Activity No. 11	
Command Line Skills	
Course Code: CPE 201A	**Program: BSCPE**
Course Title: COMPUTER SYSTEM ADMINISTRATION AND TROUBLESHOOTING	**Date Performed: 23/10/25**
Section: CPE11S1	**Date Submitted: 23/10/25**
Name: JAMES DANIEL M. VERANO	**Instructor: ENGR. JIMLORD QUEJADO**

1. Objective/s:

This activity aims to execute basic commands using command line interface of Linux.

2. Intended Learning Outcome/s:

The students should be able to:

2.1 Demonstrate how to use commands to explore BASH features.
2.2 Demonstrate how to use commands to display the values of Shell variables.
2.3 Demonstrate how to use quoting in Bash shells.

3. Discussion:

Command Line Interface

The Linux community promotes the CLI due to its power, speed and ability to accomplish a vast array of tasks with a single command line instruction. The CLI provides more precise control, greater speed and the ability to automate tasks more easily through scripting. By learning the CLI, a user can easily be productive almost instantly on ANY flavor or distribution of Linux.

The Shell

Once a user has entered a command , the terminal then accepts what the user has typed and passes to a shell. The shell is a program that enables text based communication between the operating system and the user. It is the command line interpreter that translates commands entered by a user into actions to be performed by the operating system. The Linux environment allows the use of many different shells.
There are several different shells on Linux, these are just a few:

- Bourne-again shell (Bash)
- C shell (csh or tcsh, the enhanced csh)
- Korn shell (ksh)
- Z shell (zsh)

The most commonly used shell for Linux distributions is called the **Bash** shell. When using an interactive shell, the user inputs commands at a so-called prompt. For each Linux distribution, the default prompt may look a little different, but it usually follows this structure:

```
username@hostname current_directory shell_type
```

On Ubuntu or Debian GNU/Linux, the prompt for a regular user will likely look like this:

```
carol@mycomputer:~$
```

The superuser's prompt will look like this:

```
root@mycomputer:~#
```

On CentOS or Red Hat Linux, the prompt for a regular user will instead look like this:

```
[dave@mycomputer ~]$
```

And the superuser's prompt will look like this:

```
[root@mycomputer ~]#
```

Let's explain each component of the structure:

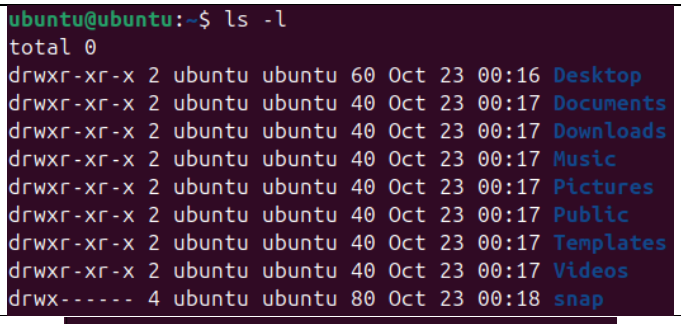
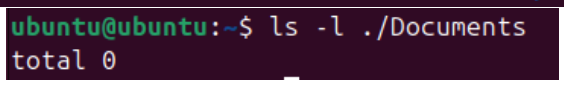
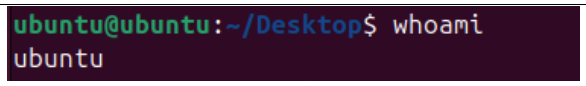
username
Name of the user that runs the shell
hostname
Name of the host on which the shell runs. There is also a command hostname , with which you can show or set the system's host name.
current_directory
The directory that the shell is currently in. A ~ means that the shell is in the current user's home directory.
shell_type
$ indicates the shell is run by a regular user.
# indicates the shell is run by the superuser root

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. Copy a screenshot as output after you execute the given command. Create a brief explanation of the command.

Command	Screenshot	Explanation
1. ls -l		Displays details information's in the current Directory
2. ls -l ./Documents		Lists of detailed contents of the Documents Directory
3. whoami		Display's the username of Linux

4. Uname	`ubuntu@ubuntu:~/Desktop$ uname` `Linux`	It prints the hostname of the network node
5. pwd	`ubuntu@ubuntu:~/Desktop$ pwd` `/home/ubuntu/Desktop`	Displays the full path of the current working directory.
6. echo Hi	`ubuntu@ubuntu:~/Desktop$ echo Hi` `Hi`	Displays Hi on the command line.

7. history	`ubuntu@ubuntu:~/Desktop$ history`
	`1 Is -I`
	`2 ls -l`
	`3 ls -l /Documents`
	`4 ls -l/Documents`
	`5 ls -l ./Documents`
	`6 ls -l./Documents`
	`7 whoami`
	`8 Uname`
	`9 uname`
	`10 ls -l ./documents`
	`11 pwd`
	`12 echo Hi`
	`13 history`
		Displays the whole command history.

8. history 5	`ubuntu@ubuntu:~/Desktop$ history 5`
	`10 ls -l ./documents`
	`11 pwd`
	`12 echo Hi`
	`13 history`
	`14 history 5`
		Displays the recent 5 command history.

9. !9	`ubuntu@ubuntu:~/Desktop$ !9`
	`uname`
	`Linux`
		Executes a a specific based by the command history number

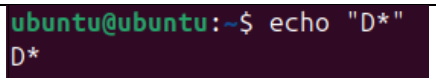
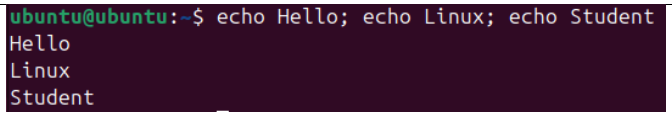
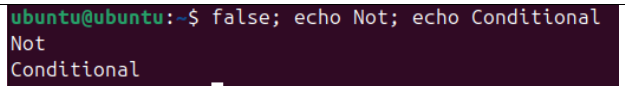
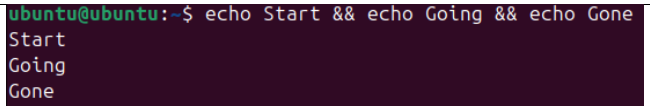
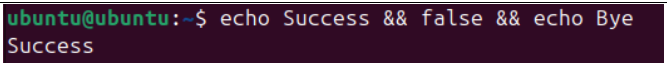
Formatted: Centered, Space After: 0 pt, Line spacing: single

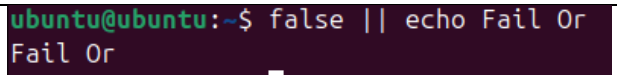
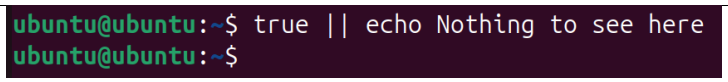
10. echo Hello Student	`ubuntu@ubuntu:~/Desktop$ echo hello Student` `hello Student`	Displays Hello Student in the command line.
11. echo $HISTSIZE	`ubuntu@ubuntu:~/Desktop$ echo $HISTSIZE` `1000`	Displays the commands that are stored in memory in a history list while bash session is on-going.
12. echo $PATH	`ubuntu@ubuntu:~/Desktop$ echo $PATH` `/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin:/snap/bin`	Tells the shell which directory to search for executable files.
13. which date	`ubuntu@ubuntu:~/Desktop$ which date` `/usr/bin/date`	Searches for the executable file named date.
14. type cd	`ubuntu@ubuntu:~/Desktop$ type cd` `cd is a shell builtin`	Displays that the CD is a shell built-in.
15. type ls	`ubuntu@ubuntu:~/Desktop$ type ls` `ls is aliased to 'ls --color=auto'`	Shows where the ls command is stored.

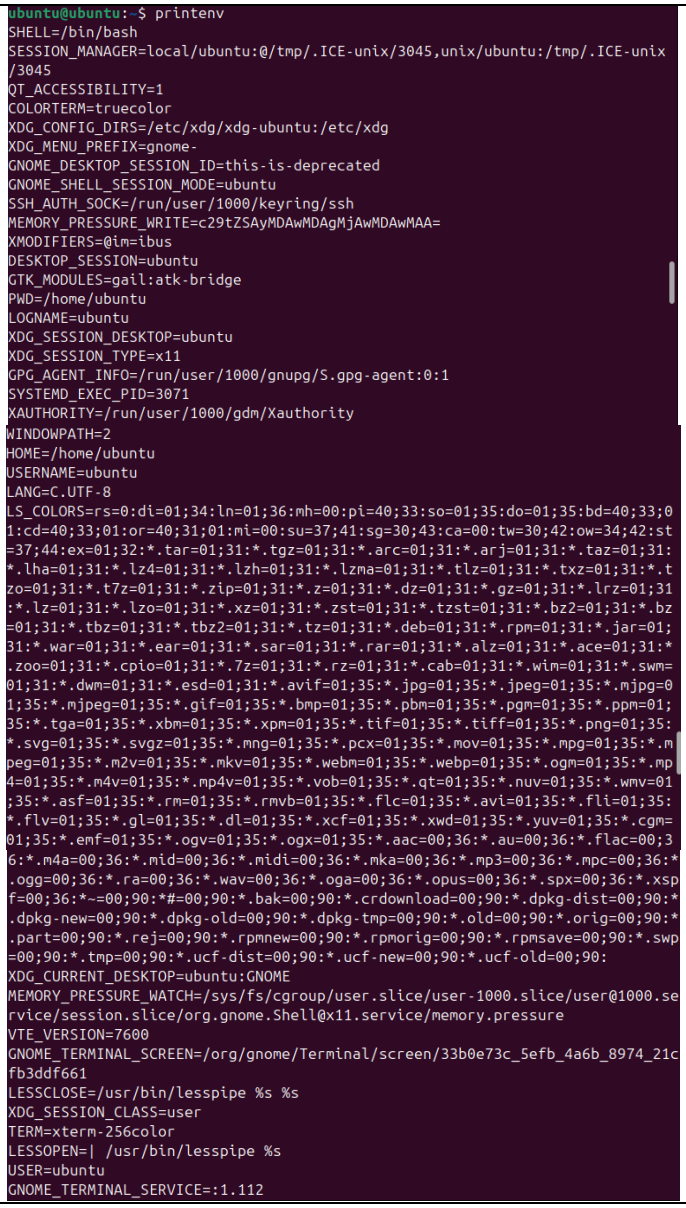
16. alias	`ubuntu@ubuntu:~/Desktop$ alias`
	`alias alert='notify-send --urgency=low -i "$([ $? = 0 ] && echo terminal \|\| echo error)" "$(history|tail -n1|sed -e '\''s/^\s*[0-9]\+\s*//;s/[;&|]\s*alert$//'\'')"'`
	`alias egrep='egrep --color=auto'`
	`alias fgrep='fgrep --color=auto'`
	`alias grep='grep --color=auto'`
	`alias l='ls -CF'`
	`alias la='ls -A'`
	`alias ll='ls -alF'`
	`alias ls='ls --color=auto'`
		Displays user-defined shortcuts for commands.

17. type vi	`ubuntu@ubuntu:~/Desktop$ type vi` `vi is /usr/bin/vi`	Shows the path to text editor.

18. cd /bin	`ubuntu@ubuntu:/bin$`	Changes the working directory to /bin.
19. type vlc	`ubuntu@ubuntu:/bin$ type vlc` `bash: type: vlc: not found`	Checks if VLC is installed.
20. cd	`ubuntu@ubuntu:~$`	Changes the working directory to ~.
21. echo Today is `date`	`ubuntu@ubuntu:~$ echo today is `date`` `today is Thu Oct 23 00:39:34 UTC 2025`	Displays the date today.
22. echo Today is $(date)	`ubuntu@ubuntu:~$ echo Today is $(date)` `Today is Thu Oct 23 00:36:58 UTC 2025`	Another way to display the date today.
23. echo This is the command ``date``	`ubuntu@ubuntu:~$ echo this is the command ``date``` `this is the command `date``	Displays the literal text with the backticks.
24. echo This is the command \`date\`	`ubuntu@ubuntu:~$ echo This is the command \`date\``  `This is the command `date``	Displays the text and backslashes the backticks to escape the execute.
25. echo This is the command "`date`"	`ubuntu@ubuntu:~$ echo This is the command "'date'"` `This is the command 'date'`	Displays the texts using double quotation to display the literal text command `date` with the backticks.
26. echo D*	`ubuntu@ubuntu:~$ echo D*` `Desktop Documents Downloads`	Displays the lists of all files and directories in the current folder that

		starts with letter D.
27. echo "D*"		Prints out the literal text with the asterisk without executing the command.
28. echo Hello; echo Linux; echo Student		This executes 3 separate commands using the command separator (;).
29. false; echo Not; echo Conditional		Prints out the echo commands sequentially even though false failed.
30. echo Start && echo Going && echo Gone		Executes the command with the && operator, which stands for execute only if the previous was successful in Linux command.
31. echo Success && false && echo Bye		Executes the echo success and disregards the echo Bye since the &&

		condition was not met.
32. false || echo Fail Or		The command displays the echo Fail Or since the or operator has not met the false.
33. true || echo Nothing to see here		The commands executes and prompts nothing since the true command always returns a success, and we have a or operator, which already outputs 1 or them possibilities. Thus, the other option does not run.

34. printenv			Displays all the environment variables currently set in the shell session.

	DISPLAY=:0 SHLVL=1 GSM_SKIP_SSH_AGENT_WORKAROUND=true QT_IM_MODULE=ibus XDG_RUNTIME_DIR=/run/user/1000 DEBUGINFOD_URLS=https://debuginfod.ubuntu.com XDG_DATA_DIRS=/usr/share/ubuntu:/usr/share/gnome:/usr/local/share/:/usr/share/:/var/lib/snapd/desktop PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin:/snap/bin GDMSESSION=ubuntu DBUS_SESSION_BUS_ADDRESS=unix:path=/run/user/1000/bus OLDPWD=/bin _=/usr/bin/printenv	
35. printenv TERM	ubuntu@ubuntu:~$ printenv TERM xterm-256color	Displays the terminal type
36. echo $TERM	ubuntu@ubuntu:~$ echo $TERM xterm-256color	This prints out the value of TERM environment variable.
37. env	ubuntu@ubuntu:~$ printenv SHELL=/bin/bash SESSION_MANAGER=local/ubuntu:@/tmp/.ICE-unix/3045,unix/ubuntu:/tmp/.ICE-unix/3045 QT_ACCESSIBILITY=1 COLORTERM=truecolor XDG_CONFIG_DIRS=/etc/xdg/xdg-ubuntu:/etc/xdg XDG_MENU_PREFIX=gnome- GNOME_DESKTOP_SESSION_ID=this-is-deprecated GNOME_SHELL_SESSION_MODE=ubuntu SSH_AUTH_SOCK=/run/user/1000/keyring/ssh MEMORY_PRESSURE_WRITE=c29tZSAyMDAwMDAgMjAwMDAwMAA= XMODIFIERS=@im=ibus DESKTOP_SESSION=ubuntu GTK_MODULES=gail:atk-bridge PWD=/home/ubuntu LOGNAME=ubuntu XDG_SESSION_DESKTOP=ubuntu XDG_SESSION_TYPE=x11 GPG_AGENT_INFO=/run/user/1000/gnupg/S.gpg-agent:0:1 SYSTEMD_EXEC_PID=3071 XAUTHORITY=/run/user/1000/gdm/Xauthority	Another way to displays all the environment variables currently set in the shell session.

Formatted: Centered, Space After: 0 pt, Line spacing: single

```
WINDOWPATH=2
HOME=/home/ubuntu
USERNAME=ubuntu
LANG=C.UTF-8
LS_COLORS=rs=0:di=01;34:ln=01;36:mh=00:pi=40;33:so=01;35:do=01;35:bd=40;33;0
1:cd=40;33;01:or=40;31;01:mi=00:su=37;41:sg=30;43:ca=00:tw=30;42:ow=34;42:st
=37;44:ex=01;32:*.tar=01;31:*.tgz=01;31:*.arc=01;31:*.arj=01;31:*.taz=01;31:
*.lha=01;31:*.lz4=01;31:*.lzh=01;31:*.lzma=01;31:*.tlz=01;31:*.txz=01;31:*.t
zo=01;31:*.t7z=01;31:*.zip=01;31:*.z=01;31:*.dz=01;31:*.gz=01;31:*.lrz=01;31
:*.lz=01;31:*.lzo=01;31:*.xz=01;31:*.zst=01;31:*.tzst=01;31:*.bz2=01;31:*.bz
=01;31:*.tbz=01;31:*.tbz2=01;31:*.tz=01;31:*.deb=01;31:*.rpm=01;31:*.jar=01;
31:*.war=01;31:*.ear=01;31:*.sar=01;31:*.rar=01;31:*.alz=01;31:*.ace=01;31:*
.zoo=01;31:*.cpio=01;31:*.7z=01;31:*.rz=01;31:*.cab=01;31:*.wim=01;31:*.swm=
01;31:*.dwm=01;31:*.esd=01;31:*.avif=01;35:*.jpg=01;35:*.jpeg=01;35:*.mjpg=0
1;35:*.mjpeg=01;35:*.gif=01;35:*.bmp=01;35:*.pbm=01;35:*.pgm=01;35:*.ppm=01;
35:*.tga=01;35:*.xbm=01;35:*.xpm=01;35:*.tif=01;35:*.tiff=01;35:*.png=01;35:
*.svg=01;35:*.svgz=01;35:*.mng=01;35:*.pcx=01;35:*.mov=01;35:*.mpg=01;35:*.m
peg=01;35:*.m2v=01;35:*.mkv=01;35:*.webm=01;35:*.webp=01;35:*.ogm=01;35:*.mp
4=01;35:*.m4v=01;35:*.mp4v=01;35:*.vob=01;35:*.qt=01;35:*.nuv=01;35:*.wmv=01
;35:*.asf=01;35:*.rm=01;35:*.rmvb=01;35:*.flc=01;35:*.avi=01;35:*.fli=01;35:
*.flv=01;35:*.gl=01;35:*.dl=01;35:*.xcf=01;35:*.xwd=01;35:*.yuv=01;35:*.cgm=
01;35:*.emf=01;35:*.ogv=01;35:*.ogx=01;35:*.aac=00;36:*.au=00;36:*.flac=00;3
6:*.m4a=00;36:*.mid=00;36:*.midi=00;36:*.mka=00;36:*.mp3=00;36:*.mpc=00;36:*
.ogg=00;36:*.ra=00;36:*.wav=00;36:*.oga=00;36:*.opus=00;36:*.spx=00;36:*.xsp
f=00;36:*~=00;90:*#=00;90:*.bak=00;90:*.crdownload=00;90:*.dpkg-dist=00;90:*
.dpkg-new=00;90:*.dpkg-old=00;90:*.dpkg-tmp=00;90:*.old=00;90:*.orig=00;90:*
.part=00;90:*.rej=00;90:*.rpmnew=00;90:*.rpmorig=00;90:*.rpmsave=00;90:*.swp
=00;90:*.tmp=00;90:*.ucf-dist=00;90:*.ucf-new=00;90:*.ucf-old=00;90:
XDG_CURRENT_DESKTOP=ubuntu:GNOME
MEMORY_PRESSURE_WATCH=/sys/fs/cgroup/user.slice/user-1000.slice/user@1000.se
rvice/session.slice/org.gnome.Shell@x11.service/memory.pressure
VTE_VERSION=7600
GNOME_TERMINAL_SCREEN=/org/gnome/Terminal/screen/33b0e73c_5efb_4a6b_8974_21c
fb3ddf661
LESSCLOSE=/usr/bin/lesspipe %s %s
XDG_SESSION_CLASS=user
TERM=xterm-256color
LESSOPEN=| /usr/bin/lesspipe %s
USER=ubuntu
GNOME_TERMINAL_SERVICE=:1.112
DISPLAY=:0
SHLVL=1
GSM_SKIP_SSH_AGENT_WORKAROUND=true
QT_IM_MODULE=ibus
XDG_RUNTIME_DIR=/run/user/1000
DEBUGINFOD_URLS=https://debuginfod.ubuntu.com
XDG_DATA_DIRS=/usr/share/ubuntu:/usr/share/gnome:/usr/local/share/:/usr/shar
e/:/var/lib/snapd/desktop
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games
:/usr/local/games:/snap/bin:/snap/bin
GDMSESSION=ubuntu
DBUS_SESSION_BUS_ADDRESS=unix:path=/run/user/1000/bus
OLDPWD=/bin
_=/usr/bin/env
```

6. Supplementary Activity:

Copy screen shot(s) of the following tasks:

1. An alias can be used to map longer commands to shorter key sequences. Use an alias to represent a very long command.

```
ubuntu@ubuntu:~$ alias loc='ls -l ./Documents'
```

2. Create a new directory in the Documents directory. Rename the directory as CPE_201A _(lastname).

```
ubuntu@ubuntu:~$ loc
total 0
drwxrwxr-x 2 ubuntu ubuntu 80 Oct 23 01:47 CPE_201A_VERANO
```

Create a new file inside the CPE_201A_(lastname) directory. Rename the file as sample1_lastname.txt.

```
ubuntu@ubuntu:~$ cd ~/Documents/CPE_201A_VERANO
ubuntu@ubuntu:~/Documents/CPE_201A_VERANO$ touch sample_Verano.txt
```

Display the content of the CPE_201A _(lastname) directory by executing one line of command only.

```
ubuntu@ubuntu:~/Documents/CPE_201A_VERANO$ ls
Sample1_Verano.odt  sample_Verano.txt
```

3. Execute a command to display the working shell.

```
ubuntu@ubuntu:~/Documents/CPE-201A_VERANO$ echo sample_Verano.txt
sample_Verano.txt
```

4. Shell variables, called environment variables, have the string data type and typically are named with capital letters and the _ (underline) character. Names are case sensitive. The env command will list all the environment variables. The printenv command will list all or will list only the names on its command line. List all environment variables. Which start with P?

```
ubuntu@ubuntu:~$ printenv | grep "^P"
PWD=/home/ubuntu
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin:/snap/bin
```

S7. Conclusion:

⇨ In this lab activity, I was able to execute and use different commands in displaying and locating files and folder directories. It also exposes us to variety ways to execute commands in creating folder directories and create file names inside the folder directories while in the command prompt. This activity, has furtherly deepen the understanding on how Linux command helps in files and folder location and identification of directories with regards to specific command function that are initiated. Linux command represents and allows users to interact directly with the operating system, which this can help users perform complex tasks efficiently.

8. Assessment (Rubric for Laboratory Performance):