Installing Windows 11: My Step-by-Step Guide

Hey! I just went through the process of installing Windows 11 on my PC, So here is step by step of how I did it.

Step 1: Check System Requirements

First things first, I made sure my PC met the Windows 11 system requirements. Here's the list I checked against:

- **Processor**: 1 GHz or faster with 2 or more cores.
- **RAM**: 4 GB or more.
- **Storage**: 64 GB or larger.
- **Firmware**: UEFI, Secure Boot capable.
- **TPM**: Version 2.0.
- **Graphics Card**: DirectX 12 compatible.
- **Display**: Over 9" with HD resolution (720p).

To confirm compatibility, I used the **PC Health Check tool**. It's quick and easy, and it gave me a green light!

Step 2: Backup Important Data

I didn't want to risk losing any files, so I backed up my important documents, photos, and other data to an external hard drive. Cloud services like OneDrive or Google Drive are also great alternatives.

Step 3: Obtain Windows 11

Option A: Via Windows Update (Upgrade)

1. I went to **Settings > Update & Security > Windows Update**.
2. Clicked on **Check for updates**.
3. Saw the option to upgrade to Windows 11 and clicked **Download and install**.

Option C: Creating a Bootable USB for a Fresh Install

1. Downloaded the Windows 11 ISO
2. Used the Rufus tool to create a bootable USB drive:
    - opened Rufus.
    - Selected the ISO file.
    - Made sure the USB drive was inserted and selected it in Rufus.
    - Clicked **Start** to create the bootable USB.

Step 4: Install Windows 11

For Windows Update or Installation Assistant:

1. I followed the on-screen instructions to complete the installation.
2. The PC restarted a few times during this process.

For Bootable USB:

1. Inserted the bootable USB drive and restarted my PC.
2. Accessed the BIOS/UEFI (I hit `F2` during boot, but this can vary; common keys include `F2`, `F12`, `DEL`, `ESC`).
3. Set the USB drive as the primary boot device.
4. Saved and exited BIOS.
5. The Windows 11 setup started. I followed these steps:
    - Selected language, time, and keyboard preferences.
    - Clicked **Install now**.
    - Entered the product key (or clicked "I don't have a product key").
    - Selected **Custom: Install Windows only (advanced)** for a clean install.
    - Choose the partition for installation. I formatted it to start fresh.

Step 5: Complete the Setup

1. After the installation, Windows 11 booted up.
2. I chose my region and keyboard layout.
3. Connected to Wi-Fi.
4. Signed in with my Microsoft account (optional but I did it for the extra features).
5. Selected my privacy settings.
6. Followed the remaining prompts to finalise the setup.

Step 6: Update and Install Drivers

1. I went to **Settings > Update & Security > Windows Update** and installed any available updates.
2. I also visited my PC manufacturer's website to download and install the latest drivers.

Step 7: Restore Data and Applications

1. I copied my backed-up data to the new Windows 11 system.
2. Reinstalled my favourite apps and tools.

Step 8: Customise and Explore

1. Customise the Start menu and Taskbar to my liking.
2. Played around with the new features, like Widgets and Snap Layouts, to make my workflow smoother.

   Here are some screenshots of installing window 11

Downloading Visual Studio Code: How I Did It

Step 1: Visit the Visual Studio Code Website

Navigated to the VS Code Website: Went to Visual Studio Code's official website.

Step 2: Download the Installer

Clicked the Download Button: The site automatically showed a **"Download for Windows"** button since I'm on Windows. If you're on macOS or Linux, it should detect your OS and give the right download link.

- I clicked the **Download for Windows** button.
- The download started, and I waited for it to finish (it was quick, just a few minutes).

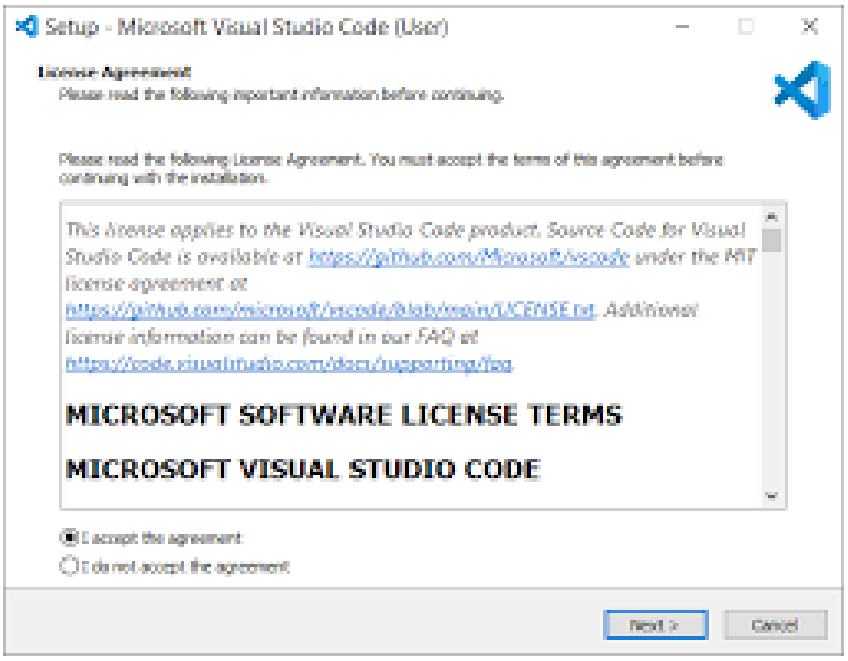
Step 3: Run the Installer

Opened the Download Folder: I found the `VSCodeSetup.exe` file in my Downloads folder.

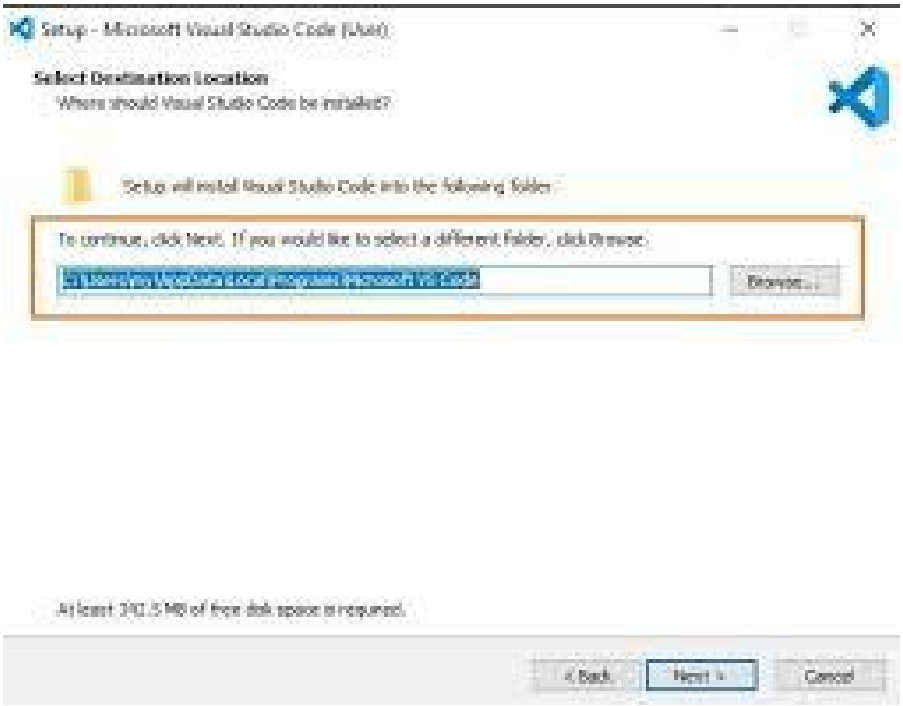
Started the Installer: Double-clicked on the `.exe` file, and the setup wizard launched.

Step 4: Follow Installation Prompts

Accepted the Licence Agreement: The installer asked me to accept the terms. I read through it quickly and clicked **Next**.

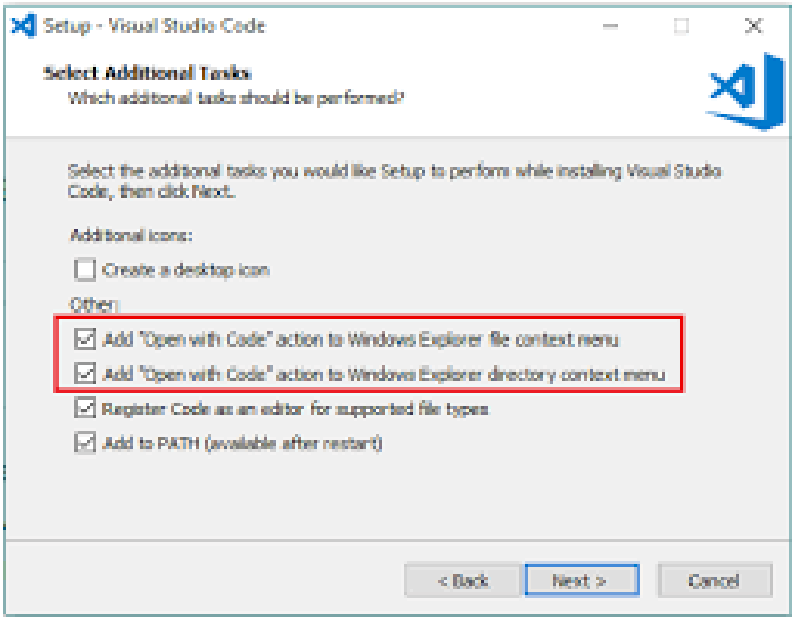


Choose Installation Location: I left the default location as is, but you can choose a different folder if you prefer. Clicked **Next**.



Select Additional Tasks: Here, I checked a few options:

- Created a desktop icon.
- Added **“Open with Code”** to the context menu.
- Added VS Code to PATH (super useful for command-line use).
- Clicked **Next**

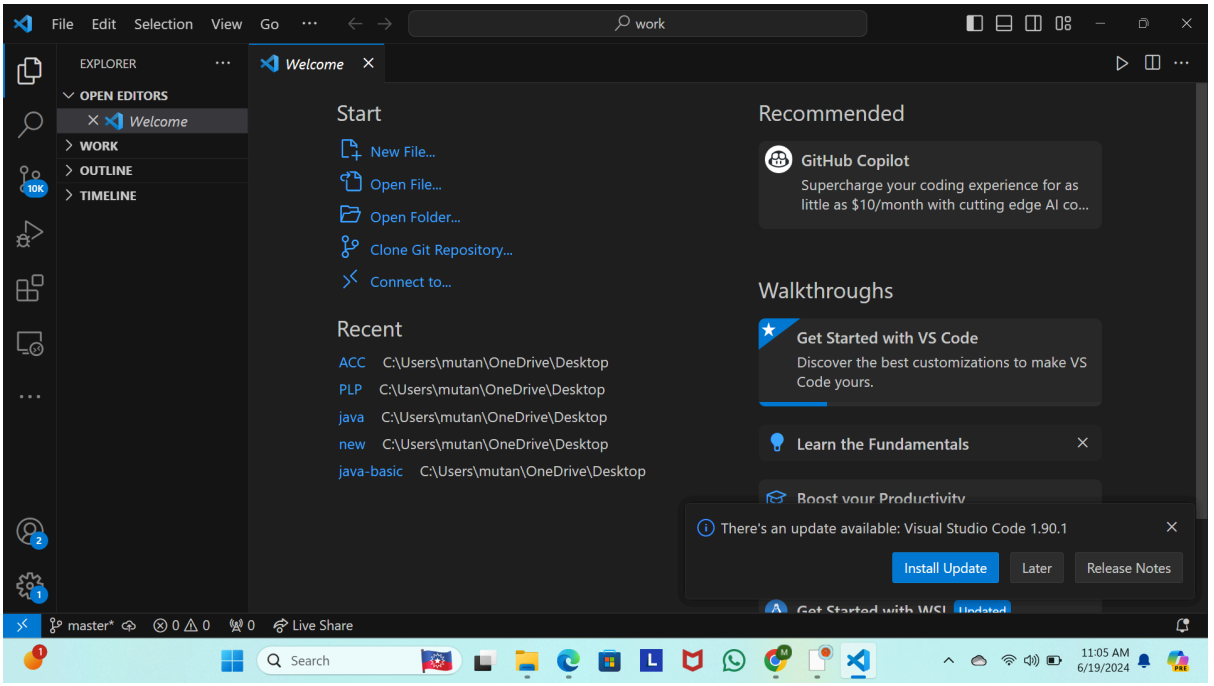


Install: Clicked the **Install** button. The installation process only took a few minutes.

Launch VS Code: When it was done, I made sure the **"Launch Visual Studio Code"** checkbox was ticked and then clicked **Finish**.

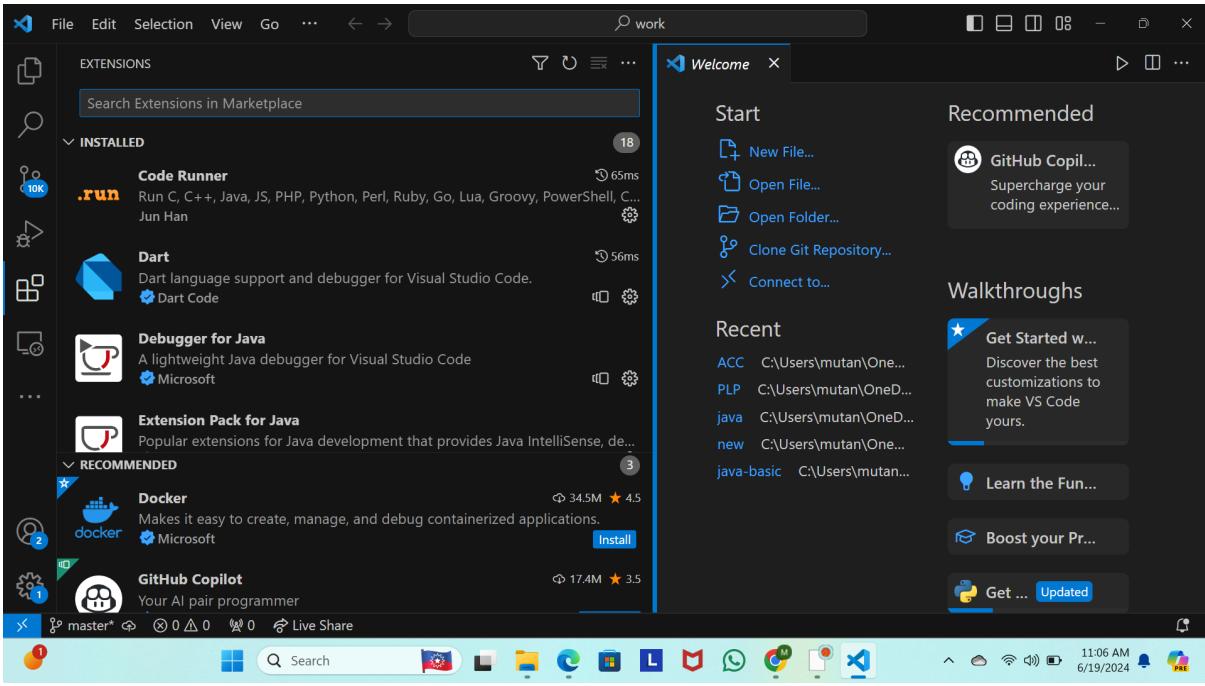
Step 5: Initial Setup and Configuration

Welcome Screen: VS Code opened up, showing the welcome screen. I took a moment to explore some of the introductory options and tutorials.



Install Extensions: I clicked the **Extensions** icon (left sidebar or press `Ctrl` + `Shift` + `X`), then searched for extensions I needed. Here's what I installed:

- Python
- Prettier
- GitLens
- Live Server



Configure Settings: I went to **File > Preferences > Settings** to tweak a few settings to my liking, like font size and theme.

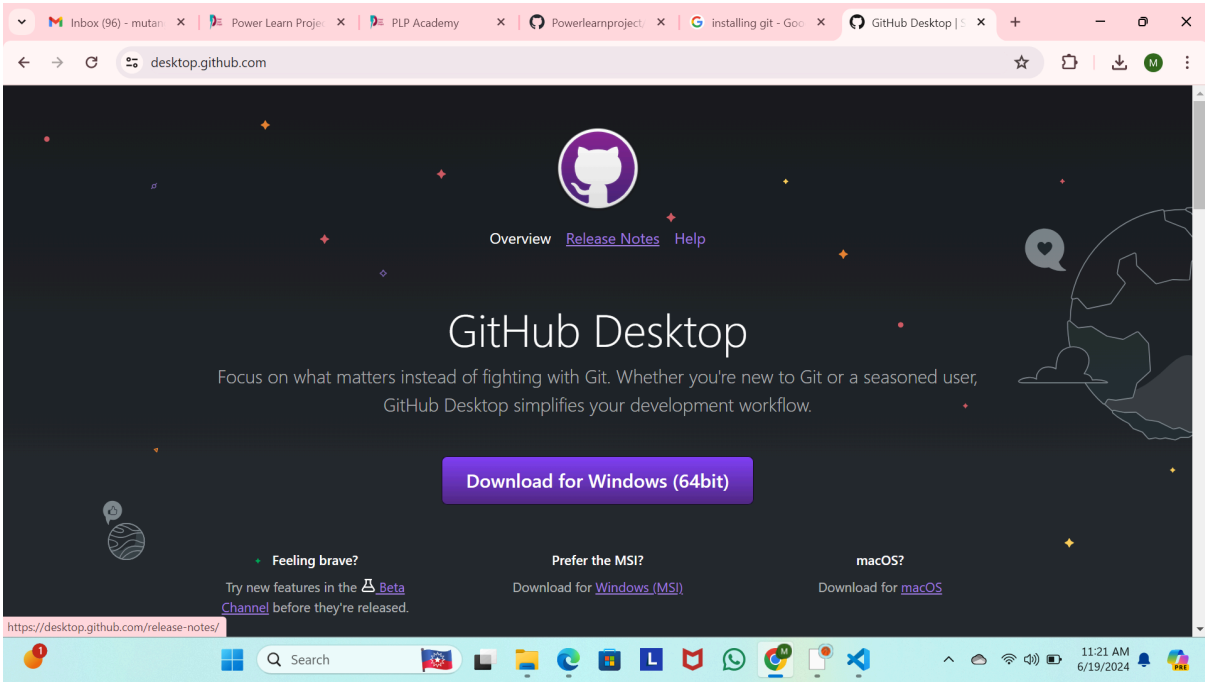
Downloading GitHub Desktop: How I Did It

Step 1: Visit the GitHub Desktop Website

1. **Opened My Browser**: I launched Chrome, but any browser will work fine.
2. **Navigated to the GitHub Desktop Site**: Went to desktop.github.com.

Step 2: Download GitHub Desktop

3. **Clicked the Download Button**: There's a big **"Download for Windows"** button right on the front page. Since I'm on Windows, I clicked that. For macOS, it'll show **"Download for macOS"**. If you're on macOS, you can click the appropriate button as well.
    - The installer (`GitHubDesktopSetup.exe` for Windows or `.dmg` for macOS) started downloading immediately.



Step 3: Run the Installer

4. **Opened the Download Folder**: Once the download finished, I went to my Downloads folder and found the `GitHubDesktopSetup.exe` file.
5. **Started the Installer**: Double-clicked the `.exe` file to launch the setup.

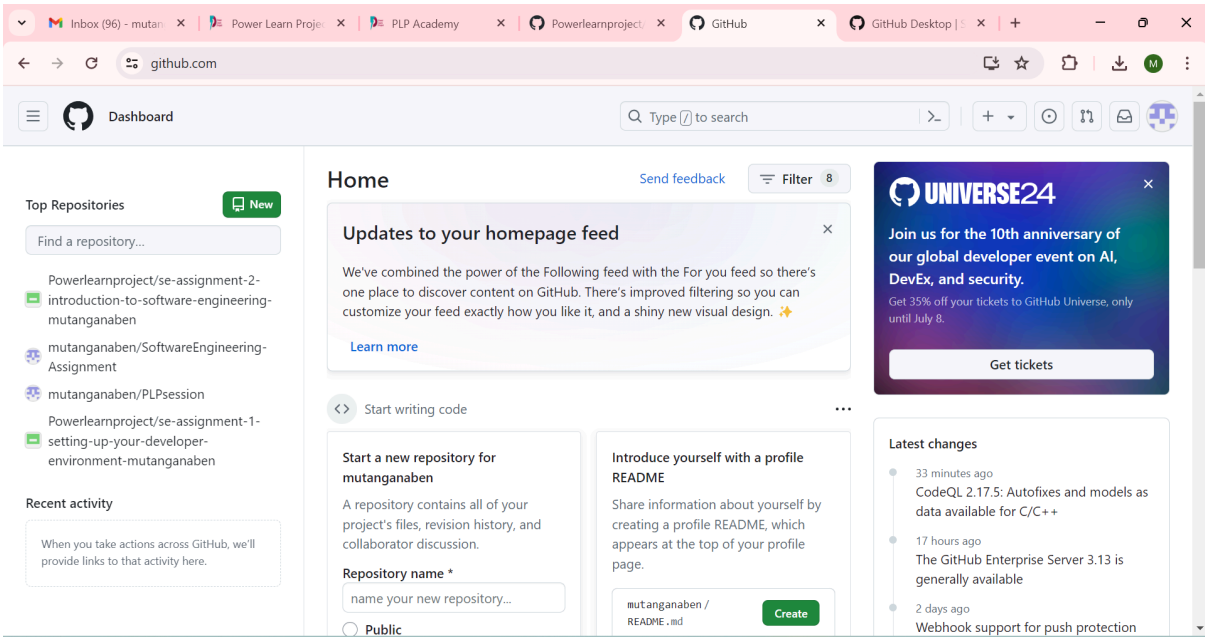
Step 4: Install GitHub Desktop

6. **Run the Installer**: The setup wizard appeared and began installing GitHub Desktop automatically. I didn't have to click through any additional prompts—it just did its thing.

Step 5: Launch and Configure GitHub Desktop

7. **Launch the App**: After the installation completed, GitHub Desktop opened up. If it doesn't open automatically, you can find it in your Start menu (or Applications folder on macOS).

8. **Sign In to GitHub**: The first screen prompted me to sign in to my GitHub account. I clicked **"Sign in to GitHub.com"**.
    - I entered my GitHub username and password, or you can use your web browser for OAuth.



9. **Configure GitHub Desktop**: After signing in, I set up my preferences:
    - **Git Config**: Entered my name and email address to be used in my commits.
    - **Default Editor**: Choose my preferred code editor for editing files (I use VS Code, but you can select whatever you like).

Step 6: Clone or Create Repositories

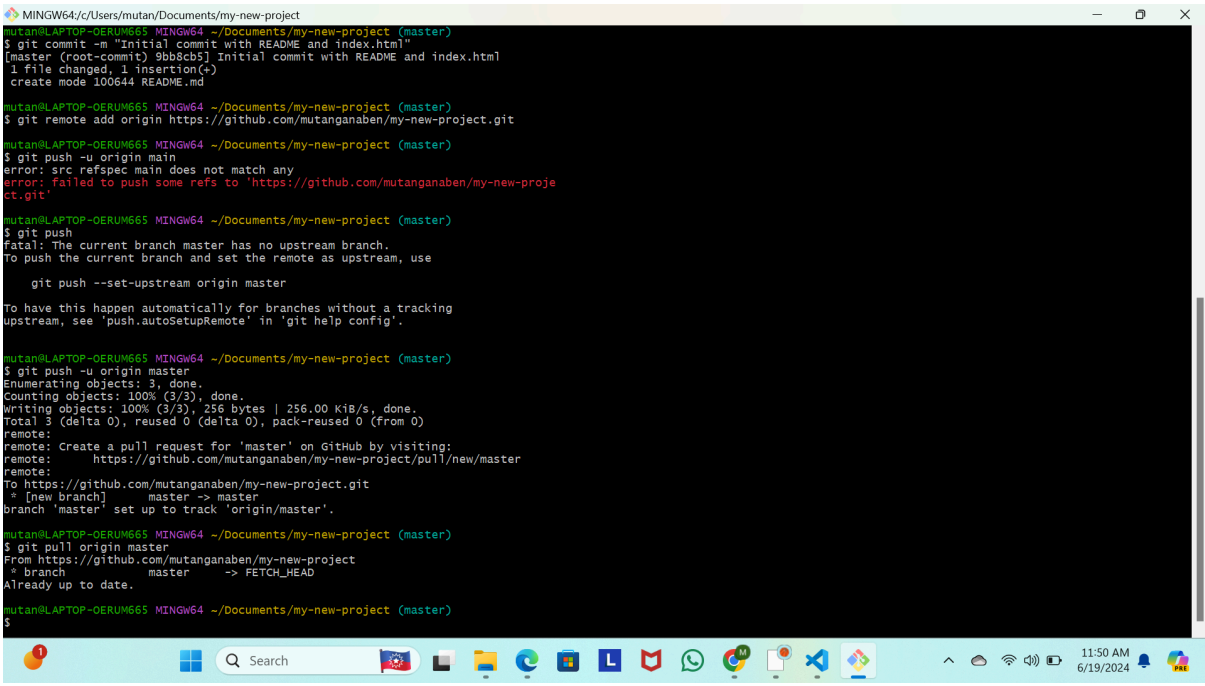
10. **Clone a Repository**:
    - Clicked on **"Clone a Repository from the Internet"**.

- Entered the repository URL I wanted to clone and selected the local directory where I wanted to save it.
- Clicked **Clone**.

11. **Create a New Repository**:
    - Alternatively, I clicked **“Create a New Repository on Your Hard Drive”**.
    - Filled in the repository name, description, and path.
    - Clicked **Create Repository**.

Step 7: Exploring GitHub Desktop

12. **Explore Features**:
    - **Commits**: I made a few changes to a file in my repository and clicked **Commit to main**.
    - **Push to GitHub**: After committing, I clicked **Push Origin** to upload my changes to GitHub.



13. **Sync Changes**: Anytime I want to sync changes, I use **Fetch Origin** or **Pull Origin** to update my local copy with the latest c

Downloading and Installing Python: How I Did It

Step 1: Download Python Installer

1. **Open Browser**: I opened my browser (I use Chrome, but any browser works fine).
2. **Go to Python's Official Website**: I navigated to Python's official download page.
3. **Download the Installer**: On the download page, I saw a big yellow button that said **“Download Python 3.X.X”** (the version number may vary). I clicked it to download

the installer.

- The `.exe` file started downloading. I found it in my Downloads folder after a few moments.

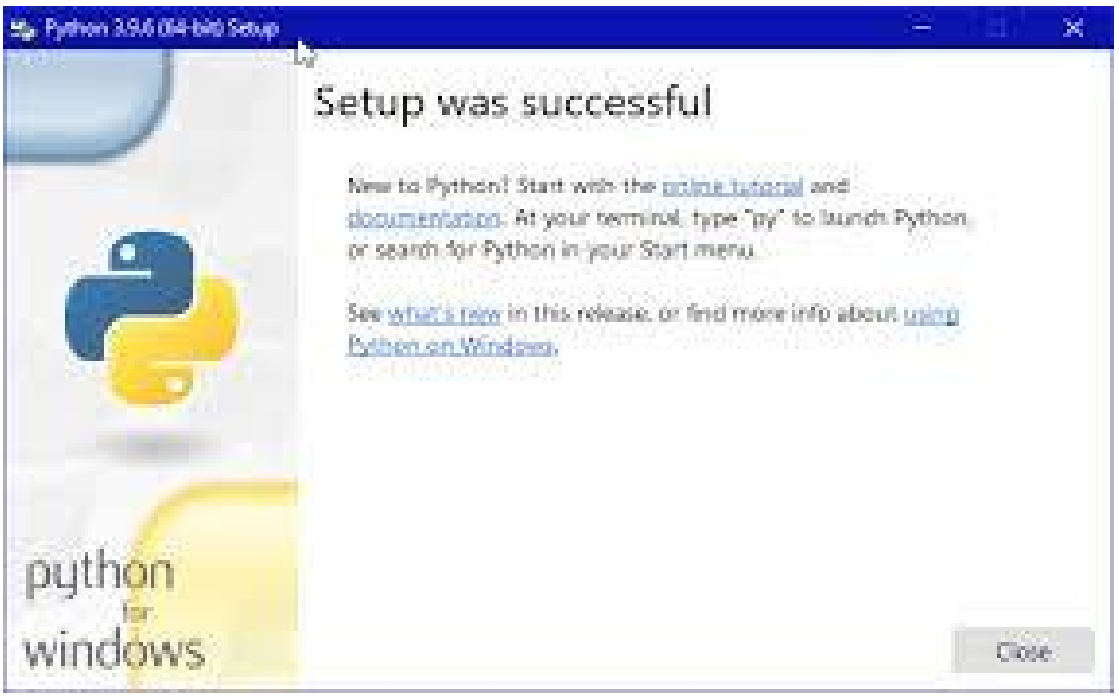
Step 2: Run the Installer

4. **Open the Installer**: I double-clicked the downloaded `python-3.X.X.exe` file to start the installation process.

Step 3: Install Python

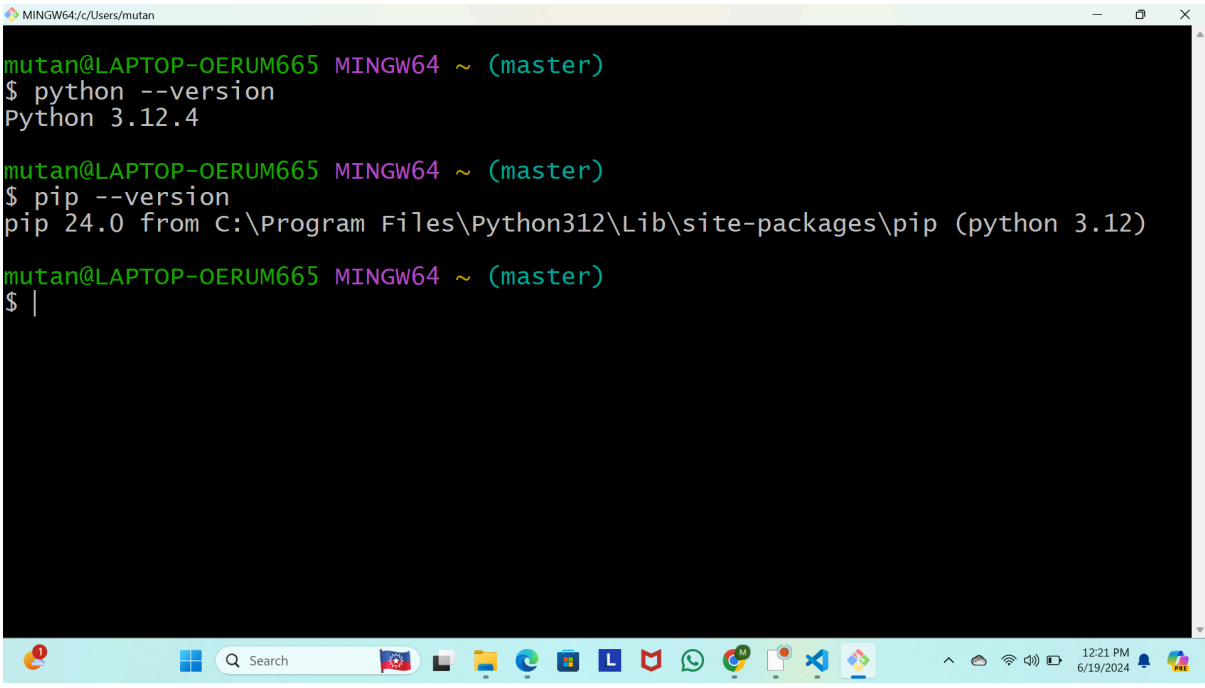
5. **Modify Installation Options**:
   - At the start of the installer, I saw an option to **"Add Python 3.X to PATH"**. I made sure to check this box because it allows me to run Python from the command line.
   - Then, I clicked **Install Now**.

6. **Wait for Installation**:
   - The installer showed a progress bar as it installed Python. This took a couple of minutes.
   - Once the installation was completed, I saw a confirmation message. I clicked **Close** to finish the process.



Step 4: Verify Python Installation

7. **Open Command Prompt**: To confirm that Python was installed correctly, I opened the Command Prompt (you can search for “cmd” in the Start menu and hit Enter).
8. **Check Python Version**:
    - In the Command Prompt, I typed



Installing MySQL Database: Step-by-Step Guide

Step 1: Download MySQL Installer

1. **Visit MySQL Community Downloads**: I opened my web browser and navigated to the MySQL Community Downloads page.
2. **Download MySQL Installer**: On the downloads page, I selected the appropriate installer version for my operating system. For Windows, I chose the MySQL Installer for Windows.
3. **Start Download**: Clicked on the download button to start downloading the MySQL Installer executable (`.exe`) file.

Step 2: Run MySQL Installer

Open the Installer: Once the download completed, I located the installer file (e.g., `mysql-installer-web-community-*.exe`) in my Downloads folder and double-clicked it to launch the installer.

Choose Setup Type: In the MySQL Installer window, I selected **Developer Default** setup type. This option instals MySQL Server, MySQL Workbench (a visual tool for database design and management), and other essential components.

Install MySQL Products: Proceeded with the default selections (MySQL Server and MySQL Workbench were selected by default) and clicked **Next**.

Check Requirements: The installer checked for any missing dependencies or additional requirements. If necessary, I followed the prompts to install any missing prerequisites.

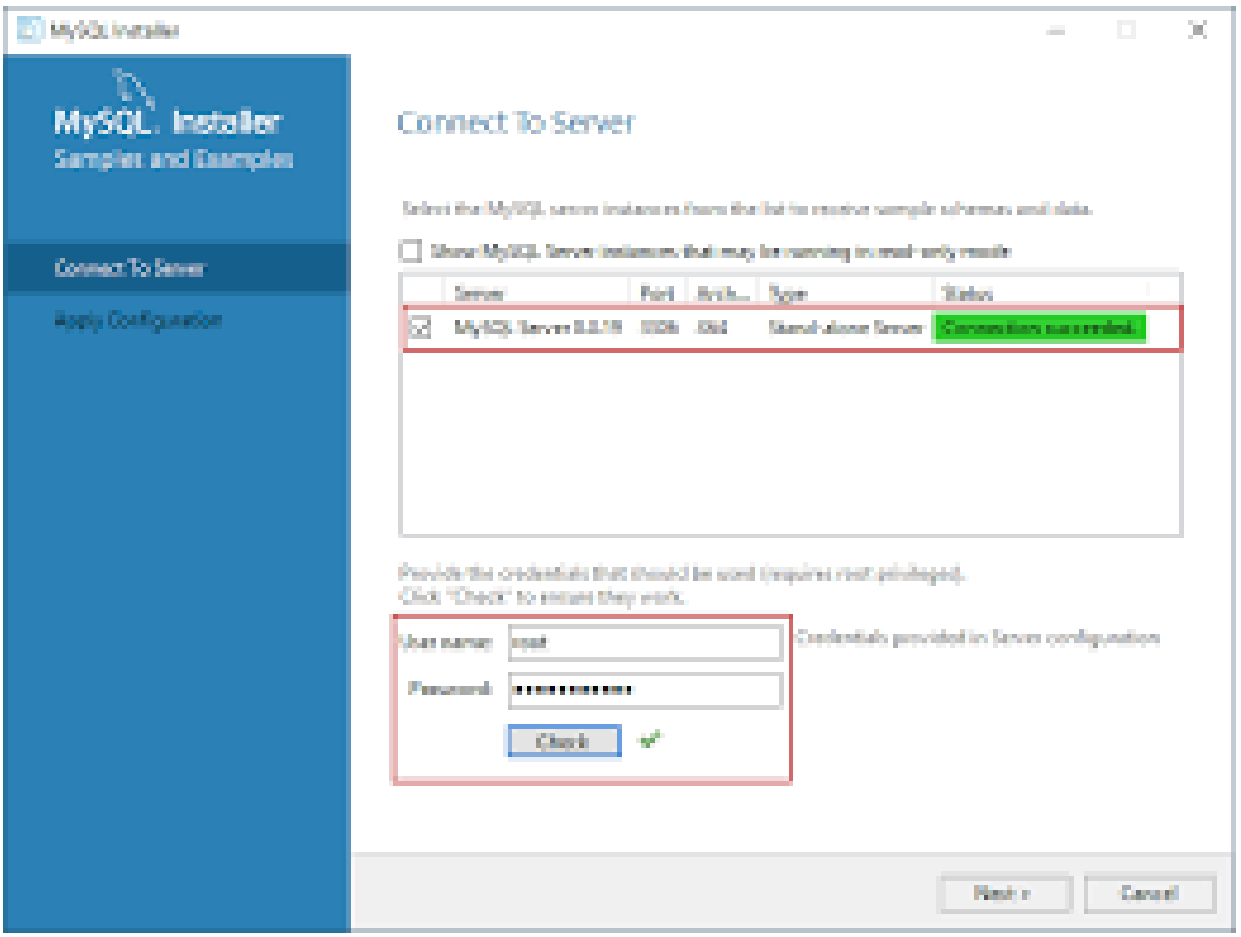
Step 3: Configure MySQL Server

Configuration: On the configuration screen, I chose the setup type (Standalone MySQL Server or MySQL InnoDB Cluster) and configured the server options.

- **Standalone MySQL Server**: Selected if I wanted a single MySQL server instance.
- **MySQL InnoDB Cluster**: Selected if I needed a high availability solution.

Port Configuration: Kept the default port (3306) unless I had a specific reason to change it.

Set Password: Set the root password for MySQL Server. Ensured to choose a strong password and noted it down securely.



Start Installation: Clicked **Execute** to begin the installation process. Waited for the installer to download and install MySQL Server and other selected components.

Step 4: Complete Installation

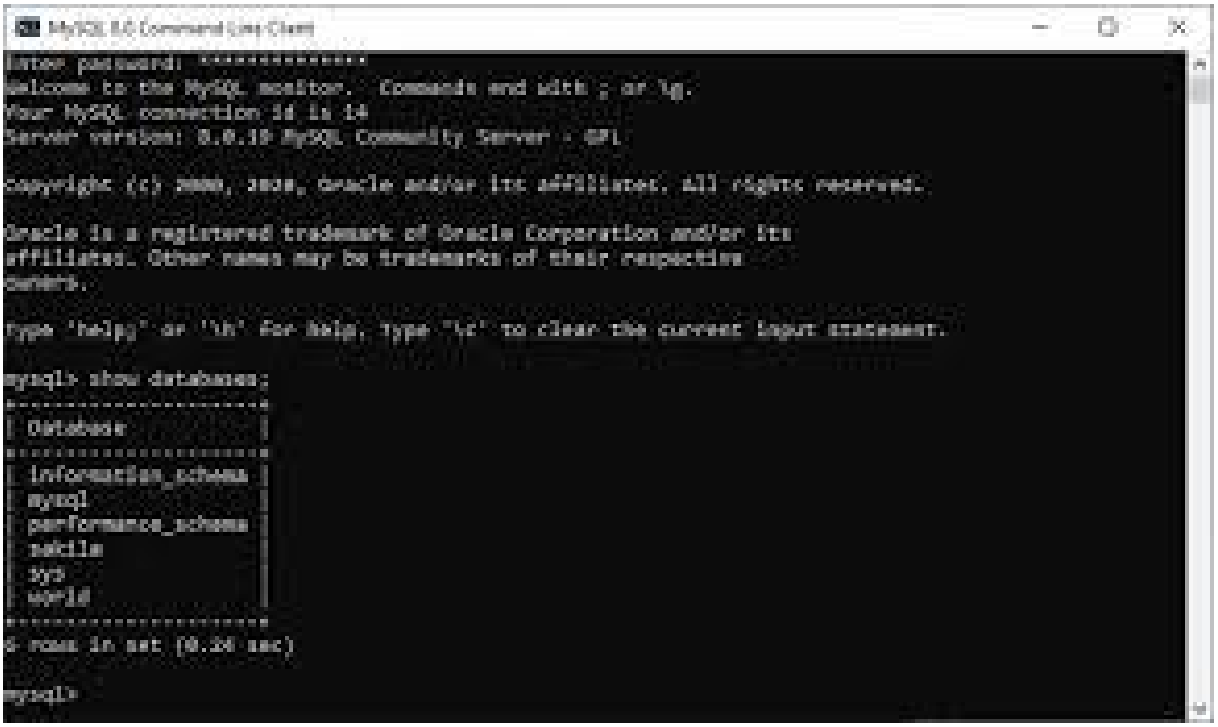
Installation Complete: Once the installation finished successfully, click **Next** to complete the setup.

Finish: Clicked **Finish** to exit the installer. MySQL Server and MySQL Workbench were now installed on my machine.

Step 5: Verify MySQL Installation

Open MySQL Workbench: To verify the installation, I opened MySQL Workbench from the Start menu.

Connect to MySQL Server: In MySQL Workbench, click on the **+** icon next to "MySQL Connections" to create a new connection.



- Enter `localhost` as the hostname, `root` as the username, and the password I set during installation.

  Test Connection: Clicked **Test Connection** to ensure MySQL Workbench could connect to MySQL Server without any issues.

  Success: If the connection was successful, I saw a confirmation message and could start using MySQL Server through MySQL Workbench.

Challenges in Installing Visual Studio Code

1. **Installation Fails**
   - **Issue**: VS Code installer fails due to permission issues or corrupted download.
   - **Solution**:
     - Run the installer as an administrator.
     - Download the installer again to ensure it's not corrupted.
2. `code` **Command Not Found**
   - **Issue**: The `code` command is not recognized in the terminal.
   - **Solution**:
     - Ensure VS Code is added to the PATH during installation.
     - Add VS Code manually to the PATH: `C:\Program Files\Microsoft VS Code\bin`.
3. **Extensions Not Installing**
   - **Issue**: Extensions may fail to install due to connectivity issues or internal errors.
   - **Solution**:
     - Check internet connection and retry installation.

        - Restart VS Code or install extensions via command line: `code --install-extension <extension-name>`.
4. **Performance Issues**
    - **Issue**: VS Code runs slowly or crashes frequently.
    - **Solution**:
        - Disable unnecessary extensions.
        - Increase memory limits in settings for handling large files.
5. **Theme and Font Customization Issues**
    - **Issue**: Changes to themes or fonts are not applied.
    - **Solution**:
        - Ensure you save changes and restart VS Code.
        - Check for conflicting settings in your settings.json file.

Challenges in Installing Python

1. **Path Environment Variable Issues**
    - **Issue**: Python or pip command is not recognized.
    - **Solution**:
        - Ensure you check **Add Python to PATH** during installation.
        - Manually add Python to PATH via System Properties > Environment Variables.
2. **Permission Issues**
    - **Issue**: Permission errors during installation or running scripts.
    - **Solution**:
        - Install Python with administrative privileges.
        - Run scripts or install packages with administrative permissions if needed.
3. **Pip Not Found**
    - **Issue**: pip is not recognized as a command.
    - **Solution**:
        - Ensure pip is installed by checking `pip --version`.
        - Use `python -m ensurepip` to install pip if missing.
4. **Virtual Environment Issues**
    - **Issue**: Problems creating or activating virtual environments.
5. **Compatibility Issues with Packages**
    - **Issue**: Some packages fail to install due to version incompatibility.
    - **Solution**:
        - Use `pip install --upgrade <package>` to update to the latest compatible version.
        - Check package documentation for compatibility information.

Challenges in Setting Up GitHub

1. **SSH Key Issues**
    - **Issue**: SSH key authentication fails.
    - **Solution**:
        - Generate a new SSH key and add it to your GitHub account.

        - Ensure the SSH agent is running: `eval "$(ssh-agent -s)"`.
2. **Git Configuration Issues**
    - **Issue**: Git configurations (user.name, user.email) are incorrect or not set.
    -
3. **Repository Initialization Problems**
    - **Issue**: Unable to initialise a local or remote repository.
    - **Solution**:
        - Ensure you have the correct permissions to create repositories on GitHub.
        - Use the correct commands to initialise and push the repository.