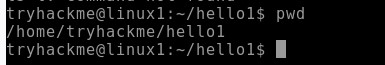
**Linux Lab -2**

**Basic Linux Command**

# 1 Pwd command :-

Displays the current working directory.

Syntax :- **pwd**



# Ls command :-

Displays information about files in the current directory.

Syntax :- **ls**



**3. Mkdir command :-**

This [mkdir command](https://www.geeksforgeeks.org/mkdir-command-in-linux-with-examples/) allows you to create fresh directories in the terminal itself.

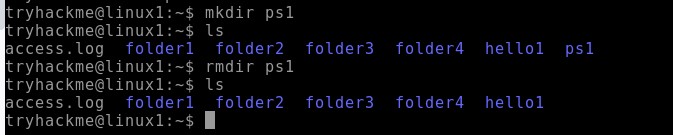
# Syntax :- mkdir <directory name>



**4 Rmdir command :-**

The [rmdir command](https://www.geeksforgeeks.org/rmdir-command-in-linux-with-examples/) is used to delete permanently an empty directory.

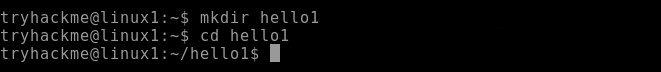
# Syntax :- rmdir <directory name>



**5. Cd command :-**

The [cd command](https://www.geeksforgeeks.org/cd-command-in-linux-with-examples/) is used to navigate between directories.

# Syntax :- cd <directory name>



**6. Cat command :-**

Display file contents on terminal

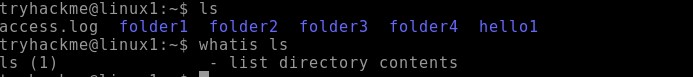
# Syntax :- cat <file name>



**7. Whatis command :-**

whatis command in Linux is used to get a one-line manual page description.

# Syntax :- whatis [option] [command\_name]



# 8. Find command :-

The find command in Linux is a dynamic utility designed for comprehensive file and directory searches within a hierarchical structure.

# Syntax :- find [path] [options] [expression]



**9. Echo command :-**

[echo command](https://www.geeksforgeeks.org/echo-command-in-linux-with-examples/) in Linux is specially used to print something in the terminal

# Syntax :- echo <Text>



# 10. Cal command :-

The [cal command](https://www.geeksforgeeks.org/cal-command-in-linux-with-examples/) is not the most famous command in the terminal but it functions to view the calendar for a particular month in the terminal. Let’s see how this works.

# Syntax :- cal <month> <Year>



# 11. Date command :-

Datecommand is used to display the system date and time. date command is also used to set date and time of the system.

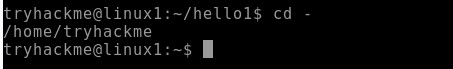
Syntax :- **date**



# 12. Cd - command :-

**Cd -** command is used to go back to previous directory in terminal of the system.

Syntax :- **cd –**



# 13. Cp command :-

The cp command copies files or directories from a source to a destination. It can handle single or multiple files and directories, and it can also overwrite existing files if specified.

# Syntax: cp [options] source destination

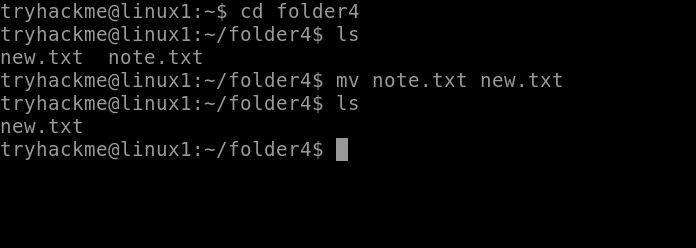
Example:



# 14. mv

The mv command in Linux is used to move or rename files and directories.

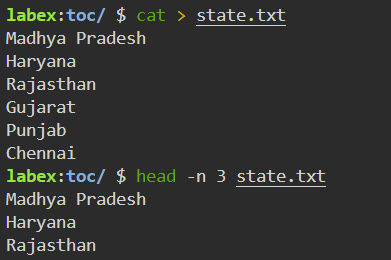
Syntax: **mv [options] source destination** Example:



# 15. head

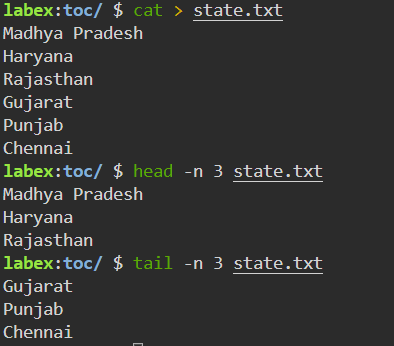
The head command outputs the first part of files or input data. It is commonly used to preview the beginning of a file or stream.

# Syntax: head [options] [file...]



# 16. tail

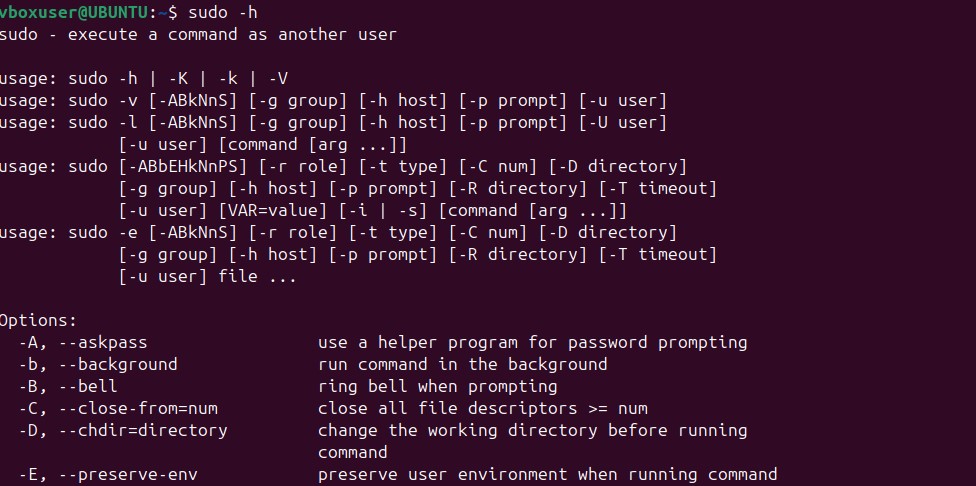
The tail command outputs the last part of files or input data. It is often used to view the most recent entries in a log file or to monitor the end of a file for changes. Syntax: **tail [options] [file...]** Example :



# 17. sudo

The sudo command grants elevated privileges to run commands that require root or administrative permissions. It's typically used to perform system administration tasks. Syntax: **sudo [options] command**

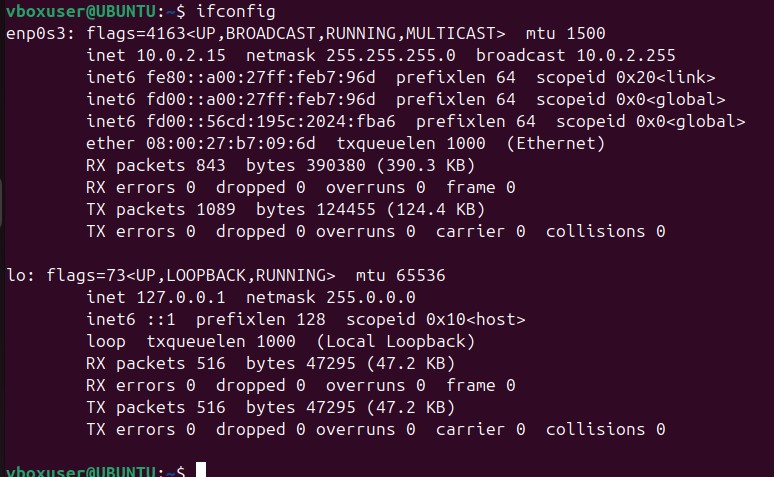
Example:



# 18. ifconfig

The ifconfig (interface configuration) command is used to display or configure a network interface.

Syntax: **ifconfig [interface] [options]** Example:

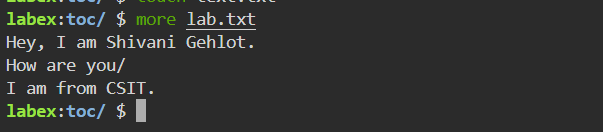


# 19. more

The more command displays the contents of a file, pausing after each screen of text. It is useful for viewing long files that don't fit on a single screen.

# Syntax: more [options] [file]

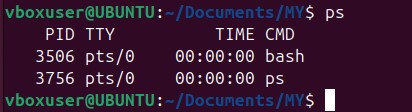
Example:



# 20. ps

The ps command provides a snapshot of current processes, showing details like process IDs (PIDs), terminal associated with the process, CPU and memory usage, and the command that started the process.

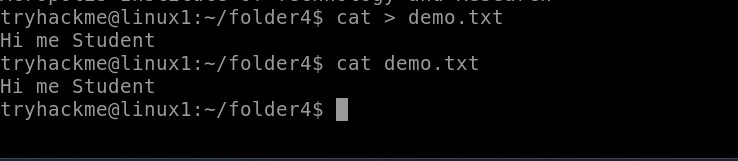
Syntax: p**s [options]** Example:



# 21. cat >

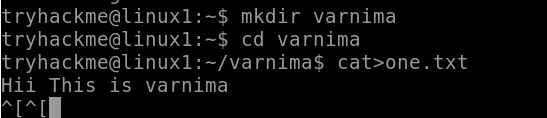
Using cat > filename, you can start typing text directly into a new file. This command redirects the terminal input into the specified file until you signal that you're done. Syntax: **cat > filename**

Example:



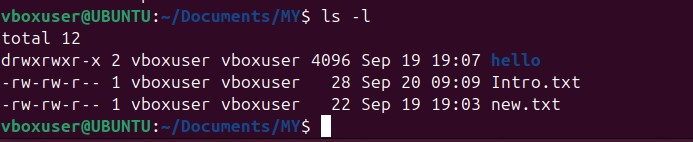
# 22. cat >>

Using cat >> filename, you can add new content to the end of a specified file. This command allows you to continue writing to the file without overwriting its current contents. Syntax: **cat >> filename**



# 23. ls -l

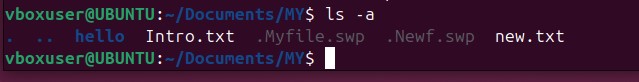
The ls -l command lists files and directories in a long format, showing detailed attributes for each item, including permissions, number of links, owner, group, size, and modification date. Syntax: **ls -l [directory]** Example:



# 24. ls -a

The ls -a command displays all entries in a directory, including those that begin with a dot (.), which are considered hidden files in Unix-like systems.

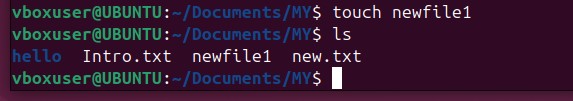
Syntax: **ls -a [directory]** Example:



# 25. touch

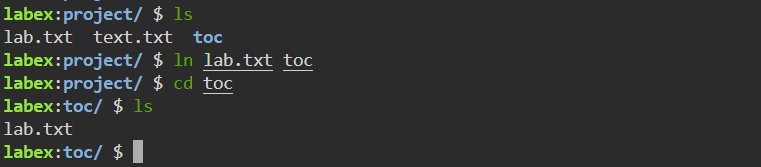
The touch command creates a new, empty file if the specified file does not exist. If the file already exists, it updates the access and modification timestamps to the current time without modifying the file's content.

# Syntax: touch [options] filename



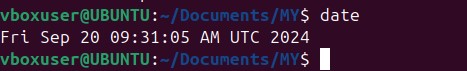
# 26. ln

The ln command in Linux is used to create links between files. There are two types of links: hard links and symbolic (soft) links. Syntax: **ln [options] target [link\_name]** Example:



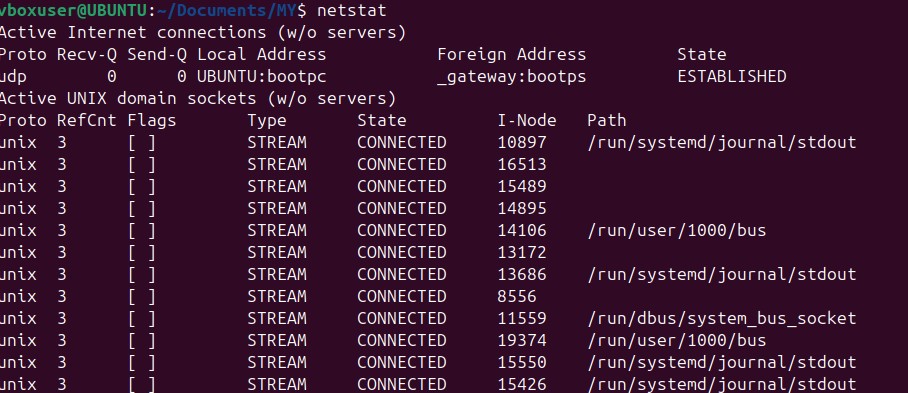
# 27. date

The date command in Linux is used to display or set the system date and time. Syntax: **date [options] [+format]** Example:



# 28. netstat

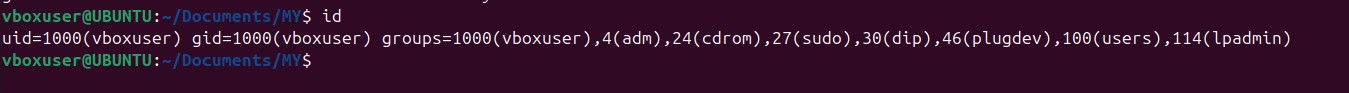
The netstat command provides information about active network connections and network interface statistics, helping users monitor and troubleshoot network issues. Syntax: **netstat [options]** Example:



# 29. gid

In Linux, GID stands for Group Identifier. It is a numeric value used to identify a specific group on the system. Each user in Linux can belong to one or more groups, and each group is assigned a unique GID. Syntax: **id username**

Example:



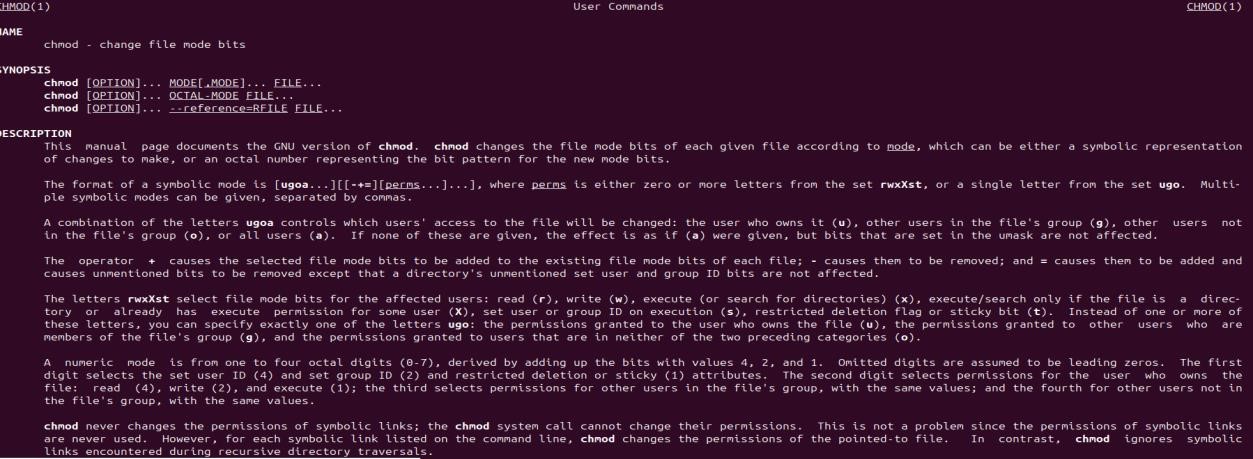
# 30. chmod

The chmod command allows users to specify who can read, write, or execute a file. Permissions can be set for three categories: the file owner, the group, and others. Syntax: **chmod [options] mode file** Example:



# 31. man

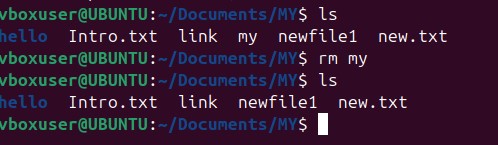
The man command is a built-in command that allows users to access the manual documentation for commands, functions, system calls, and other components in Linux. Syntax: **man [options] command** Example :



# 32. rm

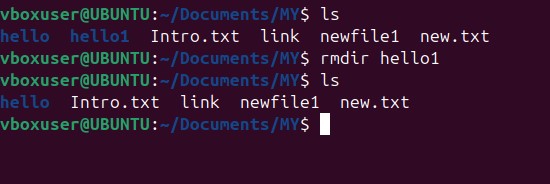
The rm command allows users to delete files and directories from the filesystem. It is a powerful command that permanently removes files without placing them in a recycle bin or trash.

Syntax: **rm [options] file** Example:



# 33. rmdir

The rmdir command allows users to delete directories, but it can only remove those that are empty. If the directory contains files or other directories, the command will fail. Syntax: **rmdir [options] directory** Example:



# 34. less

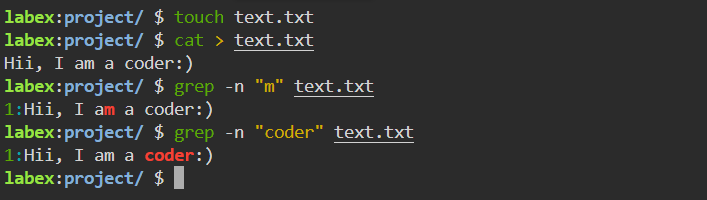
The less command provides a convenient way to scroll through text files, allowing both forward and backward navigation.

# Syntax: less [options] file

Example:



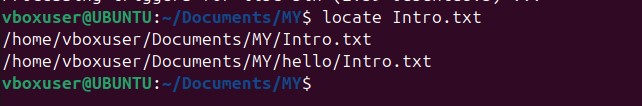
# 35. grep

The grep command searches through the input (files or standard input) and prints lines that match a specified pattern. It's commonly used for text processing and searching logs. Syntax: **grep [options] pattern [file...]** Example: 

# 36. locate

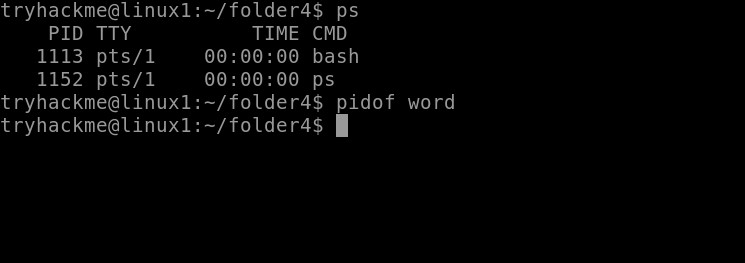
The locate command searches for files and directories in a database that contains the paths of all files on the system. This database is typically updated daily by a background service (updatedb), allowing for fast searches.

Syntax: **locate [options] pattern** Example:



# 37. pid

The PID (process identification number) is a serial number (starting from 1) given by the operating system to uniquely identify the process. Every process started either by the operating system or by the user gets a PID in order of their invocation by the kernel. Syntax: **pidof <exact\_process\_name>** Example:

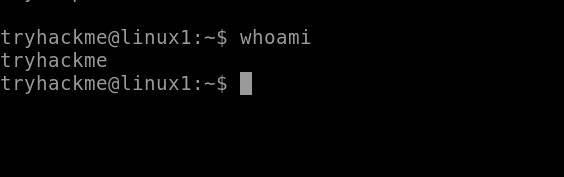


# 38. whoami

The command allows Linux users to see the currently logged-in user. The output displays the username of the effective user in the current shell. Additionally, whoami is useful in bash scripting to show who runs the script

# Syntax : whoami [OPTION]

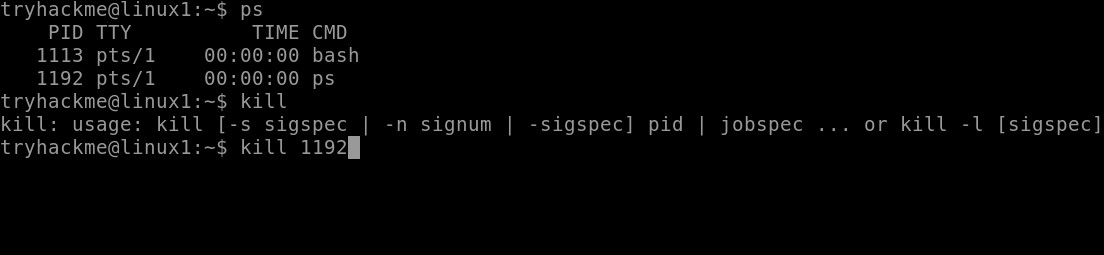
Example :



**39. kill** kill command in Linux (located in /bin/kill), is a built-in command which is used to terminate processes manually

# Syntax : kill [signal] PID

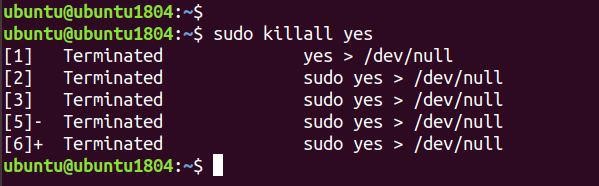
Example :



# 40. kill all

The killall command in Linux is a utility that terminates running processes based on their name. It can be useful when you need to kill multiple instances of a process or when you don't know the process ID (PID).

Syntax : **killall** [ **-** ] [ **-signal** ] Example :

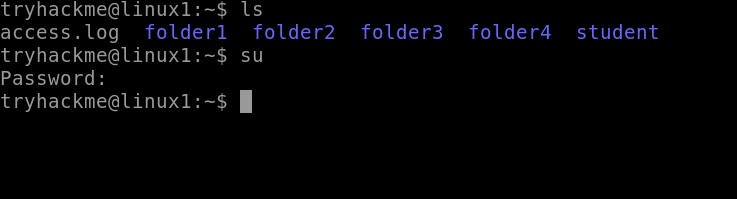


# 41. su

The su command in Linux switches users or executes commands as a different user. It's useful for administrative tasks that require elevated privileges.

# Syntax : su [options] [username]

Example :



# 42. wc

wc stands for **word count**. As the name implies, it is mainly used for counting purpose.It is used to find out **number of lines**, **word count**, **byte and characters count** in the files specified in the file arguments.

Syntax : **wc [option]... [file]...**

Example :

