

Question 1

awk

- Description:
 - a scripting language used for processing and displaying text
- Syntax/Formula:
 - `awk + options + {awk command} + file`
- Examples:
 - Print first field of `/etc/passwd` file
 - `awk -F: '{print $1}' /etc/passwd`
 - Print the last field of the `/etc/passwd` file
 - `awk -F: '{print $NF}' /etc/passwd`
 - Prints the length of a line(record)
 - `awk '{print length($0)}' /etc/passwd`

cat

- Description:
 - used for displaying the content of a file
- Syntax/Formula:
 - `cat + option + file(s) to display`
- Examples:
 - Display the content of a file located in the pwd
 - `cat todo.list`
 - How to see content of a file with line numbers
 - `cat -n ~/Documents/todo.md`
 - Display the content of a file with line numbers excluding empty lines
 - `cat -b ~/Documents/todo.md`

cp

- Description:
 - copies files/directories from a source to a destination
- Syntax/Formula:
 - `cp + files to copy + destination`
- Examples:
 - To copy a file
 - `cp Downloads/wallpaper.zip Pictures/`
 - To copy a directory with absolute path
 - `cp -r ~/Downloads/wallpapers ~/Pictures/`
 - To copy the content of a directory to another directory
 - `cp Downloads/wallpapers/* ~/Pictures/`

cut

- Description:
 - used to extract a specific section of each line of a file and display it to the screen
- Syntax/Formula:
 - `cut + option + file(s)`
- Examples:
 - Display a list of all the users in your system
 - `cut -d ':' -f1 /etc/passwd`
 - Cut a range of bytes per line
 - `cut -b 1-5 usernames.txt`
 - Cut a file excluding a given field
 - `cut -d ',' --complement -s -f3 users.txt`

grep

- Description:
 - used to search text in given file. matches the search criteria in a line by line basis
- Syntax/Formula:
 - `grep + option + search criteria + file(s)`
- Examples:
 - Search any line that contains the word 'dracula' in the given file
 - `grep 'dracula' ~/Documents/dracula.txt`
 - Search any line that contains the word 'dracula' regardless of the case
 - `grep -i 'dracula' ~/Documents/Books/dracula.txt`
 - Search for all the lines that do not contain the word 'war'
 - `grep -v 'war' ~/Documents/Books/war-and-peace.txt`

head

- Description:
 - displays the top N number of lines of a given file. By default, it prints the first 10 lines
- Syntax/Formula:
 - `head + option + file(s)`
- Examples:
 - Display the first 10 lines of a file
 - `head ~/Documents/Book/dracula.txt`
 - Display the first 5 lines of a file
 - `head -5 ~/Documents/Book/dracula.txt`

ls

- Description:
 - used for listing the content of a given directory or the file/directory itself
- Syntax/Formula:
 - `ls + option + directory to list`
- Examples:
 - List the content of the present working directory
 - `ls`
 - List all the files inside the current working directory including hidden files

- `ls -a`
- List all the files in a given directory sorted by file size
 - `ls -S ~/Documents`

man

- Description:
 - displays detailed information about a command, including its syntax, options, and examples
- Syntax/Formula:
 - `man + command`
- Examples:
 - To display the manual page for the 'grep' command
 - `man grep`
 - To display all the information containing the word 'search'
 - `man -k search`
 - To show all the available pages of a command
 - `man -a passwd`

mkdir

- Description:
 - used for creating a single directory or multiple directories
- Syntax/Formula:
 - `mkdir + the name of the directory`
- Examples:
 - Create a directory in the present working directory
 - `mkdir wallpapers`
 - Create multiple directories
 - `mkdir wallpapers/cars wallpapers/cities wallpapers/forest`
 - Create a directory in a different directory using absolute path
 - `mkdir ~/wallpapers/forest`

mv

- Description:
 - moves and renames directories
- Syntax/Formula:
 - `mv + source + destination`
- Examples:
 - To move a file from a directory to another using relative path
 - `mv Downloads/homework.pdf Documents/`
 - To move a file from one directory to another using absolute path
 - `sudo mv ~/Downloads/theme /usr/share/themes`
 - To move multiple directories/files to a different directory
 - `mv games/ wallpapers/ rockmusic/ /media/student/flashdrive/`

tac

- Description:
 - used for displaying the content of a file in reverse order
- Syntax/Formula:
 - `tac + option + file(s) to display`
- Examples:
 - Display the content of a file located in the pwd
 - `tac todo.md`
 - Display the content of a file using absolute path
 - `tac ~/Documents/todo.md`

tail

- Description:
 - displays the last N number of lines of a given file.
- Syntax/Formula:
 - `tail + option + file`
- Examples:
 - Display the last 10 lines of a file
 - `tail ~/Documents/Book/dracula.txt`
 - Display the last 5 lines of a file
 - `tail -5 ~/Documents/Book/dracula.txt`

touch

- Description:
 - used for creating files
- Syntax/Formula:
 - `touch + file name`
- Examples:
 - To create a file called list
 - `touch list`
 - To create a file using absolute path
 - `touch ~/Downloads/games.txt`
 - To create several files
 - `touch list_of_cars.txt script.py names.csv`

tr

- Description:
 - used for translating or deleting characters from a standard output
- Syntax/Formula:
 - `standard output | tr + options + set`
- Examples:
 - to change all lowercase letters in the text to uppercase
 - `$ cat linux.txt | tr [:lower:] [:upper:]`
 - To remove spaces in the domain names using the following command
 - `$ cat domains.txt | tr -d ' '`
 - How to translate one character to another

- `cat file.txt | tr '.' ','`

tree

- Description:
 - used to display the directory tree of a given directory
- Syntax/Formula:
 - `tree + directory`
- Examples:
 - To display the directory tree of a current directory, excluding hidden files and directories
 - `tree -a`
 - Display a directory in a tree format
 - `tree /Documents`

vim/nano

- Description:
 - Two command line text editors
- Syntax/Formula:
 - `vim`
 - `nano`
- Examples:
 - To open an existing file called 'myfile.txt' in vim
 - `vim myfile.txt`
 - To open an existing file called 'myfile.txt' in nano
 - `nano myfile.txt`
 - To search for text in nano
 - `Ctrl+W`

Question 2

- **How to work with multiple terminals open?**
 - open one terminal the open another terminal and set them side by side. Or user tilix and split the terminal as needed.
- **How to work with manual pages?**
 - By using the `man + command` to display how it works
- **How to parse (search) for specific words in the manual page?**
 - To find a specific word using the manual page use `man -k + file`
- **How to redirect output (> and |)?**
 - To redirect the output of '>' use `ls > myfiles.txt`
 - To redirect the output of '|' use `ls | grep`

- **How to append the output of a command to a file?**
 - To append the output of a command to a file called 'myfiles.txt' you would use `ls >> myfiles.txt`
- **How to use wildcards (For copying and moving multiple files at the same time)?**
 - You would use `cp /path/to/files*.txt /path/to/destination/`
- **How to use brace expansion (For creating entire directory structures in a single command)?**
 - `mkdir -p music/{jazz,rock}/{mp3files,videos,oggfiles}/new{1..3}`