1. What type of commands you are using in day-to-day life?

Theory:
In daily production support or system admin work, we commonly use:

- **File and directory management** → `ls`, `cd`, `pwd`, `mkdir`, `rm`, `cp`, `mv`
- **Process monitoring** → `ps`, `top`, `kill`
- **Disk/memory monitoring** → `df`, `du`, `free`, `uptime`
- **Search and filter** → `grep`, `find`, `awk`, `sed`
- **Network** → `ping`, `telnet`, `netstat`, `traceroute`
- **Archiving/Compression** → `tar`, `gzip`, `zip`
- **Scheduling** → `crontab`
- **Permission** → `chmod`, `chown`, `chgrp`

Interview Answer Example:

> "In my daily work, I mainly use commands like `grep` and `find` to locate logs, `df` and `du` to check disk usage, `ps -ef` and `top` to monitor running processes, and `chmod`/`chown` to manage file permissions. I also schedule maintenance scripts using `crontab`."

2. How to kill a process?

Theory:
Used to stop a running process.

Syntax:

```
ps -ef | grep <process_name>     # find process ID
kill <PID>                       # normal kill
kill -9 <PID>                    # forcefully kill (SIGKILL)
```

example
```
ps -ef | grep java
kill -9 4567
```

Use Case:
If an application process (say Tomcat or MySQL) hangs or uses high CPU, L3 support uses this to terminate it.

3. How to give permission to a file?

Theory:
`chmod` (change mode) command changes file permissions.
Permissions:

- **r = read (4)**
- **w = write (2)**
- **x = execute (1)**

Syntax:

```
chmod 755 script.sh
```

- Owner: read/write/execute
- Group: read/execute
- Others: read/execute

Symbolic way:

```
chmod u+x file.sh   # add execute for user
chmod g-w file.sh   # remove write for group
```

Interview Example:

> "I use `chmod` to make deployment scripts executable before running them, e.g., `chmod +x deploy.sh`."

4. `crontab`, `find`, `grep`, `ps`, `df`, `du`

Let's summarize each:

Command	Use	Example	Description
`crontab`	Schedule jobs	`crontab -e`	Edits cron jobs

`find`	Search files	`find /var/log -name "*.log"`	Finds files by name, size, date
`grep`	Search text patterns	`grep "ERROR" /var/log/app.log`	Searches content in files
`ps`	Show processes	`ps -ef`	Shows all running processes
`df`	Disk usage	`df -h`	Human-readable disk space summary
`du`	Directory usage	`du -sh /var/log`	Size of directory recursively

5. How to find old log files or temp files?

Command:

```
find /var/log -type f -mtime +30
```

- `-mtime +30` → modified more than 30 days ago
  For temp files:

```
find /tmp -type f -name "*.tmp" -mtime +7
```

Use Case: Clean up disk space in production servers.

6. Find files of 12GB size named file.sh and delete

Command:

```
find / -type f -name "file.sh" -size 12G -exec rm -f {} \;
```

Explanation:

- `-type f`: regular file
- `-size 12G`: exact 12GB

- `-exec rm -f {}`: delete the found file

7. `tar`, `gzip`, `tail`, `head`

Command	Function	Example
`tar`	Archive multiple files	`tar -cvf backup.tar /var/log/`
`gzip`	Compress a file	`gzip backup.tar`
`tail`	Show last lines of file	`tail -n 20 app.log`
`head`	Show first lines of file	`head -n 10 app.log`

Use Case:

- Use `tail -f app.log` to monitor live logs.
- Use `tar` and `gzip` for backup and compression.

8. `grep`, `find`, `sed` basics

grep → Search for patterns

```
grep "ERROR" app.log
grep -i "error" app.log      # case-insensitive
grep -r "fail" /var/log      # recursive
```

find → Locate files

```
find /etc -name "*.conf"
```

sed → Stream editor for substitution

```
sed 's/oldword/newword/g' file.txt
```

9. All UNIX commands with options (Reflex command)

This usually means **you should know command behavior instantly** — e.g.:

Task	Command
List hidden files	`ls -la`
Copy file	`cp file1 file2`
Move file	`mv old new`
Remove directory	`rm -rf dirname`
Create empty file	`touch file.txt`
Display file size	`ls -lh file.txt`
Check current directory	`pwd`
View file	`cat file.txt`, `less file.txt`

10. `find` command (deep dive)

Syntax:

```
find <path> [criteria] [action]
```

Examples:

- Find by name:
  `find /home -name "test.txt"`
- Find by size:
  `find /home -size +100M`
- Find modified in last 1 day:
  `find /var/log -mtime -1`
- Delete directly:
  `find /tmp -type f -mtime +7 -delete`

11. awk commands

Theory: Pattern scanning and text processing tool.

Examples:

```
awk '{print $1}' file.txt            # print first column
awk -F, '{print $2,$3}' file.csv     # print 2nd and 3rd field using
comma delimiter
awk '/error/ {count++} END {print count}' app.log  # count "error"
occurrences
```