

# Questions for notes 3

---

- **What is a graphical user interface (GUI)?**
  - It is a set of programs that allow users to interact with the system via icons, windows, and various other visual elements.
- **What is a desktop environment?**
  - It is an implementation of the desktop metaphor made of a bundle of programs that run on top of the computer system that share a GUI also known as a graphical shell.
- **What is the command line interface (CLI)?**
  - The command line interface is where one can input text in a shell so that the computer can read it and execute whatever commands are being input into it. It can be considered a programming language.
- **How do I access the command line interface (CLI)?**
  - Ways to access it are to click on the bottom left button on your desktop on your taskbar (In my set up) to open the apps menu and then you can either search for it and click on it or you can type in "Terminal" or other emulators such as "tilix" and click on them to open it. You can also take the system out of graphical desktop mode and put it into text mode.
- **What is a virtual console?**
  - It is a terminal session that runs in Linux system memory.
- **What is a terminal emulator?**
  - It is a program that lets you access the Linux CLI when using the GUI. Tilix is a good example of this.
- **What is bash?**
  - It is a program that provides interactive access to the Linux system. It runs as a normal program whenever the user logs in and is most often used as the default shell though others exist. Terminal emulators use this shell.
- **What is the shell prompt?**
  - The shell prompt is what comes up when you first open the shell and is ready to accept input. It usually includes your username@machinename and then the currently working directory followed by a dollar sign. If the dollar sign is a # (pound sign), you are logged into the shell with superadmin privileges.

Definition, usage, and examples of the following commands.

---

## clear

"clear" will reset or clear the interface.

### Usage

`clear`

### Example:

- I have too many on screen commands that I previously input that I do not need any more past this point. I need to clear it out so I can have an easier time seeing the new commands and information I'm inputting. `clear`
- 

## echo

"echo" will display a line of text.

### Usage

`echo + option + string`

### Example

- I wish to display a line of text because why not.
    - `echo "Hello world!"`
  - I want to display a line of text without a new line being made.
    - `echo -n "Hello world!"`
  - I want to display a line of text that include a horizontal tab.
    - `echo -e "\tHello World!"`
  - I want to display 2 lines of text in a single echo command.
    - `echo -e "Line 1\nLine2"`
  - I want to display 2 lines of text in a single echo command, with the second line having a tab.
    - `echo -e "Line1\n\tLine2"`
  - I want to display 2 lines of text in a single echo command that starts with a tab.
    - `echo -e "\tLine1\tLine2"`
- 

## date

"date" will present or set the system date and time.

### Usage

`data + option`

### Example

- I want to display the current date.
  - `date`

- I want to display the current date in rfc 5322 format.

- `date -R`

---

## free

"free" will display the free and used memory in the system

### Usage

`free + option`

### Examples

- I want to display the memory being utilized.
    - `free`
  - I want to be able to read how the memory is being used in a simpler way.
    - `free -h`
- 

## uname

"uname" will display the system information.

### Usage

`uname + option`

### Examples

- I want to print all information.
    - `uname -a`
  - I want to print the kernel's information.
    - `uname -s`
  - I want to print the node name.
    - `uname -n`
- 

## history

"history" will show all commands in the line history.

### Usage

`history + option`

### Examples

- I want to display the session's history.
  - `history`
- I want to clear the session's history.

- `history -c`
- 

## man

"man" will display an interface to the system reference manuals.

### Usage

`man + option + command`

### Examples

- I want to open the man page for history.
    - `man history`
  - I want to open a specific man page.
    - `man 5 passwd`
  - Show all available man pages
    - `man -f passwd`
- 

## apt

"apt" is a set of tools for managing debian/ubuntu packages.

### Usage

`sudo (if required) + apt + apt action + package name`

### Examples

- I want to update and upgrade.
    - `sudo apt update && sudo apt upgrade -y`
  - I want to install a program.
    - `sudo apt install firefox`
  - I want to remove a program.
    - `sudo apt remove firefox`
  - I want to search for a program.
    - `apt search "Web Browser"`
  - I want a list of all my installed programs.
    - `apt list --installed`
- 

## df

"df" will display the amount of disk space available/used on entire file systems/partitions.

### Usage

`df + option + command`

## Examples

- I want to include psuedo, duplicate, or inaccessible file systems.
    - `df -a --all`
  - I want to be able to read it in a human format.
    - `df -h`
- 

## du

"du" will display the amount of space used by specific files or directories.

### Usage

`du + option + file`

## Examples

- I want to display the write count for all files.
    - `du -a`
- 

## hostname

"hostname" shows the system's hostname which is the computer name.

### Usage

`hostname`

## Examples

- I want to see my computer's name.
    - `hostname`
- 

## figlet

"figlet" displays a name in large ASCII letters.

### Usage

`figlet + string`

## Examples

`figlet "John Doe"`

---

## cheat

"cheat" will display cheat sheets either made by you or community made.

## Usage

`cheat + option + cheatsheets`

## Examples

- I want to see all cheat sheet directories.
  - `cheat -d`
- I want to see the cheat sheet for wget.
  - `cheat wget`