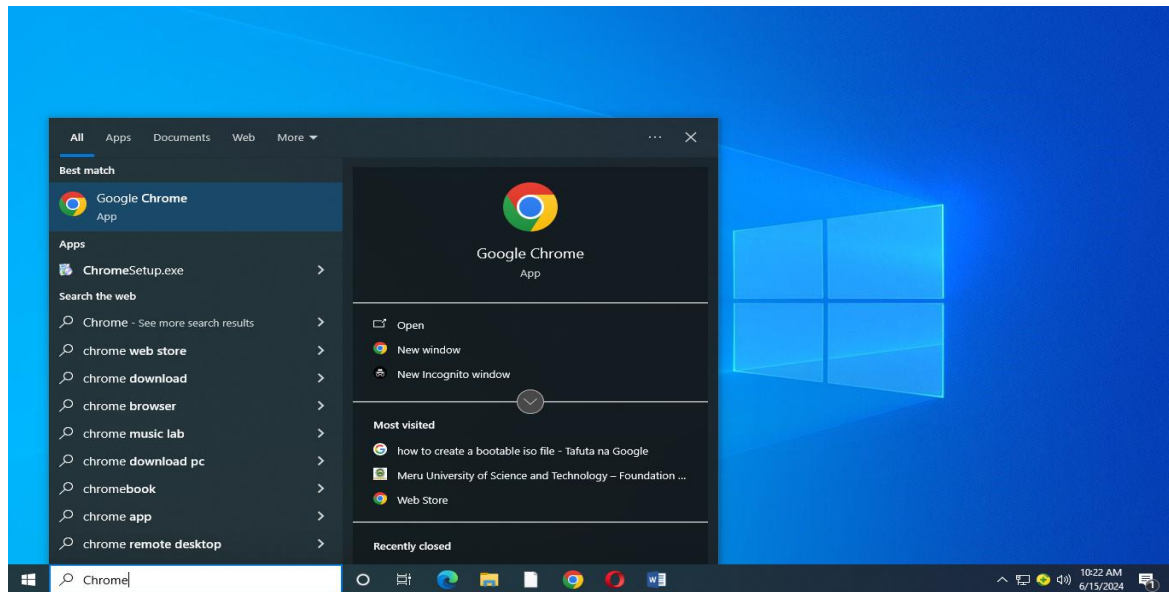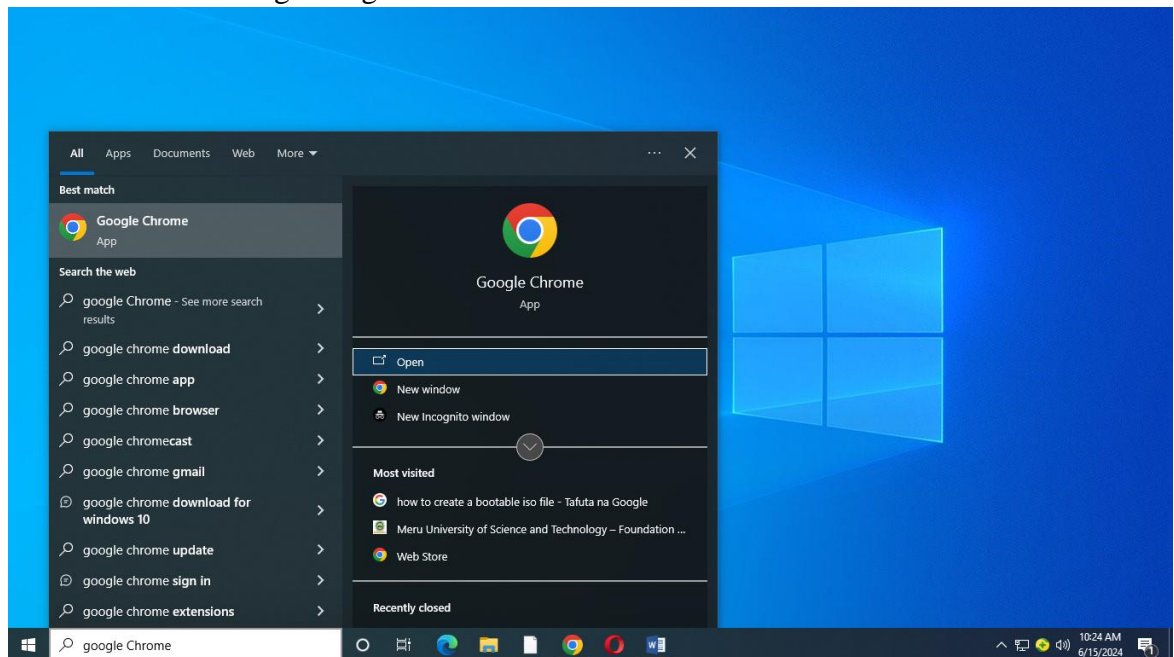# INSTALLATIONS

Installations Process for Windows 11, Visual Studio Code, Git, Python, MSQL and Creating GitHub Account, GitHub Repository commit AND Configurations of Database, virtual environment and extensions.
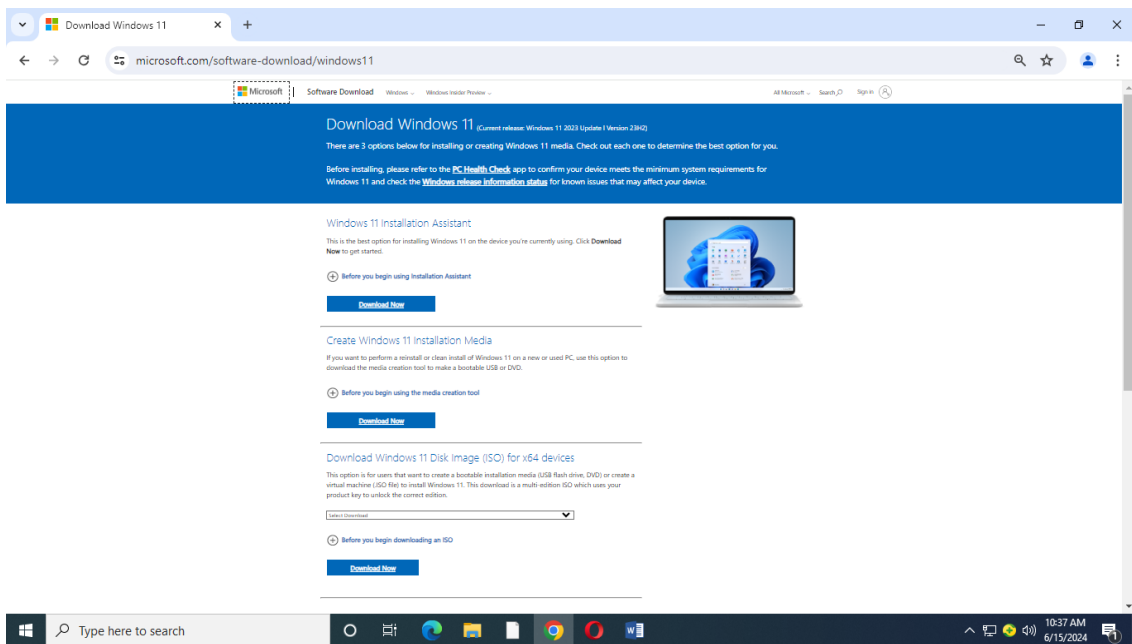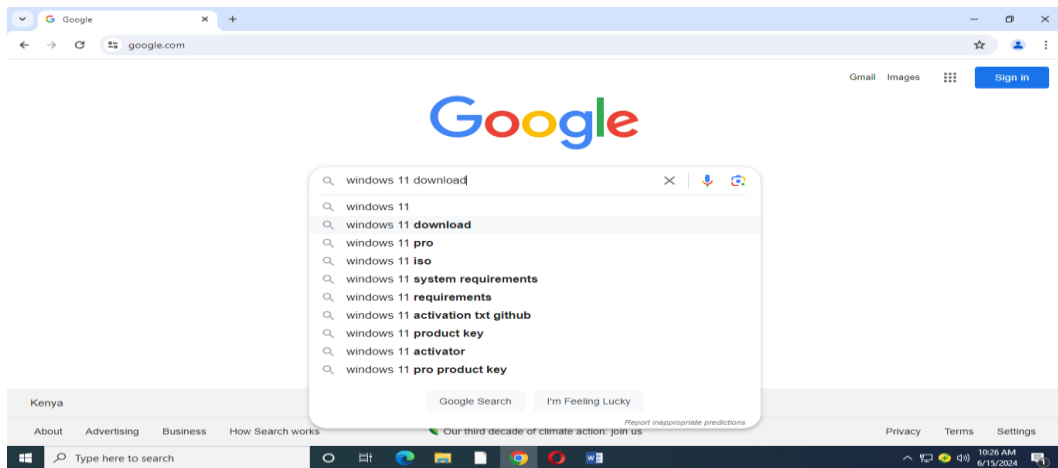
Windows 11 Download and Installation

1. From the Start Menu Search any internet browser of your choice to open.

2. Select the browser e.g. Google Chrome as shown below

3. From the search engine, search windows 11 download and load to open Microsoft website as shown https://www.microsoft.com/software-download/windows11.

5. If it's an upgrade from windows 10 to 11, Select Windows 11 Installation Assistant and download. If it's a new installation select download windows 11 disk image or create windows 11 disk media.
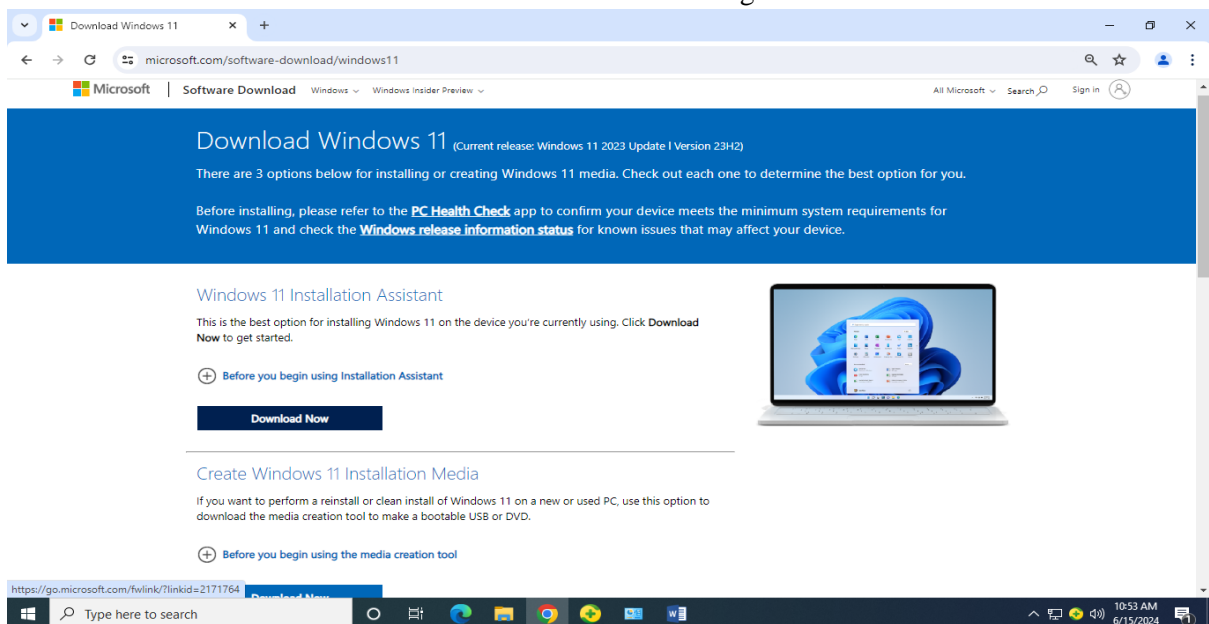
6. Open the setup and accept access request
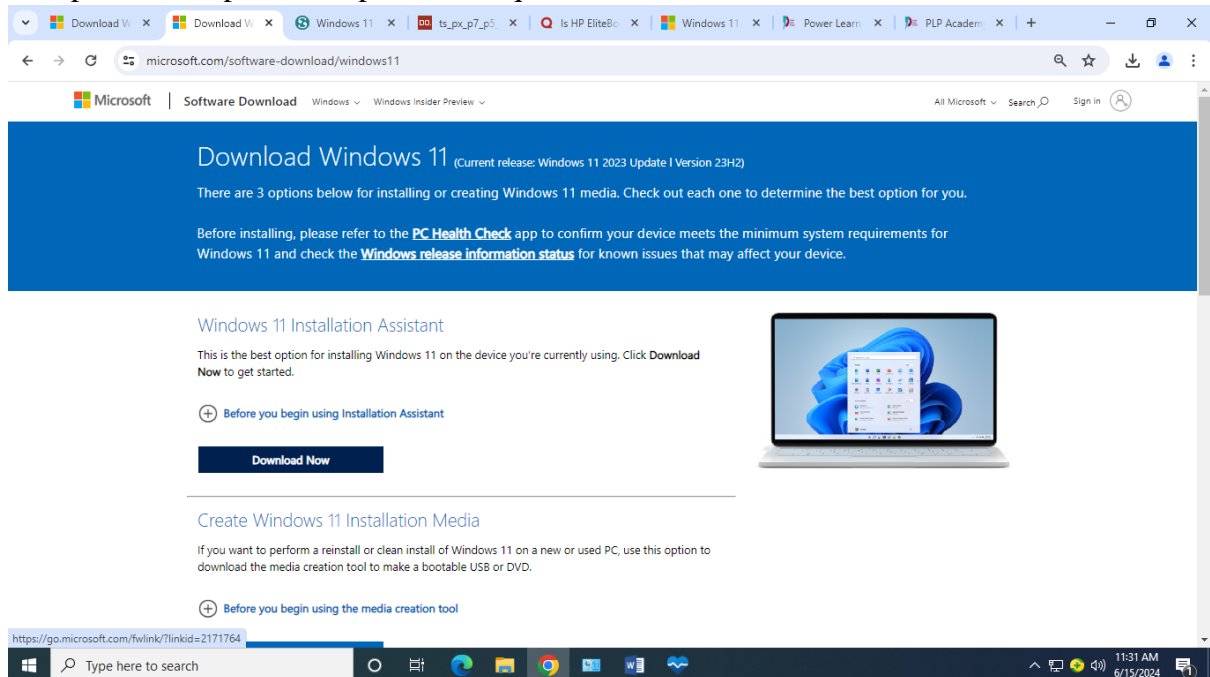
7. At the Windows 11 setup screen, select the Language, Time, and Keyboard Layout from the dropdown menus and select 'Next'.
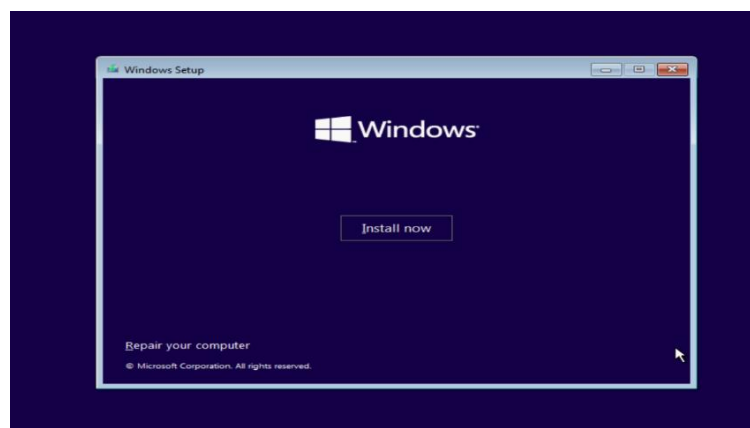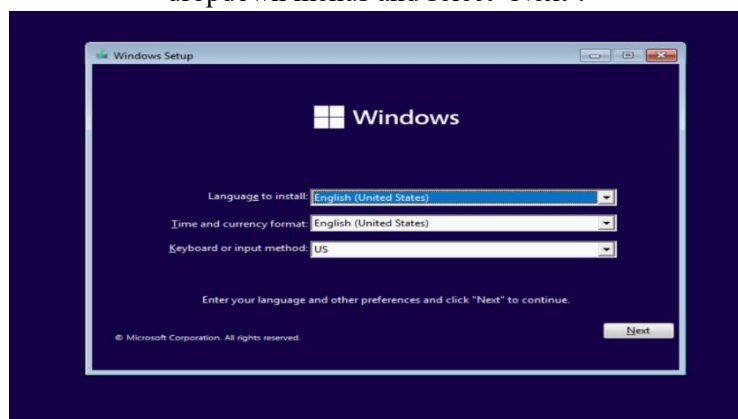
8. Select 'Install now'.

9. For a multiple Windows 11 version operating system media, select the type of Windows 11 operating system to install, then next and accept the license agreements.
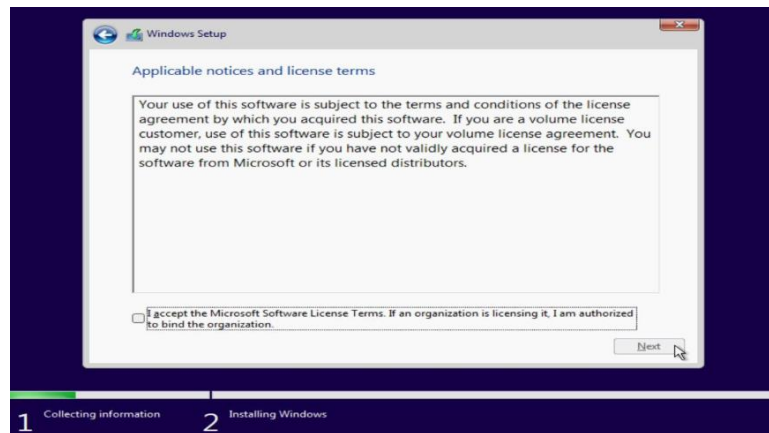
10. Choose the type of installation to perform. To upgrade to Windows 11 from an earlier version of Windows, select the "Upgrade" option below. To perform a clean OS installation, select "Custom: Install Windows only (advanced)" option below. (This option was selected below).
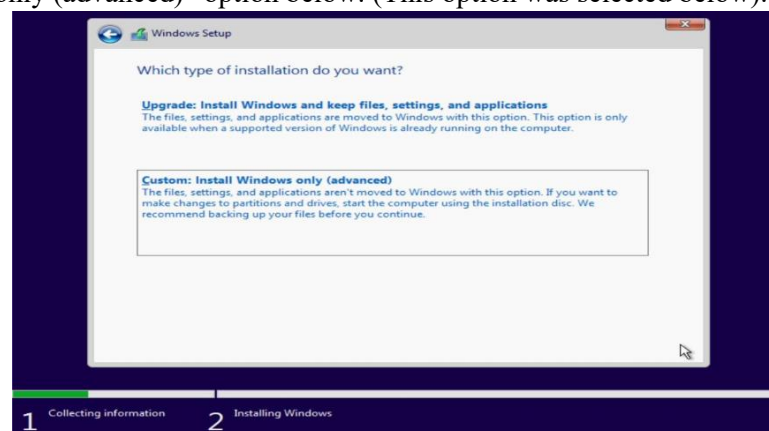
11. Select the drive to install the operating system on from the list and select 'Next'. The Windows 11 operating system will begin to install on the drive selected as shown below.
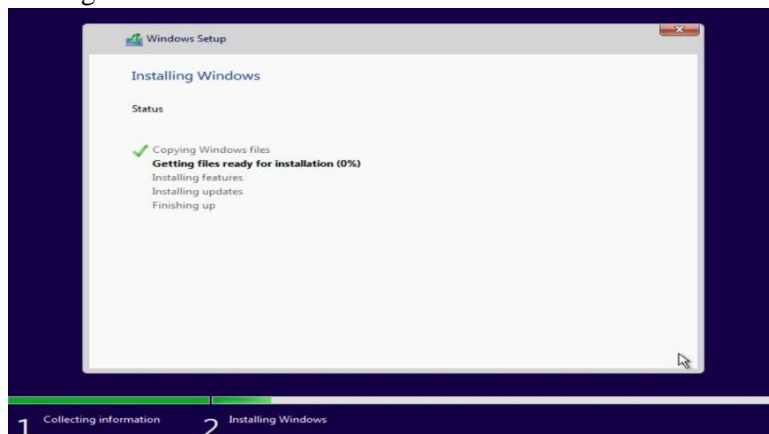
12. After the installation completes, the system should automatically reboot.
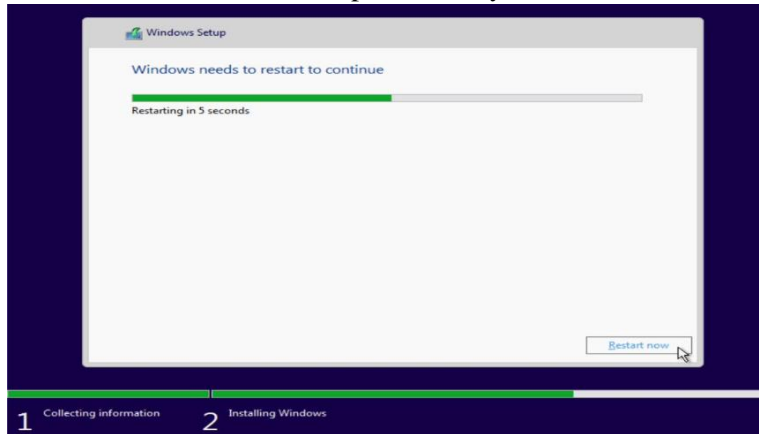
13. When done after restarting select the prompts for the appropriate location, language keyboard type. Ignore the Wi-Fi connectivity selection if not connected and Type your preferred name and password to secure the device.

14. Choose the appropriate privacy settings and select 'Next' to finish. Welcome to windows 11.

VISUAL STUDIO CODE INSTALLATION AND SETUP IN WINDOWS.

1. First, download the relevant installer from https://code.visualstudio.com .The page with figure below will open. Select on the stable version drop down arrow and select one as per your operating system version i.e. Windows, Mac Os or Ubuntu. NB: VS code requires x64 bit version.
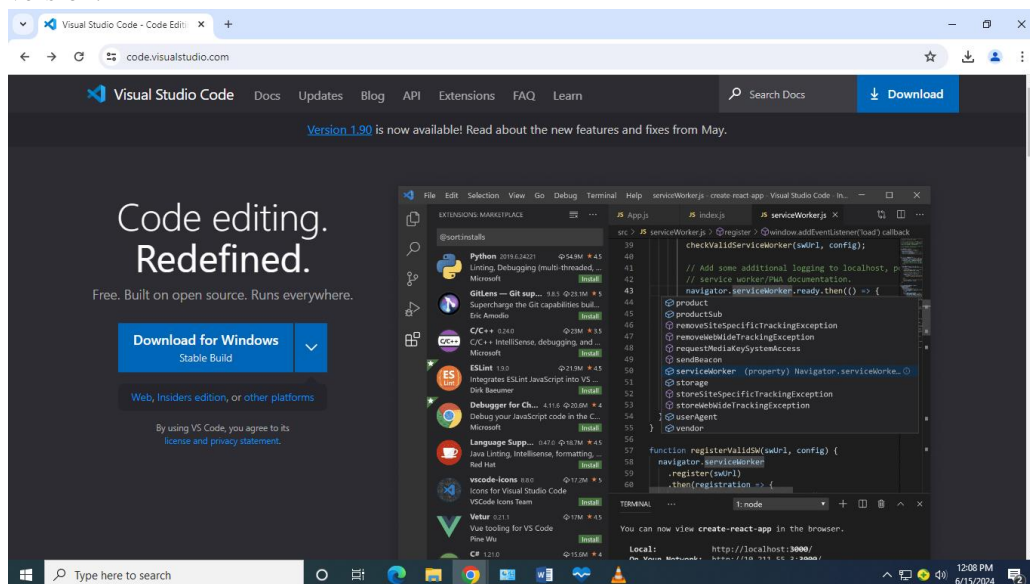
2. For the purpose of this installation I will use Windows.
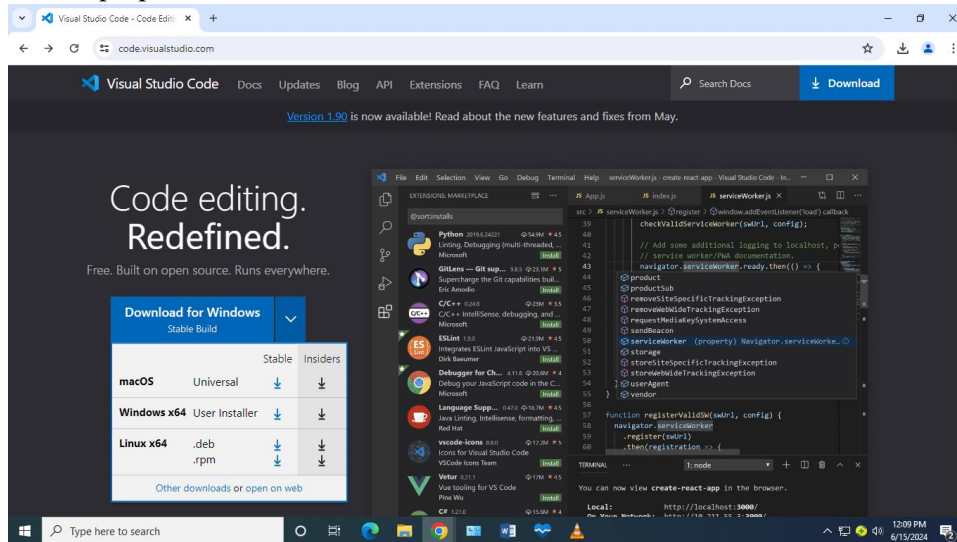
3. Click Download and open the downloads folder for installation. Run the Setup as shown below. Accept the license aggrement and click 'Next'.
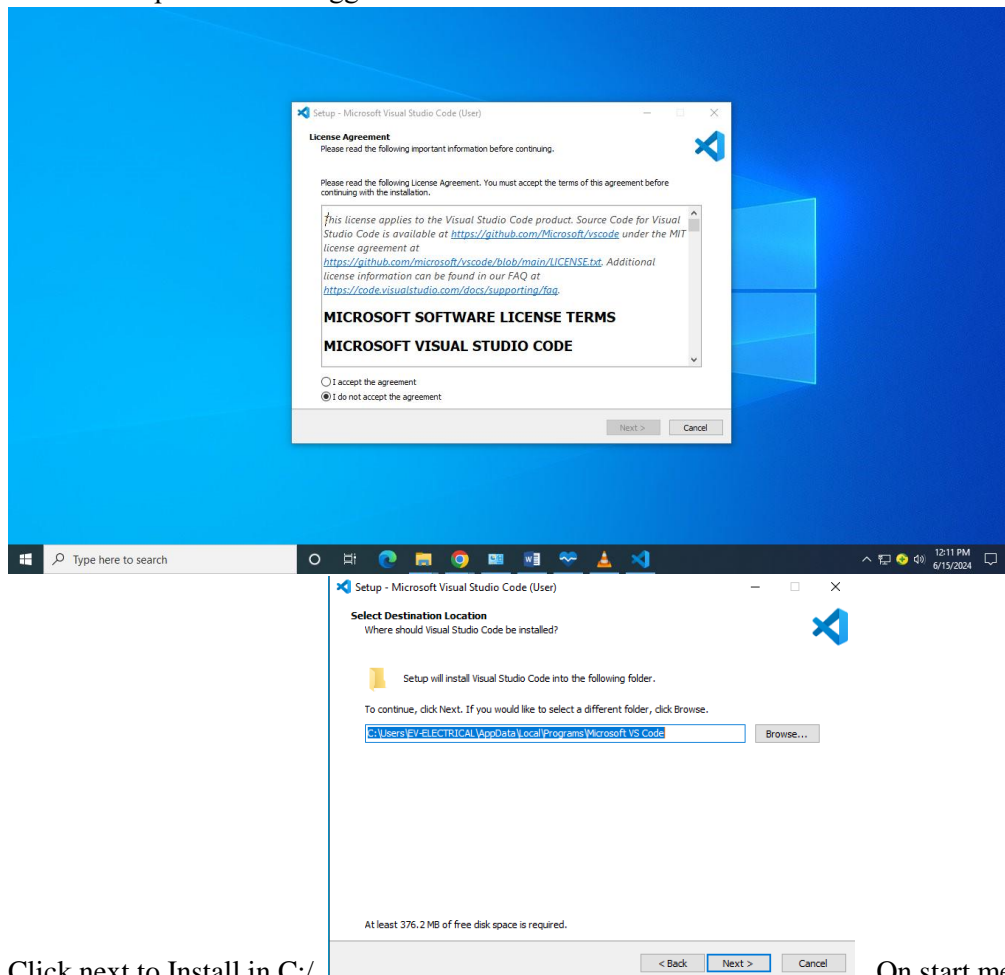
4. Click next to Install in C:/ [   ] . On start menu name leave it as Visual Studio Code and click next

5. On additional paths select Register code and Add to path.
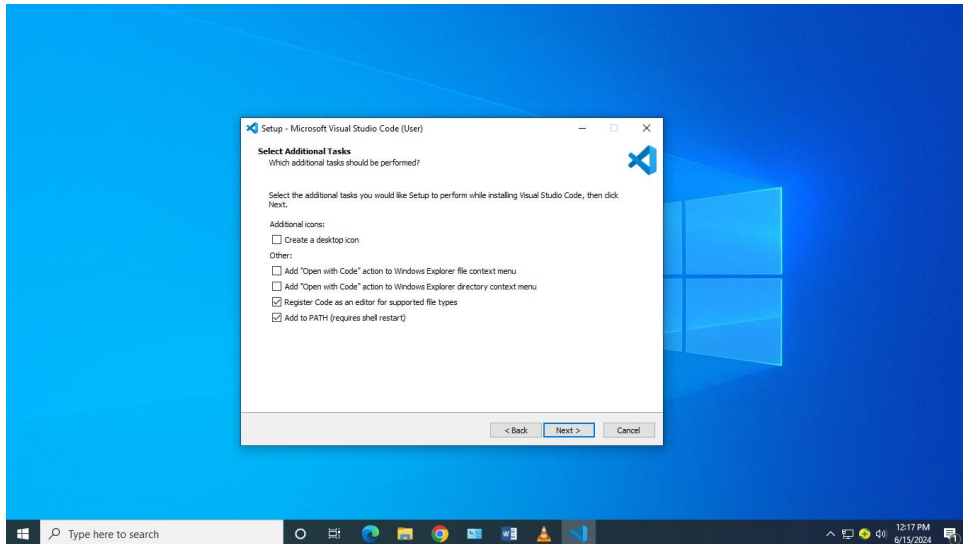
6. Click install to install VS code.
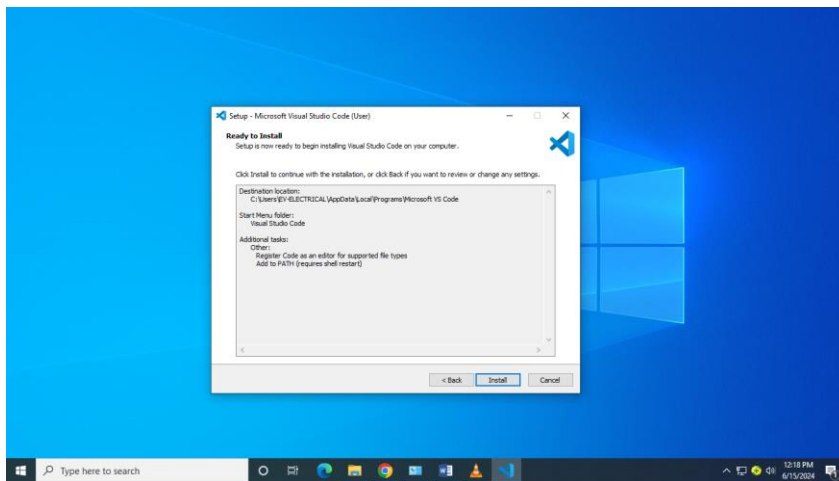
7. When done click finish as shown below and you are set.
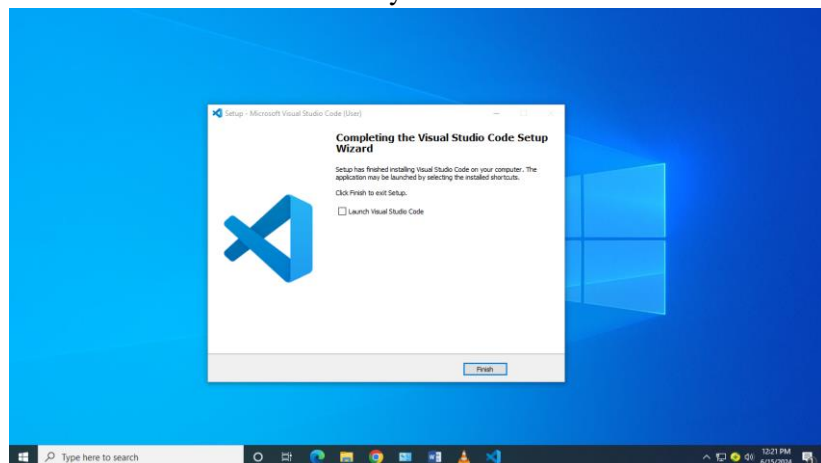
GIT INSTALLATION PROCESS

1. Open preferred search engine and search Git download as shown below. Select the link which directs to git homepage https://www.git-scm.com/downloads.
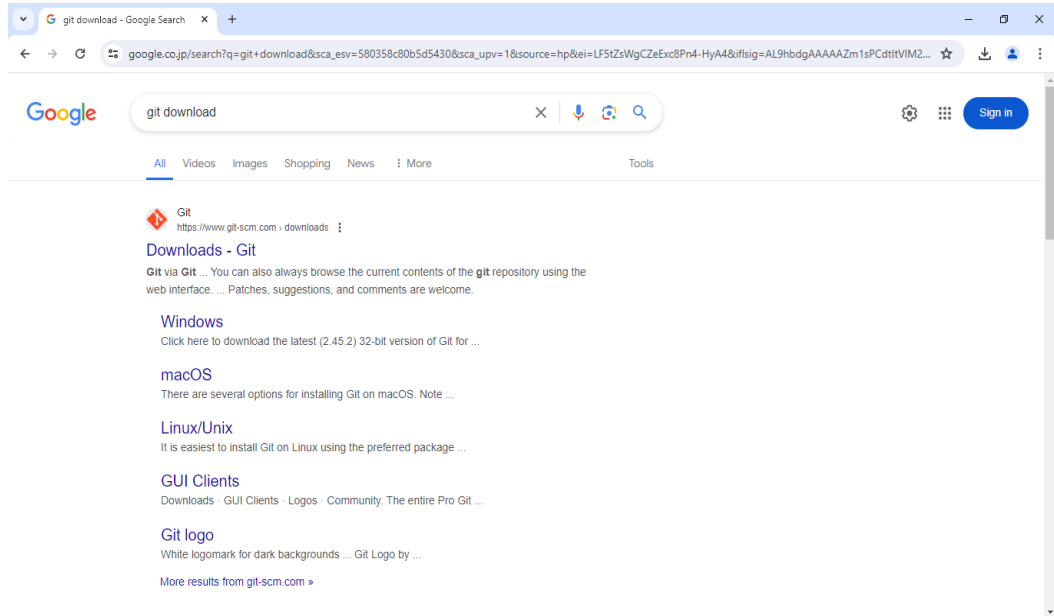
2. Under the downloads select the download as per your operating system. Fon now I will
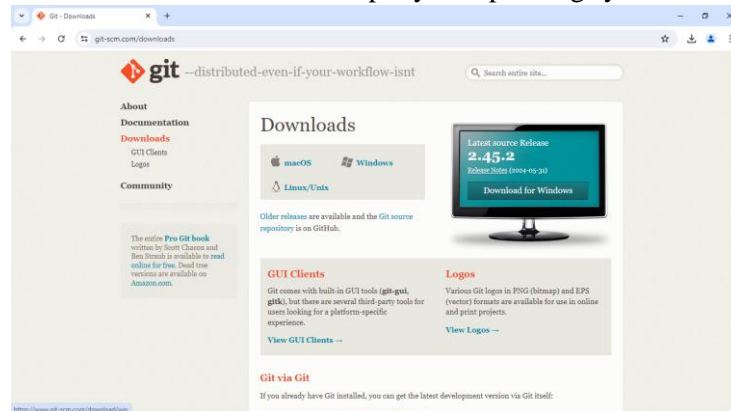
use Windows.

3. Select the version of your Operating System and download.
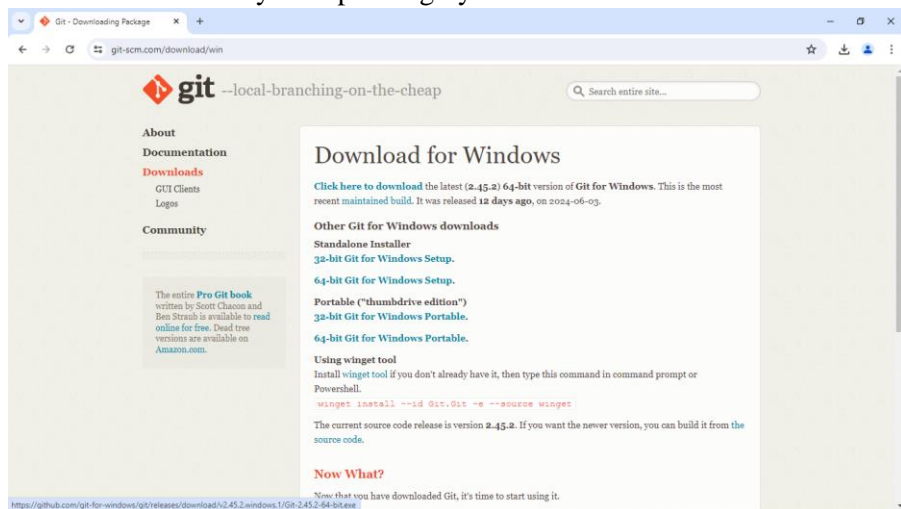
4. Go to to downloads and open the Git setup and open to install. Click next to proceed as shown below and next again to select System folder C:/
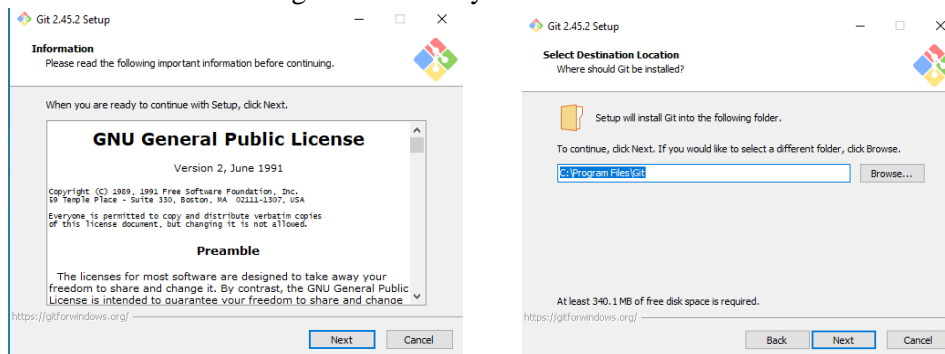
5. Under Select Components, leave the default as shown below. For menu folder name Git:

6. For default editor leave Vim as default. For initial new repository, set as let Git decide

7. For path environment select the recommended and click 'Next'. For SSH select bundled open SSH

Configuring the terminal emulator to use with Git Bash
Which terminal emulator do you want to use with your Git Bash?

◉ **Use MinTTY (the default terminal of MSYS2)**

Git Bash will use MinTTY as terminal emulator, which sports a resizable window, non-rectangular selections and a Unicode font. Windows console programs (such as interactive Python) must be launched via `winpty` to work in MinTTY.

○ **Use Windows' default console window**

Git will use the default console window of Windows ("cmd.exe"), which works well with Win32 console programs such as interactive Python or node.js, but has a very limited default scroll-back, needs to be configured to use a Unicode font in order to display non-ASCII characters correctly, and prior to Windows 10 its window was not freely resizable and it only allowed rectangular text selections.

Configuring the line ending conversions
How should Git treat line endings in text files?

◉ **Checkout Windows-style, commit Unix-style line endings**

Git will convert LF to CRLF when checking out text files. When committing text files, CRLF will be converted to LF. For cross-platform projects, this is the recommended setting on Windows ("core.autocrlf" is set to "true").

○ **Checkout as-is, commit Unix-style line endings**

Git will not perform any conversion when checking out text files. When committing text files, CRLF will be converted to LF. For cross-platform projects, this is the recommended setting on Unix ("core.autocrlf" is set to "input").
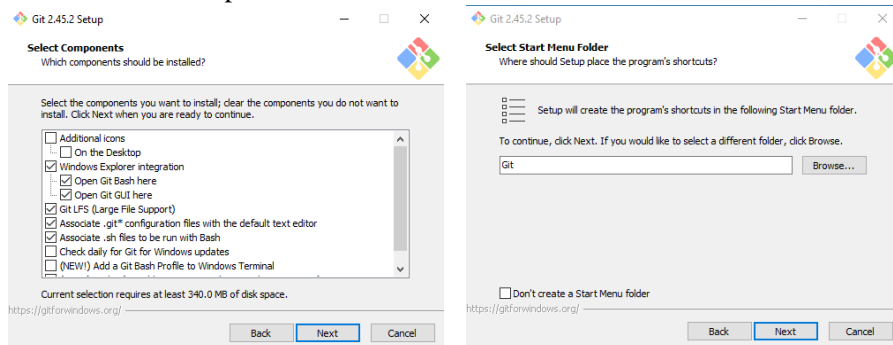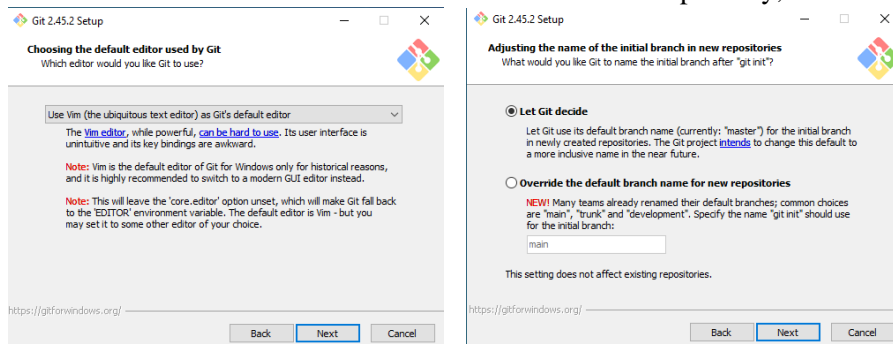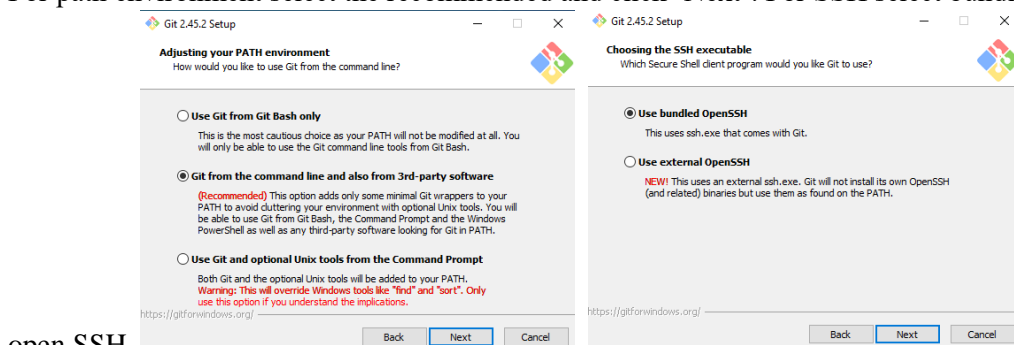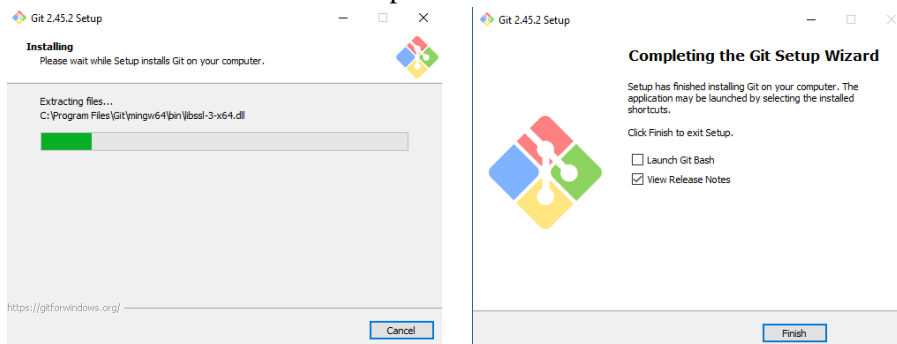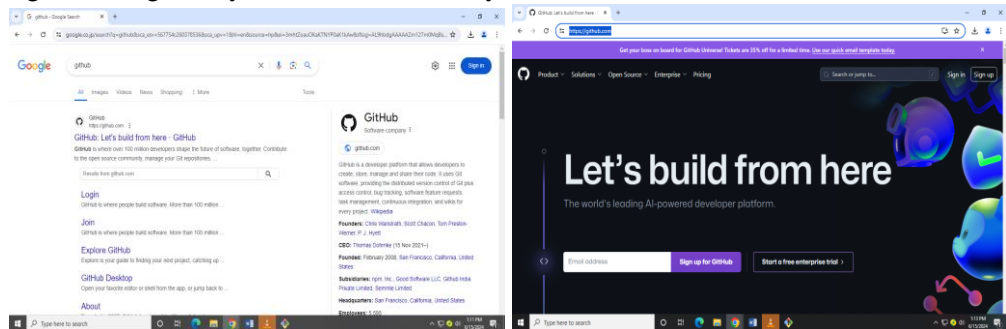
○ **Checkout as-is, commit as-is**

Git will not perform any conversions when checking out or committing text files. Choosing this option is not recommended for cross-platform projects ("core.autocrlf" is set to "false").

8. Follow prompts shown

Choose the default behavior of `git pull`
What should `git pull` do by default?

◉ **Fast-forward or merge**

Fast-forward the current branch to the fetched branch when possible, otherwise create a merge commit.

○ **Rebase**

Rebase the current branch onto the fetched branch. If there are no local commits to rebase, this is equivalent to a fast-forward.

○ **Only ever fast-forward**

Fast-forward to the fetched branch. Fail if that is not possible. This is the standard behavior of `git pull`.

Choose a credential helper
Which credential helper should be configured?

◉ **Git Credential Manager**

Use the cross-platform Git Credential Manager.
See more information about the future of Git Credential Manager here.

○ **None**

Do not use a credential helper.

Configuring extra options
Which features would you like to enable?

☑ **Enable file system caching**

File system data will be read in bulk and cached in memory for certain operations ("core.fscache" is set to "true"). This provides a significant performance boost.

☐ **Enable symbolic links**

Enable symbolic links (requires the SeCreateSymbolicLink permission). Please note that existing repositories are unaffected by this setting.

Configuring experimental options
These features are developed actively. Would you like to try them?

☐ **Enable experimental support for pseudo consoles.**

This allows running native console programs like Node or Python in a Git Bash window without using winpty, but is unfortunately not quite stable yet.

☐ **Enable experimental built-in file system monitor**

(NEW!) Automatically run a built-in file system watcher, to speed up common operations such as `git status`, `git add`, `git commit`, etc in worktrees containing many files.

9. Click Install and the installation process will continue as shown. When done click Finish.

Installing
Please wait while Setup installs Git on your computer.

Extracting files...
C:\Program Files\Git\mingw64\bin\libssl-3-x64.dll

Completing the Git Setup Wizard

Setup has finished installing Git on your computer. The application may be launched by selecting the installed shortcuts.

Click Finish to exit Setup.

☐ Launch Git Bash
☑ View Release Notes

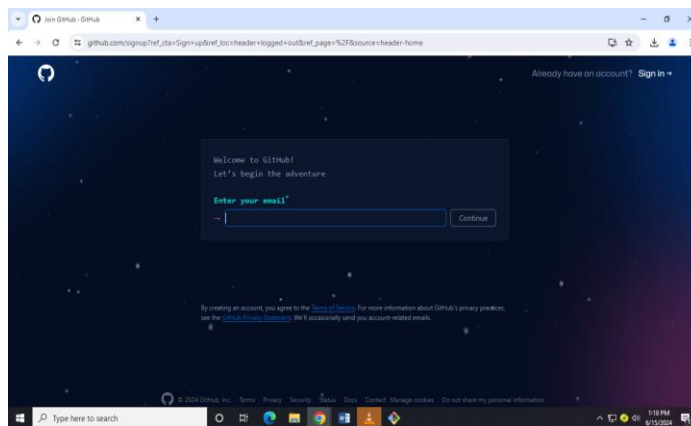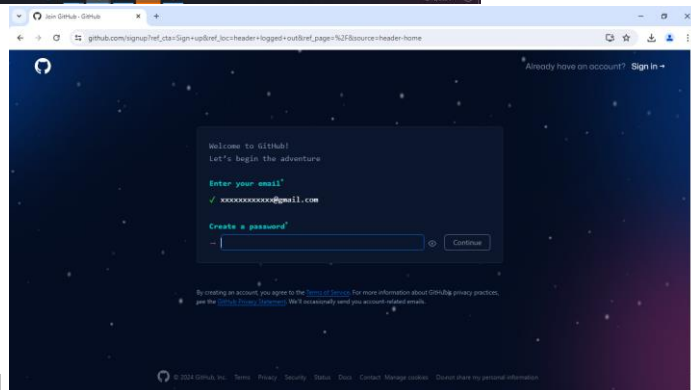**GITHUB ACCONT**

1. Open the search engine of your choice and search github as shown below. Hover to github website and open https://github.com/ . Select Sign up to open a new account or Sign in to log in to your account if already created.
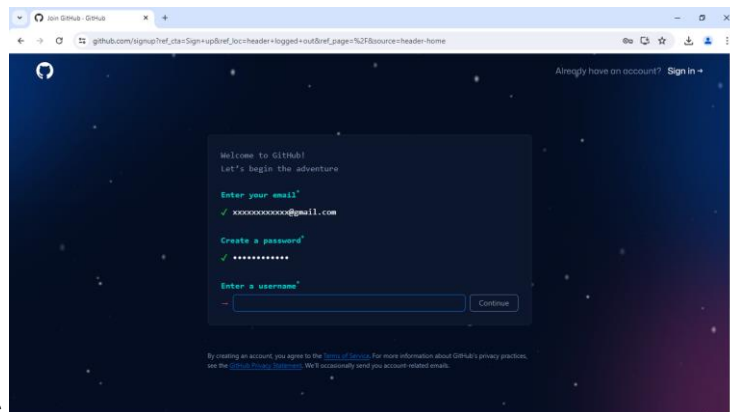

2. For new account enter your email address on the link provided below and press continue.
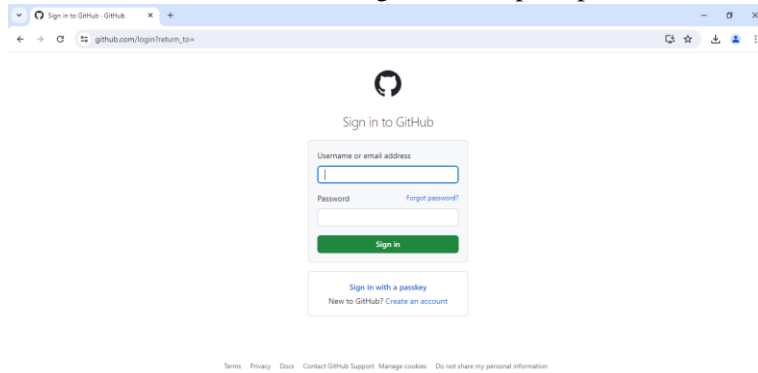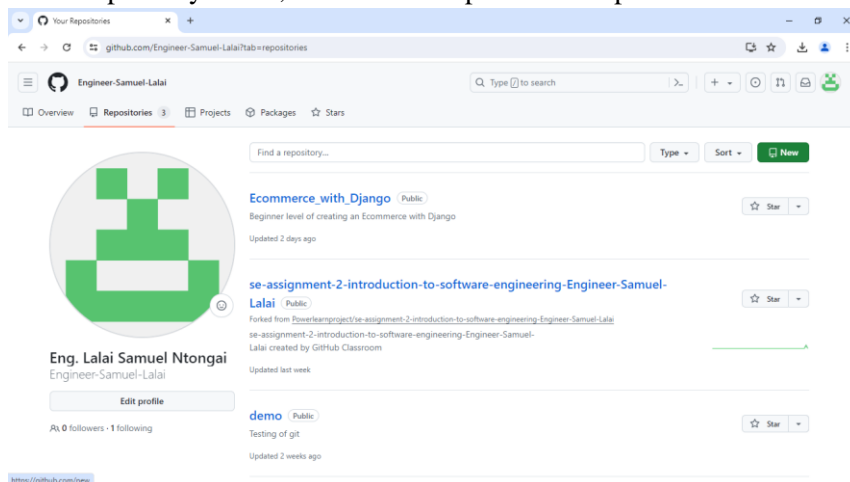

3.

4. Create a password

5. Enter User name

6. Pass the Robot test and verify account to proceed and finish. When done Click sign in to log in.

7. Enter Email and Password to sign in in the prompted window as shown below.

8. When done on the right corner click on the button and select your repository to create a new repository. Then, click New to open a new repo as shown below.


9. Under Repository name, write your preferred name for your repo e.g. FirstCommit, Add a description (Optional) to remind you in future of the content. Click **Create Repository** as shown below. Select Public (if to be viewed by everyone) or private (if for only you).
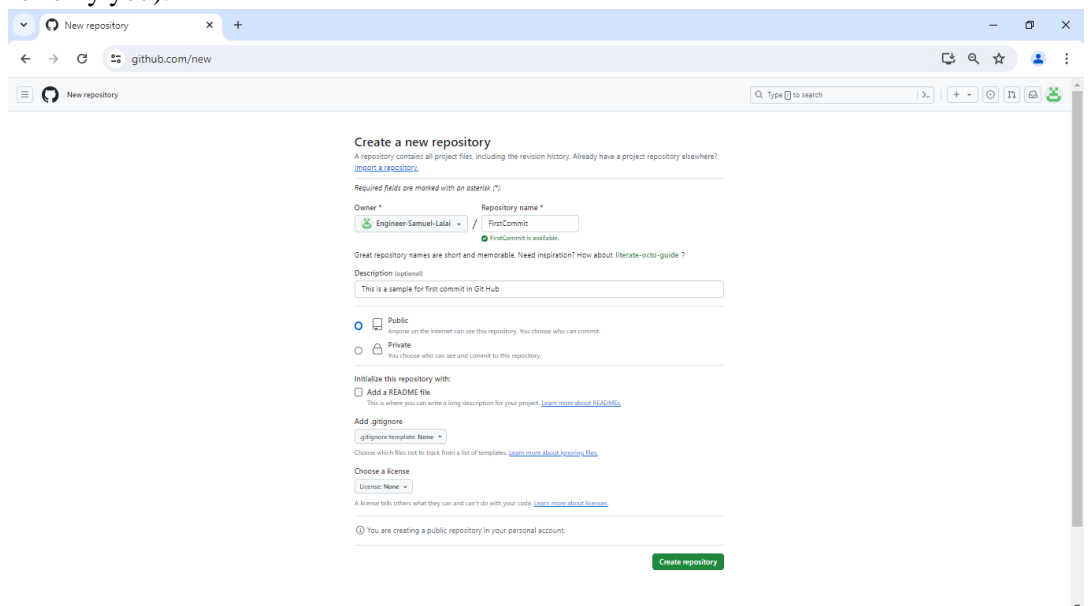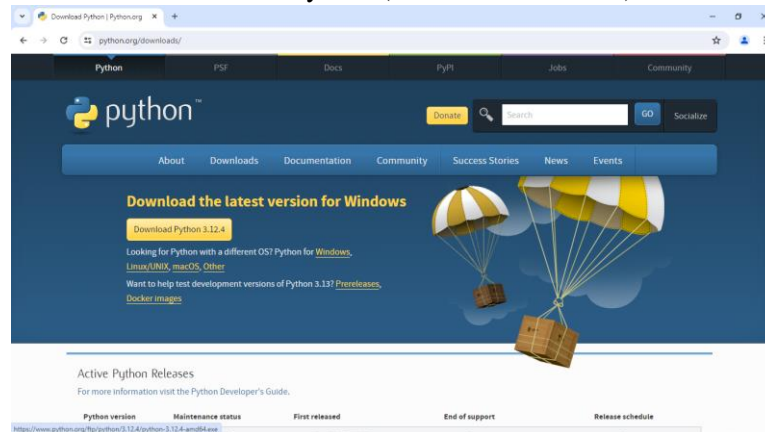

10. Congratulations for creating your first Commit

PYTHON INSTALLATION

1. Open the preferred search engine and type the url for python website http://www.python.org
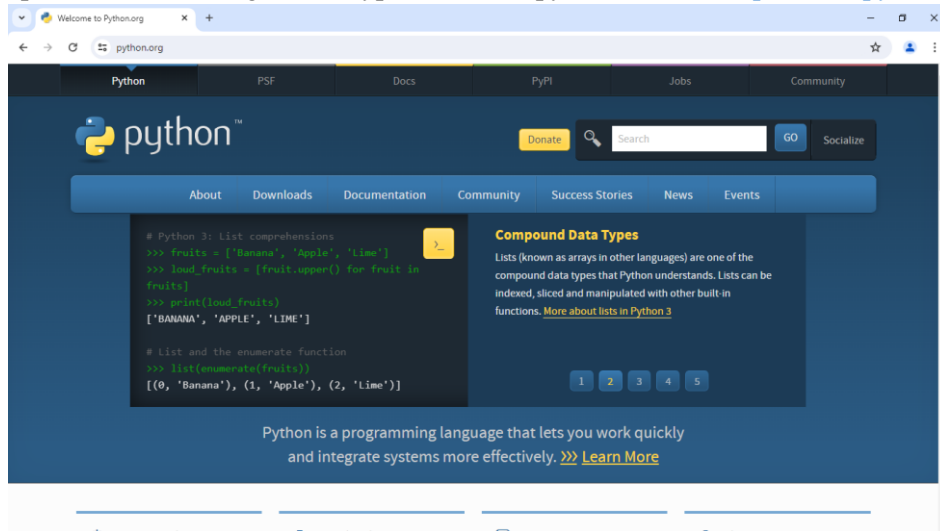
   to open.

2. Click on Dowloads. Select Download Python (Select latest version) as shown below.

3. Open the downloaded setup form the downloads folder and install. Select Use Admin priviledges and Add python to PATH.
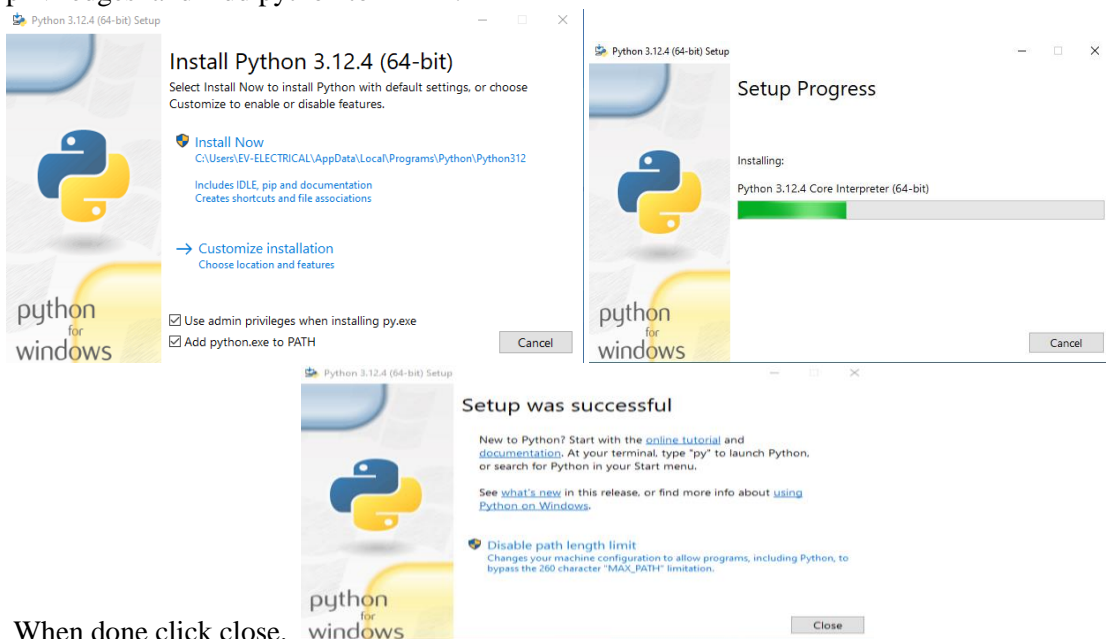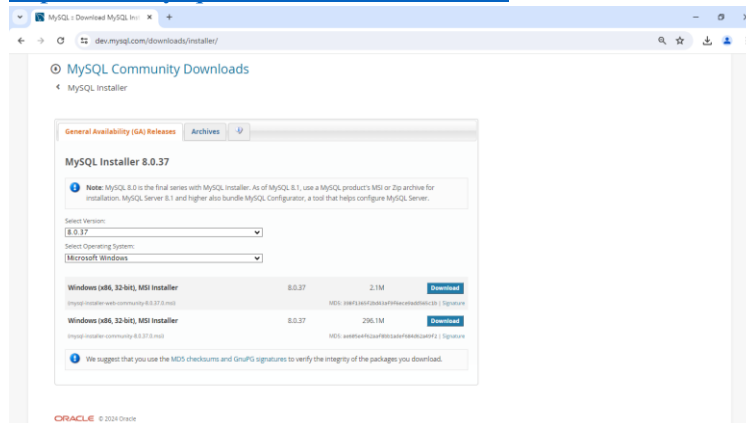

4. When done click close.

MySQL INSTALLATION FOR DATABASE

1. Open Preferred Search engine e.g. Google Chrome, Mozilla Firefox, Microsoft edge etc. and search MySQL download as shown below.
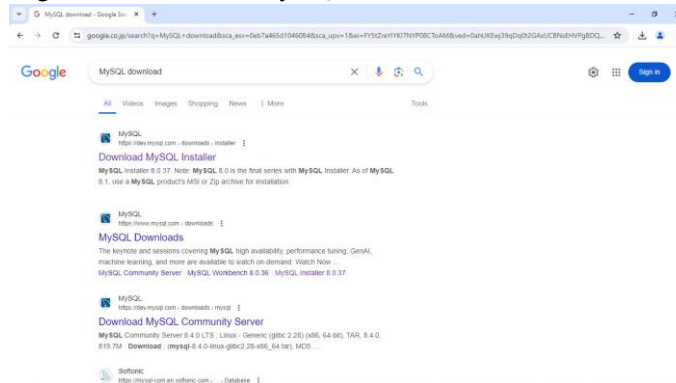
2. Select Download from the MySQL website to link https://dev.mysql.com/downloads/installer.


3. Select the latest version and operating system, in this case windows. Select the second option with more size to download. On the MySQL homepage, Click on the "No thanks, just start my download" link to proceed MySQL downloading.
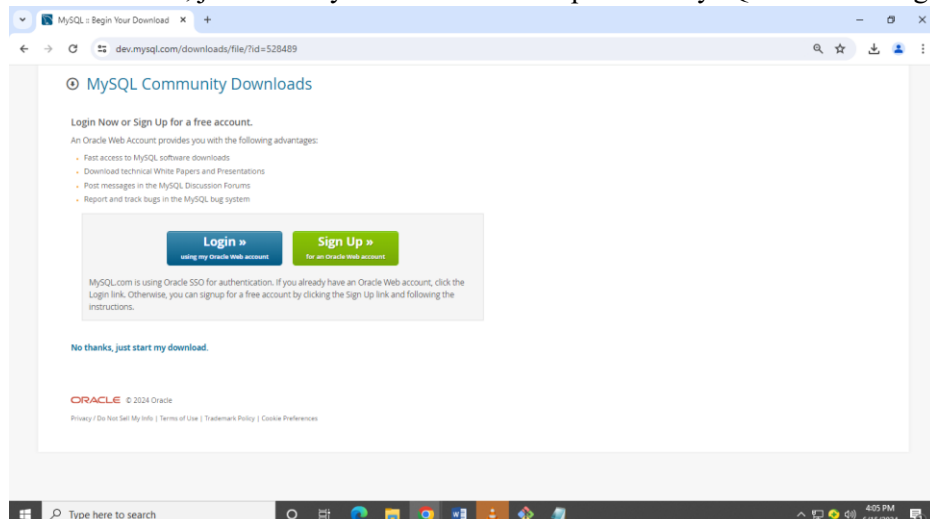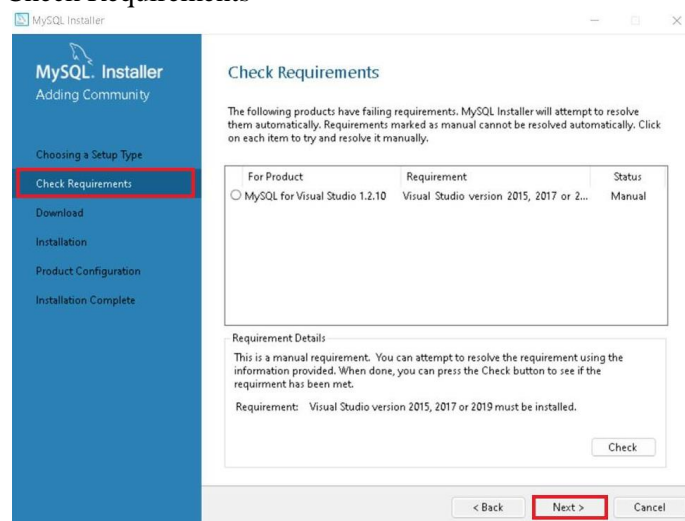

4. From downloads in your computer open the MySQL setup and install. The installer will instruct you to choose the setup type. For most users, the "Developer Default" is suitable. Click "Next" to proceed
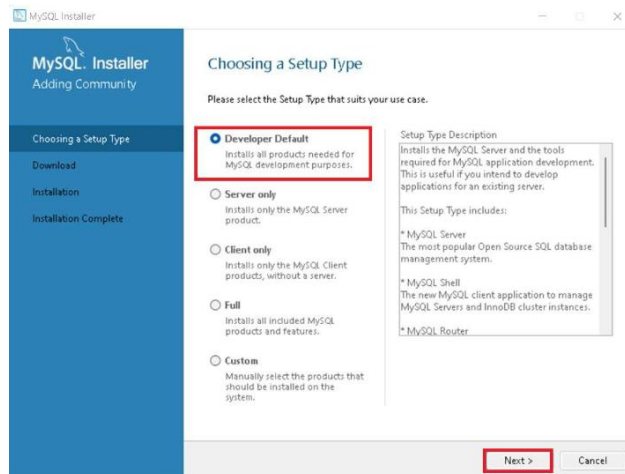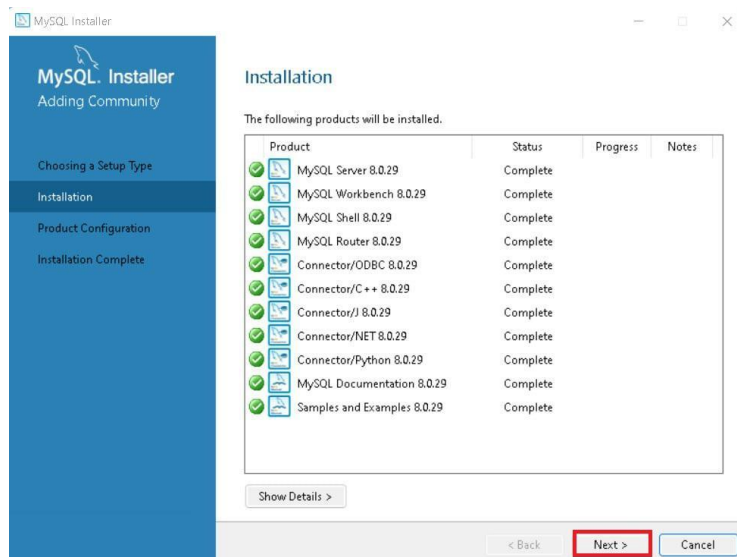
5. Check Requirements

6.
7. Click next to download the files and execute.

8. Proceed to "Product Configuration" > "Type and Networking" > "Authentication Method" Pages by clicking the "Next" button. Create a password for the MySQL root user. Ensure it's strong and memorable. Click "Next" to proceed.

9. Once done select the workbench and Shell to install and complete the installation.
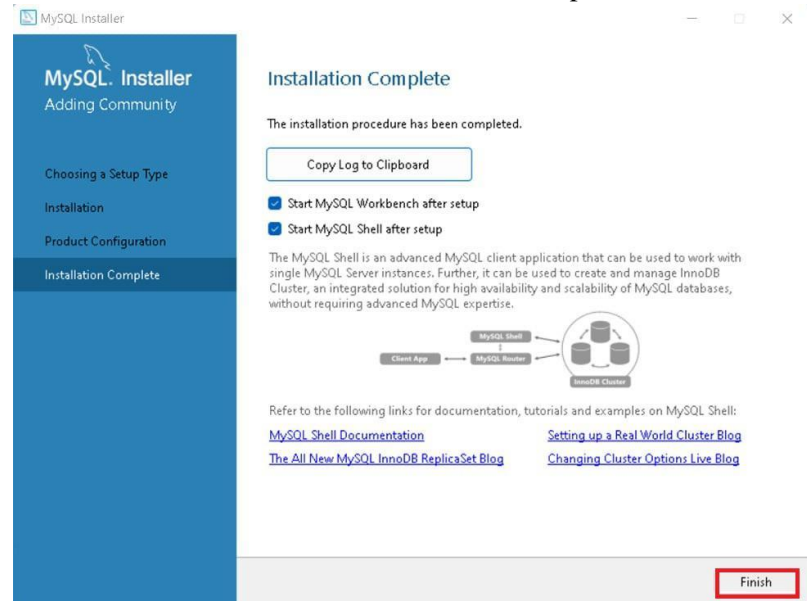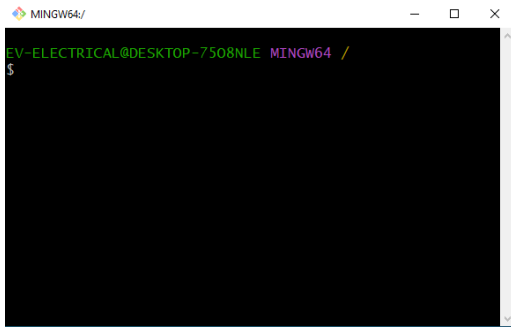

Click Finish.

10. Search MySQL Command Line on start menu to check if fully installed. Open and put the root password to open.

INITIALIZING FIRST COMMIT IN GITHUB REPOSITORY USING GIT

1. Under Start menu, Search for Git Bash and open as administrator.

2. Type cd 'directory id' to open the folder you want to open. E.g. cd h: /
3. Create a folder in the disk to save content by using mkdir 'folder name eg sample'. mkdir sample
4. Open the created folder as shown below by sample: cd sampleFile
    NB: from main folder the other folders will follow like h/.../.././././sampleFile
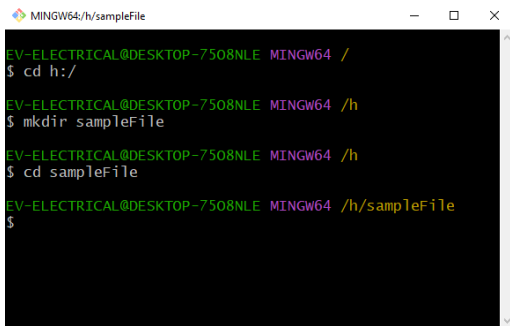
5. Install the various extensions required like python and virtual environment using the following commands.
    Install pip by first running the following command in git bash or powershell.
    curl https://bootstrap.pypa.io/get-pip.py -o get-pip.py
    Then: Run; py get-pip.py
6. After installing pip then install python and virtual environment using the following commands.
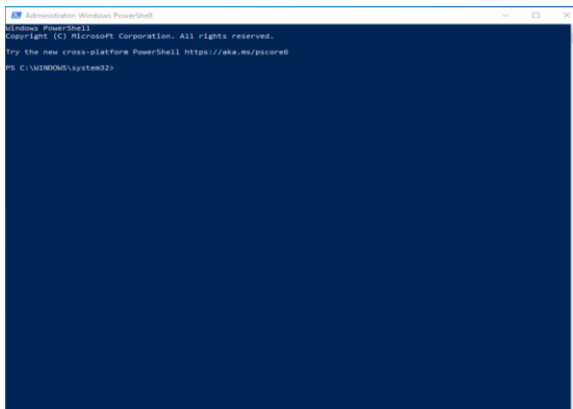   py –m pip install python   or   py –m pip install python
    py –m pip install virtualenv   or   py –m pip install virtualenv
7. When done check versions to confirm if installed fully.
    pip –version
    python –version
    Virtualenv --version

8. In the folder created, initialize git by: git init
9. Create a file e.g. index.py by: **vim index.py**   or **vim README.**md
10. Opens notepad, press i to start INSERT command for typing in file. When done press **Esc**, followed by   **: wq** to save and quit the file.
11. Check for status of file by: **git status '**you will find the files created'.
12. Add all files to commit by: git add .
13. Link git to user account to create repository by adding username as shown:
      git config --global user.name "YourUserName"
14. Link the email address registered with account: git config --global user.email "YourEmail"
15. Run: **git commit –m "My First Commit"** NB: inside the " " put the message to pass
16. Run: **git fetch** (This fetches the repository created so as to commit and push/save file to it) Open the repository and copy the link for the repository eg. git remote add origin https://github.com/..../FirstCommit.git
17. Run: **git push -u origin master** NB: If the origin name is not master but main you need to change to main to push
18. Open Github and refresh to check if updated. Done