1. `ls`

- Lists files and directories in the current directory.

- Example: `ls -l` (long listing format)

2. `cd`

- Changes the current directory.

- Example: `cd /home/user` (change to the /home/user directory)

3. `pwd`

- Prints the current working directory.

- Example: `pwd` (display the full path to the current directory)

4. \*\*`cp`\*\*

- Copies files or directories.

- Example: `cp file1.txt file2.txt` (copy file1.txt to file2.txt)

5. `mv`

- Moves or renames files or directories.

- Example: `mv file1.txt /tmp` (move file1.txt to the /tmp directory)

6. `rm`

- Removes files or directories.

- Example: `rm file1.txt` (delete file1.txt)

7. `mkdir`

- Creates a new directory.

- Example: `mkdir new\_directory` (create a directory named new\_directory)

8. `rmdir`

- Removes an empty directory.

- Example: `rmdir empty\_directory` (remove empty\_directory)

9. `touch`

- Creates an empty file or updates the timestamp of an existing file.

- Example: `touch newfile.txt` (create a new file named newfile.txt)

10. `cat`

- Concatenates and displays the content of files.

- Example: `cat file1.txt` (display the content of file1.txt)

11. `less`

- Views the content of a file one page at a time.

- Example: `less file1.txt` (view file1.txt with pagination)

12. `head`

- Displays the first few lines of a file.

- Example: `head -n 10 file1.txt` (show the first 10 lines of file1.txt)

13. `tail`

- Displays the last few lines of a file.

- Example: `tail -n 10 file1.txt` (show the last 10 lines of file1.txt)

14. `grep`

- Searches for patterns in files.

- Example: `grep 'pattern' file1.txt` (search for 'pattern' in file1.txt)

15. \*\*`find`\*\*

- Searches for files and directories.

- Example: `find /home -name '\*.txt'` (find all .txt files in /home)

16. `chmod`

- Changes file permissions.

- Example: `chmod 755 script.sh` (set read, write, and execute permissions for the owner, and read and execute permissions for others on script.sh)

17. `chown`

- Changes file ownership.

- Example: `chown user:group file1.txt` (change the owner and group of file1.txt to user and group)

18. `ps`

- Displays current running processes.

- Example: `ps aux` (show detailed information about all running processes)

19. `top`

- Displays real-time system information, including processes.

- Example: `top` (interactive display of system resource usage)

20. `df`

- Displays disk space usage.

- Example: `df -h` (show disk usage in a human-readable format)

21. `du’

- Estimates file space usage.

- Example: `du -sh /home/user` (show the total size of the /home/user directory)

22. `ifconfig`

- Configures or displays network interface parameters (mostly replaced by `ip` command).

- Example: `ifconfig eth0` (show details for the eth0 network interface)

23. `ping`

- Sends ICMP echo requests to test network connectivity.

- Example: `ping google.com` (test connectivity to google.com)

24. `tar`

- Archives files.

- Example: `tar -czvf archive.tar.gz /path/to/directory` (create a gzip-compressed archive of the directory)

25. `wget`

- Downloads files from the web.

- Example: `wget http://example.com/file.txt` (download file.txt from example.com)

These commands are foundational and are commonly used for managing files, directories, processes, and system information on Linux systems.