1. List the contents of a directory and their attributes - ( ls )
2. Long list the content with file, directory ownership, permissions,sizes, etc… - ( ls –al ) ( ls –l )
3. Display the size of the file in human readable format – ( ls –l –h )
4. Show all files and folders including hidden one – ( ls –al )
5. list directories recursively – ( ls –R )
6. Sort the files by size with largest at the top – ( ls –S )
7. Sort the files by last time modified displaying the newest first. – ( ls –sort=time )( ls –l –t )
8. Display the location of a program/command, where it is installed. ( echo $PATH )

/mnt/c/Program Files/WindowsApps/CanonicalGroupLimited.UbuntuonWindows\_2004.2021.222.0\_x64\_\_79rhkp1fndgsc

1. Which command is used to switch directory from one to another – ( cd )
2. List all the environment variables set for the current shell environment – ( printenv or env )
3. Did you notice something in the output of "env" command? – (both printenv and env gives same output )
4. which command is used to print the text or any variables value in the Console/Terminal? – ( echo )
5. print the value of the env variable "PATH" on the console – ( echo $PATH)
6. Is linux a case-sensitive operating system? – ( yes )
7. Is, "Ls" same as "ls"? – ( NO )
8. Display your currently logged in user – ( whoami )
9. how do you change the currrently logged in user to another user? – ( su - <USERNAME> )
10. Which command is used to leave a shell environment that you are currently logged in to? – ( exit )
11. How do you reboot the system? – ( sudo reboot / reboot )
12. How do you shutdown the system? – ( shutdown / poweroff )
13. Display all the major running processes in the system – ( ps -au )

USER PID %CPU %MEM VSZ RSS TTY STAT START TIME COMMAND

root 1 0.0 0.0 8892 292 ? Ssl 19:23 0:00 /init

root 7 0.0 0.0 8900 204 tty1 Ss 19:23 0:00 /init

vny\_ubu+ 8 0.0 0.0 18096 3568 tty1 S 19:23 0:00 -bash

vny\_ubu+ 118 5.0 0.0 18664 1892 tty1 R 19:38 0:00 ps aux

1. Understand the output of command used in above question of displaying processes, explain the meaning of each column and what data it displays?

USER: shows usernames

PID: Shows Process ID

%CPU: Shows %ge usage of CPU

%MEM: Shows %ge usage of memory

VSZ: Shows usage of virtual memory in bytes

RSS: Shows Resident set size for memory allocation in RAM

TTY: shows the terminal number

STAT: Semantic Variables

START: starting time of program

TIME: shows time needed to execute the command

COMMAND: shows the command

1. Display the name of the system kernel – ( uname –s ) - Linux
2. display the kernel release number – ( uname –r ) - 4.4.0-18362-Microsoft
3. display the machine type of the current kernel – ( uname -m ) - x86\_64
4. Display the name of the operating system that the kernel is running on – ( uname –o ) – GNU/Linux
5. Display all info that uname command can show – ( uname -a)

Linux DESKTOP-MFHQUNG 4.4.0-18362-Microsoft #476-Microsoft Fri Nov 01 16:53:00 PST 2019 x86\_64 x86\_64 x86\_64 GNU/Linux

1. Display the name of directory that you are currently pointing to – ( pwd ) - /home/vny\_ubuntu
2. change the current directory to another directory that you have in your system. – ( cd <dir\_path> )
3. Go up one directory – ( cd .. )
4. Return to last directory – ( cd - )
5. change the current directory to home(logged in user's) directory – ( cd ~ / cd )
6. how to check all the command used from the prompt (Command History) – ( history )
7. In which file the history of commands are stored in? – ( .bash\_history )
8. How many lines of history does the system keep and from where you can change it? – ( echo $HISTSIZE )
9. How can you modify bash's history behavior ?
10. Display all the commands entered so far, now, try to run a particular command from the history list without typing that command. – ( history -> up/down arrows )
11. What are the different types of shell and where are they used and how do we use them?

C shell ( csh ) – has built-in arthematic and c-like expression syntaxs

Bourne Shell ( sh ) – original UNIX shell and fast but lacks arthematic and logical expression handling and not have interactive use.

Korn shell ( ksh ) – supports like everything in sh and incorpates the arthematic , functions and string manipulation facilities

Bourne-Again Shell ( bash ) – combination of ksh and sh

Z-shell ( zsh ) – sh with improvements, including some features of [Bash](https://en.wikipedia.org/wiki/Bash_(Unix_shell)), [ksh](https://en.wikipedia.org/wiki/KornShell" \o "KornShell).

1. What is the difference between login shell and non-login shell?

Login shell : Shell which is launched normally and can be identified by the output “-bash” from the cmd “echo $0”

Non-Login Shell: Shell which opened when we run a program and can be identified by the output “bash” from the cmd “echo $0”

1. How do we start login shell and non-login shell?
2. What happens when you start a login shell (which files are read and used and Why)?

When [Bash](https://gnu.org/software/bash/) is invoked as a Login shell:

* Login process calls **/etc/profile**
* /etc/profile calls the scripts in **/etc/profile.d/**
* Login process calls **~/.bash\_profile**
* ~/.bash\_profile calls **~/.bashrc**
* ~/.bashrc calls **/etc/bashrc**

1. What happens when you start a non-login shell (Which files are read and used and Why)?

When bash is invoked as a Non login shell:

* Non-login process(shell)calls **~/.bashrc**
* ~/.bashrc calls **/etc/bashrc**
* /etc/bashrc calls the scripts in **/etc/profile.d/**

1. What are Shell Configuration Files, why do we need it?

To configure Bash to use our custom settings we need to configure various configuration files . Some of configuration files are .profile, .bash\_history, .bashrc and etc

1. Explain the Order of file usage from the system/user's home directory when user logs in to the System.
   * /etc/profile
   * /etc/profile.d/
   * /~/bash\_profile
   * ~/.bashrc
   * etc/bashrc
2. What are Shell Variables, list major shell variables and what do they represent?

Variable which are created and used by the shell in order to function correctly. These varibles can be used within the created shell only

1. How we see all our env variables? – ( env / printenv )
2. How we see all env variables in alphabical order? ( printenv | sort)
3. What Format does the env var and its values are stored? ( Key:Value )
4. How do you create your own varible?

<variable\_name>=<variable\_value>

1. How do you start a new bash shell?
2. Difference between Local/Shell variables to Global Variable

Local varibles will have the same value in mentioned function only but global varibles will be same from starting to ending of the program

1. Making a variable accessible from other the shell in the system.

By adding the variable to /etc/environment

1. Show the real life use case of exporting a variable
2. Convert the above script file into a command, The file should run with "myappp" instead of "myapp.sh"
3. What is Globbing? Explain in depth with examples?

 Globbing is mainly **used to match filenames or searching for content in a file**

1. List all entries with extension ".sh" – ( ls \*.sh )
2. List all entries with numbers in it. – ( ls -a | grep .[0-9] )
3. List all entries that starts with a character and ends with a number – ( ls -a |grep ^[a-zA-Z].\*[0-9]$ )
4. List all entries that name length more than 5 characters
5. What is Quoting? and Why do we need it?
6. Write few(minimum 3) unique examples that shows, how a particular problem is solved using Quoting.
7. How do you find a particular files/directories based on a particular search criteria?

HINT:- look for commands -> locate, find and whereis

1. Write major difference between locate, find and whereis?

Find – it searches the files recursively from the given directory

Locate – it wont need any directory, it searches the whole database

Whereis – it gives the installed application paths

1. How Globbing is different from locate, find and whereis?

Globbing searches for the file/directory in the given directory only but the find will search recursively in specified location. Locate searches throughout the database and gives the location. Whereis gives the installed application paths

1. Explain the Linux File System.
2. Explain absolute and Relative Paths

Absolute Path: Path for a particular location from the root directory.

Relative Path: Path for a particular location from the pwd

1. What are the different ways of creating a File in linux System? Write an example of each and the difference between them.

touch filename

* + cat > file\_name
  + echo "This is the text" > file\_name
  + grep This file.txt > new\_file1.txt
  + using a text editor like vi,nano,vm - vi file\_name

nano file\_name

1. In how many ways we can delete the files from linux system? write an example of each and the difference between them.

Unlink file\_name

rm file\_name

wipe file\_name

1. Archiving files using linux command, write a command to archive set of files from linux commands.

tar cf <file\_name.tar> <file\_name>

1. Extract the archived files from the above step.

tar xf <file\_name.tar>