Developer Environment Setup

Introduction

This document outlines the steps I took to set up a developer environment suitable for software engineering projects. The environment includes the installation and configuration of a text editor, version control system, programming language, package manager, database, and optional development environments.

Operating System

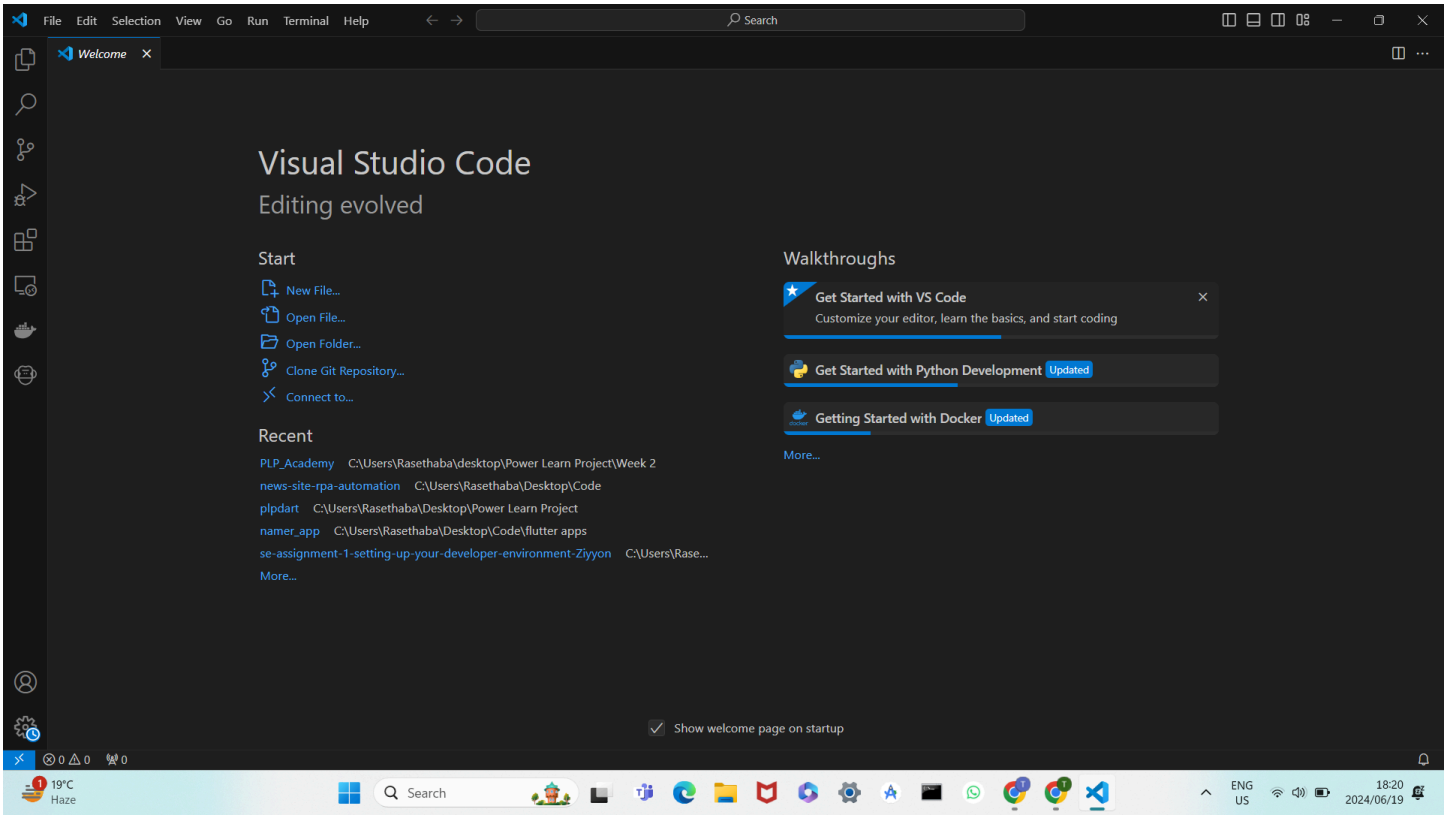
Windows 10

I am using Windows 10 for my development environment. I ensured that my system is updated with the latest updates and patches.

IDE Installation

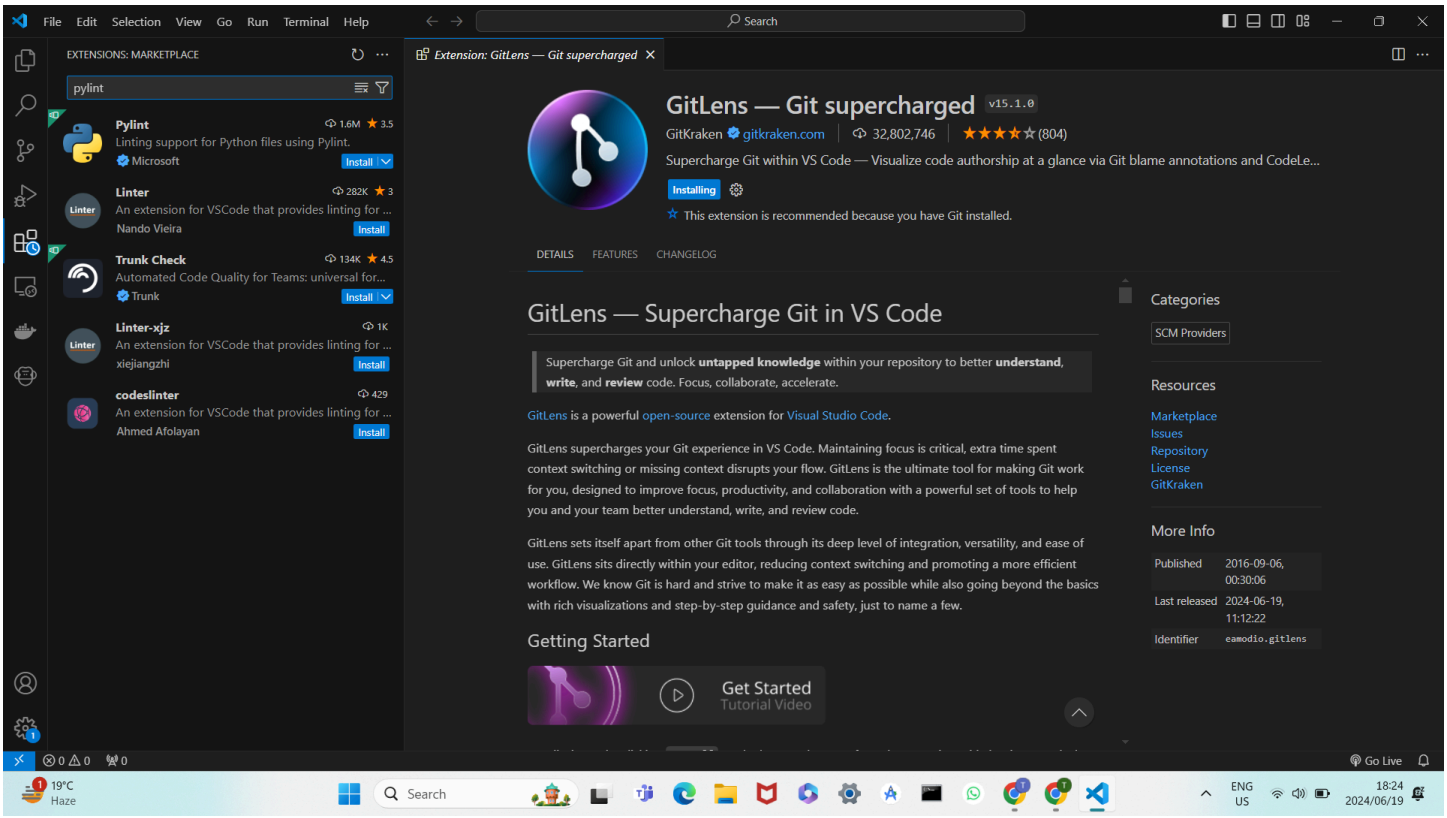
Download and Install Visual Studio Code

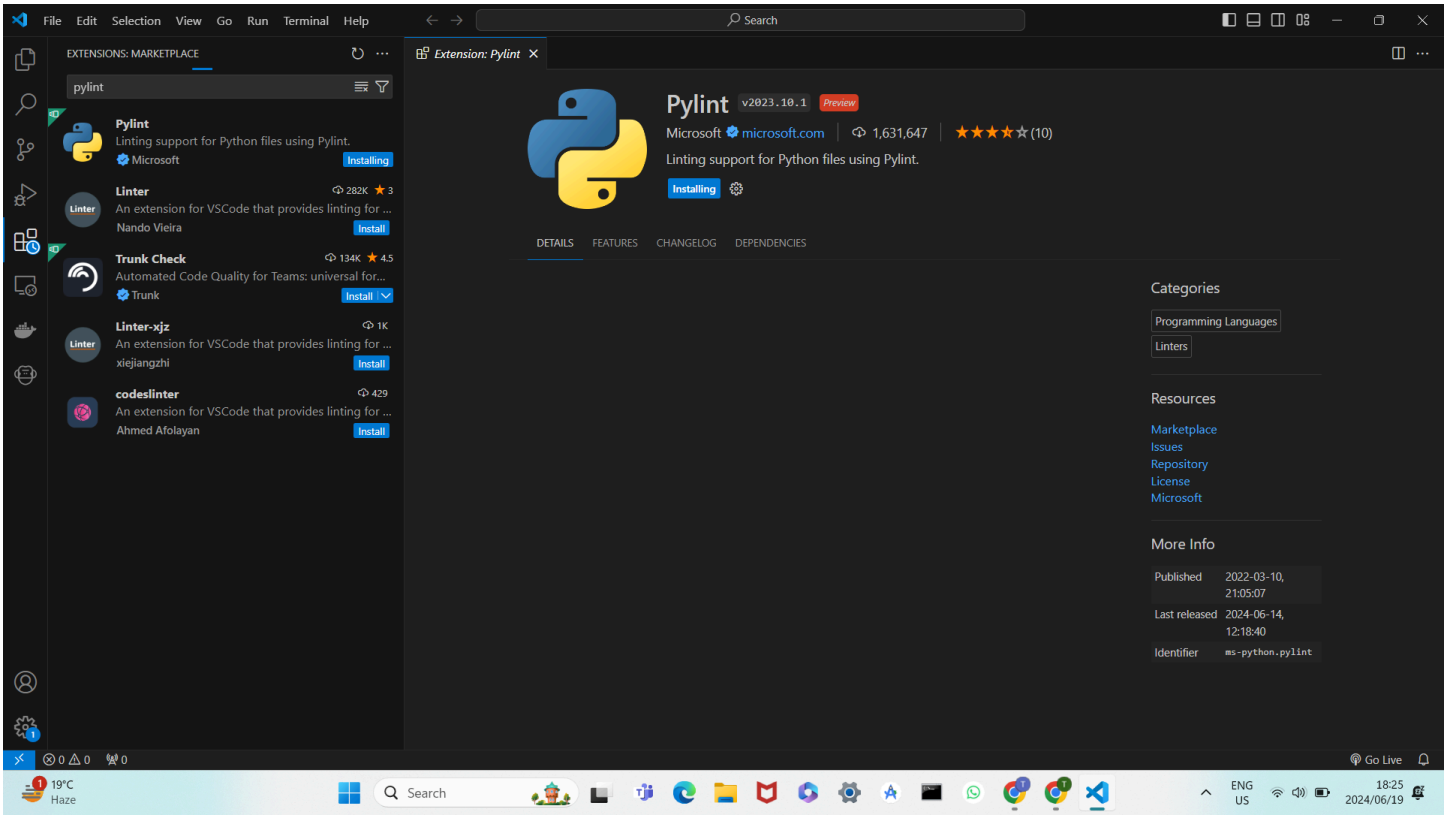
1. **I went to the [Visual Studio Code Download page](#).**
2. **I selected the appropriate version for Windows and clicked "Download".**
3. **I ran the downloaded installer and followed the installation instructions.**
4. **I launched Visual Studio Code after installation.**

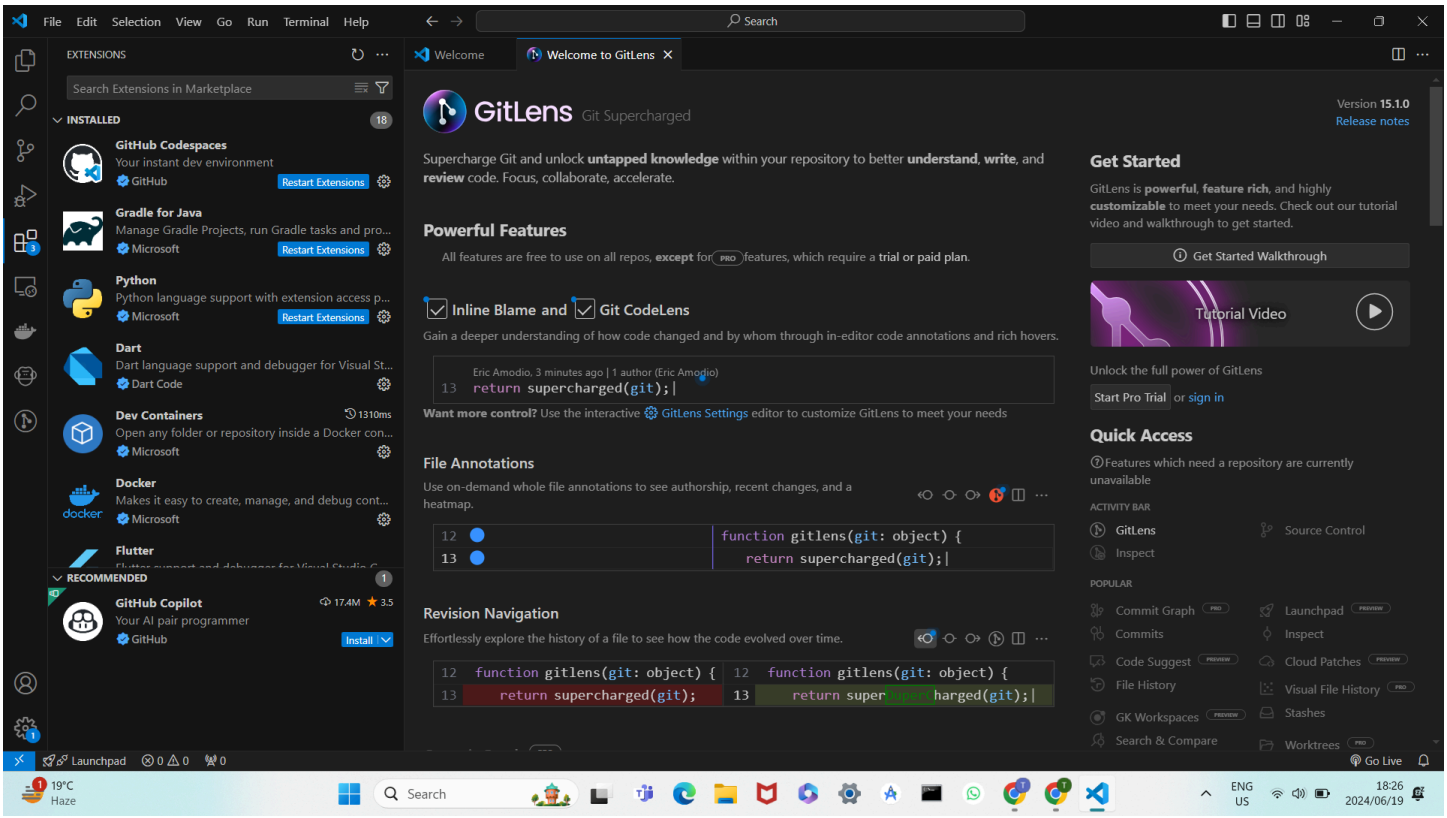


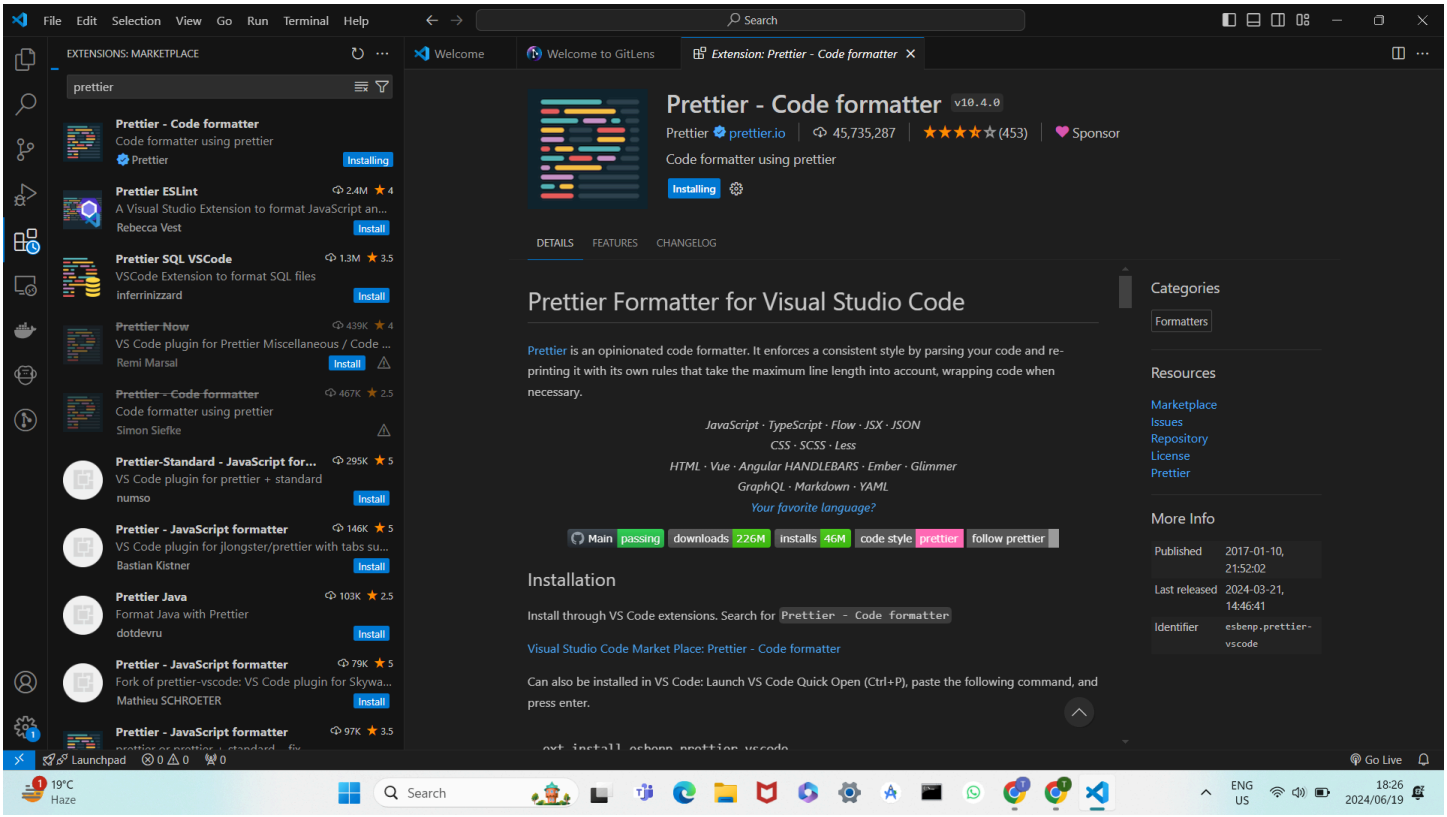
Install Extensions and Plugins

1. **I opened Visual Studio Code.**
2. **I clicked on the Extensions icon in the Activity Bar on the side of the window or pressed Ctrl+Shift+X.**
3. **I installed the following extensions:**
    - **Python: Provides support for Python programming.**
    - **GitLens: Enhances Git capabilities in VS Code.**
    - **Prettier: Code formatter.**
    - **Pylint: Linting for Python code.**




File Edit Selection View Go Run Terminal Help
Search
EXTENSIONS: MARKETPLACE
pylint
Pylint
Linting support for Python files using Pylint.
Microsoft
Installing
Linter
282K 3
An extension for VSCode that provides linting for ...
Nando Vieira
Install
Trunk Check
134K 4.5
Automated Code Quality for Teams: universal for...
Trunk
Install
Linter-xjz
1K
An extension for VSCode that provides linting for ...
xiejiangzhi
Install
codeslinter
429
An extension for VSCode that provides linting for ...
Ahmed Afolayan
Install
Extension: Pylint
Pylint
v2023.10.1
Preview
Microsoft microsoft.com
1,631,647
(10)
Linting support for Python files using Pylint.
Installing
DETAILS
FEATURES
CHANGELOG
DEPENDENCIES
Categories
Programming Languages
Linters
Resources
Marketplace
Issues
Repository
License
Microsoft
More Info
Published
2022-03-10, 21:05:07
Last released
2024-06-14, 12:18:40
Identifier
ms-python.pylint
Go Live
19°C
Haze
Search
ENG
US
18:25
2024/06/19


File Edit Selection View Go Run Terminal Help
Search
EXTENSIONS
Search Extensions in Marketplace
INSTALLED
18
GitHub Codespaces
Your instant dev environment
GitHub
Restart Extensions
Gradle for Java
Manage Gradle Projects, run Gradle tasks and pro...
Microsoft
Restart Extensions
Python
Python language support with extension access p...
Microsoft
Restart Extensions
Dart
Dart language support and debugger for Visual St...
Dart Code
Dev Containers
1310ms
Open any folder or repository inside a Docker con...
Microsoft
Docker
Makes it easy to create, manage, and debug cont...
Microsoft
Flutter
RECOMMENDED
1
GitHub Copilot
17.4M 3.5
Your AI pair programmer
GitHub
Install
Welcome
Welcome to GitLens
GitLens
Git Supercharged
Version 15.1.0
Release notes
Supercharge Git and unlock untapped knowledge within your repository to better understand, write, and review code. Focus, collaborate, accelerate.
Powerful Features
All features are free to use on all repos, except for PRO features, which require a trial or paid plan.
Inline Blame and Git CodeLens
Gain a deeper understanding of how code changed and by whom through in-editor code annotations and rich hovers.
Eric Amodio, 3 minutes ago | 1 author (Eric Amodio)
13 return supercharged(git);
Want more control? Use the interactive GitLens Settings editor to customize GitLens to meet your needs
File Annotations
Use on-demand whole file annotations to see authorship, recent changes, and a heatmap.
12 function gitlens(git: object) {
13 return supercharged(git);
Revision Navigation
Effortlessly explore the history of a file to see how the code evolved over time.
12 function gitlens(git: object) {
13 return supercharged(git);
Get Started
GitLens is powerful, feature rich, and highly customizable to meet your needs. Check out our tutorial video and walkthrough to get started.
Get Started Walkthrough
Tutorial Video
Unlock the full power of GitLens
Start Pro Trial or sign in
Quick Access
Features which need a repository are currently unavailable
ACTIVITY BAR
GitLens
Source Control
Inspect
POPULAR
Commit Graph PRO
Launchpad PREVIEW
Commits
Inspect
Code Suggest PREVIEW
Cloud Patches PREVIEW
File History
Visual File History PRO
GK Workspaces PREVIEW
Stashes
Search & Compare
Worktrees PRO
Launchpad
Go Live
19°C
Haze
Search
ENG
US
18:26
2024/06/19



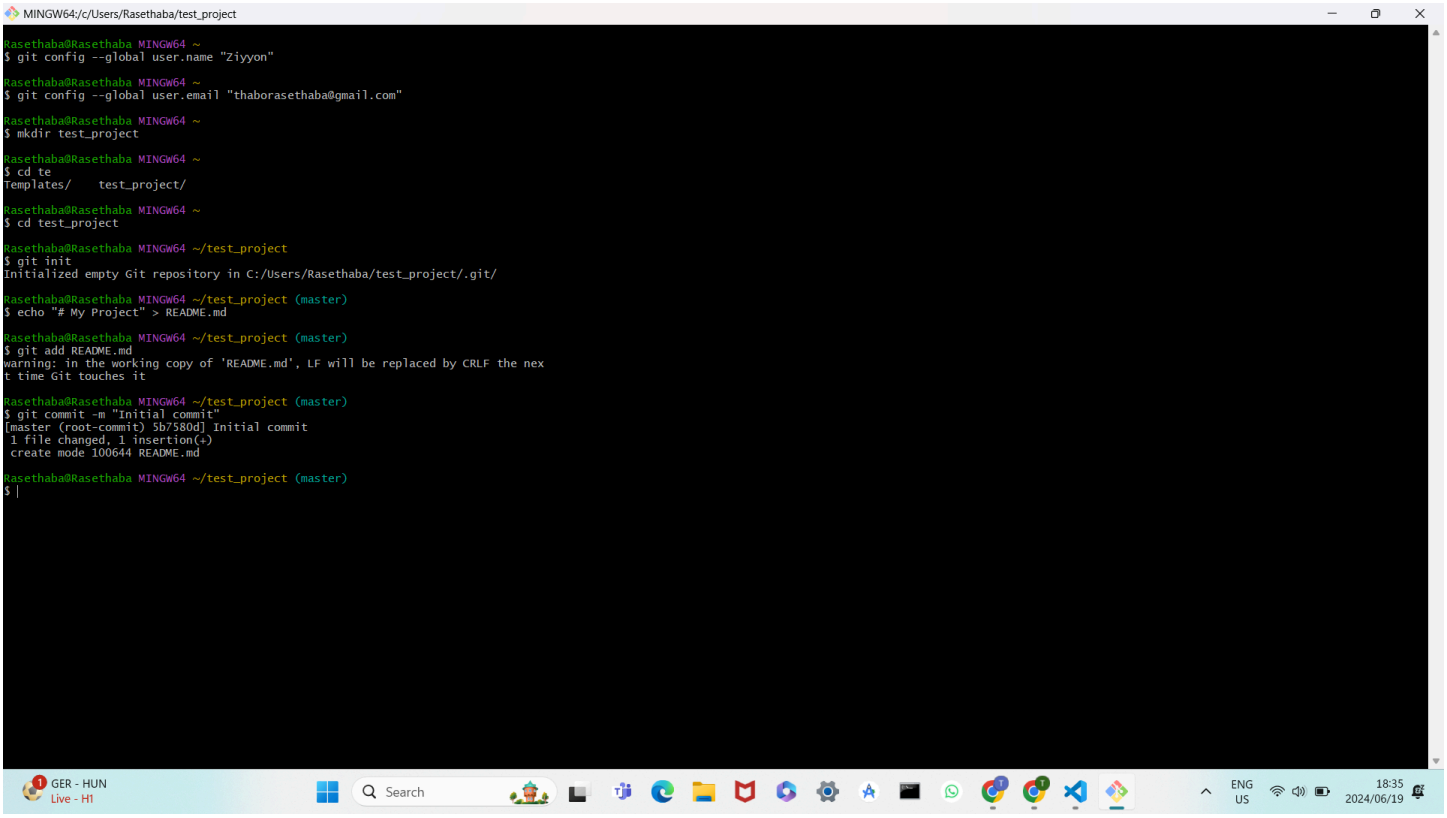
Version Control Setup

Install and Configure Git

1. I visited the [Git Download page] and downloaded the installer for Windows.
2. I ran the installer and followed the installation instructions.
3. I opened Git Bash and configured my Git username and email:
4. bash
5. Copy code
6. git config --global user.name "My Name"
7. git config --global user.email "my.email@example.com"

Initialize a Git Repository

1. I created a new directory for my project:
2. mkdir my_project
3. cd my_project
4. I initialized Git in the directory:
5. git init
6. I created a new file, e.g., README.md, and made my first commit:
7. echo "# My Project" > README.md
8. git add README.md
9. git commit -m "Initial commit"



Programming Languages and Runtimes Installation

Install Python

1. I visited the Python Download page and downloaded the latest version for Windows.
2. I ran the installer and ensured to check the option to add Python to my PATH during installation.
3. I verified the installation by opening Command Prompt and checking the Python version:
4. python --version

Package Managers Installation

Verify pip Installation

1. Pip is usually installed with Python. I verified by running:
2. pip --version

Database Installation

Download and Install MySQL

1. I visited the MySQL Download page and downloaded the MySQL Installer.
2. I ran the installer and followed the instructions to install MySQL.
3. I configured MySQL by setting up a root password during installation.

Development Environments and Virtualization (Optional)

Install Docker (Optional)

1. **I went to the Docker Download page and downloaded Docker Desktop for Windows.**
2. **I ran the installer and followed the installation instructions.**

Troubleshooting and Challenges

Challenges Faced

1. **Git Configuration:**
   - **Issue: Git configuration commands were not recognized.**
   - **Solution: I verified that Git was added to the system PATH and restarted Git Bash.**
2. **Python Installation:**
   - **Issue: Python was not recognized in Command Prompt.**
   - **Solution: I reinstalled Python and ensured the option to add Python to PATH was checked.**

Conclusion

The developer environment setup is now complete, providing a robust and productive workspace for coding, debugging, version control, and collaboration. All necessary tools and configurations have been successfully installed and verified.