Day-7

---

1. Basic System & Package Management Commands

- `sudo apt update`
  - Updates the package lists for system repositories.
- `sudo apt install ncal`
  - Installs the `ncal` (calendar) command-line tool.
- `ncal`
  - Displays the calendar in a different format compared to `cal`.
- `ncal 12 2025`
  - Shows the calendar for **December 2025**.
- `cal 2025`
  - Displays the full calendar for the year **2025**.
- `cal 1990`
  - Displays the full calendar for the year **1990**.

---

2. System Information Commands

- `uname -a`
  - Shows complete system information, including kernel version.
- `uname -s`
  - Displays the **kernel name**.

---

3. Finding File Locations

- `whereis python`
  - Locates installed Python binaries and related files.
- `whereis java`
  - Locates Java installation paths.
- `whereis ls`
  - Locates the **ls** (list directory) binary.

---

4. Process Management Commands

- `ps aux`
  - Displays all running processes with detailed information.
- `kill -9 <PID>`
  - Forcefully terminates a process using its **Process ID (PID)**.
  - Example: `kill -9 119675`

---

5. File Operations

- `ls -lrt`
  - Lists files in long format, sorted by modification time (oldest first).
- `wc -l name.txt`
  - Counts the number of **lines** in `name.txt`.
- `wc -w name.txt`
  - Counts the number of **words** in `name.txt`.
- `wc -c name.txt`
  - Counts the number of **characters** in `name.txt`.
- `cat name.txt`
  - Displays the contents of `name.txt`.
- `vi name.txt`
  - Opens `name.txt` in the **vi editor** for editing.

---

6. Searching Files and Directories

- `find -name "*.txt"`
  - Finds all `.txt` files in the current directory and subdirectories.
- `find . -type d`
  - Finds all directories inside the current directory.
- `find . -name "*.tmp" -exec rm {} \;`
  - Finds and removes all `.tmp` files.

---

7. System Monitoring Commands

- `top`
  - Displays real-time system performance and running processes.
- `du -sh *`
  - Displays the disk usage of files and directories in the current folder.
- `ncdu .`
  - A more interactive disk usage viewer (`ncdu` needs to be installed first).
- `sudo apt install ncdu`
  - Installs `ncdu` (a disk usage analyzer).

---

8. Network Commands

- `ip a`
  - Shows network interface details (IP addresses, MAC addresses, etc.).
- `grep -Ril "ranawat"`
  - Searches for the word **"ranawat"** inside files in the current directory and **lists filenames only**.
- `telnet google.com`
  - Attempts to connect to **Google** via Telnet (used for testing network connectivity).
- `dig google.com`
  - Queries DNS records for `google.com`.
- `iperf -s -f M`
  - Runs an `iperf` server to measure network performance (in Megabytes).
- `sudo tcpdump -i any`
  - Captures network traffic on all interfaces.

---

9. Git Commands

- `git pull`
  - Fetches the latest changes from the remote repository.
- `git reset --hard origin/main`
  - Resets the current branch to match the remote **main** branch.
- `history | grep git`
  - Searches command history for Git-related commands.

---

10. File Compression & Archiving

- `tar -cvf etc.tar Invoice-9.pdf`
  - Creates a tar archive named `etc.tar` containing `Invoice-9.pdf`.
- `tar -cvf etc1.tar .`
  - Archives everything in the current directory into `etc1.tar`.
- `tar -xzvf etc1.tar`
  - Extracts the contents of `etc1.tar`.
- `tar -cvf etc12.tar.gz .`
  - Creates a **compressed** archive (`.tar.gz`) of the current directory.
- `tar -xzvf etc12.tar.gz`
  - Extracts the `.tar.gz` file.
- `tar -tzvf etc12.tar.gz`
  - Lists the contents of `etc12.tar.gz` without extracting.
- `man tar`
  - Displays the manual (documentation) for the `tar` command.

---

11. Shell Scripting & Variables

Creating & Using Variables

```
namej="jinesh"
echo $namej
```

- Assigns `"jinesh"` to variable `namej` and prints it.

Concatenating Variables

```
var_1="jinesh"
var_2="ranawat"
echo "$var_1$var_2"
```

- Prints `"jineshranawat"`.

Using Readonly Variables

```
readonly var_2
# var_2="ranawatjinesh"  # This will give an error because var_2 is
now readonly
```

- **Using Conditional Statements**

```
time=$(date +%H)

echo $time
if [ $time -lt 12 ];then
    message="Good morning user"
elif [ $time -lt 18 ];then
    message="Good afternoon user"
else
    message="Good evening user"
fi
echo "$message $time"
```

- Checks the current hour and prints a time-based greeting.

Unsetting Variables

```
unset var_age
echo "Age is after unsetting $var_age"  # Prints an empty value
```

---

12. Loops in Shell Scripting

While Loop

```
i=1
while [ $i -lt 5 ]
do
   echo "Jinesh"
   i=`expr $i + 1`
done
```

- Prints `"Jinesh"` 4 times.

For Loop with Break

```
for a in 1 2 3 4 5 6 7 8 9
do
   if [ $a == 5 ]
   then
     break
   fi
```

```
   echo "Iteration is $a"
done
```

- Prints numbers **1 to 4** and exits when `a == 5`.

---

13. Aliases & Custom Commands

- `alias j1="ls -lrt"`
  - Creates an alias `j1` that runs `ls -lrt`.
- `j1`
  - Runs the alias (`ls -lrt`).
- `alias c="clear"`
  - Creates an alias `c` for the `clear` command.
- `c`
  - Clears the terminal screen.

---

14. File & Directory Removal

- `find . -name "*.tmp" -exec rm {} \;`
  - Finds and **deletes** all `.tmp` files.
- `rmdir [-rf / --ignore-fail-on-non-empty]`
  - Removes directories forcefully, even if they contain files.

---

15. Searching in Man Pages

- `/keyword`
  - Searches for a **keyword** inside a man page.
  - Press `n` to find the next occurrence.

Key Takeaways

✅ **Package Management:** `apt install`, `apt update`
✅ **System Information:** `uname`, `top`, `ps aux`
✅ **Process Management:** `kill`, `grep`, `find`
✅ **Networking:** `dig`, `iperf`, `tcpdump`
✅ **Git & SSH:** `git pull`, `git reset`
✅ **Archiving & Compression:** `tar` commands
✅ **Shell Scripting:** Variables, loops, conditionals
✅ **Aliases:** Creating shortcuts for commands
