Linux Admin

- In Linux, Admin is called as "super-user".
- Admin's login name is "root".
- Most of modern Linux, disable "root" login (for security).
- To execute commands with admin privileges use "sudo" (if approved by system admin).
    - cmd> sudo apt update
    - cmd> sudo apt install vim gcc python3 python3-pip
    - cmd> sudo snap install --classic code

Directory commands

- pwd -- print present working directory (current directory)
- cd -- change directory (syntax> cd dirpath)
- ls - list directory contents (syntax> ls dirpath)
- mkdir -- make directory (syntax> mkdir dirpath)
- rmdir -- remove empty directory (syntax> rmdir dirpath)
- cd
    - cd ~ - change working directory to home directory
    - cd - - change working directory to old working directory
    - cd .. - change working directory to parent directory

File commands

- cat
    - cat > filepath <-- create new file
    - cat filepath <-- display file contents
- rm
    - rm filepath <-- delete given file
    - rm -r dirpath <-- delete dir with all contents
- mv
    - mv filepath destdirpath <-- move given file into given dest directory
    - mv dirpath destdirpath <-- move given dir into given dest directory
    - mv oldname newname <-- rename given file
- cp
    - cp filename newfilename <-- copy file with new name/path.
    - cp filepath destdirpath <-- copy file into given dest dir with same name.
    - cp -r dirpath destdirpath <-- copy file into given dest dir with same name.

Linux commands

- cd
    - cd ~ - change working directory to home directory
    - cd - - change working directory to old working directory
    - cd .. - change working directory to parent directory
- ls

    - ls - list the contents of present working directory
    - ls path - list the contents of given path
    - ls -l - list the contents in detail format
        - type and permissions
            - Types of files
                - Regular file (-)
                - Directory file (d)
                - Link file (l)
                - pipe file (p)
                - socket file (s)
                - character special file (c)
                - block specical file (b)
            - Permissions of files
                - r - read, w - write, x - execute
                - (rwx)user/owner, (rwx)group, (rwx)others
        - link count
        - user/owner
        - group
        - size
        - timestamp
        - name
    - ls -a - list all contents along with hidden
    - ls -A - list all contents along with hidden except . and ..
    - ls -i - list contents with indoe number
        - inode number is unique number given to every file
    - ls -s - list content with size (number of blocks)
    - ls -S - list content in descending order of their sizes

- touch
    - if file does not exist, empty file is created
    - if file exist, timestamp of that file is changed

- stat
    - stat file - display information of file
    - stat file1 file2 - display information of file1 and file2
    - stat -c "format" file - display file information in given format

- head
    - head file - display first 10 lines
    - head -5 file - display first 5 lines

- tail
    - tail file - display last 10 lines
    - tail -4 file - display last 4 lines

- sort

    - sort file - sort the content by alphabetically
    - sort -n file - sort the content by their value
        - sort command do not modify file content

- uniq
    - uniq file - display contents uniquely (truncate duplicate)
        - truncate duplicate content if it is consecutive

- rev filepath
    - Print each line reversed.
    - File contents are not modified.

- tac filepath
    - Print all lines in reverse order. The first line printed at last, while last line printed first.

- stat path
    - Display info about file or directory.

- alias
    - alias list="ls -l"
        - list will be alias/nick name to ls -l
        - list will give output same as ls -l

- unalias
    - unalias list

- which
    - which command
        - display the location of command executable.

- whereis
    - whereis command
        - display the location of command executable and also manual paage location.

Redirection

- for every command input is taken from terminal, output is printed on terminal and error is also printed on terminal
- Standard streams (by defualt for every process, three files are opened)
    - stdin
    - stdout
    - stderr
- There are three types of redirections
    - input redirection

        - input will be taken from file instead of stdin
        - to do input redirection '<' symbol is used
        - command < file
    - output redirection
        - output will be written into file instead of stdout
        - to do output redirection '>' or '>>' symbol is used
        - command > file
            - older content of file will be over written
        - command >> file
            - content will be appneded into file at the end
    - error redirection
        - error will be written into file instead of stderr
        - to do output redirection '2>' or '2>>' symbol is used
        - command 2> file
            - older content of file will be over written
        - command 2>> file
            - content will be appneded into file at the end

Pipe

- Using pipe, we can redirect output of any command to the input of any other command.
- Two processes are connected using pipe operator (|).
- Two processes runs simultaneously and are automatically rescheduled as data flows between them.
- If you don't use pipes, you must use several steps to do single task.
- command1 | command2
    - output of command1 will be given as input to command 2
- E.g.
    - who | wc

Shell meta characters

- '*' - zero or more occurances of any character
- '?' - one occurance of any character

Regular Expressions

- Find a pattern in text file(s).
- Regular expressions are patterns used to match character combinations in strings.
- A regular expression pattern is composed of simple characters, or a combination of simple and special characters e.g. /abc/, /ab*c/
- Pattern is given using regex wild-card characters.
    - Basic wild-card characters
        - $ - find at the end of line.
        - ^ - find at the start of line.
        - [ ] - any single char in give range or set of chars

        - [^] - any single char not in give range or set of chars
        - . - any single character
        - zero or more occurrences of previous character
    - Extended wild-card characters
        - ? - zero or one occurrence of previous character
        - one or more occurrences of previous character
        - {n} - n occurrences of previous character
        - {,n} - max n occurrences of previous character
        - {m,} - min m occurrences of previous character
        - {m,n} - min m and max n occurrences of previous character
        - () - grouping (chars)
        - (|) - find one of the group of characters

grep

- Regex commands
    - grep - GNU Regular Expression Parser - Basic wild-card
    - egrep - Extended Grep - Basic + Extended wild-card
    - fgrep - Fixed Grep - No wild-card
- Command syntax
    - grep "pattern" filepath
    - grep [options] "pattern" filepath
        - -c : count number of occurrences
        - -v : invert the find output
        - -i : case insensitive search
        - -w : search whole words only
        - -R : search recursively in a directory
        - -n : show line number.

Classification of OS

- OS can be categorized based on the target system (computers).
    - Desktop systems
    - Server systems
    - Embedded systems
    - Distributed systems
    - Hand-held systems
    - Real-time systems

Desktop systems

- Personal computers -- desktop and laptops
- User convinence and Responsiveness
- Examples: Windows, Mac, Linux, few UNIX, ...

Handheld systems

- OS installed on handheld devices like mobiles, PDAs, iPODs, etc.

- Challenges:
  - Small screen size
  - Low end processors
  - Less RAM size
  - Battery powered
- Examples: Symbian, iOS, Linux, PalmOS, WindowsCE, etc.

Realtime systems

- The OS in which accuracy of results depends on accuracy of the computation as well as time duration in which results are produced, is called as "RTOS".
- If results are not produced within certain time (deadline), catastrophic effects may occur.
- These OS ensure that tasks will be completed in a definite time duration.
- Time from the arrival of interrupt till begin handling of the interrupt is called as "Interrupt Latency".
- RTOS have very small and fixed interrupt latencies.
- RTOS Examples: uC-OS, VxWorks, pSOS, RTLinux, FreeRTOS, etc.

Distributed systems

- Multiple computers connected together in a close network is called as "distributed system".
- Its advantages are high availability (24x7), high scalability (many clients, huge data), fault tolerance (any computer may fail).
- The requests are redirected to the computer having less load using "load balancing" techniques.
- The set of computers connected together for a certain task is called as "cluster". Examples: Linux.