LINUX COMMANDS
(BASIC TO ADVANCED)

Basic Commands

1. **pwd** – Shows the current working directory's path
2. **ls** – Lists a directory's content
3. **cd** – Changes the working directory
4. **echo** – Prints a message as a standard output
5. **cal** – Displays a calendar in Terminal
6. **hostname** – Shows your system's hostname
7. **time** – Calculates commands' execution time
8. **man** – Shows a command's manual

File and Directory Operations

9. **mkdir** – Creates a new directory
10. **touch** – Creates a new empty file
11. **cp** – Copies files and directories, including their content
12. **mv** – Moves or renames files and directories
13. **rm** – Deletes a file
14. **file** – Checks a file's type
15. **cat** – Lists, combines, and writes a file's content as a standard output
16. **head** – Displays a file's first ten lines
17. **tail** – Prints a file's last ten lines
18. **ln** – Links files or directories
19. **chmod** – Modifies a file's read, write, and execute permissions
20. **chown** – Changes a file, directory, or symbolic link's ownership

---

Text Manipulation

21. **grep** – Searches a string within a file
22. **awk** – Finds and manipulates patterns in a file
23. **sed** – Finds, replaces, or deletes patterns in a file
24. **sort** – Reorders a file’s content
25. **cut** – Sections and prints lines from a file
26. **diff** – Compares two files’ content and their differences

---

Archiving and Compression

27. **zip and unzip** – Creates and extracts a ZIP archive
28. **tar** – Archives files without compression in a TAR format

---

System Monitoring and Management

29. **df** – Displays the system’s overall disk space usage
30. **du** – Checks a file or directory’s storage consumption
31. **top** – Displays running processes and the system’s resource usage
32. **htop** – Works like top but with an interactive user interface
33. **ps** – Creates a snapshot of all running processes
34. **systemctl** – Manages system services
35. **watch** – Runs another command continuously
36. **jobs** – Displays a shell’s running processes with their statuses
37. **kill** – Terminates a running process
38. **shutdown** – Turns off or restarts the system

User and Permission Management

39. **useradd** – Creates a user account
40. **userdel** – Removes a user account
41. **sudo** – Runs a command as a superuser
42. **su** – Runs programs in the current shell as another user

Networking and Connectivity

43. **ping** – Checks the system’s network connectivity
44. **ifconfig** – Displays the system’s network interfaces and their configurations
45. **netstat** – Shows the system’s network information, like routing and sockets
46. **traceroute** – Tracks a packet’s hops to its destination
47. **nslookup** – Queries a domain’s IP address and vice versa
48. **dig** – Displays DNS information, including record types
49. **wget** – Downloads files from a URL
50. **curl** – Transmits data between servers using URLs
51. **scp** – Securely copies files or directories to another system
52. **rsync** – Synchronizes content between directories or machines

File Search and Database Operations

53. **locate** – Finds files in a system’s database
54. **find** – Outputs a file or folder’s location

Command History and Aliases

55. **history** – Lists previously run commands
56. **alias and unalias** – Sets and removes an alias for a file or command

Package Management (Debian-based distros)

57. **apt-get** – Manages Debian-based distros package libraries

System Information

58. **uname** – Prints information about your machine’s kernel, name, and hardware
59. **locate** – Finds files in a system’s database

REALTIME USECASES

Basic Commands

1. **pwd (Print Working Directory)**: Shows the absolute path of your current location in the file system.
   - *Example:* If you're in /home/user/documents, pwd will output /home/user/documents.
2. **ls (List)**: Lists the contents of a directory.
   - *Subcommands:*
     - -l: Long listing format (permissions, owner, size, date, etc.). ls -l
     - -a: Show all files, including hidden ones (starting with .). ls -a
     - -h: Human-readable sizes (e.g., KB, MB, GB). ls -lh
     - -t: Sort by modification time (newest first). ls -lt
     - -r: Reverse order. ls -ltr (list in reverse order of time, with details)
   - *Example:* ls (lists files in the current directory), ls /home/user (lists files in /home/user).
3. **cd (Change Directory)**: Changes the current working directory.
   - *Subcommands:*
     - cd: Go to the home directory.
     - cd ..: Go up one directory.
     - cd -: Go to the previous directory.
   - *Example:* cd documents (goes to the "documents" directory in the current directory), cd /var/log (goes to /var/log).
4. **echo**: Displays a line of text/string.
   - *Example:* echo "Hello, world!" (prints "Hello, world!"), echo $HOME (prints the path to your home directory).
5. **cal (Calendar)**: Displays a calendar.
   - *Subcommands:*
     - cal: Displays the current month's calendar.
     - cal <month> <year>: Displays the calendar for a specific month and year.

- *Example:* cal 10 2024 (displays the calendar for October 2024).

6. **hostname**: Displays the system's hostname.
   - *Example:* Running hostname might output "mycomputer".
7. **time**: Measures the execution time of a command.
   - *Example:* time ls -l (shows how long it takes to list files with details).
8. **man (Manual)**: Displays the manual page for a command.
   - *Example:* man ls (shows the manual for the ls command).

File and Directory Operations

9. **mkdir (Make Directory)**: Creates a new directory.
   - *Subcommands:*
     - -p: Create parent directories as needed.
   - *Example:* mkdir my_new_directory, mkdir -p path/to/new/directory.
10. **touch**: Creates an empty file or updates the timestamp of an existing file.
    - *Example:* touch new_file.txt.
11. **cp (Copy)**: Copies files and directories.
    - *Subcommands:*
      - -r: Recursively copy directories.
      - -i: Interactive (prompts before overwriting).
    - *Example:* cp file1.txt file2.txt (copies file1.txt to file2.txt), cp -r directory1 directory2 (copies directory1 and its contents to directory2).
12. **mv (Move)**: Moves or renames files and directories.
    - *Example:* mv file.txt new_location/, mv old_name.txt new_name.txt.
13. **rm (Remove)**: Deletes files and directories. *Use with caution!*
    - *Subcommands:*
      - -r: Recursively remove directories.
      - -f: Force (don't prompt).
      - -i: Interactive (prompts before deleting).
    - *Example:* rm file.txt, rm -rf directory (forcefully and recursively removes a directory).
14. **file**: Determines the file type.

- *Example:* file image.jpg (might output "image.jpg: JPEG image data").

15. **cat (Concatenate)**: Displays the contents of a file.

    - *Example:* cat text.txt.

16. **head**: Displays the first few lines of a file (default 10).

    - *Subcommands:*

        - -n <number>: Specify the number of lines.

    - *Example:* head -n 5 file.txt (shows the first 5 lines).

17. **tail**: Displays the last few lines of a file (default 10).

    - *Subcommands:*

        - -n <number>: Specify the number of lines.

        - -f: Follow (useful for log files, shows new lines as they are added).

    - *Example:* tail -f logfile.txt.

18. **ln (Link)**: Creates links between files.

    - *Subcommands:*

        - -s: Create a symbolic (soft) link.

    - *Example:* ln file.txt link_to_file.txt (creates a hard link), ln -s file.txt symbolic_link.txt (creates a symbolic link).

19. **chmod (Change Mode)**: Changes file permissions. Uses octal or symbolic notation.

    - *Example:* chmod 755 script.sh (sets permissions to rwxr-xr-x), chmod +x script.sh (adds execute permission).

20. **chown (Change Owner)**: Changes file ownership.

    - *Example:* chown user:group file.txt.

Text Manipulation

21. **grep (Global Regular Expression Print)**: Searches for patterns in files.

    - *Subcommands:*

        - -i: Ignore case.

        - -r: Recursive search in directories.

        - -n: Show line numbers.

    - *Example:* grep "keyword" file.txt, grep -r "pattern" directory/.

22. **awk**: Powerful text processing tool for pattern scanning and processing.

- *Example:* awk '{print $1}' file.txt (prints the first field of each line).

23. **sed (Stream Editor)**: Stream editor for filtering and transforming text.
    - *Example:* sed 's/old/new/g' file.txt (replaces all occurrences of "old" with "new").
24. **sort**: Sorts lines of text files.
    - *Example:* sort file.txt.
25. **cut**: Removes sections from each line of files.
    - *Example:* cut -d',' -f1 file.csv (cuts the first field from a comma-separated file).
26. **diff (Difference)**: Compares files line by line.
    - *Example:* diff file1.txt file2.txt.

Archiving and Compression

27. **zip and unzip**: Creates and extracts ZIP archives.
    - *Example:* zip archive.zip file1.txt file2.txt, unzip archive.zip.
28. **tar (Tape Archive)**: Archives files (often combined with compression).
    - *Subcommands:*
        - -c: Create an archive.
        - -x: Extract an archive.
        - -v: Verbose (list files processed).
        - -f: Specify the archive file name.
        - -z: Compress with gzip (.tar.gz or .tgz).
        - -j: Compress with bzip2 (.tar.bz2 or .tbz).
    - *Example:* tar -cvf archive.tar files/, tar -xvzf archive.tar.gz.

System Monitoring and Management

29. **df (Disk Free)**: Shows disk space usage.
    - *Subcommands:*
        - -h: Human-readable format (KB, MB, GB).
        - -T: Show file system types.
    - *Example:* df -h (shows disk usage in human-readable format).
30. **du (Disk Usage)**: Shows disk space used by files and directories.

- *Subcommands:*
    - -h: Human-readable format.
    - -s: Summarize total usage.
    - -a: Show usage for all files.
- *Example:* du -sh /home/user (shows the total disk usage of the /home/user directory in human-readable format).

31. **top**: Displays dynamic real-time view of running processes.
    - *Interactive Commands (within top):*
        - q: Quit.
        - h: Help.
        - P: Sort by CPU usage.
        - M: Sort by memory usage.
    - *Example:* Just type top in the terminal.
32. **htop**: An interactive process viewer (requires installation).
    - Similar to top but with a more user-friendly interface using colors and mouse interaction.
    - *Example:* htop.
33. **ps (Process Status)**: Shows a snapshot of current processes.
    - *Subcommands:*
        - aux: Show all processes, including those run by other users.
        - -ef: Same as aux, but with full command lines.
    - *Example:* ps aux | grep firefox (shows processes related to Firefox).
34. **systemctl**: Manages systemd services (used on most modern Linux systems).
    - *Subcommands:*
        - start <service>: Start a service.
        - stop <service>: Stop a service.
        - restart <service>: Restart a service.
        - status <service>: Check the status of a service.
        - enable <service>: Enable a service to start on boot.
        - disable <service>: Disable a service from starting on boot.

- *Example:* systemctl restart apache2.

35. **watch**: Executes a command repeatedly and displays the output.
    - *Example:* watch -n 1 'date' (runs the date command every second).
36. **jobs**: Lists background jobs in the current shell.
    - *Example:* After running a command in the background (e.g., sleep 100 &), use jobs to see its status.
37. **kill**: Terminates a process.
    - *Example:* kill <PID> (where PID is the process ID). kill -9 <PID> (forcefully kills a process).
38. **shutdown**: Shuts down or restarts the system.
    - *Subcommands:*
        - -h: Halt (power off).
        - -r: Reboot.
        - <time>: Time until shutdown (e.g., now, +5, 10:00).
    - *Example:* shutdown -h now (shuts down immediately), shutdown -r +10 (reboots in 10 minutes).

User and Permission Management

11. **useradd**: Creates a new user account.
    - *Example:* sudo useradd newuser.
12. **userdel**: Deletes a user account.
    - *Subcommands:*
        - -r: Remove the user's home directory and mail spool.
    - *Example:* sudo userdel -r olduser.
13. **sudo (Super User Do)**: Executes a command as the superuser (root).
    - *Example:* sudo apt update.
14. **su (Switch User)**: Changes the current user.
    - *Example:* su username (switches to the specified user). su (switches to root).

Networking and Connectivity

15. **ping**: Checks network connectivity to a host.
    - *Example:* ping google.com.

16. **ifconfig (Interface Configuration) / ip (newer alternative)**: Displays and configures network interfaces.
    - ➢ *Example (using ip):* ip a (shows network interfaces and addresses), ip route (shows routing table)
    - ➢ *Example (using ifconfig):* ifconfig
17. **netstat (Network Statistics) / ss (newer alternative)**: Displays network connections, routing tables, etc.
    - ➢ *Example (using ss):* ss -tulnp (shows listening TCP and UDP ports with process information).
    - ➢ *Example (using netstat):* netstat -tulnp
18. **traceroute**: Traces the route packets take to a destination.
    - ➢ *Example:* traceroute google.com.
19. **nslookup**: Queries DNS servers to find IP addresses or domain names.
    - ➢ *Example:* nslookup google.com.
20. **dig (Domain Information Groper)**: More advanced DNS lookup utility.
    - ➢ *Example:* dig @8.8.8.8 google.com (queries Google's public DNS server).
21. **wget**: Downloads files from the web.
    - ➢ *Example:* wget https://www.example.com/file.zip.
22. **curl**: Transfers data with URLs (more versatile than wget).
    - ➢ *Example:* curl https://www.example.com (displays the website's HTML), curl -O https://www.example.com/file.zip (downloads the file).
23. **scp (Secure Copy)**: Securely copies files between systems over SSH.
    - ➢ *Example:* scp file.txt user@remotehost:/path/.
24. **rsync (Remote Sync)**: Synchronizes files and directories between locations.
    - ➢ *Example:* rsync -avz /local/directory/ user@remotehost:/remote/directory/.

File Search and Database Operations

25. **locate**: Finds files by name (uses a pre-built database).
    - ➢ *Example:* locate myfile.txt. (Update database with sudo updatedb)
26. **find**: Powerful file searching based on various criteria.

- *Example:* find . -name "myfile.txt" (finds files named "myfile.txt" in the current directory and subdirectories), find / -type d -name "mydir" (finds directories named "mydir" in the root directory and below).

Command History and Aliases

27. **history**: Displays command history.
    - *Example:* history.
28. **alias and unalias**: Creates and removes command aliases.
    - *Example:* alias la='ls -la' (creates an alias la for ls -la), unalias la (removes the alias).

Package Management (Debian-based distros)

29. **apt-get / apt (newer version)**: Manages packages on Debian/Ubuntu systems.
    - *Subcommands:*
        - update: Update package lists.
        - upgrade: Upgrade installed packages.
        - install <package>: Install a package.
        - remove <package>: Remove a package.
        - purge <package>: Remove a package and its configuration files.
    - *Example:* sudo apt update, sudo apt install firefox.

System Information

30. **uname**: Displays system information.
    - *Subcommands:*
        - -a: All information.
        - -r: Kernel release.
        - -m: Machine hardware name.
    - *Example:* uname -a.