***Task 1: -*** *Dual boot or install a VM with Linux OS. Refer to a YouTube video for the same.*

* Installed **Oracle VM VirtualBox**: Oracle VM VirtualBox is cross-platform virtualization software. It allows users to extend their existing computer to run multiple operating systems including Microsoft Windows, Mac OS X, Linux, and Oracle Solaris, at the same time.
* Creating a Virtual Machine using Virtual Box: -
* Click Machine → New → Enter Name (of machine) → Select Machine Folder → Type as Linux → Version (of our choice say “Ubuntu 64bit”) → Next
* Selected Hardware: Base Memory 8192MB(8GB) and Processors 16 CPUs
* Hard Disk options: Created a Virtual Hard disk of 100GB
* Click Finish and machine is created
* Click Start and the machine boots
* Installing Ubuntu: -
* Select startup disk: Selected the .iso file of Ubuntu predownloaded from official website.
* Click Install Ubuntu → move forward with default options.
* Virtual Machine created with Linux OS(Ubuntu).

***Task 2: -*** *Create an account on GitHub. Familiarize with Git and GitHub commands using videos.  Make a repository for writeups. Make a detailed write-up for each task completed.*

* GIT Commands: -

1. Configuring name and email address:

* **git config –global user.name “…”**
* **git config –global user.email “…”**

1. For empty git repository: **gitint**
2. To take the file to staging area: **git add <file\_name.extension>**

* Add multiple files to staging area: **git add -A**

1. Status of the file created: **git status**
2. Create a new file: **touch <file\_name.extension>**
3. Commit files: **git commit -m “…. your msg ….”**
4. To match file with last commit: **git checkout <file\_name.extension>**

* Match multiple files with last commit: **git checkout -f**

1. Log/Record of all files committed: **git log**

* Filter o/p of git log: **git log -p -<no. of last commits we want to see>**

1. Compares working directory with staging area: **git diff**
2. Compares last commit with staging area: **git diff --staged**
3. Skip stagging area & directly commit: **git commit -a -m “…. your msg ….”**
4. List the files present: **ls**
5. Delete a file completely(from working tree): **git rm <file\_name.extension>**

* File only removed from staging area: **git rm --cached <file\_name.extension>**