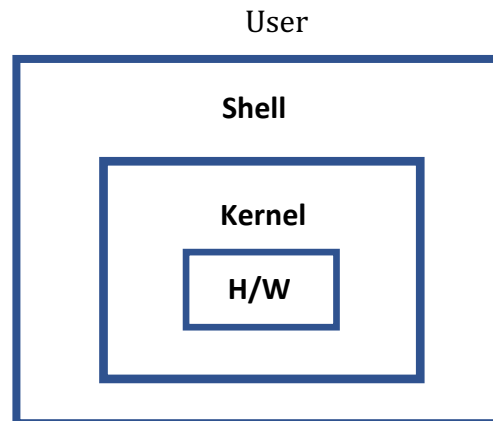


Unix

It is portable operating system that is designed for multi-tasking and multi user functions. It was written in C and it is control under shell



Kernel :- Kernel in Unix OS is the master program that control the computer resource. Kernel does not deal directly with user. User need to write the program / command in shell

Shell :- it is interface between user and system. And it is command interpreter that execute the command

There are two types of logging screen

- 1) CUI (character user interface) Virtual console
- 2) GUI (Graphical user interface) Graphical console

Unix :- It run on CUI and it is licensed and Free ware depends on the flavor

Linux :- It is like Unix but not Unix, freely available to everyone (RedHat, Ubuntu, Fedora) Mac

Terminal :- Used to write the command and get output

Student@localhost ~]\$

User, machine_name home_directory Unix prompt

In Unix folder is said to be directory

Feature of Unix

- Multiuser :- multiple user access the system by connecting to points is known as terminal
- Multitasking :- several user can run multiple program on one system
- Provide better security by specific user permission

Unix Commands

- 1) man :- It give the manual of command

To quit press q button

- 2) pwd :- shows current working directory (present working directory)

- 3) ls :- shows the non-hidden content of current directory

ls -a :- Shows hidden content of current directory

ls -l :- gives the details of files in directory

ls -al :- gives the details of hidden files in directory

ls -r :- gives the file list in reverse order (oldest first)

ls -R :- gives details of directory, sub directory and present files in directory

- 4) cd :- change current working directory to specific directory

Syntax :- cd directory_name => cd Desktop

cd .. :- move one step back

cd **OR** cd ~ :- Navigate to home directory

- 5) mkdir folder1 :- create new directory of name folder1

mkdir {folder1,folder2,folder3} :- create multiple new directory

mkdir folder1/test1 :- create sub directory if directory is available

mkdir folder1/{test1,test2} :- create multiple sub directory

mkdir -v folder10 :- created directory with **success message**

mkdir -p folder10/test1 :- create directory with sub directory **when directory hasn't been created** previously

- 6) rmdir folder10 :- remove empty directory

rmdir (folder2,folder3) :- remove multiple empty directory

7) `rm a.txt` :- remove the file
`ls`

`rm -i a.txt` :- remove the file with confirmation
`y`

`rm -r folder1` :- remove non empty directory

8) `wc automation.txt` :- return no of lines, words and characters of file

9) `vi java.txt` :- It is editor in Unix, create new file and if present then open it
`esc + : + wq + enter`

10) `cat java.txt` :- It read the data from file

`cat java.txt javaprogram.txt` :- Read data from multiple files

`cat -n java.txt` :- It read the data from file with line number

11) `grep` :- global regular expression pattern, used to search the text
Syntax :- `grep "automation" java.txt`

Syntax :- `grep -c "am" java.txt` :- gives count of line where search word exist

12) `touch abc.txt` => create blank file, but it has no editor window

13) `head automation.txt` => shows first 10 lines

`head -11 automation.txt` => shows first 11 lines

`tail automation.txt` => shows last 10 lines

`tail -11 automation.txt` => shows last 11 lines

14) `cp automation.txt automation1.txt` => copy content of one file to other file
(replace)

`cp abc.txt /Users/Zenith/Desktop/Suraj` => copy one file to directory

15) `mv automation.txt automationByJava.txt` => Rename the file
`cat automationByJava.txt`

`mv Unix InnerUnix` => Rename the directory

ls

mv a.txt /Users/Zenith/Desktop/UnixDes => Moving file from one directory to other

16) less/more => display content of file page by page
less abc.txt => shows content of file
space => next page
b => back page
G => last page
g => first page
To exit from file press q

17) top => Display all running process

18) kill processed => kill or terminate the process
kill process_d => kill 123

19) Ps => show current working process

20) chmod

drwxr-xr-x 11 Zenith staff 352 Feb 19 20:08 Learning
permission-field-user_group-group_owner-size-date-file_name

There are three types of permission

r – read w – write x – execute

at the start of file details

“-”Indicate file

“d” – indicate directory

“l” indicate link

There are three types of owner

User owner => u

Group owner => g

Others => o

Changing permission by symbolic method

Syntax :- chmod whowhatwhich file/directory

Ex. chmod o+w file.txt

Where who => u, g, o for user, group, other user

what => +, -, = for add remove assign

which => r, w, x for read write execute permission

to add permission => `chmod g+rw file.txt`

to remove permission => `chmod g-rw file.txt`

to assign permission => `chmod o=u file.txt`

set all the permission => `chmod o+rwx file.txt`

Changing permission by numeric way

4 – read 2 – write 1 – execute

Ex. r w x
 ↓ ↓ ↓
 4 2 1 Total = 7

If there is a permission which is 640 then

6	4	0
user	group	other
(4 + 2 + 0)	(4 + 0 + 0)	(0 + 0 + 0)

User 547 permission

`chmod 547 file.txt`

o/p =>

Output redirection (>) (>>)

Syntax :- `command > output file name`

Ex. `ls > file.txt`

All the content of ls command will be copied to file.txt and if file exist then it will overwrite

If we want previous content too then use below syntax

`ls >> file.txt`

Soft link and hard link.

Soft link => It is symbolic link called as shortcut of file

Hard link => It is copy of file called as backup of file

Both link work as pointers in linux

Creation of soft link

Syntax :- `ln -s original_file_name softlink_file_name`

Ex. `ln -s file.txt newfile.txt`

Creation of hard link

Syntax :- `ln original_file_name hardlink_file_name`

Ex. `ln file.txt backupfile.txt`

Shows all the links

`ls -li` => show all nodes

- inode (index node) of backup file and original file are same
- inode of softlink file and original file is different
- if we made changes in original file same changes will be observed in softlink and not in hardlink file
- if we delete original file then hardlink file will be as it is and short link file not get deleted (but it is unable to navigate/open the original file)