

# Question 1

---

## awk

- Description:
  - The awk command is a scripting language used for processing and displaying text.
- Syntax/Formula:
  - `awk + options + {awk command} + file + file to save`
- Example
  - `awk '{print}' jobs.txt`  
`awk '{print $1,$4}' jobs.txt`  
`awk '{print $1,NF}' jobs.txt`

## cat

- Description:
  - The cat command is used for displaying the content of a file.
- Syntax/Formula:
  - `cat + option + files`
- Example
  - `cat tree.lst`  
`cat -n ~/Documents/tree.md`  
`cat -E ~/Documents/tree.md`

## cp

- Description:
  - The cp command is used to copy files and directories from a source to a destination.
- Syntax/Formula:
  - `cp + file to copy + destination`
- Example
  - `cp Downloads/screenshots.jpg Pictures/`  
`cp -r ~/Downloads/screenshots ~/Pictures/`  
`cp Downloads/screenshots/* ~/Pictures/`

## cut

- Description:
  - The cut command is used to extract a specific section of each line of a file and display it to the screen.
- Syntax/Formula:
  - `cut + option + file`
- Example
  - `cut -b 1-10 employeenames.txt`  
`cut -d ',' --complement -s -f7 employeenames.txt`  
`cut -b 1,2,3 employeenames`

## grep

- Description:
  - The grep command is used to search text in given file. Grep works line by line basis.
- Syntax/Formula:

- `grep + option + search criteria + file`
- Example
  - `grep -c "test" final test.txt` `grep -i 'finaltest' ~/Documents/tests/finaltest.txt` `grep -o 'finaltest' ~/Documents/tests/finaltest.txt`

## head

- Description:
  - The head command displays the top N number of lines of given file. It prints the first 10 lines.
- Syntax/Formula:
  - `head + option + file`
- Example
  - `head -n 20 state.txt` `head -v state.txt` `head -q state.txt`

## ls

- Description:
  - The ls command is used for displaying all the files inside a given directory.
- Syntax/Formula:
  - `ls + option + directory to list`
- Example
  - `ls -als` `-l ls` `-A`

## man

- Description:
  - The man command in Linux is used to display the user manual of any command that we can run on the terminal.
- Syntax/Formula:
  - `man + option + command name`
- Example
  - `man drinks` `man 5 drinks` `man -f drinks`

## mkdir

- Description:
  - The mkdir command is used for creating a single directory or multiple directories.
- Syntax/Formula:
  - `mkdir + name of the directory`
- Example
  - `mkdir games` `mkdir -p action/horror/comedy` `mkdir -v games`

## mv

- Description:
  - The mv command is used to moves and rename directories.
- Syntax/Formula:

- `mv + source + destination`
- Example
  - `mv food /home/grocery mv Download/food.txt Documents/ mv phones/ music/ sounds/ media/iphone/`

## tac

- Description:
  - The `tac` command is used for displaying the content of a file in a reverse order.
- Syntax/Formula:
  - `tac + option + files`
- Example
  - `tac bike.md tac ~/Documents/bike.md tac -r bike.md`

## tail

- Description:
  - The `tail` command displays the last N number of lines of a given file. It prints the last 10 lines.
- Syntax/Formula:
  - `tail + option + file`
- Example
  - `tail -n 7 streets.txt tail -c 7 streets.txt tail -v streets.txt`

## touch

- Description:
  - The `touch` command is used for creating files.
- Syntax/Formula:
  - `touch + list`
- Example
  - `touch videos touch -a videos touch -c videos`

## tr

- Description:
  - The `tr` command is used for translating or deleting characters from standard output.
- Syntax/Formula:
  - `standard output | tr + option + set + set`
- Example
  - `cat movies.mov | tr [a-z] [A-Z] cat movies.mov | tr '.' ',' cat movies.mov | tr "[:space:]" '\t'`

## tree

- Description:
  - The `tree` command is used for displays directory paths and files in each subdirectory.
- Syntax/Formula:
  - `tree + option`

- Example
  - `tree tables tree -p clothes.txt tree -a ./brands`

## Question 2

---

### How to work with multiple terminals open?

Having several open terminals can be helpful for a variety of tasks, like monitoring various processes or executing multiple commands at once.

### How to work with manual pages?

Man pages, is an essential ability for navigating and using command-line utilities on operating systems similar to Unix.

### How to parse (search) for specific words in the manual page

Just hit `/`, and type your search pattern or find all of the long arguments

### How to redirect output (`>` and `|`)

The `>` operator is used to redirect the standard output of a command to a file. `command output + > + file` To pass the output of one command as the input of another, use the pipe operator `|`. `ls | grep "pattern"`

### How to append the output of a command to a file

`command >> filename` If the file doesn't already exist, this operator creates it and appends the output to it.

### How to use wildcards

`ls -A *apple*` this command will list all the files that have 'apple' in the name.

### How to use brace expansion

`mkdir -p video/{movies,shows}/{movfiles,films}/new{1..2}` creates 2 different directories inside video directory