Activity No. 11	
Command Line Skills	
Course Code: CPE 201A	**Program:**
Course Title: COMPUTER SYSTEM ADMINISTRATION AND TROUBLESHOOTING	**Date Performed:**
Section:	**Date Submitted:**
Name:	**Instructor:**

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:

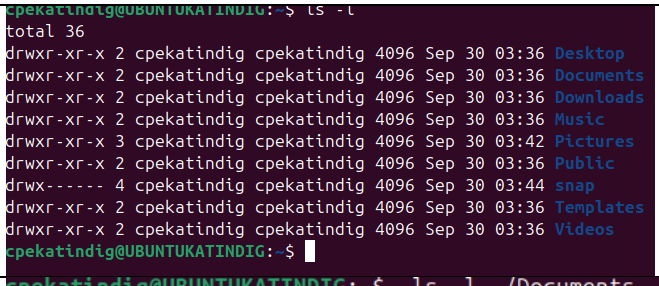
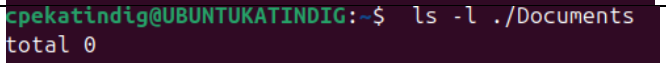
username
Name of the user that runs the shell
hostname
Name of the host on which the shell runs. There is also a command hostname , with which you can show or set the system's host name.
current_directory
The directory that the shell is currently in. A ~ means that the shell is in the current user's 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 detailed file information in the current directory
2. ls -l ./Documents		Lists detailed contents of the Documents directory
3. whoami		Shows the current username

4. Uname	`cpekatindig@UBUNTUKATINDIG:~$ uname` `Linux`	Displays the OS kernel name
5. pwd	`cpekatindig@UBUNTUKATINDIG:~$ pwd` `/home/cpekatindig`	Prints the current working directory
6. echo Hi	`cpekatindig@UBUNTUKATINDIG:~$ echo Hi` `Hi`	Print text "Hi" to the screen

7. history	`cpekatindig@UBUNTUKATINDIG:~$ history`
	`1 sudo apt-get install bzip2`
	`2 ls- l`
	`3 ls -l`
	`4 ls -l/Documents`
	`5 ls -l /Documents`
	`6 ls -l ./Documents`
	`7 whoami`
	`8 Uname`
	`9 uname`
	`10 pwd`
	`11 echo hi`
	`12 echo HI`
	`13 echo Hi`
	`14 history`
		Displays the command history

8. history 5	`cpekatindig@UBUNTUKATINDIG:~$ history 5`
	`11 echo hi`
	`12 echo HI`
	`13 echo Hi`
	`14 history`
	`15 history 5`
		Displays 5 command history

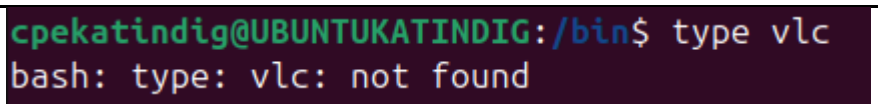
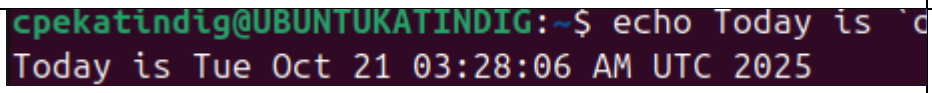
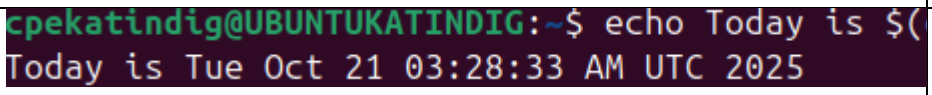
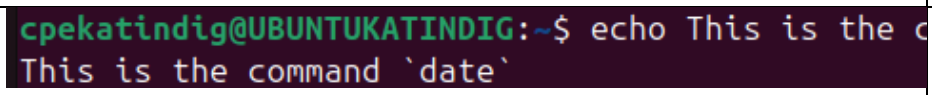
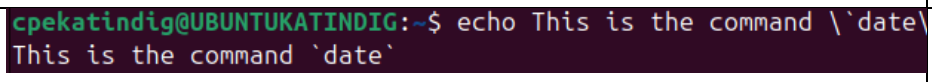
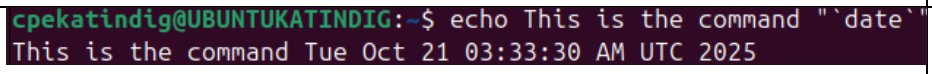
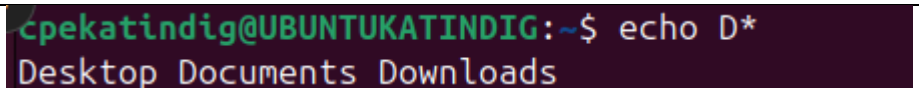
9. !9	`cpekatindig@UBUNTUKATINDIG:~$ !9`
	`uname`
	`Linux`
		Executes a command based by the history number

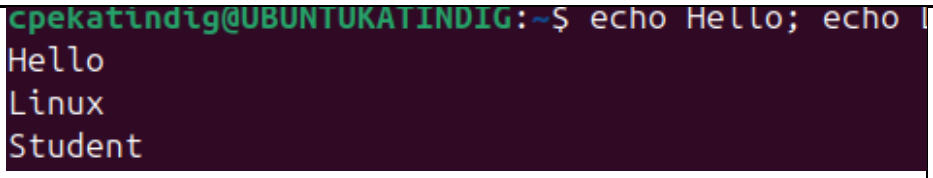
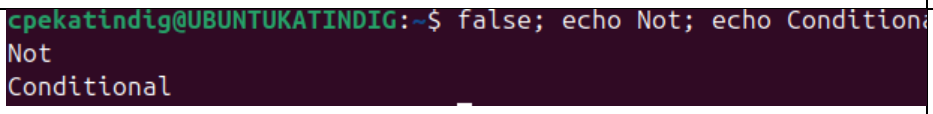
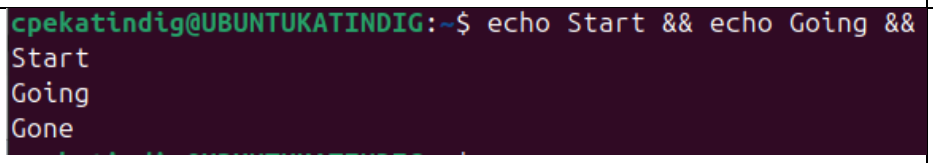
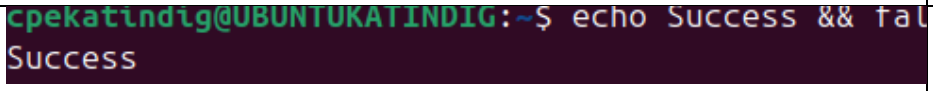
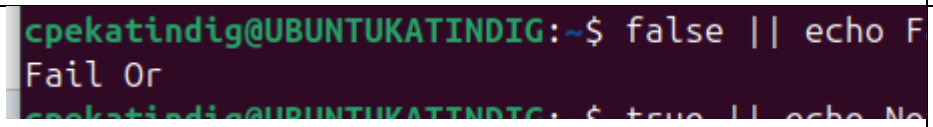
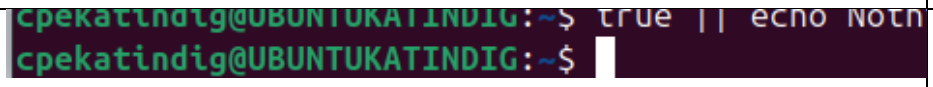
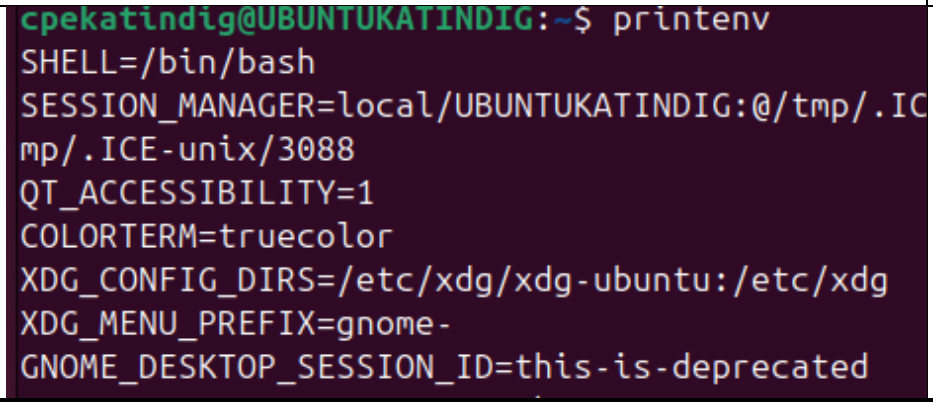
10. echo Hello Student	`cpekatindig@UBUNTUKATINDIG:~$ echo Hello Stude` `Hello Student`	Prints "Hello Student"

11. echo $HISTSIZE	`cpekatindig@UBUNTUKATINDIG:~$ echo $HISTSIZE` `1000`	Shows how many commands history stores
12. echo $PATH	`cpekatindig@UBUNTUKATINDIG:~$ echo $PATH` `/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/usr/games:/snap/bin:/snap/bin`	Shows directories where executables are searched
13. which date	`cpekatindig@UBUNTUKATINDIG:~$ which date` `/usr/bin/date`	Shows the path to the date command
14. type cd	`cpekatindig@UBUNTUKATINDIG:~$ type cd` `cd is a shell builtin`	Shows that cd is a built -in shell command
15. type ls	`cpekatindig@UBUNTUKATINDIG:~$ type ls` `ls is aliased to 'ls --color=auto'`	Shows where the ls command is stored

16. alias	`cpekatindig@UBUNTUKATINDIG:~$ 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	`cpekatindig@UBUNTUKATINDIG:~$ type vi` `vi is /usr/bin/vi`	Shows the path to the text editor vi
18. cd /bin	`cpekatindig@UBUNTUKATINDIG:~$ cd /bin` `cpekatindig@UBUNTUKATINDIG:/bin$`	Changes directory to /bin (system binaries)

19. type vlc		Checks if VLC is installed and where
20. cd		Returns to home directory
21. echo Today is `date`		Uses backticks to run the date command
22. echo Today is $(date)		Another syntax for command substitution
23. echo This is the command "date"		Prints literal text with quotes around backticks
24. echo This is the command \`date\`		Escapes backticks so they don't execute
25. echo This is the command "`date`"		Displays text with the result of the date command using command substitution inside quotes
26. echo D*		Displays files/directories starting with D
27. echo "D*"		Displays the text D*

28. echo Hello; echo Linux; echo Student		Runs multiple commands in one line
29. false; echo Not; echo Conditional		false fails but next commands still run
30. echo Start && echo Going && echo Gone		Runs all only if previous succeeds
31. echo Success && false && echo Bye		Stops when a command fails
32. false || echo Fail Or		echo runs if previous command fails
33. true || echo Nothing to see here		Redirected to ready a new command
34. printenv		Shows environment variables and their values

35. printenv TERM	`cpekatindig@UBUNTUKATINDIG:~$ printenv TERM` `xterm-256color`	Shows the terminal type
36. echo $TERM	`cpekatindig@UBUNTUKATINDIG:~$ echo $TERM` `xterm-256color`	Displays same terminal environment variable

37.
 	
env
	`cpekatindig@UBUNTUKATINDIG:~$ env`
	`SHELL=/bin/bash`
	`SESSION_MANAGER=local/UBUNTUKATINDIG:@/tmp/.IC`
	`mp/.ICE-unix/3088`
	`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=c29tZSAyMDAwMDAgMjAwMDAw`
		Shows all environment variables in the shell

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.

```
cpekatindig@UBUNTUKATINDIG:~$ alias lsg='ls -l --group-director
=auto'
cpekatindig@UBUNTUKATINDIG:~$ alias
alias alert='notify-send --urgency=low -i "$([ $? = 0 ] && echo
 error)" "$(history|tail -n1|sed -e '\''s/^\s*[0-9]\+\s*//;s/[;
')"'
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'
alias lsd='ls -la --color=auto --group-directories-first'
alias lsg='ls -l --group-directories-first --color=auto'
cpekatindig@UBUNTUKATINDIG:~$
```

2. Create a new directory in the Documents directory. Rename the directory as CPE_201A _(lastname). Create a new file inside the CPE_201A_(lastname) directory. Rename the file as sample1_lastname.txt. Display the content of the CPE_201A _(lastname) directory by executing one line of command only.

```
cpekatindig@UBUNTUKATINDIG:~$ cd Documents
mkdir CPE_201A_KATINDIG
cd CPE_201A_KATINDIG
touch sample1_Katindig.txt
ls ~/Documents/CPE_201A_KATINDIG
sample1_Katindig.txt
cpekatindig@UBUNTUKATINDIG:~/Documents/CPE_201A_KATINDIG$
```

3. Execute a command to display the working shell.

```
cpekatindig@UBUNTUKATINDIG:~/Documents/CPE_201A_KATINDIG$ echo $SHELL
/bin/bash
```

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?

```
cpekatindig@UBUNTUKATINDIG:~/Documents/CPE_201A_KATINDIG$ printenv
SHELL=/bin/bash
SESSION_MANAGER=local/UBUNTUKATINDIG:@/tmp/.ICE-unix/3088,unix/UBU
mp/.ICE-unix/3088
QT_ACCESSIBILITY=1
COLORTERM=truecolor
XDG_CONFIG_DIRS=/etc/xdg/xdg-ubuntu:/etc/xdg
XDG_MENU_PREFIX=gnome-
NOME_DESKTOP_SESSION_ID=this-is-deprecated
```

```
cpekatindig@UBUNTUKATINDIG:~/Documents/CPE_201A_KATINDIG$ printenv | grep ^P
PWD=/home/cpekatindig/Documents/CPE_201A_KATINDIG
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/u
r/local/games:/snap/bin:/snap/bin
cpekatindig@UBUNTUKATINDIG:~/Documents/CPE_201A_KATINDIG$
```

7. Conclusion:

Through this activity, I learned how to use basic Linux commands in the terminal. I practiced creating aliases, managing files and folders, and checking environment variables. Using commands like alias, ls, echo, env, and printenv helped me understand how the Bash shell works. Overall, this activity made me more comfortable and confident in using the command line.

8. Assessment (Rubric for Laboratory Performance):