

Activity No. 11	
Command Line Skills	
Course Code: CPE 201A	Program:
Course Title: COMPUTER SYSTEM ADMINISTRATION AND TROUBLESHOOTING	Date Performed: 10/23/2025
Section: CPE11S1	Date Submitted: 10/23/2025
Name: Cenar, Marqui	Instructor: Engr. Jimlord Quejado
<b>1. Objective/s:</b>	
This activity aims to execute basic commands using command line interface of Linux.	
<b>2. Intended Learning Outcome/s:</b>	
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.	
<b>3. Discussion:</b>	
<p><b>Command Line Interface</b></p> <p>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.</p> <p><b>The Shell</b></p> <p>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:</p> <ul style="list-style-type: none"> <li>• Bourne-again shell (Bash)</li> <li>• C shell (csh or tcsh, the enhanced csh)</li> <li>• Korn shell (ksh)</li> <li>• Z shell (zsh)</li> </ul> <p>The most commonly used shell for Linux distributions is called the <b>Bash</b> 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:</p> <p><b>username@hostname current_directory shell_type</b></p> <p>On Ubuntu or Debian GNU/Linux, the prompt for a regular user will likely look like this:</p> <p><b>carol@mycomputer:~\$</b></p> <p>The superuser's prompt will look like this:</p> <p><b>root@mycomputer:~#</b></p> <p>On CentOS or Red Hat Linux, the prompt for a regular user will instead look like this:</p> <p><b>[dave@mycomputer ~]\$</b></p> <p>And the superuser's prompt will look like this:</p>	

**[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.

**DISCLAIMER** – On my macbook the logged in account is my mother's account, so that's why it shows her name "Gie Cengar", not mine.

Command	Screenshot	Explanation
1. <code>ls -l</code>		This command lists all files and folders in the current directory in long format. It shows details like permissions, size, and date modified.
2. <code>ls -l ./Documents</code>		Same as the first command that shows all the files with full details and the <code>./Documents</code> means look inside the current directory.
3. <code>whoami</code>		This command shows my macOS username, Helpful to know what account is currently logged in

4. Uname	<pre>giecenar@M-FVFWQP9XJ1WL ~ % uname Darwin</pre>	The command shows basic information about your system like the OS type. On a macbook, it shows "Darwin".
5. pwd	<pre>giecenar@M-FVFWQP9XJ1WL ~ % pwd /Users/giecenar</pre>	pwd means "print working directory." Shows the folder path where you are currently located.
6. echo Hi	<pre>giecenar@M-FVFWQP9XJ1WL ~ % echo hi hi</pre>	This command displays/shows the word "Hi" like printing out a text.
7. history	<pre>giecenar@M-FVFWQP9XJ1WL ~ % history 1  xcode-select --install 2  xcode-select --install 3  clang --version 4  ifconfig 5  ls -l 6  ls -l ./Documents 7  ls -l ./Documents 8  whoami 9  uname 10 pwd 11 echo hi</pre>	Displays a list of all the commands you've recently used in the terminal.
8. history 5	<pre>giecenar@M-FVFWQP9XJ1WL ~ % history 5 5  ls -l 6  ls -l ./Documents 7  ls -l ./Documents 8  whoami 9  uname 10 pwd 11 echo hi 12 history</pre>	Shows only the 5 most recent commands you've executed.
9. !9	<pre>giecenar@M-FVFWQP9XJ1WL ~ % !9 zsh: command not found: 19</pre>	This command is likely a typo or alias, by default this will cause an error unless !9 is defined.
10. echo Hello Student	<pre>giecenar@M-FVFWQP9XJ1WL ~ % echo Hello Student Hello Student</pre>	Displays "Hello Student" on the screen. The echo command repeats whatever follows it.
11. echo \$HISTSIZE	<pre>giecenar@M-FVFWQP9XJ1WL ~ % echo \$HISTSIZE 2000</pre>	Displays the number of commands stored in your history
12. echo \$PATH	<pre>giecenar@M-FVFWQP9XJ1WL ~ % echo \$PATH /usr/local/bin:/usr/bin:/bin:/usr/sbin:/sbin</pre>	Shows directories where the Mac looks for executable programs.
13. which date	<pre>giecenar@M-FVFWQP9XJ1WL ~ % which date /bin/date</pre>	Shows the exact file path of the date command used by the system.
14. type cd	<pre>giecenar@M-FVFWQP9XJ1WL ~ % type cd cd is a shell builtin</pre>	This command tells the user what kind of command "cd" is, on a macbook usually it's a shell builtin.

15. type ls	<pre>giecenar@M-FVFWQP9XJ1WL ~ % type ls ls is /bin/ls</pre>	This command shows whether it's a builtin command or a system executable.
16. alias	<pre>giecenar@M-FVFWQP9XJ1WL ~ % alias run-help=man which-command=whence</pre>	This command shows some command shortcuts that the user can use.
17. type vi	<pre>giecenar@M-FVFWQP9XJ1WL ~ % type vi vi is /usr/bin/vi</pre>	It tells what kind of command vi is
18. cd /bin	<pre>giecenar@M-FVFWQP9XJ1WL ~ % cd /bin giecenar@M-FVFWQP9XJ1WL /bin %</pre>	This command changes the working directory to /bin, where system commands are stored.
19. type vlc	<pre>giecenar@M-FVFWQP9XJ1WL /bin % type vlc vlc not found</pre>	The command checks if "vlc" is installed.
20. cd	<pre>giecenar@M-FVFWQP9XJ1WL /bin % cd giecenar@M-FVFWQP9XJ1WL ~ %</pre>	This command moves back to your home directory.
21. echo Today is `date`	<pre>giecenar@M-FVFWQP9XJ1WL ~ % echo Today is `date` Today is Thu Oct 23 21:23:27 PST 2025</pre>	The command displays the exact date
22. echo Today is \$(date)	<pre>giecenar@M-FVFWQP9XJ1WL ~ % echo Today is \$(date) Today is Thu Oct 23 21:23:43 PST 2025</pre>	This command is similar to the first one but instead of using backticks, it uses "\$()".
23. echo This is the command "date"	<pre>giecenar@M-FVFWQP9XJ1WL ~ % echo This is the command "date" This is the command date</pre>	The command displays the actual text and the word inside of the quotation quotes.
24. echo This is the command `date`	<pre>giecenar@M-FVFWQP9XJ1WL ~ % echo This is the command `date` This is the command 'date'</pre>	Displays the exact text but it escaped the single quotes.
25. echo This is the command "date"	<pre>giecenar@M-FVFWQP9XJ1WL ~ % echo This is the command "\"date\"" This is the command "date"</pre>	Displays the same message using escaped double quotes.
26. echo D*	<pre>giecenar@M-FVFWQP9XJ1WL ~ % echo D* Desktop Documents Downloads</pre>	This command shows all files and folders that start with the letter "D".
27. echo "D"	<pre>giecenar@M-FVFWQP9XJ1WL ~ % echo "D*" D*</pre>	Displays the text "D*" literally, without showing files.
28. echo Hello; echo Linux; echo Student	<pre>giecenar@M-FVFWQP9XJ1WL ~ % echo Hello; echo Linux; echo Student Hello Linux Student</pre>	This command runs multiple echo commands on the same line and not next to each other using the semicolon symbol

29. false; echo Not; echo Conditional	<pre>giecenar@M-FVFWQP9XJ1WL ~ % false; echo Not; echo Conditional Not Conditional</pre>	This command has a false command that returns failure, but still prints the next echoes.
30. echo Start && echo Going && echo Gone	<pre>giecenar@M-FVFWQP9XJ1WL ~ % echo Start &amp;&amp; echo Going &amp;&amp; echo Gone Start Going Gone</pre>	This command allows the user to display each command only if the previous one was successful.
31. echo Success && false && echo Bye	<pre>giecenar@M-FVFWQP9XJ1WL ~ % echo Success &amp;&amp; false &amp;&amp; echo Bye Success</pre>	The command displays "Success" then stops since false fails. "Bye" won't show.
32. false    echo Fail Or	<pre>giecenar@M-FVFWQP9XJ1WL ~ % false    echo Fail Or Fail Or</pre>	This command has a false command so it displays nothing, but continues to display the next command only if the first one fails.
33. true    echo Nothing to see here	<pre>giecenar@M-FVFWQP9XJ1WL ~ % true    echo Nothing to see here giecenar@M-FVFWQP9XJ1WL ~ %</pre>	This command did not display any text because it has a "true" command and the "  " only makes the second command run if the first one fails.
34. printenv	<pre>giecenar@M-FVFWQP9XJ1WL ~ % printenv __CFBundleIdentifier=com.apple.Terminal TMPDIR=/var/folders/8f/3w19_98j4k9cjdjwk64qxg0000gn/T/ XPC_FLAGS=0x0 LaunchInstanceID=692E37FF-6438-452C-B25A-097B865D8A1F TERM=xterm-256color SSH_AUTH_SOCK=/private/tmp/com.apple.launchd.e0wk6dKlpd/Listeners SECURITYSESSIONID=186a3 XPC_SERVICE_NAME=0 TERM_PROGRAM=Apple_Terminal TERM_PROGRAM_VERSION=443 TERM_SESSION_ID=DC82C5CE-B98A-4AC6-8248-AC55FAC7E962 SHELL=/bin/zsh HOME=/Users/giecenar LOGNAME=giecenar USER=giecenar PATH=/usr/local/bin:/usr/bin:/bin:/usr/sbin:/sbin SHLVL=1 PWD=/Users/giecenar OLDPWD=/bin LC_CTYPE=UTF-8 _/usr/bin/printenv</pre>	This command shows all environment variables in your system.
35. printenv TERM	<pre>giecenar@M-FVFWQP9XJ1WL ~ % printenv TERM xterm-256color</pre>	The command displays the terminal type
36. echo \$TERM	<pre>giecenar@M-FVFWQP9XJ1WL ~ % echo \$TERM xterm-256color</pre>	This command also shows the terminal type, but uses a variable.
37. env	<pre>giecenar@M-FVFWQP9XJ1WL ~ % env __CFBundleIdentifier=com.apple.Terminal TMPDIR=/var/folders/8f/3w19_98j4k9cjdjwk64qxg0000gn/T/ XPC_FLAGS=0x0 LaunchInstanceID=692E37FF-6438-452C-B25A-097B865D8A1F TERM=xterm-256color SSH_AUTH_SOCK=/private/tmp/com.apple.launchd.e0wk6dKlpd/Listeners SECURITYSESSIONID=186a3 XPC_SERVICE_NAME=0 TERM_PROGRAM=Apple_Terminal TERM_PROGRAM_VERSION=443 TERM_SESSION_ID=DC82C5CE-B98A-4AC6-8248-AC55FAC7E962 SHELL=/bin/zsh HOME=/Users/giecenar LOGNAME=giecenar USER=giecenar PATH=/usr/local/bin:/usr/bin:/bin:/usr/sbin:/sbin SHLVL=1 PWD=/Users/giecenar OLDPWD=/bin LC_CTYPE=UTF-8 _/usr/bin/env</pre>	This command shows all environment variables and their current values.

## 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.

```
giecenar@M-FVFWQP9XJ1WL ~ % alias hi='echo Hello, this is an alias'
giecenar@M-FVFWQP9XJ1WL ~ % hi
Hello, this is an alias
```

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.

```
giecenar@M-FVFWQP9XJ1WL ~ % cd ~/Documents
giecenar@M-FVFWQP9XJ1WL Documents % mkdir CPE_201A_Cenar
giecenar@M-FVFWQP9XJ1WL Documents % touch CPE_201A_Cenar/sample1_Cenar.txt
giecenar@M-FVFWQP9XJ1WL Documents % cd CPE_201A_Cenar && ls
sample1_Cenar.txt
giecenar@M-FVFWQP9XJ1WL CPE_201A_Cenar %
```

3. Execute a command to display the working shell.

```
giecenar@M-FVFWQP9XJ1WL ~ % echo $SHELL
/bin/zsh
```

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?

```
giecenar@M-FVFWQP9XJ1WL ~ % echo P*
Pictures Public
```

## 7. Conclusion:

In this activity, I learned how different types of commands work in the terminal and how it helps the user to know certain commands that can help out a lot and control the system. I learned that some commands or symbols are important in certain texts and commands for them to run, for example the "|", it can be used to run commands based on their success or failure depending on what the user inputted. This activity improved my understanding of how the terminal interprets commands and executes them in sequence. Overall, I was able to explore basic terminal functions of my macbook.

## 8. Assessment (Rubric for Laboratory Performance):

TIP Rubric E (1) (1)			
Criteria	Ratings		Pts
Performance Indicators 1. Apply appropriate techniques, skills, and modern tools to perform a discipline-specific engineering task.	<b>4 pts</b> <b>Very Satisfactory</b> Applies the most appropriate modern technique in performing discipline-specific engineering task exceeding the requirements.	<b>0 pts</b> <b>No Marks</b>	4 pts
Performance Indicators 2. Demonstrate skills in applying different techniques and modern tools to solve engineering problems.1. Apply appropriate techniques, skills, and modern tools to perform a discipline-specific engineering task.	<b>4 pts</b> <b>Very Satisfactory</b> Applies the most appropriate modern technique in performing discipline-specific engineering task exceeding the requirements.	<b>0 pts</b> <b>No Marks</b>	4 pts
Performance Indicators 3. Recognize the benefits and constraints of modern engineering tools.Demonstrate skills in applying different techniques and modern tools to solve engineering problems.1. Apply appropriate techniques, skills, and modern tools to perform a discipline-specific engineering task.	<b>4 pts</b> <b>Very Satisfactory</b> Applies the most appropriate modern technique in performing discipline-specific engineering task exceeding the requirements.	<b>0 pts</b> <b>No Marks</b>	4 pts
			Total Points: 12