Week Report 7

Question

Question 1

- The 'cat' command is used for displaying the content of a file. Cat is short for concatenate which is its command intended use.
- The formula of the command is "cat + option + file(s) to display"
- Examples:
  - cat apple.txt
  - cat ~/Documents/apple.txt

Question 2

- The 'tac' command is used for displaying the content of a file in reverse order. Just like 'cat', 'tac' concatenates files and displays the output of the concatenation
- The formula of the command is "tac + option + file(s) to display"
- Example:
  - tac todo.md
  - tac ~/Documents/todo.md

Question 3

- 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 that data from each file is preceded by its file name.
- The formula of the command is "head + option + file(s)"
- Example:
  - head ~/Documents/Book/bible.txt
  - head -5 ~/Documents/Book/bible.txt

Question 4

- 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
- The formula of the command is "tail + option + file"
- Example:
  - tail ~/Documents/Book/bible.txt
  - tail -5 ~/Documents/Book/bible.txt

Question 5

- The 'cut' command is used to extract a specific section of each line of a file and display it to the screen
- The formula of the command is "cut + option + file(s)"
- Example:
  - cut -d ':' -f1 /etc/passwd

    - cut -d ':' -f1,7 /etc/passwd

Question 6

- The 'paste' command is used for joining files horizontally in columns
- The formula of the command is "paste + option + files"
- Example:
    - paste users.lst ip_address.lst
    - paste -d ":" users1.lst ip_addresses.lst

Question 7

- The 'sort' command is used for sorting files. The sort command supports sorting:
    - alphabetically, in reverse order, by number, and by month.
- The 'sort command follows this order unless specified otherwise:
    - Lines starting with a number will appear before ines starting with a letter
    - Lines starting with a letter that appears earlier in the alphabet will appear before lines starting with a letter that appears later in the alphabet
    - Lines starting with a lowercase letter will appear before ines starting with the same letter in uppercase
- The formula of the command is "sort + option + file"
- Example:
    - sort -r bible.txt
    - sort -k 2 bible.txt

Question 8

- The 'wc' command is used for printing the number of lines, characters and bytes in a file
- The formula of the command is "wc + option + file(s)"
- Example:
    - wc -m bible.txt
    - wc -l bible.txt

Question 9

- The 'tr' command is used for translating or deleting characters from standard output.
- The formula of the command is "standard output | tr + option + set + set"
- Example:
    - cat bible.txt | tr '.' ','
    - cat bible.txt | tr -s "[:space:]" ' '

Question 10

- The 'diff' command compares files and displays the differences between them
- The formula of the command is "diff + option + file1 + file2"
- Example:
    - diff bible.txt dracula.txt
    - diff -y bible.txt dracula.txt

Question 11

- 'grep' is used to search text in given file. 'grep' works line by line basis (it matches the search criteria in a line by line basis)
- The formula of the command is "grep + option + search criteria + file(s)"
- Example:
  - grep 'bible' ~/Documents/Books/bible.txt
  - grep -n 'dracula' ~/Documents/Books/dracula.txt

Question 12

- 'awk' is a scripting language used for processing and displaying text. 'awk' can work with a text file or form standard output. 'awk' was created in Bell Labs during the 70s by Alfred Aho, Peter Weinberger, and Brian Kernighan and its name comes from its authors' initials. There are several implementations of 'awk': nawk, mawk, gawk, and busybox.
- 'awk' preforms operations line by line
- The formula of the command is "awk + options + {awk command} + files + files to save (optional)
- Example:
  - awk '{print $1}' ~/Documents/Csv/cars.csv
  - awk -F: '{print $1}' /etc/passwd
  - awk -F: '{print $NF}' /etc/passwd
  - awk -F: '{print $FILENAME}' /etc/passwd
  - awk -F: '{print toupper($1)}' /etc/passwd

Question 13

- 'sed' is a stream editor that preform operations on files and standard output. For instance it can search, find and replace, insert, and deletion. By using 'sed' you can edit files without opening them
- The formula of the command is "sed options + sed script + file"
- Example:
  - sed 's/pizza/rice/' shopping-list.lst
  - sed 's/pizza/rice/4' shopping-list.lst
  - sed '3 s/pizza/rice/' shopping-list.lst
  - sed '5d' shopping-list.lst
  - sed '$d' shopping-list.lst