

Experiment #	<TO BE FILLED BY STUDENT>	Student ID	<TO BE FILLED BY STUDENT>
Date	<TO BE FILLED BY STUDENT>	Student Name	[@KLWKS_BOT THANOS]

Experiment Title: Basic Commands with Examples

Aim/Objective:

The aim of learning basic commands with examples in operating systems is to familiarize, oneself with the fundamental commands and their functionalities. These commands allow users to interact with the operating system and perform various tasks efficiently.

Description:

Basic commands in operating systems refer to the essential commands used to interact with the operating system via the command-line interface (CLI) or terminal. These commands allow users to perform various tasks, such as navigating directories, managing files and processes, accessing system information, and configuring system settings.

Pre-Requisites:

- general idea of what an Operating System is (an interface between User and Hardware)
- How do users communicate with the hardware? (Using commands)
- What is a Shell?
- How commands can be executed (through Shell)?

Pre-Lab:

1. What is the purpose of following commands

BASIC COMMANDS	FUNCTIONALITY
Man	Displays manual pages for commands.
Who	Shows who is logged in.
pwd	Prints the current working directory.

Course Title	OPERATING SYSTEMS	ACADEMIC YEAR: 2024-25
Course Code(s)	23CS2104A	Page 1 of 226

Experiment #	<TO BE FILLED BY STUDENT>	Student ID	<TO BE FILLED BY STUDENT>
Date	<TO BE FILLED BY STUDENT>	Student Name	[@KLWKS_BOT THANOS]

mkdir	Creates a new directory.
cat	Concatenates and displays file content.
Touch	Creates an empty file or updates timestamp.
Nano	Opens a terminal-based text editor.
Tar	Archives files into a compressed file.
mv	Moves or renames files or directories.
sort	Sorts lines in a file or input.
grep	Searches for patterns within files.

Course Title	OPERATING SYSTEMS	ACADEMIC YEAR: 2024-25
Course Code(s)	23CS2104A	Page 2 of 226

Experiment #	<TO BE FILLED BY STUDENT>	Student ID	<TO BE FILLED BY STUDENT>
Date	<TO BE FILLED BY STUDENT>	Student Name	[@KLWKS_BOT THANOS]

ls	Lists files and directories.
chmod	Changes file permissions.
Head	Displays the first part of a file.
Date	Displays or sets the system date.
cp	Copies files or directories.
echo	Displays a message or variable value.
cal	Displays the current month's calendar.

Course Title	OPERATING SYSTEMS	ACADEMIC YEAR: 2024-25
Course Code(s)	23CS2104A	Page 3 of 226

Experiment #	<TO BE FILLED BY STUDENT>	Student ID	<TO BE FILLED BY STUDENT>
Date	<TO BE FILLED BY STUDENT>	Student Name	[@KLWKS_BOT THANOS]

In-Lab:

Problem Description:

Stanley wants to get started with terminal commands in Linux.

Help him out toper from the following set of statements:

a. He wants to know the current directory that he is working with, in the system. After identifying the current directory, he desires to create a folder called Marvel.

Ans:

- **Pwd**
- **mkdir Marvel**

b. Now, he wants to list out all the Avengers of the “Marvel” universe. He adds the following set of Avengers to Avengers.txt:

- Ironman
- Captain America
- Thor
- Hulk
- Black widow

Ans:

- **echo -e "Ironman\nCaptain America\nThor\nHulk\nBlack widow" > Avengers.txt**

c. After adding the names displayed above check whether the names are inserted or not.

Ans:

- **cat Avengers.txt**

Course Title	OPERATING SYSTEMS	ACADEMIC YEAR: 2024-25
Course Code(s)	23CS2104A	Page 4 of 226

Experiment #	<TO BE FILLED BY STUDENT>	Student ID	<TO BE FILLED BY STUDENT>
Date	<TO BE FILLED BY STUDENT>	Student Name	[@KLWKS_BOT THANOS]

d. Stanley wants to relocate the file (Avengers.txt) from Marvel to Desktop, after relocating the file give all the permissions to the user and group and give only read permission to others. Verify the permissions when done.

Ans:

- **mv Marvel/Avengers.txt ~/Desktop/**
- **chmod 770 ~/Desktop/Avengers.txt**
- **ls -l ~/Desktop/Avengers.txt**

e. Stanley now wants to add more Avengers to the Marvel, Add the following set of new Avengers to Avengers.txt.

1. Black Panther
2. Groot
3. Captain Marvel
4. Spiderman

Ans:

- **echo -e "Black Panther\nGroot\nCaptain Marvel\nSpiderman" >> ~/Desktop/Avengers.txt**

f. Sort the names of the file in lexicographical order export the result to Sortedavengers.txt and display the content of it.

Ans:

- **sort ~/Desktop/Avengers.txt > ~/Desktop/Sortedavengers.txt**
- **cat ~/Desktop/Sortedavengers.txt**

g. Now, Stanley sends the first Avenger from Sortedavengers.txt to visit Wakanda.txt (another file on the desktop) as a part of a mission to kill Thanos. After sending, move the wakanda.txt to marvel.

Ans:

- **head -n 1 ~/Desktop/Sortedavengers.txt > ~/Desktop/Wakanda.txt**
- **mv ~/Desktop/Wakanda.txt ~/Marvel**

Course Title	OPERATING SYSTEMS	ACADEMIC YEAR: 2024-25
Course Code(s)	23CS2104A	Page 5 of 226

Experiment #	<TO BE FILLED BY STUDENT>	Student ID	<TO BE FILLED BY STUDENT>
Date	<TO BE FILLED BY STUDENT>	Student Name	[@KLWKS_BOT THANOS]

Data and Results:

Data:

Commands for directory navigation, file manipulation, permissions, sorting, and file transfer in Linux.

Results:

Successfully created folders, added Avengers, sorted, transferred files, and set proper permissions.

Analysis and inferences:

Analysis:

Commands demonstrate efficient file handling, permissions setting, sorting, and transferring within Linux operations.

Inferences:

Linux commands streamline file management, ensure security through permissions, and enable organized data handling.

Course Title	OPERATING SYSTEMS	ACADEMIC YEAR: 2024-25
Course Code(s)	23CS2104A	Page 6 of 226

Experiment #	<TO BE FILLED BY STUDENT>	Student ID	<TO BE FILLED BY STUDENT>
Date	<TO BE FILLED BY STUDENT>	Student Name	[@KLWKS_BOT THANOS]

POST LAB

BASIC COMMANDS	FUNCTIONALITY
name	Identifies the name of a file or command.
mount	Mounts a filesystem or storage device to the directory.
umount	Unmounts a mounted filesystem or storage device.
more	Displays file contents one page at a time.
less	Allows scrolling through file contents forward and backward.
Diff	Compares two files and shows differences between them.

Course Title	OPERATING SYSTEMS	ACADEMIC YEAR: 2024-25
Course Code(s)	23CS2104A	Page 7 of 226

Experiment #	<TO BE FILLED BY STUDENT>	Student ID	<TO BE FILLED BY STUDENT>
Date	<TO BE FILLED BY STUDENT>	Student Name	[@KLWKS_BOT THANOS]

ln	Creates a hard or symbolic link to a file.
rm	Removes files or directories.
cp	Copies files or directories.
rmdir	Removes an empty directory.
gzip	Compresses files using the gzip algorithm.
find	Searches for files or directories in a specified location.
telnet	Connects to a remote server using the Telnet protocol.

Course Title	OPERATING SYSTEMS	ACADEMIC YEAR: 2024-25
Course Code(s)	23CS2104A	Page 8 of 226

Experiment #	<TO BE FILLED BY STUDENT>	Student ID	<TO BE FILLED BY STUDENT>
Date	<TO BE FILLED BY STUDENT>	Student Name	[@KLWKS_BOT THANOS]

nslookup	Queries domain name servers to obtain domain information.
df	Displays disk space usage of file systems.
du	Shows disk usage of files and directories.
free	Displays system memory usage statistics.
top	Displays real-time system process and resource usage.
ps	Displays information about active processes.
kill	Terminates a running process by its process ID.

Course Title	OPERATING SYSTEMS	ACADEMIC YEAR: 2024-25
Course Code(s)	23CS2104A	Page 9 of 226

Experiment #	<TO BE FILLED BY STUDENT>	Student ID	<TO BE FILLED BY STUDENT>
Date	<TO BE FILLED BY STUDENT>	Student Name	[@KLWKS_BOT THANOS]

Data and Results:

Data:

Commands for file management, system monitoring, and process control in Linux environment operations.

Result:

Successfully identified, mounted, copied, removed files, and monitored system resources using commands.

Analysis and inferences:

Analysis:

Commands effectively manage files, monitor system resources, and control processes in Linux environment efficiently.

Inferences:

Linux commands offer efficient file management, system monitoring, and process control solutions.

Course Title	OPERATING SYSTEMS	ACADEMIC YEAR: 2024-25
Course Code(s)	23CS2104A	Page 10 of 226

Experiment #	<TO BE FILLED BY STUDENT>	Student ID	<TO BE FILLED BY STUDENT>
Date	<TO BE FILLED BY STUDENT>	Student Name	[@KLWKS_BOT THANOS]

Sample VIVA-VOCE Questions (In-Lab):

1. How does a system call work?

- A user program requests an OS service via a system call.
- The CPU switches from **user mode** to **kernel mode** to execute the request.
- The OS performs the task and returns the result to the program, switching back to **user mode**.

2. Why do you need system calls in the Operating System?

- To provide controlled, secure access to hardware and system resources.
- They abstract hardware complexities and ensure OS stability and isolation.

3. What do you mean by file operations?

- Actions like **create**, **open**, **read**, **write**, **close**, **delete**, and **seek** performed on files to manage and access data.

Course Title	OPERATING SYSTEMS	ACADEMIC YEAR: 2024-25
Course Code(s)	23CS2104A	Page 11 of 226

Experiment #	<TO BE FILLED BY STUDENT>	Student ID	<TO BE FILLED BY STUDENT>
Date	<TO BE FILLED BY STUDENT>	Student Name	[@KLWKS_BOT THANOS]

4. Differentiate between system calls and library calls.

System Calls	Library Calls
Access OS services directly.	Use functions provided by libraries.
Switch to kernel mode.	Run in user mode.
Slower (mode switching).	Faster (no OS interaction).

5. Differentiate between function and system call.

Function	System Call
Defined by users or libraries.	Defined by the OS kernel.
No mode switch needed.	Requires user-to-kernel mode switch.
Example: <code>printf()</code>	Example: <code>read()</code>

Evaluator Remark (if any):	Marks Secured _____ out of 50
	Signature of the Evaluator with Date

Note: Evaluator MUST ask Viva-voce before signing and posting marks for each experiment.

Course Title	OPERATING SYSTEMS	ACADEMIC YEAR: 2024-25
Course Code(s)	23CS2104A	Page 12 of 226