UNIX COMMANDS :

1. $ data : Show the current date & time
2. $ who : Show all the login users
3. $ man cmd : Show the use of any cmd
4. $ head & tail : Show the initial 10 lines and end 10 lines by default.
   $tail - 4 : show 4 lines only form the end
   $tail +4 : show 10 lines after first 4 lines
5. $ pwd : Show the path to present working directory
6. $ ls : Show all the subdirectories inside current directory
   $ ls-l : show all subdirectories in long format
   $ ls-t : show recently modified once
   $ ls-a: show hidden files or dir
7. $ cd : change directory
8. $ rmdir: remove a directory if its empty
9. $ cat filename: show the content in the file
10. $ cat > filename: create if file is not there and edit its content to
11. $ cat >> filename: Edit the content of any file
12. $ cp filename target file : copy the content to target file
13. $ ln filename second name : create a another link for the filename with the second name
14. $ mv filename dirname: move or rename file from one directory to another
15. $ rm filename : remove the file form the directory
16. $ clear : clear the terminal
17. $ cal : calendar
18. $ wall : Send common message to all connected users.
19. $ ls ch * : will show all the files with prefix ch
    -> $ ls *ing : will show files with suffix ing
    -> $ ls ch.? : will search for file type
20. $ wc file : by default shows line word char count.
    -> wc -l -w -c : line, word, char count for different options.

21. $ chmod 777 filename : will grant all permission to everyone
    will change the read write & execute permission for that file for user, groups, others
    -> $**chmod u+rw,go+r [file_name]** we can separately change the different person with options check out the internet

Operators	Definition
`+`	Add permissions
`-`	Remove permissions
`=`	Set the permissions to the specified values

The following letters that can be used in symbolic mode:

Letters	Definition
`r`	Read permission
`w`	Write permission
`x`	Execute permission

The following Reference that are used:

Reference	Class
u	Owner
g	Group
o	Others
a	All (owner,groups,others)

Examples of Using the Symbolic mode:

- Read, write and execute permissions to the file owner:

chmod u+rwx [file_name]

- Remove write permission for the group and others:

chmod go-w [file_name]

- Read and write for Owner, and Read-only for the group and other:

chmod u+rw,go+r [file_name]

2) Octal mode

It is also a method for specifying permissions. In this method we specify permission using three-digit number. Where..

- **First digit** specify the permission for Owner.
- **Second digit** specify the permission for Group.
- **Third digit** specify the permission for Others. The digits

NOTE: The digits are calculated by adding the values of the individual permissions.

Value	Permission
4	Read Permission
2	Write Permission

1	Execute Permission

0	0 0 0
1	0 0 1
2	0 1 0
3	0 1 1
4	1 0 0
5	1 0 1
6	1 1 0
7	1 1 1

Examples of Using the Octal mode:
Suppose if we to give read and write permission to the file Owner. Read, write and executable permission to the Group. Read-only permission to the Other. They our command would be.
chmod 674 [file_name]

-> VI Editor

$ vi filename -> create or open file in edit mode
-r open file in reading mode
- view filename - open in reading mode

22. Pipe - $ cat file | wc -l -> show the lines count in file
We can use pipe symbol '|' with other cods output & it will use the previous output and use other operation on it.

Filter -

A filter can accept standard input or even a file as an input and will filter out stuff according to the queries.
We can sort, extract some words from lines, can merge two files into one etc with filters

1. **Sort Filter :** By default it arranges the file in alphabetic order (a-z)
– '-r' option - sort in recursive order

- ‘-f’ option - sort according to ASCII value
- ‘-n’ option - sort file numerically as ASCII values $ cat file | sort -n

2. Grep Filter - global search for regular expression
   $ grep ‘word’ file.txt -> by default it finds the word in the file.txt , case sensitivity
- ‘- i’ option - non-case sensitive
- $ egrep “India | word2” file.txt -> can search more words with pipe in the file
- $ fgrep “sentence ..... “ -> yeah sure we can search an entire sentence now .....

21. $ tee
    - It’s used between the pipe cods where we use multiple cmds at a same time ....
    - $ cat file1.txt | sort | tee temp.txt -> will store the input in temp.txt in between on pipelines

22. $ tr
    - $ cat file.txt | tr “[a-z}” “[A-Z]” -> will capitalize the chars in file.txt and print

23. $ paste file1 file2. -> merge the same lines of both files and print
24. $ passwd -> used to change password
25. $unique data -> prints only unique entries include space, -c print with count, -u print only non duplicate entries
26. $ cut -b 1,2,3 file.txt -> will print first 3 letters on each lines in that file
27. $ bc -> used to perform arithmetic operations

```
$ bc
>>> 5*2
  10
ctr+d
$

$ bc
>>> scale=3
>>> 10/2
  5.000
```

$ bc
>>> obase=2
5
101. -> converts decimal to binary
obase -> output value base
ibase -> input value base

$ echo "obase=16;255" | bc

- FF

28. $ echo “text” -> print the text “text” just simple

29. $ echo “Hello World” | tr “Hello” “Hi” -> WIll translate the “Hello” word to Hi
    Hii Wirid

30. $ column -t -s "," file

- DATA
- India 6890 Asia
- China 8765 Asia
- France 3243 Europe

31. $cmp f2 f1. -> Compares two file and return the first difference
    f2 f1 differ: char 9, line 1

32. $ps

- PID TTY TIME CMD
- 3458 ttys000 0:00.44 -zsh

34. $kill PID. -> PID -> process If

35. $awk -F, '{ $2=""; print $0 }' OFS=, temp > newTemp. -> remove 2nd column from that file and print the output in new file
36. $lines=$(wc -l < filename) random_line=$((RANDOM % lines + 1)) awk "NR==$random_line" filename
    -> Prints a random line of filename
    Name,Age,Location
37. $ zip file.zip file. -> makes a zip file

- adding: file (deflated 6%)

38. $ cmp f1 f2. Return the difference in files
- f1 f2 differ: char 7, line 1

39. $ diff f1 f2
- 1c1
- < Hello Guys
- ---
- > Hello World!

40. $gunzip f1.gz
- nik21@Nikhils-MacBook-Air unixTrial % gzip f1
- nik21@Nikhils-MacBook-Air unixTrial % cat f1.gz
- d\;gf1?H???Wp/?,?<V~

41. $tar -> bundle all files for admin side backup. -> $tar -cvf my_archive.tar file1 file2 directory1
42. $ps
- PID TTY TIME CMD
- 49935 ttys000 0:00.69 -zsh

43. $nice 3 process_id -> set priority of process to 3, 1-15
44. $kill process_id -> terminate a process