Most Important / Commonly Used Commands

Command	Description
ls	List files and directories
cd	Change directory
pwd	Print current working directory
mkdir	Create a new directory
rm -rf	Remove files and directories recursively
cp	Copy files and directories
mv	Move or rename files and directories
cat	Display file contents
echo	Print text to the console
grep	Search text using patterns
find	Search for files and directories
chmod	Change file permissions
chown	Change file ownership
df -h	Show disk space usage
du -sh	Show directory size
tar -czvf	Compress files using tar with gzip
unzip	Extract zip files
curl	Transfer data from URLs
wget	Download files from the internet
scp	Securely copy files between systems

📘 Basic Commands

Command	Description
touch	Create an empty file
head	Display the first few lines of a file
tail	Display the last few lines of a file
history	Show command history
clear	Clear the terminal screen
whoami	Display the current logged-in user
uptime	Show system uptime
date	Display current date and time
cal	Show calendar
exit	Close the terminal session

🛠 Intermediate Commands

Command	Description
ps aux	Show running processes
top	Display active processes dynamically
kill -9 <PID>	Force kill a process
pkill <process>	Kill a process by name
service <service> status	Check service status
systemctl restart <service>	Restart a service
netstat -tulnp	Show network connections and listening ports
ss -tulnp	Alternative to netstat
iptables -L	List firewall rules
journalctl -xe	View system logs
nohup <command> &	Run a command in the background
alias ll='ls -la'	Create a shortcut for a command

🚀 Advanced Commands

Command	Description
awk '{print $1}' file.txt	Extract specific columns from a file
sed 's/old/new/g' file.txt	Replace text in a file
cut -d':' -f1 /etc/passwd	Extract fields from a file
crontab -e	Schedule a cron job
tmux	Terminal multiplexer for multiple sessions
screen	Keep terminal sessions running in the background
rsync -avz	Sync files between systems
docker ps	List running Docker containers
docker logs <container>	Show logs of a Docker container
kubectl get pods	List Kubernetes pods
terraform apply	Apply Terraform configurations
ansible-playbook <file>.yml	Run an Ansible playbook
git clone <repo>	Clone a Git repository
git pull	Fetch latest changes from a remote repo
git commit -m "message"	Commit changes in Git

man folowed by any command

man ls	Gives all possible combinati

on and details

Different Ways to Use grep Command

Command	**Description**
grep "text" file.txt	Search for "text" in a file
grep -i "text" file.txt	Case-insensitive search
grep -v "pattern" file.txt	Exclude lines matching a pattern
grep -r "pattern" /dir/	Recursively search in directories
grep -w "word" file.txt	Match whole words only
grep -A3 "pattern" file.txt	Show **3 lines after** the match
grep -B2 "pattern" file.txt	Show **2 lines before** the match
grep -C4 "pattern" file.txt	Show **4 lines before & after** the match
grep -n "pattern" file.txt	Show **line numbers**
grep -o "pattern" file.txt	Show only matched text
grep -l "pattern" *.txt	Show only file names with matches
grep -c "pattern" file.txt	Count the number of matches
`grep -E "pattern1	pattern2" file.txt`
grep -f pattern.txt file.txt	Use a file with multiple patterns
grep "^pattern" file.txt	Match lines starting with pattern
grep "pattern$" file.txt	Match lines ending with pattern
grep "[0-9]" file.txt	Match lines containing digits
grep "[A-Za-z]" file.txt	Match lines containing letters
`grep "ERROR" /var/log/syslog	tee errors.log`

Example

```
grep "error" logfile.txt
grep -i "Warning" logs.txt
grep -v "error" logs.txt
grep -r "TODO" /home/user/Projects
grep -w "fail" logs.txt
grep -A3 "error" logs.txt
grep -B2 "failed" logs.txt
grep -C4 "critical" logs.txt
grep -n "404" access.log
grep -o "hello" file.txt
grep -l "error" *.log
grep -c "error" logs.txt
```

Search multiple patterns (Extended Regex)

```
grep -f patterns.txt logfile.txt
grep "^root" /etc/passwd
grep "done$" script.log
grep "[0-9]" data.txt
grep "[A-Za-z]" text.txt
```

Save grep output to a file

Most Important / Commonly Used Commands

Command
ls
cd
pwd
mkdir
rm -rf
cp
mv
cat
echo
grep
find
chmod
chown
df -h
du -sh
tar -czvf
unzip
curl/ curl -O curl -o output.html https://example.com
wget
scp

🔷 Basic Commands

Command
touch
head
tail
history
clear
whoami
uptime
date

cal
exit

🛠 Intermediate Commands

Command
ps aux
top
kill -9 <PID>
pkill <process>
service <service> status
systemctl restart <service>
netstat -tulnp
ss -tulnp
iptables -L
journalctl -xe
nohup <command> &
alias ll='ls -la'

🚀 Advanced Commands

Command
awk '{print $1}' file.txt
sed 's/old/new/g' file.txt
cut -d':' -f1 /etc/passwd
crontab -e
tmux
screen
rsync -avz
docker ps
docker logs <container>
kubectl get pods
terraform apply
ansible-playbook <file>.yml
git clone <repo>
git pull
git commit -m "message"

Description	Syntax / Example
List files and directories	ls -l (detailed list), ls -a (show hidden files)
Change directory	cd /var/www/html
Print current working directory	pwd
Create a new directory	mkdir new_folder
Remove files and directories recursively	rm -rf test_folder
Copy files and directories	cp file1.txt /home/user/docs/
Move or rename files and directories	mv file1.txt /var/tmp/ or mv old.txt new.txt
Display file contents	cat myfile.txt
Print text to console	echo "Hello World"
Search text using patterns	grep "error" log.txt
Search for files and directories	find /var/log -name "*.log"
Change file permissions	chmod 755 script.sh
Change file ownership	chown user:group file.txt
Show disk space usage	df -h
Show directory size	du -sh /home/user/docs/
Compress files using tar with gzip	tar -czvf archive.tar.gz folder/
Extract zip files	unzip archive.zip
Transfer data from URLs -o output.html: Saves the response as output.html.	curl -O http://example.com/file.zip
Download files from the internet	wget http://example.com/sample.pdf
Securely copy files between systems	scp file.txt user@192.168.1.1:/home/user/

Description	Syntax / Example
Create an empty file	touch myfile.txt
Display the first few lines of a file	head -5 log.txt
Display the last few lines of a file	tail -10 log.txt
Show command history	` history
Clear the terminal screen	clear
Display the current logged-in user	whoami
Show system uptime	uptime
Display current date and time	date "+%Y-%m-%d %H:%M:%S"

Show calendar	cal
Close the terminal session	exit

Description	Syntax / Example
Show running processes	\`ps aux
Display active processes dynamically	top
Force kill a process	kill -9 1234
Kill a process by name	pkill -f java
Check service status	service apache2 status
Restart a service	systemctl restart nginx
Show network connections and listening ports	\`netstat -tulnp
Alternative to netstat	ss -tulnp
List firewall rules	iptables -L -n -v
View system logs	\`journalctl -xe
Run a command in the background	nohup python script.py &
Create a shortcut for a command	alias gs='git status'

Description	Syntax / Example
Extract specific columns from a file	awk '{print $2}' access.log
Replace text in a file	sed 's/error/ERROR/g' log.txt
Extract fields from a file	cut -d' ' -f1 names.txt
Schedule a cron job	0 3 * * * /backup.sh
Terminal multiplexer for multiple sessions	tmux new -s session1
Keep terminal sessions running in the background	screen -S my_session
Sync files between systems	rsync -avz /src/ user@remote:/dst/
List running Docker containers	docker ps -a
Show logs of a Docker container	docker logs nginx_container
List Kubernetes pods	kubectl get pods -n kube-system
Apply Terraform configurations	terraform apply -auto-approve
Run an Ansible playbook	ansible-playbook deploy.yml
Clone a Git repository	git clone https://github.com/user/repo.git
Fetch latest changes from a remote repo	git pull origin main
Commit changes in Git	git commit -m "Fixed bug"

Different Ways to Use cd Command

Command	Description
cd <directory>	Change to a specific directory
cd ..	Move up **one level** (parent directory)
cd ../..	Move up **two levels**
cd /	Go to **root directory**
cd ~	Go to **home directory**
cd ~/Desktop	Go to a **specific directory inside home**
cd -	Switch back to the **previous directory**
cd /etc	Move to an **absolute path**
cd ./folder	Move to a directory **inside the current directc**
cd "$(pwd)/subdir"	Use pwd to navigate
cd --	Works like cd -, switches to previous directory
cd ~/..	Moves **one level up** from home directory

Example

cd /home/user/Documents

cd .. (If in /home/user/Documents, it moves to /home/user/)

cd ../.. (Moves up twice)

cd /

cd ~ (or simply cd)

cd ~/Downloads

cd - (Useful when switching between two directories)

cd /etc (Goes to /etc folder)

cd ./Projects

cd "$(pwd)/Documents/Work"

cd --

If home is /home/user, cd ~/.. moves to /home/