Basic Linux Commands Quick Reference

whoami

Shows the current logged-in username.

```
whoami
```

hostname

Shows the name of the computer.

```
hostname
```

pwd

Shows the full path of the current directory.

```
pwd
```

ls

Lists files and folders in the current directory.

```
ls
ls -l
ls -a
```

cd

Changes the current directory.

```
cd foldername
cd ..
cd ~
```

mkdir

Creates a new directory.

```
mkdir test
```

cp

Copies files or directories.

```
cp file.txt backup.txt
```

mv

Moves or renames files or directories.

```
mv old.txt new.txt
```

rm

Deletes files or directories.

```
rm file.txt
rm -r folder/
```

chmod +x

Makes a file executable.

```
chmod +x script.sh
```

cat

Displays the contents of a file.

```
cat file.txt
```

less

Views large files one screen at a time.

```
less file.txt
```

head

Shows the first 10 lines of a file.

```
head file.txt
```

tail

Shows the last 10 lines of a file.

```
tail file.txt
```

wc

Counts lines, words, and characters in a file.

```
wc file.txt
```

grep

Searches for matching text patterns in files.

```
grep "error" log.txt
```

cut

Extracts specific columns from text.

```
cut -d "," -f 1 file.csv
```

sort

Sorts lines of text alphabetically or numerically.

```
sort names.txt
```

uniq

Removes duplicate adjacent lines.

```
sort names.txt | uniq
```

echo

Prints text to the terminal.

```
echo "Hello"
```

>

Redirects output to a file (overwrites).

```
echo "Hello" > file.txt
```

>>

Appends output to a file.

```
echo "World" >> file.txt
```

<

Redirects input from a file.

```
sort < file.txt
```

read

Reads input from the user into a variable.

```
read name
echo $name
```

|

Sends output of one command to another command.

```
ls | wc -l
```

date

Displays the current date and time.

```
date
```

cal

Shows a calendar in the terminal.

```
cal
```

uptime

Shows how long the system has been running.

```
uptime
```

clear

Clears the terminal screen.

```
clear
```

which

Shows the full path of a command.

```
which ls
```

man

Displays the manual page for a command.

```
man ls
```

history

Shows previously executed commands.

```
history
```

IFS - Internal Field Separator

IFS is special shell variable which is used to recognize word boundaries while splitting a sequence of character strings.
The default value of IFS is a three-character string comprising a space, tab, and newline.

```
names="Raj,Thej,Abdul"
IFS=","
for name in $names; do echo "$name is an individual"; done
```

Thank you!

Any questions?

Thejesh GN