Setup Development Environment

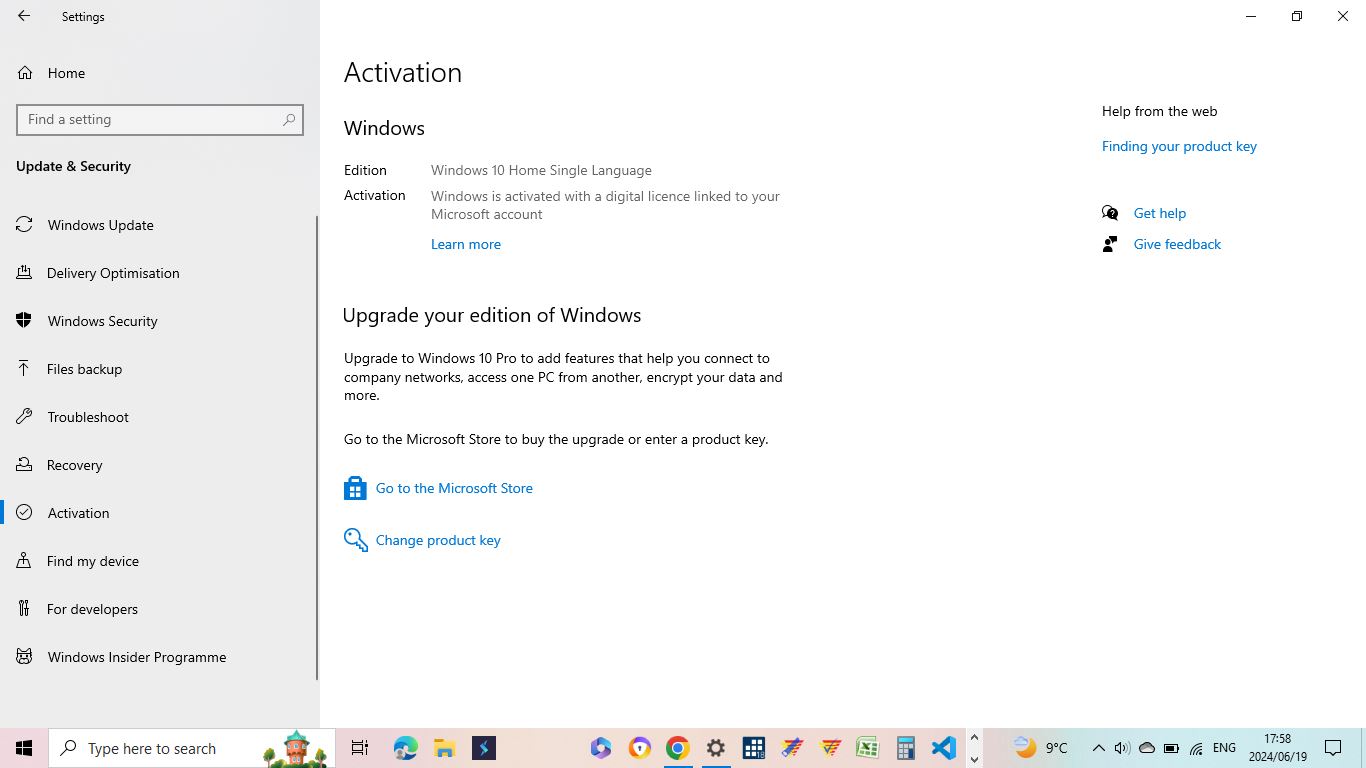
#Assignment: Setting Up Your Developer Environment

#Objective: This assignment aims to familiarize you with the tools and configurations necessary to set up an efficient developer environment for software engineering projects. Completing this assignment will give you the skills required to set up a robust and productive workspace conducive to coding, debugging, version control, and collaboration.

#Tasks:

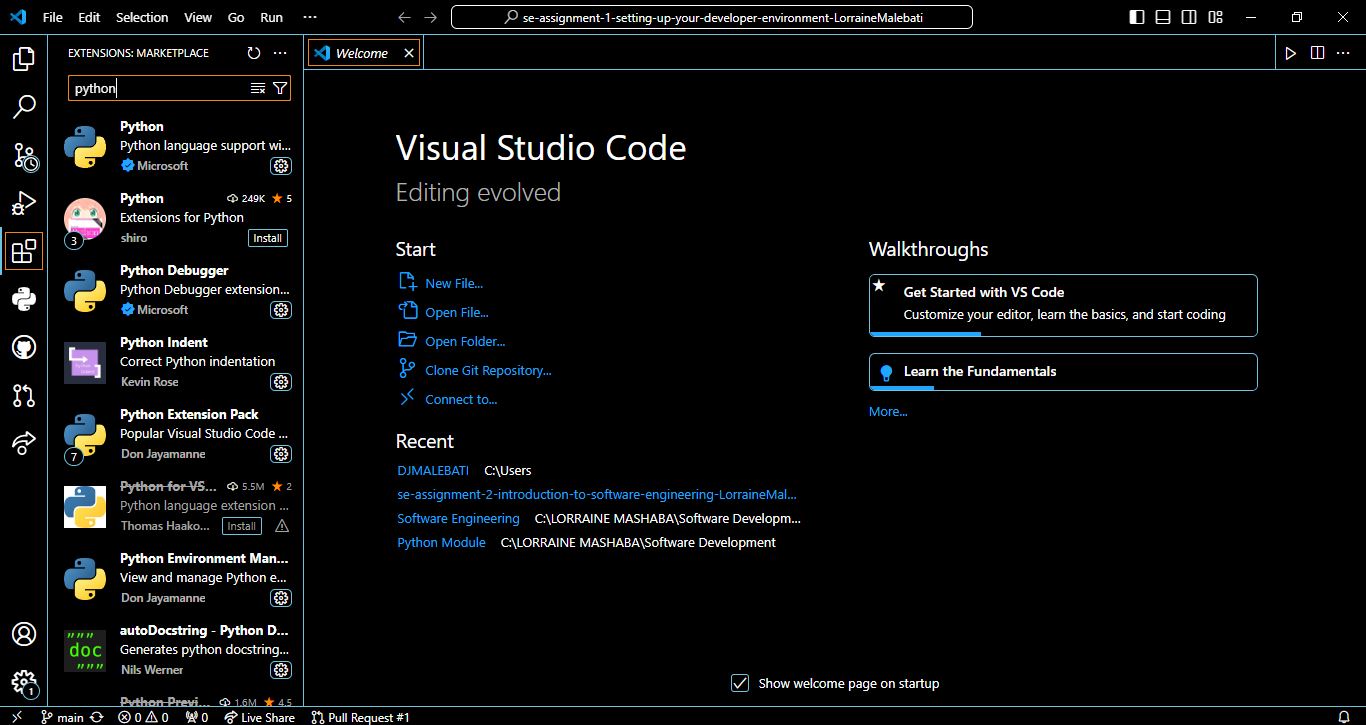
1. Select Your Operating System (OS): Choose an operating system that best suits your preferences and project requirements. Download and Install Windows 11. <https://www.microsoft.com/software-download/windows11>

The steps to be followed to download and install Windows 11 were to determine if my laptop is compatible with the minimum system requirements. Then the next was to use the Windows Update tool to check for the latest version. If upgrading from Windows 10, ensure it is fully updated and compatible. Then downloaded the Windows 11 installation media creation tool from Microsoft's website and run it. The on-screen instructions were carefully and accepted the license terms and customizing privacy settings.



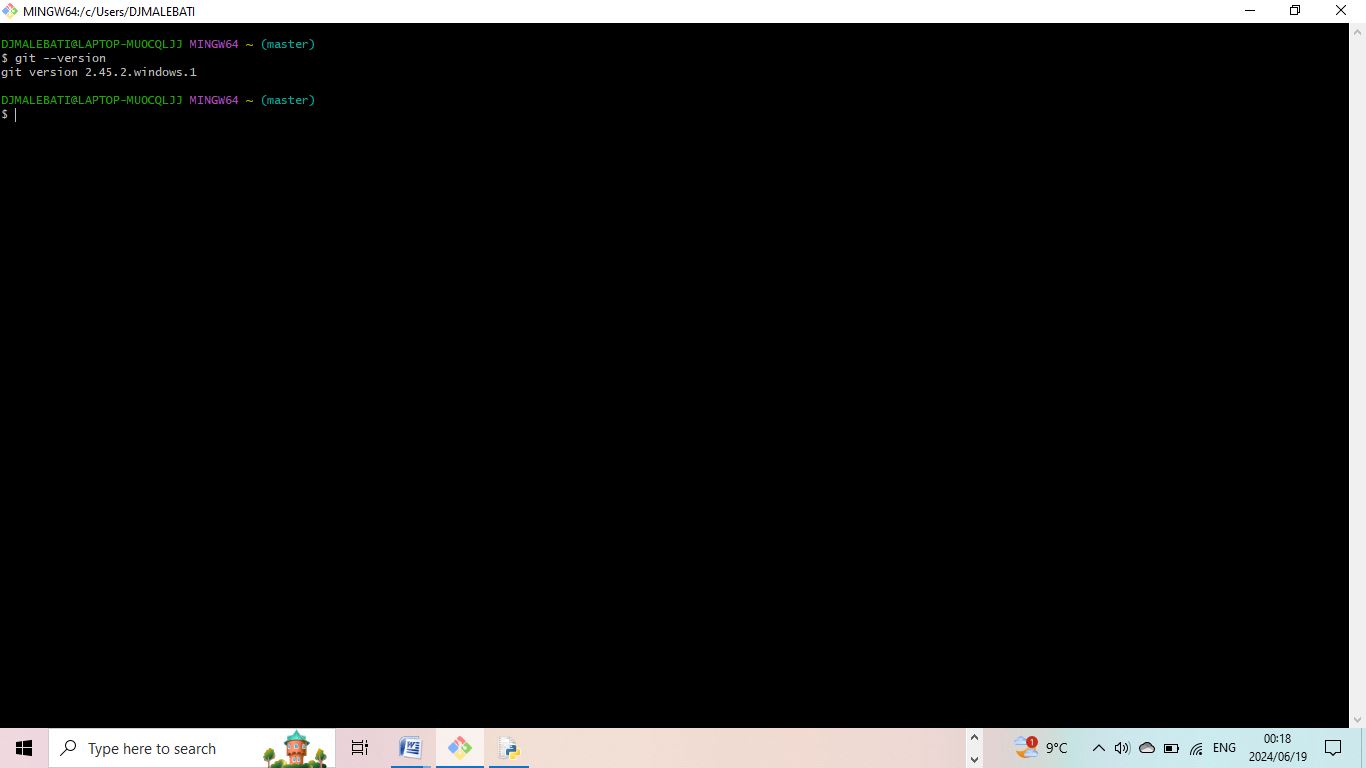
1. Install a Text Editor or Integrated Development Environment (IDE): Select and install a text editor or IDE suitable for your programming languages and workflow. Download and Install Visual Studio Code. <https://code.visualstudio.com/Download>

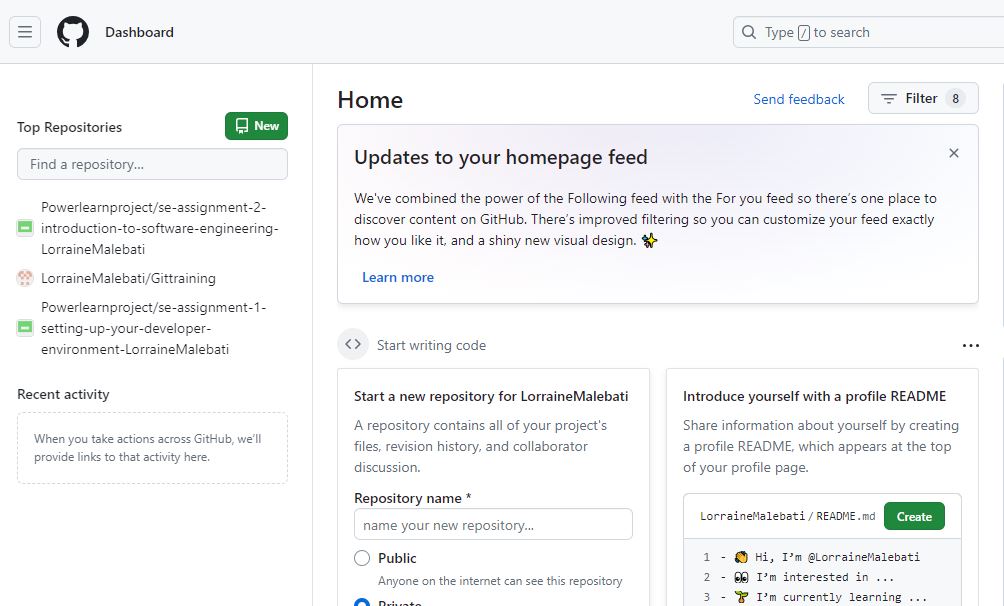
I went to VS Code website at <https://code.visualstudio.com>. I clicked a download button for windows. I ran the downloaded .exe file and follow the installation prompts, and checked "Add to PATH". Once installed and launched VS Code, I customize it by installing extensions, adjusting settings, and configuring keybindings to suit your development needs.



1. Set Up Version Control System: Install Git and configure it on your local machine. Create a GitHub account for hosting your repositories. Initialize a Git repository for your project and make your first commit. [https://github.com](https://github.com/)

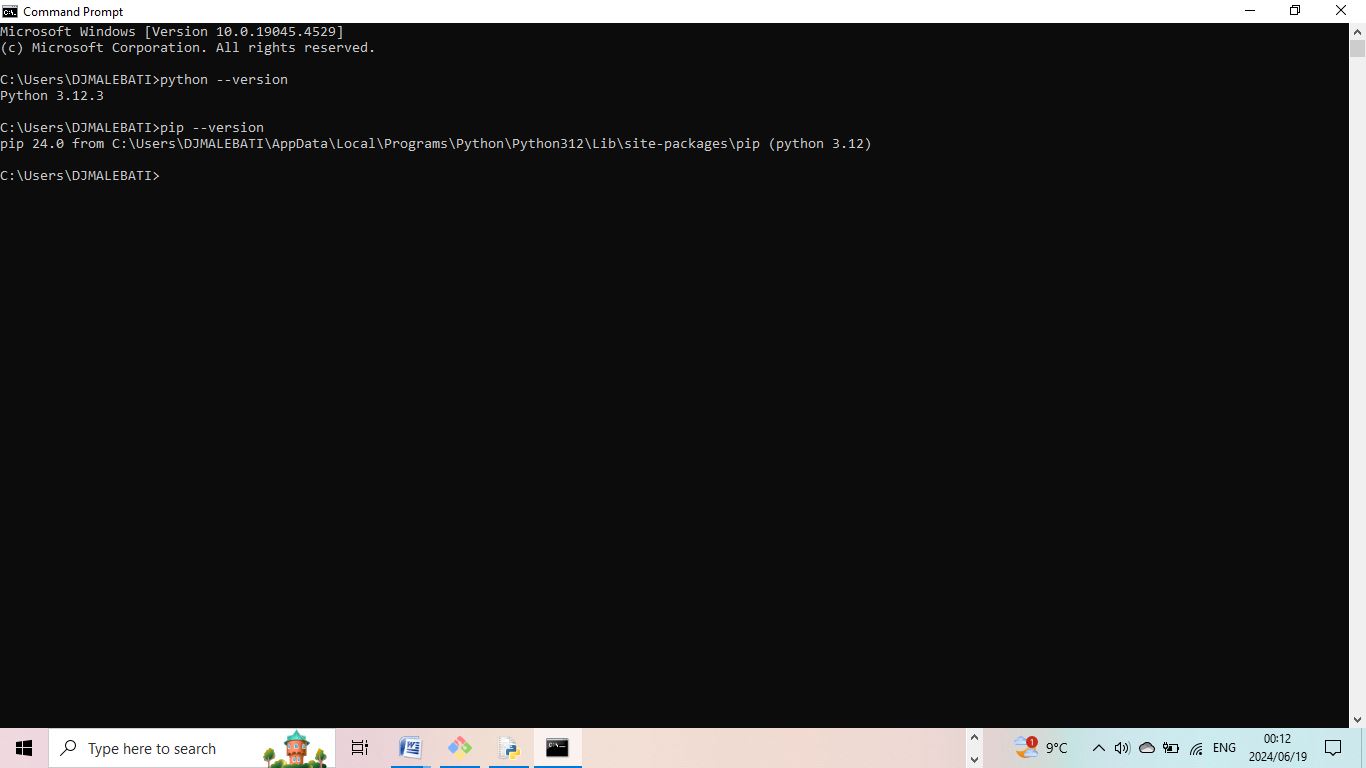
I went to GitHub website (https://github.com) and clicked on the "Sign up" button. I entered my email address, create a password, and choose a username. After filling in these details, I solved a puzzle to verify that you I am not a robot. I create an account and follow the on-screen instructions, which included verifying my email address by clicking a link sent to your inbox. After email verification, I started creating repositories or collaborating on projects.





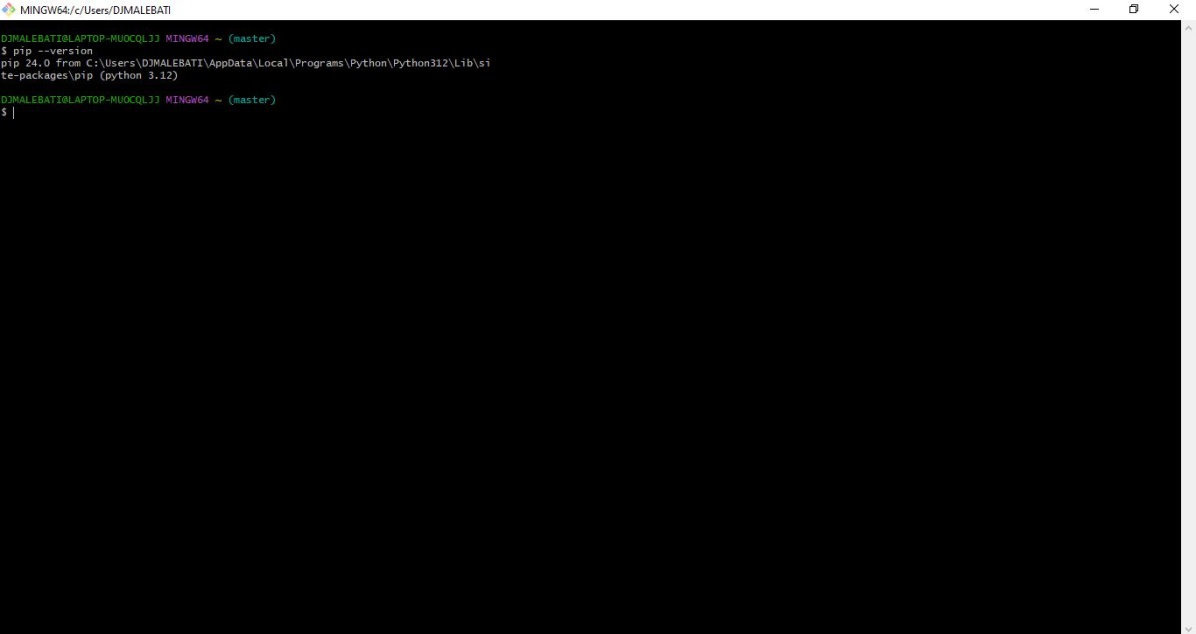
1. Install Necessary Programming Languages and Runtimes: Install Python from [http://wwww.python.org](http://wwww.python.org/) programming language required for your project and install their respective compilers, interpreters, or runtimes. Ensure you have the necessary tools to build and execute your code.

I went to Python website at https://www.python.org. Navigate to the Downloads section, which automatically suggests the appropriate version for your operating system which is windows for my PC. Click the download button for the recommended version. Once the installer is downloaded, open it and follow the on-screen instructions. On Windows, I made sure to check the box that says "Add Python to PATH" before clicking "Install Now." After installation, verify that Python is installed correctly by opening a terminal (or Command Prompt on Windows) and typed python –version.



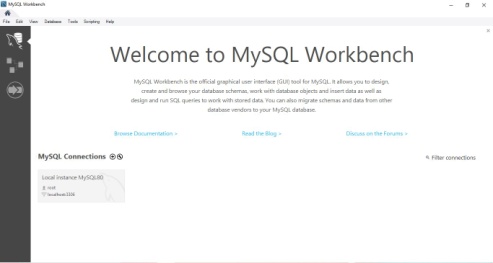
1. Install Package Managers: If applicable, install package managers like pip (Python).

To check if pip is already installed, I opened Git and typed pip –version



1. Configure a Database (MySQL): Download and install MySQL database. <https://dev.mysql.com/downloads/windows/installer/5.7.html>

I went to MySQL website at https://dev.mysql.com/downloads/ and download the MySQL Community Server for windows. I ran the downloaded installer and follow the setup wizard, making sure to choose the appropriate configuration settings such as server type and root password.

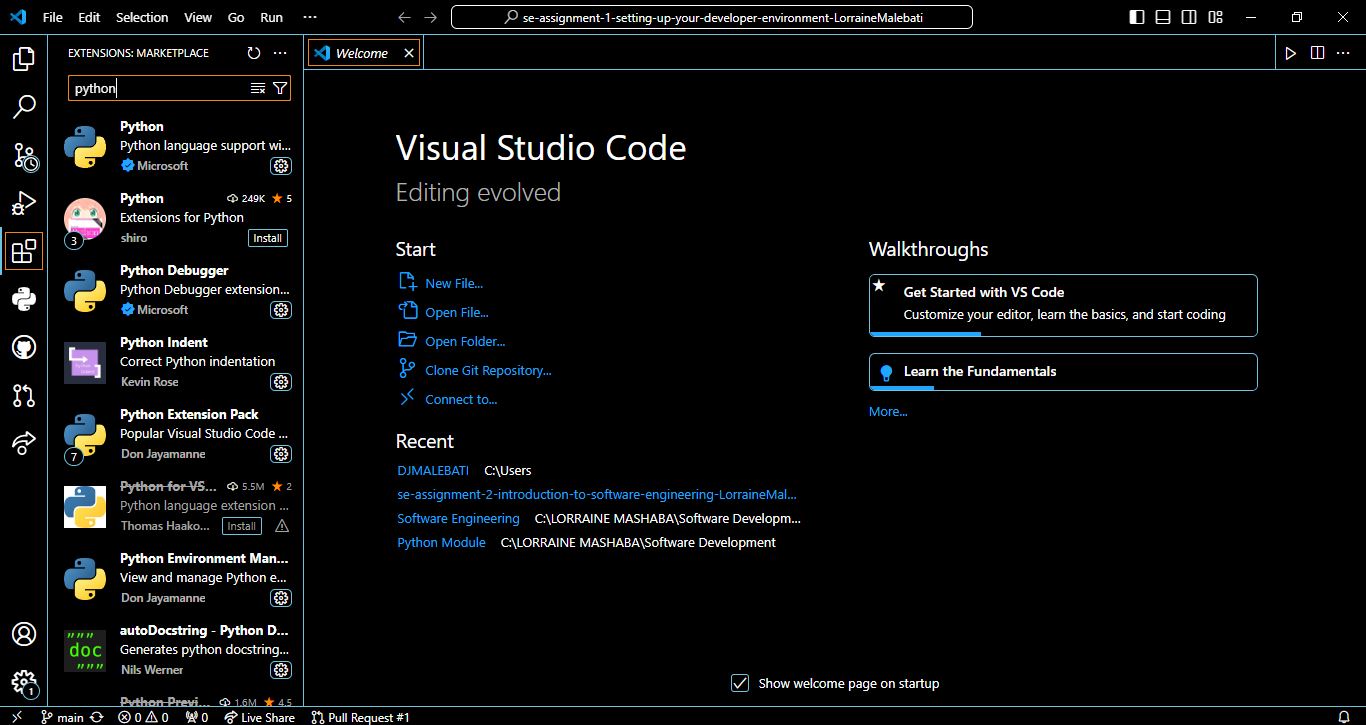


1. Set Up Development Environments and Virtualization (Optional): Consider using virtualization tools like Docker or virtual machines to isolate project dependencies and ensure consistent environments across different machines.

Setting up development environments using virtualization tools like Docker or virtual machines can greatly enhance project isolation and consistency across different machines. Docker allows you to create lightweight containers that package your application and its dependencies, ensuring it runs the same regardless of the environment. To get started, install Docker from https://www.docker.com, then create a Dockerfile that specifies the environment setup for your project. Use docker-compose to manage multi-container applications if needed. Virtual machines (VMs), on the other hand, provide a full-fledged virtualized operating system using tools like VirtualBox or VMware. This approach is useful for running different OS environments on the same hardware. Install your preferred VM software, create a new VM, and install the necessary development tools and dependencies within it. Both Docker and VMs ensure that your development environment is consistent, reproducible, and isolated from the host system, mitigating issues caused by differing setups and configurations.

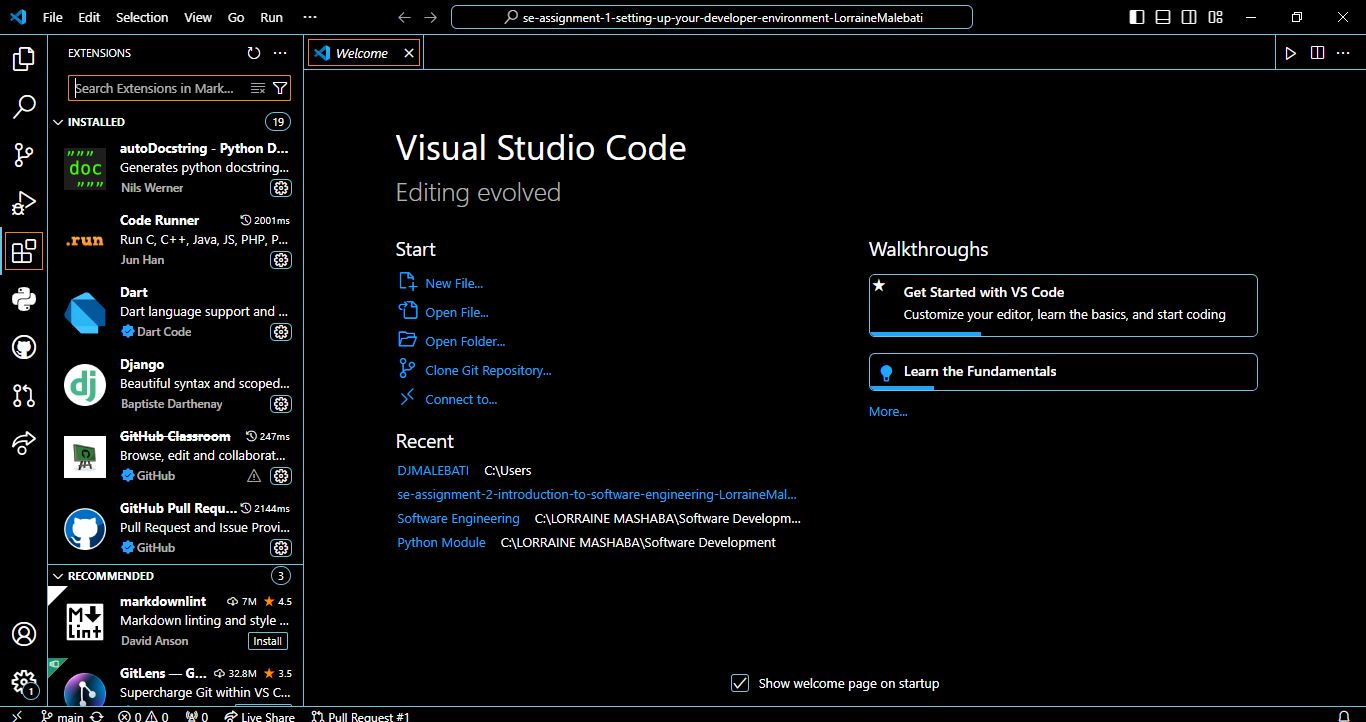
1. Explore Extensions and Plugins: Explore available extensions, plugins, and add-ons for your chosen text editor or IDE to enhance functionality, such as syntax highlighting, linting, code formatting and version control integration.

To achieve this, it is vital to open visual studio code, navigate to the extensions. On the search bar located at the top, search for the extension of your preference then select it, It is important to select a Microsoft verified extension, then install.



1. Document Your Setup: Create a comprehensive document outlining the steps you've taken to set up your developer environment. Include any configurations, customizations, or troubleshooting steps encountered during the process.

The initial step was installation of Visual Studio Code, which was followed by ensuring that the user interface was customized according to my preferences. The next step was to navigate to the extensions and installed essential extensions I will require for my project (s). Such extensions included and were not limited to Python, Code Runner, Dart, Django and GitHub to name a few.



#Deliverables:

* Document detailing the setup process with step-by-step instructions and screenshots where necessary.
* A GitHub repository containing a sample project initialized with Git and any necessary configuration files (e.g., .gitignore).
* A reflection on the challenges faced during setup and strategies employed to overcome them.

#Submission: Submit your document and GitHub repository link through the designated platform or email to the instructor by the specified deadline.

#Evaluation Criteria:\*\*

* Completeness and accuracy of setup documentation.
* Effectiveness of version control implementation.
* Appropriateness of tools selected for the project requirements.
* Clarity of reflection on challenges and solutions encountered.
* Adherence to submission guidelines and deadlines.

Note: Feel free to reach out for clarification or assistance with any aspect of the assignment.