

OS Practicals codes

- Krishna Patil

* OSPr1 :- Demonstration of Linux commands with attributes:-
pwd, cd, ls, more, less, echo, clear, kill, ps, man,
cat, date, who, whoami, wc, mbatis, ematis, rm
and sort.

1. ls (list) :- (Lists contents of directory)

a) -l : Long format, displays detailed information about files, including permission, owner, size, modification date, etc.

b) -a : All, shows hidden files starting with a dot (e.g., .bashrc).

Syntax :- ls

2. cd (Change directory) :- (Changes directory)

a) directory-name :- Changes to the specific directory.

b) .. :- Move one level up in the directory hierarchy.

c) - :- Switch to previous directory.

Syntax :- cd directory-name



3. pwd (Print working directory) :-

- display's the current working directory's full path.

Syntax :- pwd

4. cp (copy) :-

a) -r :- Recursive, copy directory & their contents.

b) -i :- Interactive prompt before overwriting existing files.

Syntax :- cp source destination

5. mv (move or rename) :-

-i :- Interactive prompt before overwriting existing files.

Syntax :- mv source destination

6. rm (remove) :- delete

- a) -r : Recursive, remove directory & their contents
- b) -f : Force, suppress prompt before removing write-protected files.
- Syntax :- rm filename

7. mkdir (make directory) :-

- -p : Create parent directories if they don't exist.
- Syntax :- mkdir directory-name

8. rmdir (remove directory) :-

- Removes empty directory.
- Syntax :- rmdir directory-name

9. cat (concatenate) :-

- Displays the contents of a file
- Syntax :- cat filename

10. less :-

- A page program for viewing file content interactively.
- Syntax :- less filename

11. head & tail :-

- a) Display the first or last few lines of a file.

b) -n : Specify the number of lines to display (e.g., head -n 10 file.txt). (10 is default).

• head -n no filename

• tail -n no filename

c) -f : Outputs appended data as the file grows. Useful for monitoring log files in real-time (only for tail).

12.

(2) more :-

- i) 'more' command is used to display the contents of a text file one screen at a time.
- ii) Allows user to read through large files or long outputs in a controlled manner.

iii) Syntax :- more [options] [file]

iv) Attributes :-

a) '-d' :-

Display the filename & line number at the top of the screen.

b) '-F' :-

Count logical lines rather than screen lines.

c) '-p' :-

Do not scroll, clear the whole screen, display the file.

d) '-c' :-

Clear the screen before displaying the file.

e) '+n' :-

Start displaying the file from line number n.

f) '/pattern' :-

Search for a pattern in the file & display the lines containing the pattern.

e.g.: - ↴ attributes are

more filename.txt

(3) echo :-

- i) Used to display line of text or string on screen.

ii) Syntax :- echo [options] [string(s)]

iii) Attributes :-

a) '-n' :-

i) suppresses the newline.

ii) By default 'echo' adds a newline character at end of each present line.

b) '-e' :-

(13)

III) b) '-e' :-

i) rarely use of escape sequence character

ii) e.g.: \n for newline, \t for tab.

c) '-E' :-

Directly interpretable of escape sequence character

IV) e.g.:-

echo "Hello World!"

(14)

clear :-

Clear screen.

(15)

kill :-

i) Used to terminate or send signals to processes, allowing users to manage running programs & programs effectively.

ii) Syntax :-

kill [options] [PID]
 | process ID

iii) Attributes :-

a) '-<SIGNAL>' or '--signal=<SIGNAL>' :-

i) specifies the signal to send.

ii) Common signals include 'TERM' (15, default) for termination, 'HUP' (1) for hang-up & 'KILL' (9) for immediate termination.

b) Same as this followed by SIG :-

i) e.g.: 'kill SIGTERM'.

ii) same as specifying the signal number directly but with the 'SIG' prefix.

c) '-l' or '--list' :-

lists available signal names

iv) e.g.,

kill -SIGKILL 5678

| process ID

16) ps :-

- i) Used to provide information about the currently running processes in the system.
- ii) Syntax :- `ps [options]`
- iii) Attributes :-
 - a) '-a' :- Shows process for all users.
 - b) '-u' :- Displays detailed information, including the user owning the process, CPU usage, memory usage, etc.
 - c) '-x' :- Shows process without controlling terminals.
 - d) '-e' :- Displays information about all the processes.
 - e) '-f' :- provides full-format listing.
- iv) E.g.,
`ps`

17) man :-

- i) Used to display the manual pages of other commands and provides documentation of that Linux command.
- ii) Syntax :-
`man [Command]`
- iii) Attributes :-
 - a) '-k' or '-- apropos' :- searches the manual page names & descriptions for the specified keyword.
 - b) '-f' or '-- whatis' :- Displays a concise description of the specified command.
 - c) '-C' :- specifies the configuration file to use for formatting the output.
 - d) '-S' :- specifies the section of the manual to search. By default, it searches all sections.

iv) E.g.,

`man ls`

18)

cal :-

i) used to display calendar on the terminal.

ii) syntax :- `cal [options] [month] [year]`

iii) Attributes :-

a) '`-1`' :- Display one month only.b) '`-3`' :- Display the previous, current & next month.c) '`-y`' :- Display a calendar for the entire year.d) '`-m`' :- Display Monday as first day of week.

(Default is Sunday).

e) '`-j`' :- Display Julian date (days since January 1, 4713 BC, Greenwich noon).f) '`-h`' :- suppress the header.g) '`-s`' :- Display Sunday as first day of week (default).iv) `log:``cal`

19)

date :-

i) Used to display & set the system date & time.

ii) syntax :- `date [option] [+FORMAT]`

iii) Attributes :-

a) '`-u`' :- Displays or sets the Coordinated Universal Time (UTC).b) '`-R`' :- Displays the date & time in RFC 2822 format.c) '`-d`' :- y m d a string.d) '`-s`' :- Sets the system date & time. Requires superuser privilege ('sudo').e) '`+FORMAT`' :- Customizing the output format specifying (e.g., '%Y' for year, '%m' for month, '%d' for day).iv) `log:``date`

20) who :-

- i) Used to display information about currently logged-in users on the system.
- ii) Syntax :- who [options]
- iii) Attributes :-
 - a) '-a' :- Displays all information same as who "-bdprt".
 - b) '-b' :- Displays the time of the last system boot.
 - c) '-d' :- Shows dead processes (processes that have ended).
 - d) '-l' :- Displays login name & terminal line numbers.
 - e) '-p' :- Shows active processes spawned by 'init'.
 - f) '-r' :- Displays the current runlevel.
 - g) '-T' :- Shows the last change in Terminal.
 - h) '-t' :- Shows last system clock change.
- iv) E.g.,
who

21) whoami :-

- i) Used to display username of currently logged in user.
- ii) Syntax :- whoami
- iii) E.g., whoami

22) wc :- (word count) :-

- i) Used to count number of words, lines and characters in a text file or input provided through standard input stream.
- ii) Syntax :- wc [Options] [file]
- iii) Attributes :-
 - a) '-w' :- count no of words in a file
 - b) '-l' :- count no of lines
 - c) '-c' :- count no of characters
 - d) '-m' :- count the number of characters, including spaces.
 - e) '-L' :- display length of longest file.

22) `wc :-`

IV) E.g.,

`wc -w filename.txt`

23) `sort :-`

I) Used to sort lines of text files or command output in ascending or descending order.

II) Syntax :- `sort [options] [file]`

III) Attributes :-

a) '`-r`' :- Sort in descending order (reverse).

b) '`-n`' :- Perform a numeric sort.

c) '`-k`' :- 'KEYDEF' :- Sort based on a 'KEYFIELD' (e.g. '`-k2,2`' sorts based on second field).

d) '`-t`' :- 'DELIMITER' :- Use 'DELIMITER' as the field separator character.

e) '`-b`' :- Ignore leading blanks.

f) '`-u`' :- Output only the unique lines after sorting.

g) '`-c`' :- Check if the file is already sorted: do not sort if it's already sorted.

h) '`-o`' :- 'OUTPUT-FILE' :- Write the sorted output to 'OUTPUT-FILE'.

IV) E.g.,

`sort filename.txt`