UNIVERSITI TENAGA NASIONAL

COLLEGE OF COMPUTING AND INFORMATICS

CMPD393 – SYSTEM ADMINISTRATION

SEMESTER 1, YEAR 2023/2024

GROUP ASSIGNMENT: LINUX COMMAND

SECTION: 01A

LECTURER'S NAME: SIR AHMAD IKHSAN BIN ABD AZIZ

SUBMISSION DATE: 14TH JANUARY 2024

GROUP NO: GROUP 3

GROUP MEMBERS:

NO	NAME	ID
1	IZZA AMIRA BINTI SUHAIRI	DC98788
2	SITI FATIMAH ZAHRA BINTI ABD RAHMAN	DC98883
3	MUHIIN RAO A/L JEGATIS RAO	DC98860
4	NISA MITHALINA BINTI MOHD NAZRI	DC98891
5	MUHAMMAD HAIZAD BIN ROSLI	DC98817

Linux Command: du

(Prepared by: Izza Amira binti Suhairi - DC98788)

Background/ Function:

The function of the Linux “du” command is to estimate file space usage, showing the disk space occupied by files and directories. It represents "disk usage." When you run du, it calculates the size of each file and directory and outputs a summary of the disk space they take by recursively walking around the directory tree from the given location (or the current directory if none is provided).

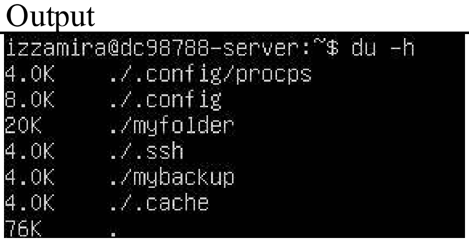
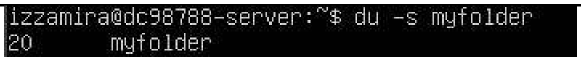
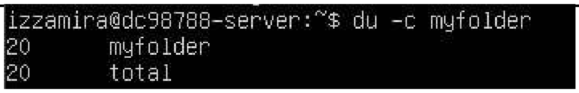
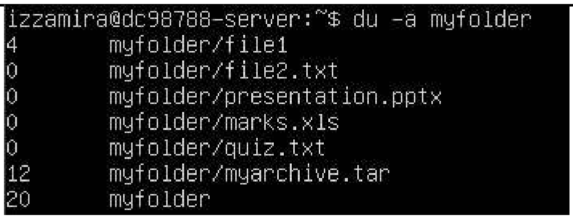
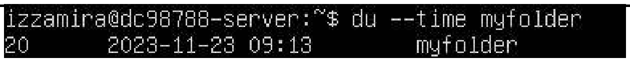
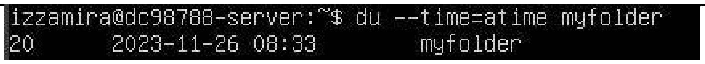
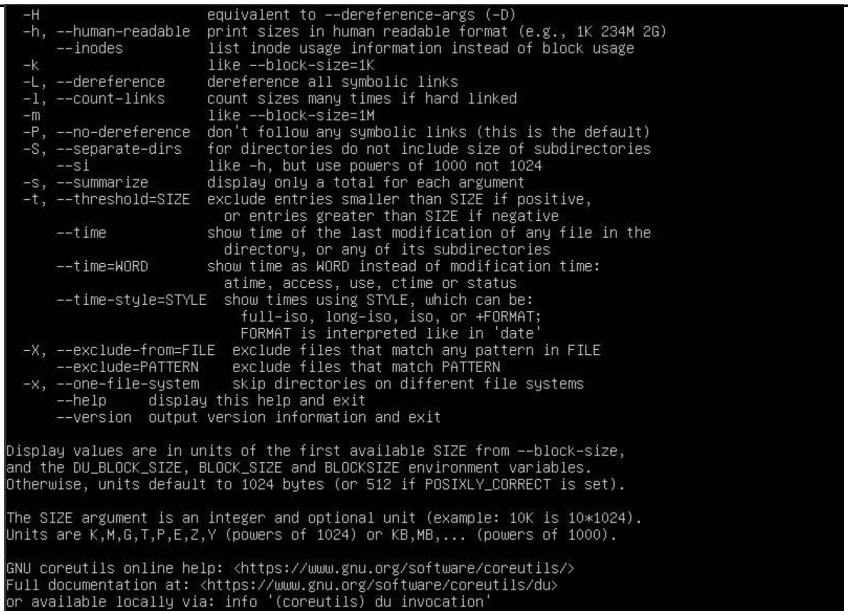
How to Use/ Full Command:

Basic syntax for ‘du’:

- du [options]
- du [options] [directories/files]

Option	Usage	Description
‘-h’	du ***-h***	Human-readable output. Print sizes in a more easily readable format (e.g., KB, MB, GB).
‘-s’	du ***-s [directory]***	Display only the total for each specified directory.
‘-c’	du ***-c [directory]***	Produce a grand total at the end of the listing.
‘-a’	du ***-a [directory]***	Display an entry for each file in the specified directory.
‘--time’	du ***--time [directory]***	Show the time of the last modification of any file in the directory.
‘--time=atime’	du ***--time=atime [directory]***	Show the time of the last access of any file in the directory.
‘--help’	du ***--help***	Display help information about ‘du’.

Sample Output:

Option	Output
'-h'	 izzamira@dc98788-server:~$ du -h 4.0K ./.config/procps 8.0K ./.config 20K ./myfolder 4.0K ./.ssh 4.0K ./mybackup 4.0K ./.cache 76K .
'-s'	 izzamira@dc98788-server:~$ du -s myfolder 20 myfolder
'-c'	 izzamira@dc98788-server:~$ du -c myfolder 20 myfolder 20 total
'-a'	 izzamira@dc98788-server:~$ du -a myfolder 4 myfolder/file1 0 myfolder/file2.txt 0 myfolder/presentation.pptx 0 myfolder/marks.xls 0 myfolder/quiz.txt 12 myfolder/myarchive.tar 20 myfolder
'--time'	 izzamira@dc98788-server:~$ du --time myfolder 20 2023-11-23 09:13 myfolder
'--time=atime'	 izzamira@dc98788-server:~$ du --time=atime myfolder 20 2023-11-26 08:33 myfolder
'--help'	 -H equivalent to --dereference-args (-D) -h, --human-readable print sizes in human readable format (e.g., 1K 234M 2G) --inodes list inode usage information instead of block usage -k like --block-size=1K -L, --dereference dereference all symbolic links -l, --count-links count sizes many times if hard linked -m like --block-size=1M -P, --no-dereference don't follow any symbolic links (this is the default) -S, --separate-dirs for directories do not include size of subdirectories --si like -h, but use powers of 1000 not 1024 -s, --summarize display only a total for each argument -t, --threshold=SIZE exclude entries smaller than SIZE if positive, or entries greater than SIZE if negative --time show time of the last modification of any file in the directory, or any of its subdirectories --time=WORD show time as WORD instead of modification time: atime, access, use, ctime or status --time-style=STYLE show times using STYLE, which can be: full-iso, long-iso, iso, or +FORMAT; FORMAT is interpreted like in 'date' -X, --exclude-from=FILE exclude files that match any pattern in FILE --exclude=PATTERN exclude files that match PATTERN -x, --one-file-system skip directories on different file systems --help display this help and exit --version output version information and exit Display values are in units of the first available SIZE from --block-size, and the DU_BLOCK_SIZE, BLOCK_SIZE and BLOCKSIZE environment variables. Otherwise, units default to 1024 bytes (or 512 if POSIXLY_CORRECT is set). The SIZE argument is an integer and optional unit (example: 10K is 10*1024). Units are K,M,G,T,P,E,Z,Y (powers of 1024) or KB,MB,... (powers of 1000). GNU coreutils online help: <https://www.gnu.org/software/coreutils/> Full documentation at: <https://www.gnu.org/software/coreutils/du> or available locally via: info '(coreutils) du invocation'

Linux Command: cp

(Prepared by: Siti Fatimah Zahra - DC98883)

Background/ Function:

The “cp” command is the fundamental tool in Linux for copying files and directories. It allows you to duplicate data efficiently, manage your file system organization, and even handle entire directory structure. By specifying source and destination paths, you can copy both individual files and entire directory structures with various options like recursive copying, preserving symbolic links and file attributes, and handling hidden files.

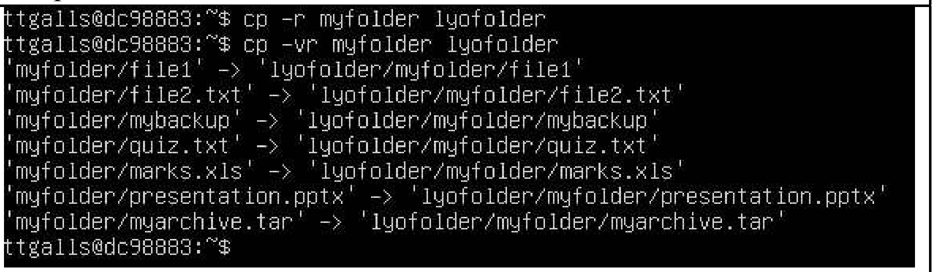
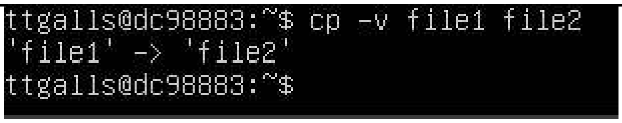
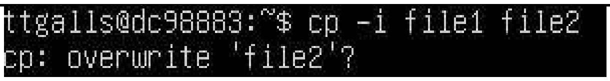
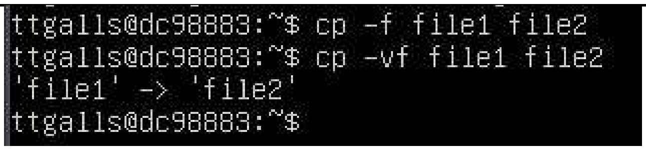
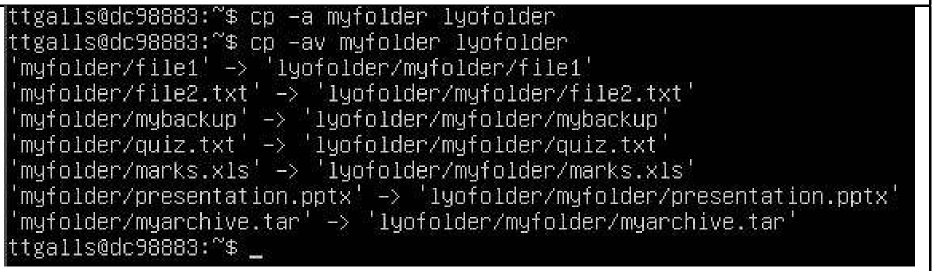
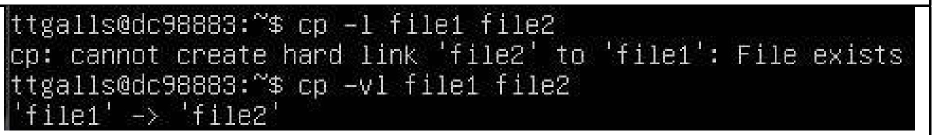
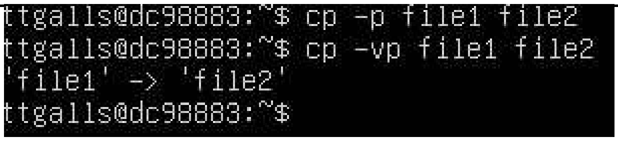
How to Use/ Full Command:

Basic syntax for ‘cp’:

- cp [options]
- cp [options] [directories/files]

Option	Usage	Description
‘-r’	***cp-r [directory]***	Copies directories recursively, including all subdirectories and files.
‘-v’	***cp-v [file]***	Displays detailed information about the copying process.
‘-i’	***cp-i [file]***	Prompts for confirmation before overwriting existing files.
‘-f’	***cp-f [file]***	Forces the copy operation, even if the destination file already exists.
‘-a’	***cp-a [directory]***	Copy all files includes hidden files starting with a dot (.).
‘-l’	***cp-l [file]***	Copies symbolic links as links instead of dereferencing them.
‘-p’	***cp-p [file]***	Copies file attributes like permissions, timestamps, and ownership.

Sample Output:

Option	Output
‘-r’	 ttgalls@dc98883:~$ cp -r myfolder lyofolder ttgalls@dc98883:~$ cp -vr myfolder lyofolder 'myfolder/file1' -> 'lyofolder/myfolder/file1' 'myfolder/file2.txt' -> 'lyofolder/myfolder/file2.txt' 'myfolder/mybackup' -> 'lyofolder/myfolder/mybackup' 'myfolder/quiz.txt' -> 'lyofolder/myfolder/quiz.txt' 'myfolder/marks.xls' -> 'lyofolder/myfolder/marks.xls' 'myfolder/presentation.pptx' -> 'lyofolder/myfolder/presentation.pptx' 'myfolder/myarchive.tar' -> 'lyofolder/myfolder/myarchive.tar' ttgalls@dc98883:~$
‘-v’	 ttgalls@dc98883:~$ cp -v file1 file2 'file1' -> 'file2' ttgalls@dc98883:~$
‘-i’	 ttgalls@dc98883:~$ cp -i file1 file2 cp: overwrite 'file2'?
‘-f’	 ttgalls@dc98883:~$ cp -f file1 file2 ttgalls@dc98883:~$ cp -vf file1 file2 'file1' -> 'file2' ttgalls@dc98883:~$
‘-a’	 ttgalls@dc98883:~$ cp -a myfolder lyofolder ttgalls@dc98883:~$ cp -av myfolder lyofolder 'myfolder/file1' -> 'lyofolder/myfolder/file1' 'myfolder/file2.txt' -> 'lyofolder/myfolder/file2.txt' 'myfolder/mybackup' -> 'lyofolder/myfolder/mybackup' 'myfolder/quiz.txt' -> 'lyofolder/myfolder/quiz.txt' 'myfolder/marks.xls' -> 'lyofolder/myfolder/marks.xls' 'myfolder/presentation.pptx' -> 'lyofolder/myfolder/presentation.pptx' 'myfolder/myarchive.tar' -> 'lyofolder/myfolder/myarchive.tar' ttgalls@dc98883:~$ _
‘-l’	 ttgalls@dc98883:~$ cp -l file1 file2 cp: cannot create hard link 'file2' to 'file1': File exists ttgalls@dc98883:~$ cp -vl file1 file2 'file1' -> 'file2'
‘-p’	 ttgalls@dc98883:~$ cp -p file1 file2 ttgalls@dc98883:~$ cp -vp file1 file2 'file1' -> 'file2' ttgalls@dc98883:~$

<u>Linux Command: useradd</u>

(Prepared by: Nisa Mithalina binti Mohd Nazri - DC98891)

Background/ Function:

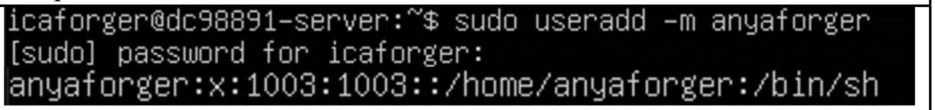
The ‘useradd’ command in Linux is used to add a new user to the system. It’s commonly used with various options to specify details such as the user’s home directory, default shell, user ID(UID). And group ID(GID).

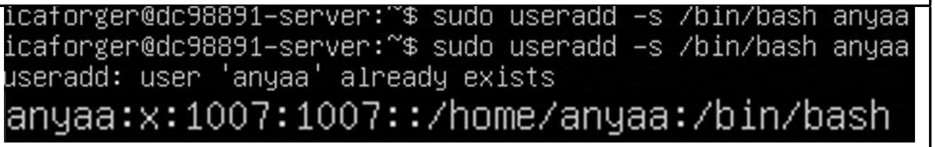
How to Use/ Full Command:

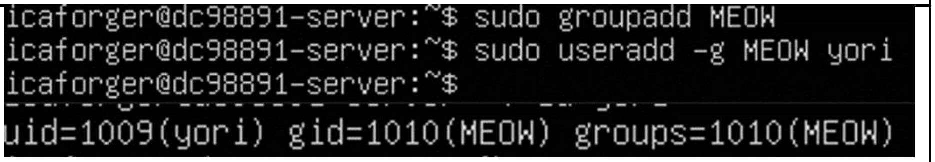
Basic syntax for ‘useradd’: • useradd [options]		
Option	Usage	Description
‘-m’ or ‘--create-home'	***useradd -m***	Ensure the creation of the user’s home directory. If the home directory doesn’t exist, it will be created.
‘-s’ or ‘--shell’	***useradd -s /bin/bash***	Specifies the login shell for the user. The login shell is the program that interprets commands entered by user.
‘-g’ or ‘--gid’	***useradd -g primary_group***	Sets the initial login group of the user to the specified group. The group must already exist.
‘-G’ or ‘--groups’	***useradd -G group1,group2***	Specifies additional groups to which the user should belong. Multiple groups can be specified, separated by commas.
‘-u’ or ‘--uid’	***Useradd –u (e.g., 1001)***	Sets the user ID for the new user. The UID is a unique numerical identifier assigned to each other.

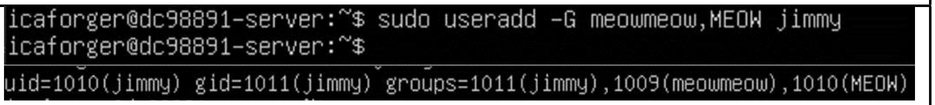
Sample Output:

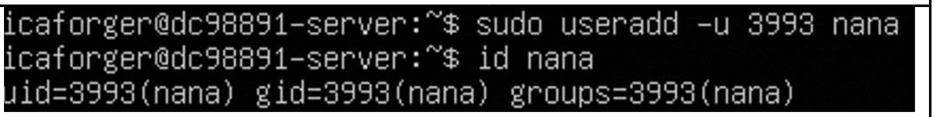
Option	Output

‘-m’	
	`icaforger@dc98891-server:~$ sudo useradd -m anyaforger`
	`[sudo] password for icaforger:`
	`anyaforger:x:1003:1003::/home/anyaforger:/bin/sh`

‘-s’	
	`icaforger@dc98891-server:~$ sudo useradd -s /bin/bash anyaa`
	`icaforger@dc98891-server:~$ sudo useradd -s /bin/bash anyaa`
	`useradd: user 'anyaa' already exists`
	`anyaa:x:1007:1007::/home/anyaa:/bin/bash`

‘-g’	
	`icaforger@dc98891-server:~$ sudo groupadd MEOW`
	`icaforger@dc98891-server:~$ sudo useradd -g MEOW yori`
	`icaforger@dc98891-server:~$`
	`uid=1009(yori) gid=1010(MEOW) groups=1010(MEOW)`

‘-G’	
	`icaforger@dc98891-server:~$ sudo useradd -G meowmeow,MEOW jimmy`
	`icaforger@dc98891-server:~$`
	`uid=1010(jimmy) gid=1011(jimmy) groups=1011(jimmy),1009(meowmeow),1010(MEOW)`

‘-u’	
	`icaforger@dc98891-server:~$ sudo useradd -u 3993 nana`
	`icaforger@dc98891-server:~$ id nana`
	`uid=3993(nana) gid=3993(nana) groups=3993(nana)`

Linux Command: cat

(Prepared by: Muhammad Haizad Bin Rosli- DC98817)

Background/ Function:

- Cat is short for concatenate. This command displays the contents of one or more files without having to open the file for editing.
- Concatenation is the process of joining two or more files together.
- The cat command displays the contents of a file in the terminal window.
- Concatenating files, numbering lines, and displaying non-printing characters.

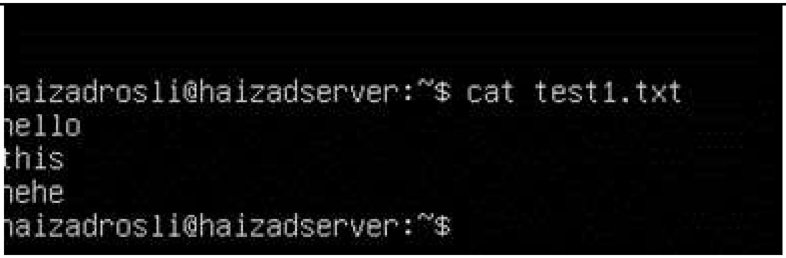
How to Use/ Full Command:

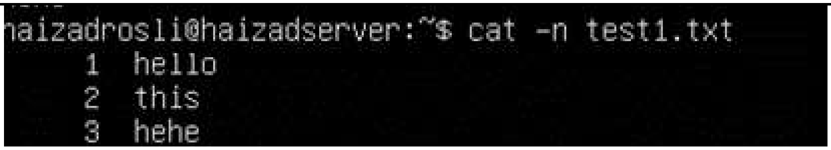
Basic syntax for 'cat':

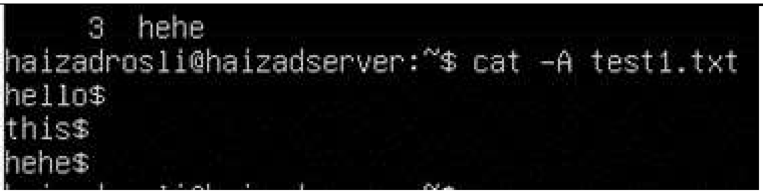
- cat [options] filename(s)

Option	Usage	Description
'-n'	***cat-n file.txt***	Number all output lines.
'-A'	***cat-A file.txt***	Show all characters, including non-printing characters.
'file.txt'	***cat file.txt***	Display the content of a file.

Sample Output:

Option	Output
'test1.txt'	 haizadrosli@haizadserver:~$ cat test1.txt hello this hehe haizadrosli@haizadserver:~$

'-n'	 haizadrosli@haizadserver:~$ cat -n test1.txt
	     1 hello
	     2 this
	     3 hehe

'-A'	 3 hehe haizadrosli@haizadserver:~$ cat -A test1.txt hello$ this$ hehe$

Linux Command: rm

(Prepared by: MUHIIN RAO A/L JEGATIS RAO-DC98860)

Background/function:

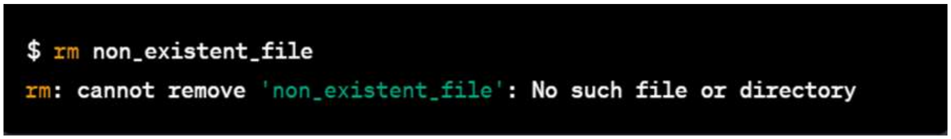
The rm command in Linux is used to remove or delete files and directories. rm command is irreversible, and deleted files are usually not recoverable unless you have a backup.

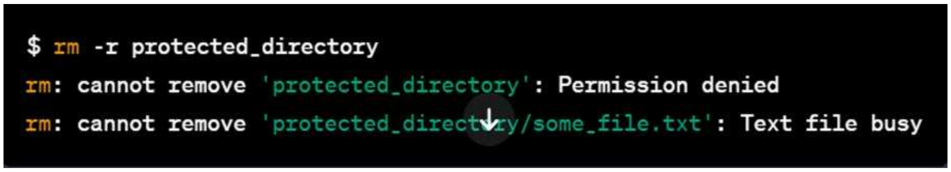
- Remove Files: Deletes specified files from the filesystem.
- Remove Multiple Files: Allows deletion of multiple files in a single command.
- Remove Directories: Deletes specified directories and their contents.
- Recursive Deletion: When used with the -r or -R option, removes directories and their contents recursively.

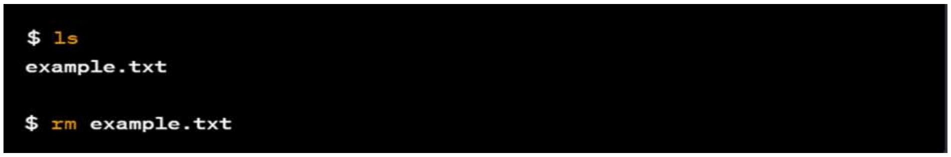
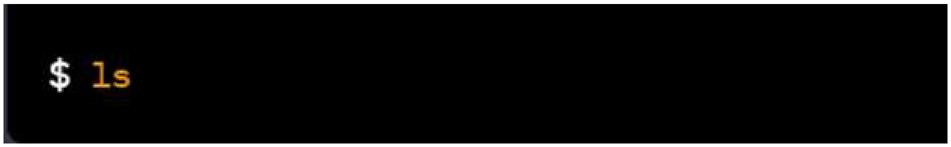
How to Use/ Full Command:

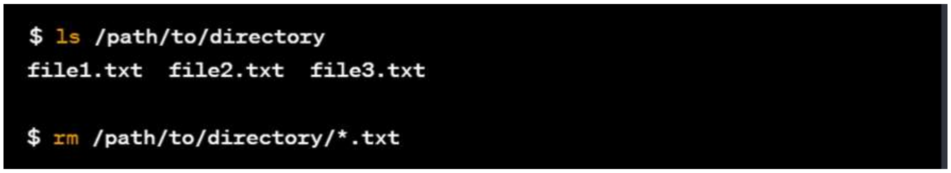
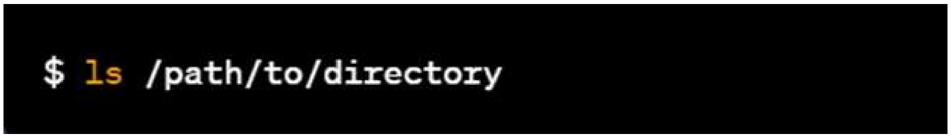
<table>
<tr><td colspan="3">Basic syntax for ‘rm’:
• rm[options] filename(s)</td></tr>
<tr><th>Option</th><th>Usage</th><th>Description</th></tr>
<tr><td>‘Filename’</td><td><b><i>rm filename</i></b></td><td>Remove a file</td></tr>
<tr><td>‘-r directory_name’</td><td><b><i>rm -r directory_name</i></b></td><td>Remove a directory and its contents recursively:</td></tr>
<tr><td>‘-v filename’</td><td><b><i>rm -v filename</i></b></td><td>Provides detailed information about each file being removed.</td></tr>
<tr><td>‘/path/to/directory/*.txt’</td><td><b><i>rm /path/to/directory/*.txt</i></b></td><td>Removes all files with the .txt extension in the specified directory and its subdirectories.</td></tr>
<tr><td>‘-d empty_directory’</td><td><b><i>rm -d empty_directory</i></b></td><td>Removes the specified empty directory.</td></tr>
</table>

Sample Output:

Option	Output
‘Filename’	 `$ rm non_existent_file` `rm: cannot remove 'non_existent_file': No such file or directory`

‘-r directory_name’	 `$ rm -r protected_directory`
	`rm: cannot remove 'protected_directory': Permission denied`
	`rm: cannot remove 'protected_directory/some_file.txt': Text file busy`

‘-v filename’	 `$ ls`
	`example.txt`
	`$ rm example.txt`
	 `$ ls`

‘/path/to/directory/*.txt’	 `$ ls /path/to/directory`
	`file1.txt  file2.txt  file3.txt`
	`$ rm /path/to/directory/*.txt`
	 `$ ls /path/to/directory`