**2. Basic Structure of Commands**

A Linux command typically follows this format:

bash

Copy code

command [options] [arguments]

* **command**: The action you want to perform (e.g., ls, cd).
* **options**: Modify the behavior of the command (e.g., -l, -a).
* **arguments**: The target of the command (e.g., a file or directory name).

**3. Basic Commands**

**a. Navigating the File System**

* pwd (Print Working Directory): Shows your current directory.

bash

Copy code

pwd

* ls (List): Lists files and directories in the current directory.

bash

Copy code

ls

Options:

* + ls -l: Detailed list.
  + ls -a: Includes hidden files.
* cd (Change Directory): Moves you between directories.

bash

Copy code

cd /path/to/directory

Examples:

* + cd .. (Go up one directory).
  + cd ~ (Go to your home directory).

**b. Working with Files and Directories**

* mkdir (Make Directory): Creates a new directory.

bash

Copy code

mkdir myfolder

* rmdir (Remove Directory): Removes an empty directory.

bash

Copy code

rmdir myfolder

* touch: Creates an empty file.

bash

Copy code

touch myfile.txt

* rm (Remove): Deletes files.

bash

Copy code

rm myfile.txt

To delete a directory and its contents:

bash

Copy code

rm -r myfolder

**c. Viewing File Contents**

* cat: Displays the content of a file.

bash

Copy code

cat myfile.txt

* less: Opens a file for viewing, one screen at a time.

bash

Copy code

less myfile.txt

* head: Shows the first few lines of a file.

bash

Copy code

head myfile.txt

* tail: Shows the last few lines of a file.

bash

Copy code

tail myfile.txt

**d. File Permissions**

* chmod: Changes file permissions.

bash

Copy code

chmod 755 myfile.txt

* ls -l: Shows the permissions of files and directories.

**e. Searching for Files**

* find: Searches for files and directories.

bash

Copy code

find /path -name "filename"

* grep: Searches within files.

bash

Copy code

grep "search\_term" myfile.txt

**4. Working with Processes**

* ps: Displays currently running processes.

bash

Copy code

ps

* top: Displays live processes and resource usage.

bash

Copy code

top

* kill: Ends a process by its ID.

bash

Copy code

kill PID

**7. Practice Time!**

* Try creating a directory, navigating to it, creating a file, writing into it using echo, and then viewing the content:

bash

Copy code

mkdir practice

cd practice

touch myfile.txt

echo "Hello, Linux!" > myfile.txt

cat myfile.txt

Would you like to dive deeper into any of these commands or proceed to more advanced topics?

4o

**You said:**

what is echo and like other commands

**ChatGPT said:**

ChatGPT

The echo command in Linux is a basic yet versatile command that prints (or "echoes") text to the terminal. It’s often used to display messages, variables, or results of commands.

Let’s explore echo and some other commands similar in utility and function.

**1. echo Command**

**What It Does:**

* Outputs the string or text provided to it.

**Basic Usage:**

bash

Copy code

echo Hello, Linux!

**Output:**

Copy code

Hello, Linux!

**Common Use Cases:**

* **Displaying Messages:**

bash

Copy code

echo "Welcome to Linux!"

* **Printing Variables:**

bash

Copy code

name="Jenith"

echo "Hello, $name"

**Output:**

Copy code

Hello, Jenith

* **Redirecting Output to a File:**

bash

Copy code

echo "This is a test file." > test.txt

* + This creates a file test.txt with the content: This is a test file.
* **Appending to a File:**

bash

Copy code

echo "Adding another line." >> test.txt

* **Escape Characters:** Use the -e option for special characters like \n (new line), \t (tab):

bash

Copy code

echo -e "Line1\nLine2"

**Output:**

Copy code

Line1

Line2

**2. cat Command**

**What It Does:**

* Concatenates and displays the contents of files.

**Basic Usage:**

bash

Copy code

cat file.txt

**Output:**

* Displays the content of file.txt.

**Use Cases:**

* **Combine Files:**

bash

Copy code

cat file1.txt file2.txt > combined.txt

**3. head and tail Commands**

**What They Do:**

* **head:** Shows the first lines of a file.
* **tail:** Shows the last lines of a file.

**Usage:**

* Display the first 5 lines of a file:

bash

Copy code

head -n 5 file.txt

* Display the last 3 lines of a file:

bash

Copy code

tail -n 3 file.txt

**Live Log Monitoring:**

Using tail with the -f option:

bash

Copy code

tail -f logfile.txt

* Continuously watches a file for updates (useful for logs).

**4. touch Command**

**What It Does:**

* Creates an empty file.

**Usage:**

bash

Copy code

touch newfile.txt

* Creates a file named newfile.txt if it doesn’t already exist.

**5. mkdir and rmdir Commands**

**What They Do:**

* **mkdir:** Creates directories.
* **rmdir:** Removes empty directories.

**Usage:**

* Create a directory:

bash

Copy code

mkdir myfolder

* Remove an empty directory:

bash

Copy code

rmdir myfolder

**6. rm Command**

**What It Does:**

* Deletes files or directories.

**Usage:**

* Remove a file:

bash

Copy code

rm file.txt

* Remove a directory with contents:

bash

Copy code

rm -r myfolder

**7. cp Command**

**What It Does:**

* Copies files or directories.

**Usage:**

* Copy a file:

bash

Copy code

cp file1.txt file2.txt

* Copy a directory:

bash

Copy code

cp -r folder1 folder2

**8. mv Command**

**What It Does:**

* Moves (or renames) files or directories.

**Usage:**

* Rename a file:

bash

Copy code

mv oldname.txt newname.txt

* Move a file:

bash

Copy code

mv file.txt /path/to/destination/

**9. find Command**

**What It Does:**

* Searches for files and directories.

**Usage:**

* Find files by name:

bash

Copy code

find /path -name "file.txt"

**10. grep Command**

**What It Does:**

* Searches for a specific text within files.

**Usage:**

* Search for a word in a file:

bash

Copy code

grep "Linux" file.txt

* Search recursively in all files of a directory:

bash

Copy code

grep -r "word" /path

**Quick Recap Table**

| **Command** | **Purpose** | **Example** |
| --- | --- | --- |
| echo | Print text or variables | echo "Hello, Linux!" |
| cat | Display file contents | cat file.txt |
| head | Show first lines of file | head -n 5 file.txt |
| tail | Show last lines of file | tail -n 3 file.txt |
| touch | Create empty file | touch newfile.txt |
| mkdir | Create directory | mkdir myfolder |
| rmdir | Remove empty directory | rmdir myfolder |
| rm | Remove files/directories | rm -r myfolder |
| cp | Copy files/directories | cp file1.txt file2.txt |
| mv | Move or rename files | mv oldname.txt newname.txt |
| find | Search files/directories | find /path -name "file.txt" |
| grep | Search within files | grep "text" file.txt |

Would you like me to provide examples of using these commands together in a small project?

4o

**You said:**

more commands

**ChatGPT said:**

ChatGPT

Sure! Here’s a list of more useful Linux commands, grouped by their functionality, to help you expand your knowledge. Each command includes a brief explanation and an example.

**1. File and Directory Management**

* **basename**: Strips the directory and returns only the file name.

bash

Copy code

basename /path/to/file.txt

# Output: file.txt

* **dirname**: Returns the directory part of the path.

bash

Copy code

dirname /path/to/file.txt

# Output: /path/to

**2. Viewing and Editing Files**

* **nano**: Opens a file in the Nano text editor.

bash

Copy code

nano file.txt

* **vi or vim**: Opens a file in the Vi/Vim text editor.

bash

Copy code

vim file.txt

* **wc**: Counts lines, words, and characters in a file.

bash

Copy code

wc file.txt

# Output: lines words characters

* **sort**: Sorts the contents of a file alphabetically or numerically.

bash

Copy code

sort file.txt

* **uniq**: Removes duplicate lines from a sorted file.

bash

Copy code

sort file.txt | uniq

* **diff**: Compares the contents of two files line by line.

bash

Copy code

diff file1.txt file2.txt

**3. Disk Usage and File System**

* **df**: Displays available disk space.

bash

Copy code

df -h

# -h: Shows output in human-readable format (e.g., GB, MB).

* **du**: Displays the size of files and directories.

bash

Copy code

du -h myfolder

# -h: Human-readable format.

* **stat**: Displays detailed information about a file (e.g., size, permissions, last modified date).

bash

Copy code

stat file.txt

* **lsblk**: Lists information about block devices (e.g., disks, partitions).

bash

Copy code

lsblk

**4. Permissions and Ownership**

* **chmod**: Changes file permissions.

bash

Copy code

chmod 644 file.txt

**1. Display top 10 processes in descending order**

bash

Copy code

ps aux --sort=-%cpu | head -n 10

**2. Display processes with the highest memory usage**

bash

Copy code

ps aux --sort=-%mem | head -n 10

**3. Display current logged-in user and logname**

bash

Copy code

whoami

logname

**4. Display current shell, home directory, operating system type, current path setting, and current working directory**

bash

Copy code

echo $SHELL # Current shell

echo $HOME # Home directory

uname -o # Operating system type

echo $PATH # Current path setting

pwd # Current working directory

**5. Display OS version, release number, and kernel version**

bash

Copy code

cat /etc/os-release # OS version and release info

uname -r # Kernel version

**6. Display the first 15 columns from each line in the file**

bash

Copy code

cut -c 1-15 filename

**7. Cut specified columns from a file and display them**

bash

Copy code

cut -f1,3,5 -d',' filename # Example: Extract columns 1, 3, and 5 (comma-separated file)

**8. Sort a given file ignoring upper and lower case**

bash

Copy code

sort -f filename

**9. Display only directories in the current working directory**

bash

Copy code

ls -d \*/

**10. Copy files from one place to another**

bash

Copy code

cp source\_file /path/to/destination/

**11. Move files from one place to another**

bash

Copy code

mv source\_file /path/to/destination/

**12. Remove a specific directory with various options**

* **Remove an empty directory:**

bash

Copy code

rmdir directory\_name

* **Remove a non-empty directory:**

bash

Copy code

rm -r directory\_name

**13. List the numbers of users currently logged in and then sort it**

bash

Copy code

who | wc -l

**14. Merge two files into one file**

bash

Copy code

cat file1 file2 > merged\_file

**15. Change the access mode of one file**

bash

Copy code

chmod 644 filename # Example: Read and write for the owner, read-only for others

**16. Display the last ten lines of the file**

bash

Copy code

tail filename

**17. Locate files in a directory and subdirectory**

bash

Copy code

find /path/to/directory -name "filename"

**18. Display the contents of all files starting with "ap" followed by any number of characters**

bash

Copy code

cat ap\*

**19. Rename a file from aaa to aaa.aa1, where aa1 is the user's login name**

bash

Copy code

mv aaa aaa.$(whoami)

**20. Search for the word picture in a file and display the lines**

bash

Copy code

grep "picture" filename

**21. Search for all occurrences of Rebecca or rebecca in a file**

bash

Copy code

grep -i "rebecca" filename

**22. Search all four-letter words starting with b and ending with k**

bash

Copy code

grep -E "\bb[a-zA-Z]k\b" filename

**23. Display only lines that do not contain the search pattern**

bash

Copy code

grep -v "pattern" filename

**Illustrating the use of sort, grep, and awk**

* **sort**: Sort lines alphabetically.

bash

Copy code

sort filename

* **grep**: Search for specific patterns.

bash

Copy code

grep "pattern" filename

* **awk**: Print specific columns or process text.

bash

Copy code

awk '{print $1, $3}' filename