# whoami
root
root@amr:/home/vagrant# echo $SHELL
/bin/bash
root@amr:/home/vagrant# echo $HOME
/root
root@amr:/home/vagrant# exit
exit
```

11. Switch to the root user in a login shell and explore the new shell environment then exit .

```
vagrant@amr:~$ su -
Password:
root@amr:~# whoami
root
root@amr:~# echo $SHELL
/bin/bash
root@amr:~# echo $HOME
/root
root@amr:~# exit
logout
```

12. Create a user named “ devops “ and confirm that it exists in the system.
13. Set the password for devops user and switch to it

```
vagrant@amr:~$ sudo useradd -m devops
vagrant@amr:~$ sudo passwd devops
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
vagrant@amr:~$ cat /etc/passwd | grep devops
devops:x:1002:1002::/home/devops:
vagrant@amr:~$ su - devops
Password:
devops@amr:~$ cd $HOME
devops@amr:~$ pwd
/home/devops
devops@amr:~$ whoami
devops
```

14. Update the devops user account to include the “ Test user “ comment. and Verify that the comment exist for the user account.

```
vagrant@amr:~$ usermod --help | grep comment
  -c, --comment COMMENT         new value of the GECOS field
vagrant@amr:~$ sudo usermod -c "Test user" devops
vagrant@amr:~$ cat /etc/passwd | grep devops
devops:x:1002:1002:Test user:/home/devops:
```

15. Create the operators secondary group with a GID of 30000.

```
vagrant@amr:~$ sudo groupadd -g 30000 operators
vagrant@amr:~$ cat /etc/group | grep operators
operators:x:30000:
```

16. Create the admin secondary group without specifying a GID.

```
vagrant@amr:~$ sudo groupadd admin2
vagrant@amr:~$ cat /etc/group | grep admin2
admin2:x:30001:
```

17. Verify that both the operators and admin secondary groups exist.

```
vagrant@amr:~$ getent group operators admin2
operators:x:30000:
admin2:x:30001:
```

18. Add your user to the operators group.
19. Confirm that the user are in the group.

```
vagrant@amr:~$ sudo usermod -aG operators devops
vagrant@amr:~$ groups devops
devops : devops operators
```

20. Create a new user named “sys admin” and add it to admin group .

```
vagrant@amr:~$ sudo useradd -m sys_admin
vagrant@amr:~$ sudo passwd sys_admin
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
vagrant@amr:~$ sudo usermod -aG admin2 sys_admin
vagrant@amr:~$ groups sys_admin
sys_admin : sys_admin admin2
```

21. Delete the devops user along with any personal data of the user. Confirm that the devops user does not exist .

```
vagrant@amr:~$ sudo userdel -r devops
userdel: devops mail spool (/var/mail/devops) not found
vagrant@amr:~$ cat /etc/passwd | grep devops
```