Activity No. 13	
Working with Files and Directories in Linux	
Course Code: CPE 201A	**Program: BSCPE**
Course Title: COMPUTER SYSTEM ADMINISTRATION AND TROUBLESHOOTING	**Date Performed: 11/11/2025**
Section: CPE11S5	**Date Submitted: 11/11/2025**
Name: Andee Ramon V. Salgado	**Instructor: Lloyd Aldrin Pornobi**

1. Objective/s:

This activity aims to navigate and manage files and directories using command line interface.

2. Intended Learning Outcome/s:

The students should be able to:

2.1 Demonstrate how to navigate home and system directories
2.2 Demonstrate how to how to list files and directories.
2.3 Demonstrate how to use globbing to manipulate files and directories.
2.4 Demonstrate how to create, move and delete files and directories.

3. Discussion:

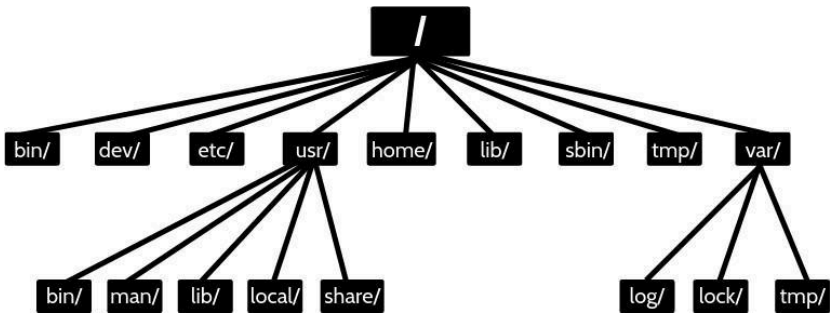
Files and Directories

The Linux filesystem is similar to other operating system's filesystems in that it contains *files* and *directories*. Files contain data such as human-readable text, executable programs, or binary data that is used by the computer.

Directories are used to create organization within the filesystem. Directories can contain files and other directories.

Directory Structure

On a Windows system, the *top level* of the directory structure is called ***My Computer.*** The Linux directory structure, called a ***filesystem***, also has a top level called the ***root directory*** (symbolized by the **slash /character**).



Home Directory

On most Linux distributions there is a directory called **home** under the root /directory. Under this **/home** directory there is a directory for each user on the system. When a user opens a shell, they should

automatically be placed in their home directory. The user has the full control to create and delete additional files and directories in their home directory. Most other directories in a Linux filesystem are protected with *file permissions*. The home directory has a special symbol used to represent it, the **tilde ~ character**.

The directory name is the same as the name of the user. So, a user named sysadmin would have a *home directory* called /home/sysadmin:

File and Directory Names

File and directory names in Linux can contain lower case and upper case letters, numbers, spaces and special characters. However, since many special characters have a special meaning in the Linux shell, it is good practice to not use spaces or special characters when naming files or directories. Spaces, for example, need the *escape character* \ to be entered correctly

Paths

A path is a list of directories separated by the / character. There are two types of paths: *absolute* and *relative*. For example, /home/sysadmin is a path to the home directory

A. Absolute Paths

Absolute paths allow the user to specify the exact location of a directory. Absolute paths always starts at the root directory, and therefore it always begins with the / character.

B. Relative Paths

A relative path gives directions to a file relative to the current location in the filesystem. The user must currently be in a directory that contains objects in the path. Relative paths start with the name of a directory.

Special Relative Paths

The single period . character always represents the current directory.
Two period .. characters always represents one directory higher relative to the current directory, sometimes referred to as the parent directory.

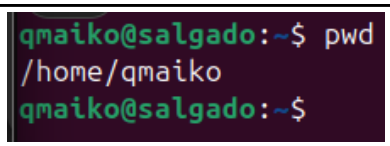
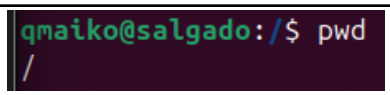
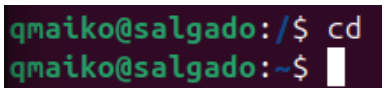
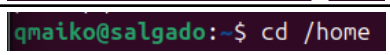
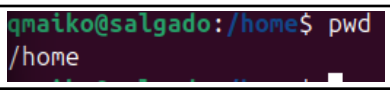
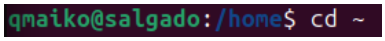
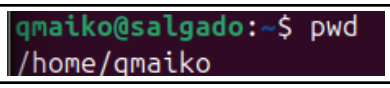
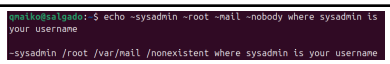
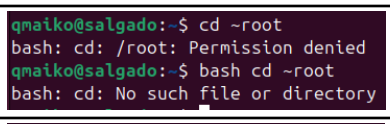
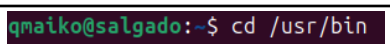
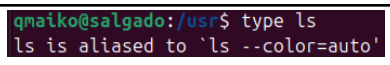
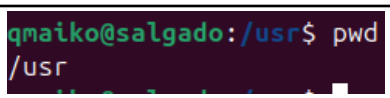
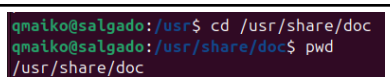
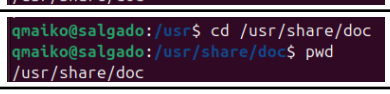
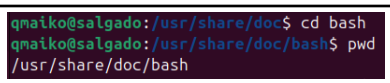
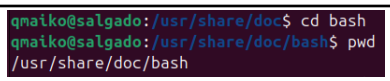
4. Resources:

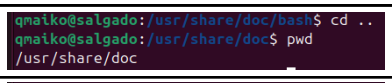
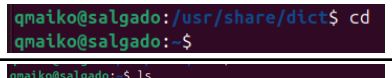
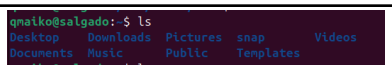
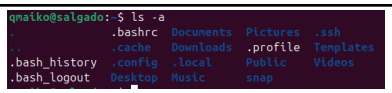
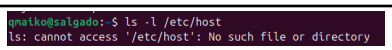
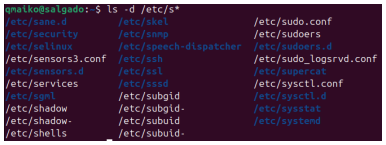
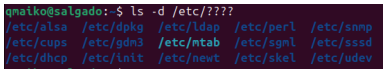
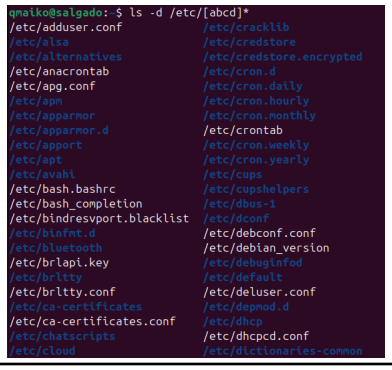
Personal Computer with installed Virtual Box
Ubuntu Server or Desktop virtual machine

5. Procedure:

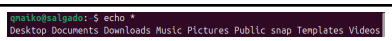
1. Login using your username and password.
2. Use terminal emulator application (if you are using desktop version)
3. Execute the following commands to navigate files and directories. Copy a screenshot as output after you execute the given command. Create a brief explanation of the command.

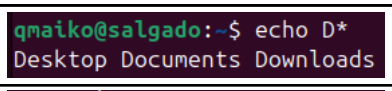
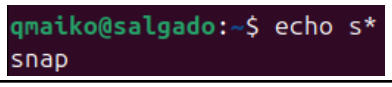
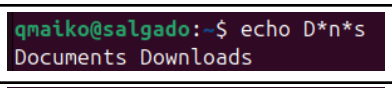
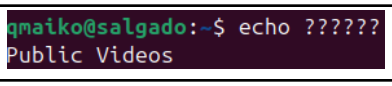
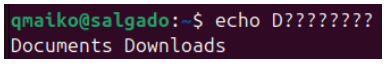
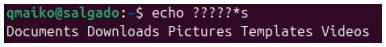
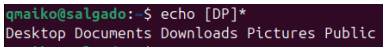
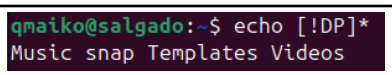
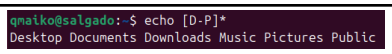
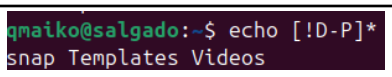
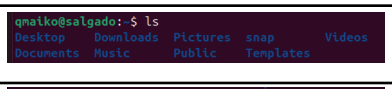
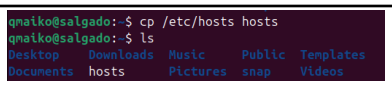
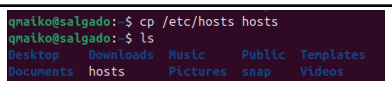
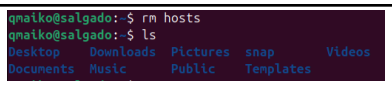
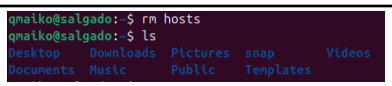
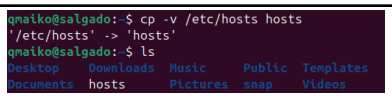
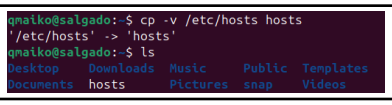
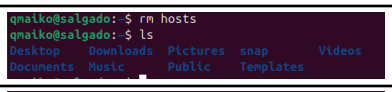
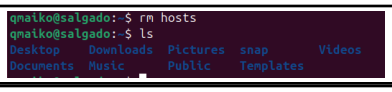
Command	Screenshot	Explanation

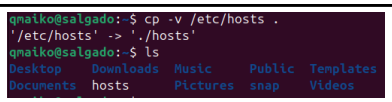
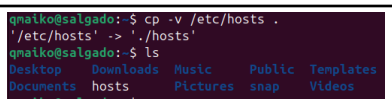
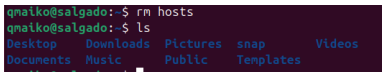
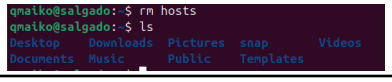
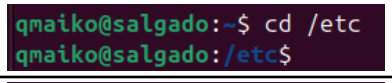
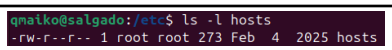
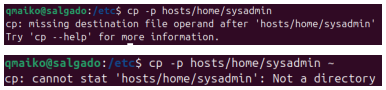
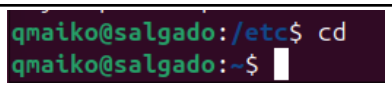
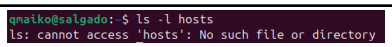
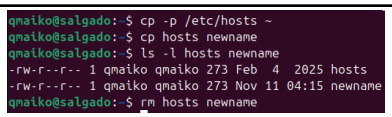
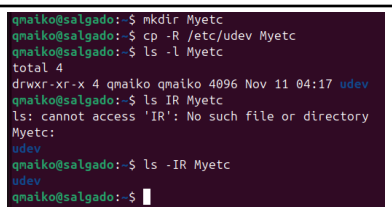
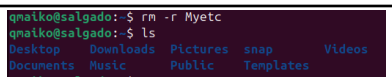
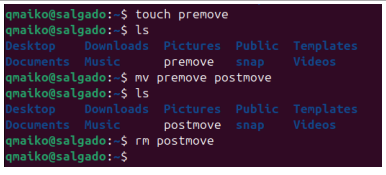
1. pwd		Shows the current directory
2. cd /		moves to the root directory
3. pwd		shows current directory to be in root
4. cd		moves to the default directory
5. pwd		shows current directory to be in default directory
6. cd /home		moves the directory to the home directory
7. pwd		shows the current directory is the home directory
8. cd ~		moves the directory to the default directory
9. pwd		shows the current directory
10. echo ~ ~sysadmin ~root ~mail ~nobody where sysadmin is your username		shows the home directory of each of the users
11. cd ~root		This should bring you to the directory for root, but my system does not allow me
12. cd /usr/bin		brings you to the user bin directory
13. pwd		shows current directory
14. cd /usr		brings you to the users folder
15. type ls		shows what type the ls function is
16. pwd		shows current directory
17. cd /usr/share/doc		goes to the shared documents folder
18. pwd		shows current directory
19. cd bash		goes into the bash directory
20. pwd		shows current directory

21. cd ..		goes back one directory
22. pwd		shows current directory
23. cd ../dict		goes to the /dict directory
24. pwd		shows current directory
25. cd		goes back to default directory
26. ls		shows all present directories
27. ls -a		shows all present directories, including the ones hidden
28. ls -l /etc/hosts		this should attempt to get the list of files in /host, but my system does not have such a file.
29. ls -R /etc/udev		shows all the files in the directories and subdirectories
30. ls -d /etc/s*		shows all directories within /etc that has an S
31. ls -d /etc/????		shows all directories within /etc with a matching amount of ?s (in this case 4)
32. ls –d /etc/[abcd]*		shows all directories with matching starting names as the symbols in the brackets

4. Execute the following commands to manage files and directories. Copy a screenshot as output after you execute the given command. Create a brief explanation of the command.

Command	Screenshot	Explanation
1. echo *		prints all available files in the current directory

2. echo D*		prints only the ones starting with D
3. echo P*		prints only the ones starting with P
4. echo *s		prints only the ones starting with s
5. echo D*n*s		prints only the ones starting with D, contains n, and s
6. echo ??????		prints out directories with matching amount of ?s
7. echo D????????		prints out directories that start with a D and contains the matching amount of ?s
8. echo ?????*s		prints out directories that have at least 5 characters and ends with an s
9. echo [DP]*		prints out directories that have a D or P
10. echo [!DP]*		prints out directories that do not have a D or P
11. echo [D-P]*		prints out directories that have the letters D to P
12. echo [!D-P]*		prints out directories that do not contain the letters D to P
13. ls		lists all directories in active directory
14. cp /etc/hosts hosts		copies files from hosts, and names the copied file hosts
15. ls		lists all directories in active directory
16. rm hosts		deletes the file named hosts
17. ls		lists all directories in active directory
18. cp –v /etc/hosts hosts		-v just shows you where the file copied came from and what it was named to
19. ls		lists all directories in active directory
20. rm hosts		deletes the file named hosts
21. ls		lists all directories in active directory

22. cp –v /etc/hosts .		copies the contents of hosts into the current directory
23. ls		lists all directories in active directory
24. rm hosts		deletes the file named hosts
25. ls		lists all directories in active directory
26. cd /etc		moves into the /etc directory
27. ls -l hosts		shows information about the specified file
28. cp –p hosts/home/sysadmin		this command is improperly formatted. even when providing the destination directory, hosts/home/sysadmin does not exist on my system.
29. cd		returns to the default directory
30. ls –l hosts		supposed to show a long list of the hosts directory/file, but that file does not exist on my system
31. rm hosts 32. cp -p /etc/hosts ~ 33. cp hosts newname 34. ls –l hosts newname 35. rm hosts newname		removes the present hosts file. copies the hosts file to the default directory, copies hosts to new name, gets the information of hosts newname, then deletes it.
36. mkdir Myetc 37. cp –R /etc/udev Myetc 38. ls –l Myetc 39. ls –lR Myetc		makes a new directory in current directory, copies a file into the new directory, then it lists the files in the directory
40. ls 41. rm -r Myetc 42. ls		just deletes the Myetc directory that we created
43. touch premove 44. ls 45. mv premove postmove 46. ls 47. rm postmove		makes a file, moves the file into a new file, then deletes the new file

6. Supplementary Activity:

Copy screen shot(s) of the following tasks:

1. Create a directory CPE231 in the Documents directory.

```
qmaiko@salgado:~/Documents$ mkdir CPE11S5
```

2. Create a directory <lastname> and <firstname> under CPE231 directory.

```
qmaiko@salgado:~/Documents$ mkdir CPE11S5/Salgado || mkdir CPE11S5/Andee
```

3. Create two text files: (1) A.txt and (2) B.txt in the <lastname> directory.

```
qmaiko@salgado:~/Documents/CPE11S5$ touch Salgado/A.txt
qmaiko@salgado:~/Documents/CPE11S5$ touch Salgado/B.txt
```

4. Copy the content of <lastname> directory to <firstname> directory.

```
qmaiko@salgado:~/Documents/CPE11S5$ cp Salgado/* Andee
qmaiko@salgado:~/Documents/CPE11S5$ ls /Andee
ls: cannot access '/Andee': No such file or directory
qmaiko@salgado:~/Documents/CPE11S5$ ls Andee
A.txt  B.txt
qmaiko@salgado:~/Documents/CPE11S5$
```

5. Delete the content of <lastname> directory.

```
qmaiko@salgado:~/Documents/CPE11S5$ rm Salgado/*
```

6. Create a directory CPE231_backup in the Documents directory.

```
qmaiko@salgado:~/Documents$ mkdir CPE11S5_backup
```

7. Copy the content of CPE231 to CPE231_backup directory.

```
qmaiko@salgado:~/Documents$ cp -r  CPE11S5/* CPE11S5_backup
```

8. Remove the CPE231 directory.

```
qmaiko@salgado:~/Documents$ rm -r CPE11S5
```

7. Conclusion:

This would be useful for managing many files at once, but I do somewhat struggle to see how this would beat using the included file manager in Ubuntu. Now if this was Arch linux, I would understand as Arch linux really is just a terminal andn nothing else. As for now, I think I will stick to the file manager]

8. Assessment (Rubric for Laboratory Performance):