

Question 1

Awk

- Description: * Awk is a scripting language used for processing and displaying text. Awk can work with a text file or from standard output.
- Formula:
 - 'awk + options + {awk command} + file'
- 'command output' | 'awk + options + {awk command}'
- Examples:
- how to print the first field of a file:
 - `awk -F:' ' '{print $1}' /etc/passwd`
- how to start printing from a different line
 - `'awk 'NR > 3 {print}' /etc/passwd`
- how to change a field to upper case:
 - `awk -F: '{print toupper($1)}'`

cat

- Description:
 - the cat is used for displaying the content of a file. cat is short for concatenate which is its command intended use.
- Formula: * cat + option +file(s) to display
 - Example: Display the content of a file located in the pwd
 - `cat todo.lst`
- Display the content of a file using absolute path * `cat ~/Documents/todo.lst`

cp

- Description:
- cp copies files/directories from a source to a destination.
- Formula:
 - `cp + file to copy + destination`
 - `cp Downloads/wallpapers.zip pictures/`

- Example:
 - `cp Downloads/wallpapers.zip pictures/`
 - `cp -r ~/Downloads/wallpapers ~/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.
- formula:
 - `cut + file(s)`
- Example:
 - Display a list of all the users in your system:
 - `cut -d ';' -f1 /etc/passwd`
 - Display a list of all the users in your system with their login shell * `cut -d ';' -f1,7 /etc/passwd`

grep

- Description: Grep is used to search text in given file. Grep work line by basis(it matches the search criteria in a line by line basis).
 - Formula:
 - `grep + option + search criteria + file(s)`

*Example: Search any line that contains the word "dracula" in the given file: * `grep 'dracula' ~/Documents/dracula.txt`

head

- Description:
 - The head command displays the top N number of lines of a given file. By default, it prints the first 10 lines. If more than one file name is provided then data from each file is preceded by its file name.
- Formula:
 - `'head + option + file(s)`
- Example:
 - Display the first 10 lines of a file
 - `head ~/Documents/Book/dracula.txt`
 - Display the first 5 lines of a file: * `head -5 lines of a file`

ls

**** Description:**

- Used for displaying all the files inside a given directory. When no directory specified, ls displays the files in the current working directory
- Formula:
 - ls + option + directory to list
- Example:
 - ls
 - ls -a
 - ls -a ~/pictures
 - ls -lR ~/Pictures

man

- Description:
 - man(manual) pages are documentation files that describe Linux shell commands, executable programs, system calls, special files, and so forth.
 - Example: man ls

mkdir

- Description:
- mkdir is used for creating a single directory or multiple directories.
- Formula:
 - mkdir + the name of the directory
- Example:
 - mkdir wallpapers
 - mkdir wallpapers/ocean
 - mkdir ~/wallpapers/forest

mv

- Description:
 - mv moves and renames directories.
 - formula: mv + source + destination
 - mv + file/directory to rename + new name.
- Example:
- mv Downloads/homework.pdf Documents/
- sudo mv ~/Downloads/theme /usr/share/themes

tac

- Description:
 - The cat command is used for displaying the content of a file. Cat is short for concatenate which its command intended use.
- Formula:
 - 'tac + option + file(s) to display
- Example: 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:
 - the tail command displays the last N number of lines of a given file. By default, it prints the last 10 lines. If more than one file name is provided then data from each file is preceded by its file name.
- Formula: * `'tail ~/Documents/Book/dracula.txt`
- Example:
 - 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:
 - touch is used for creating.
 - Formula:
 - `touch list`
- Example:
 - `touch list of_cars.txt script.py name.csv`

tr

- description:
 - Deletes or substitutes characters from input and writes the result to standard output.
 - example:

- `cat domain.txt`

tree

- A recursive directory listing programs which generates depth inside indented file's list.
 - Example:
 - `tree -a DirectoryNamePath`

How to work with multiple terminals open?

The work with multiple terminals open the shift + ctr; + o keyboard to split the window horizontally and shift + ctrl + eE to split vertically.

How to work with manual pages?

- the work wit manual page is opening your terminal emulator, or logging into into your server via ssh.

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

The parse (search) for specific words in the manual page to help in terminal with -i help has different way to search with specific word.

How to redirect output (> and |)

- The redirect output (> and|) the symbol> is used to redirect output by taking the output from the command on left and passing as input to the file on the right

How to append the output of a command to a file

- the append the output of a command to a file append text to end using echo command: `echo "text here" >> filename` and append command `~name >> filename`.

How to use wildcards

****** For copying and moving multiple files at the same time

- The main wildcard is a start, or asterisk(*) character.
- for example, `ls *.txt` will match all files that end in .txt regardless of the size of the file name.

How to use brace expansion

****** For creating entire directory structures in a single command ****** Brace expansion {}is not a wildcard but another feature of bash that allows you to generate arbitrary string to use with commands.