

Day 2 Summary - Linux & Shell Scripting Commands

1 Directory & File Management

- `rmdir / rmdir -p a/b/c/d/e/f`
 - **Removes empty directories.** The `-p` flag removes parent directories if they become empty.
 - `cat content.txt`
 - **Displays the content of a file** on the terminal.
 - `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`.
-

2 System Information & Monitoring

- `uname`
 - Displays system information (kernel name, version, etc.).
 - `uname -a`
 - Shows detailed system info including OS, kernel version, machine type, etc.
 - `whoami`
 - Displays the **current logged-in user**.
 - `ps aux`
 - Shows **all running processes** with details like user, PID, CPU usage, etc.
 - `top`
 - Shows **real-time processes & system resource usage** (CPU, memory).
 - `du -sh *`
 - Displays the **disk usage** of all files and directories in the current folder.
 - `ncdu .`
 - **Interactive disk usage analyzer** (requires `sudo apt install ncdu`).
-

3 Package Management & Software Installation

- `sudo apt install ncal`
 - Installs the **ncal calendar utility**.
- `cal 2025 / ncal 12 2025`
 - Displays a calendar for a specific year/month.
- `sudo apt update`
 - Updates the **package lists** for available updates.
- `sudo apt install`

- Installs new software packages.
-

4 Searching & Finding Files

- `find . -name "*.txt"`
 - Finds all `.txt` files in the current directory and subdirectories.
 - `find . -type d`
 - Lists all **directories** inside the current folder.
 - `find . -name "*.tmp" -exec rm {} \;`
 - Finds and **deletes all .tmp files**.
 - `whereis python`
 - Locates the binary, source, and manual pages for Python.
 - `where python`
 - Finds **all paths where Python is installed**.
-

5 File Archiving & Compression

- `tar -cvf etc.tar Invoice-9.pdf`
 - Creates a **tar archive** (`etc.tar`) containing `Invoice-9.pdf`.
 - `tar -cvf etc1.tar .`
 - Creates a tar archive of the **current directory**.
 - `tar -xzvf etc1.tar`
 - Extracts the **tar archive**.
 - `tar -cvf etc12.tar.gz .`
 - Creates a **compressed tar archive** (`.tar.gz`).
 - `tar -xzvf etc12.tar.gz`
 - Extracts a **compressed tar.gz archive**.
 - `tar -tzvf etc12.tar.gz`
 - Lists the **contents** of the archive without extracting.
-

6 Shell Scripting Basics

Variables

```
var_name="Hitesh"
var_age=23
echo "Name is $var_name and age is $var_age"
```

- Defines **variables** and prints their values.

Readonly Variables

```
var_blood_group="O-"
```

```
readonly var_blood_group
```

- **Marks a variable as readonly**, preventing modification.

Unset Variables

```
unset var_age  
echo "Age is after unsetting: $var_age"
```

- **Removes the variable** from memory.

Time-Based Greetings

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

- Uses `date +%H` to fetch **current hour** and print a **greeting** accordingly.
-

7 Process Management

- `kill -9 119675`
 - **Terminates process** with PID 119675 forcefully.
 - `alias j1="ls -lrt"`
 - Creates an **alias** (j1) for `ls -lrt`.
 - `alias c="clear"`
 - Creates an alias `c` to **clear the terminal**.
-

8 AWK - Text Processing

- `awk '{print $1}' data.txt`
 - Prints the **first column** of `data.txt`.
- `awk '{print $2}' data.txt`
 - Prints the **second column** of `data.txt`.
- `awk '{print "AWS name " $1, "AWS profession " $3}' data.txt`
 - Formats output with **custom labels**.
- `awk '/Engineer/' data.txt`
 - Searches for **lines containing "Engineer"**.
- `awk '$2 > 25 {print $1, "is older than 25"}' data.txt`
 - **Conditional filtering**: Prints names of people **older than 25**.

✓ Key Takeaways

1. **File & Directory Management:** Use `rmdir`, `cat`, `wc`, and `find` for organizing files.
2. **System Monitoring:** Use `ps aux`, `top`, `whoami`, `uname`, and `du` to check system details.
3. **Package Management:** Update & install packages using `apt`.
4. **Shell Scripting Basics:** Learn **variables, conditions, and automation**.
5. **Process Management:** Use `kill`, `alias`, and `history` to manage processes.
6. **Text Processing with AWK:** Use `awk` to extract & format data from files.