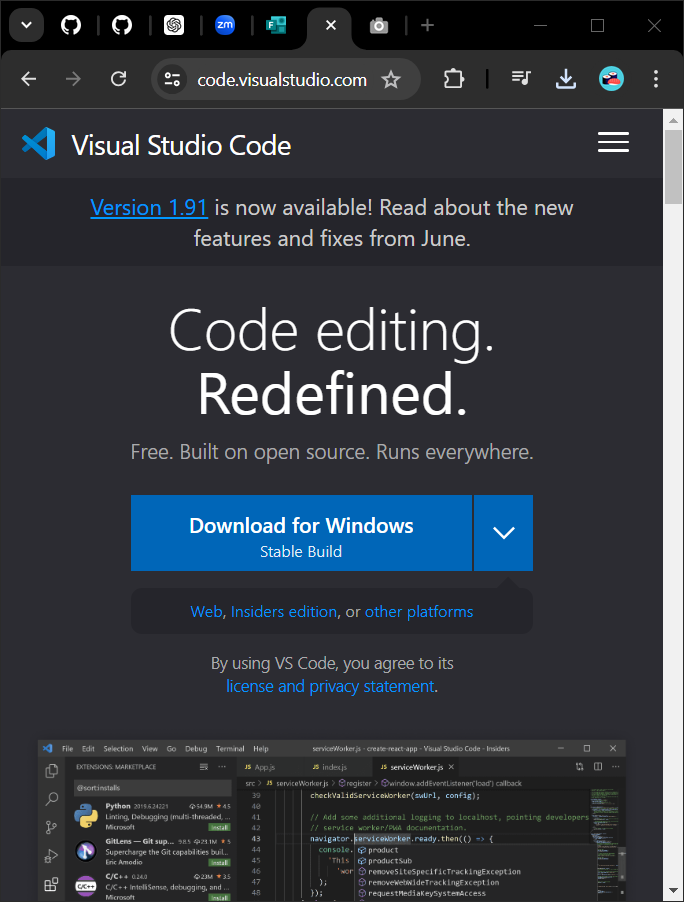
Software Engineering Assignment 5

# 1. Installation of VS Code

1. Visit the VS Code Website: Open your web browser and go to [<https://code.visualstudio.com/>



2. Download the Installer: Click on the "Download for Windows" button. This will download the VS Code installer for Windows.

3. Run the Installer: Once the download is complete, open the downloaded `.exe` file.

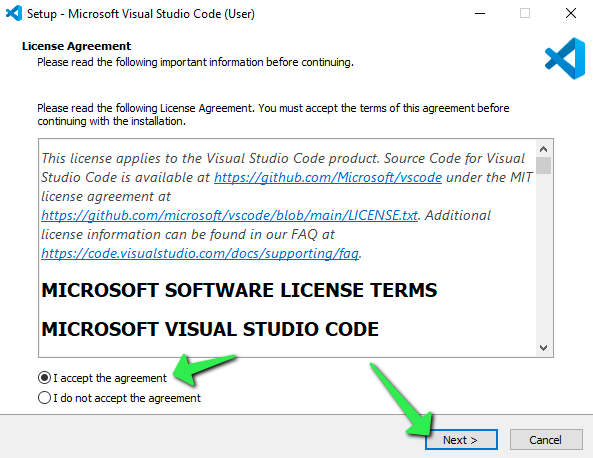
4. Follow the Installation Wizard:

- License Agreement: Read and accept the license agreement.

- Select Destination Location: Choose the installation location or leave it as the default.

- Select Additional Tasks: Check the boxes for additional tasks like creating a desktop icon and adding to PATH for easier command line access.

- Install: Click on the "Install" button.



5. Launch VS Code: Once the installation is complete, you can choose to launch VS Code immediately or find it in your Start menu.

**Prerequisites**

- Operating System: Windows 11

- Administrator Access: Required to install applications.

-Internet Connection: To get from the web.

-Space: Enough memory and storage for it to download and install

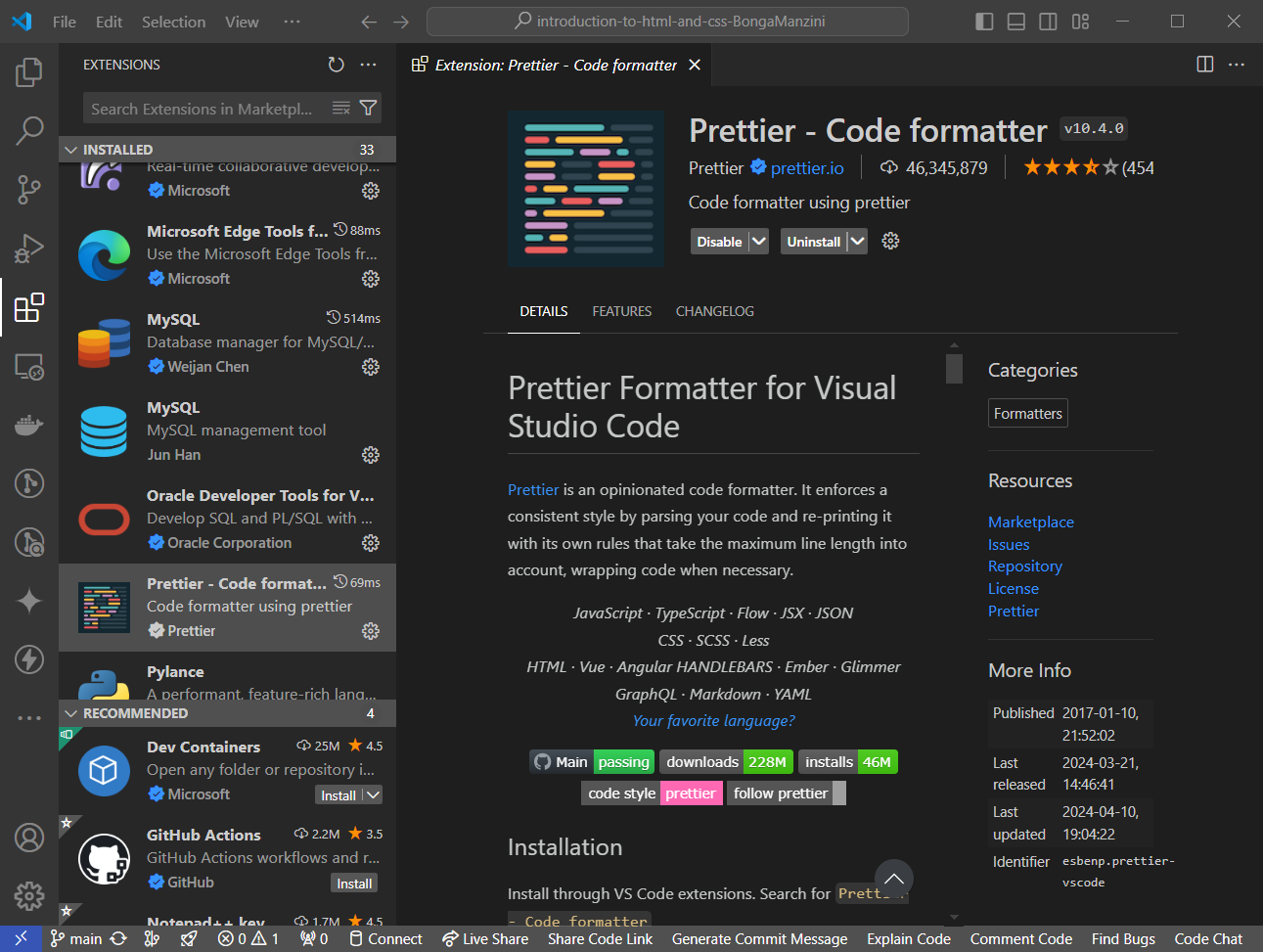
# 2. First-time Setup

After installing VS Code, you can optimize your coding environment with the following configurations and settings:

1. Update VS Code: Ensure you are running the latest version of VS Code by going to `Help` > `Check for Updates`.

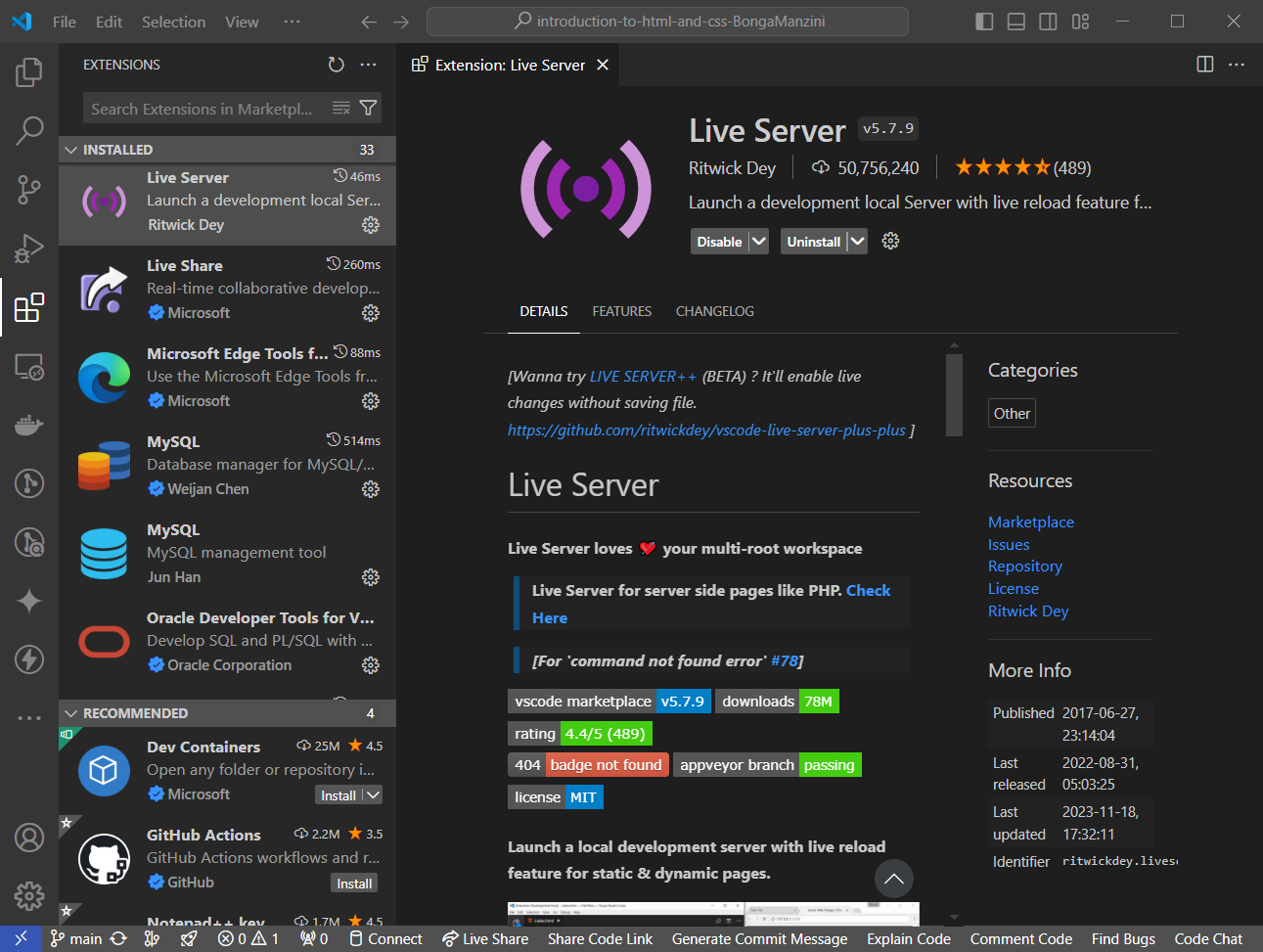
2. Install Essential Extensions:

- Prettier - Code formatter: For code formatting.

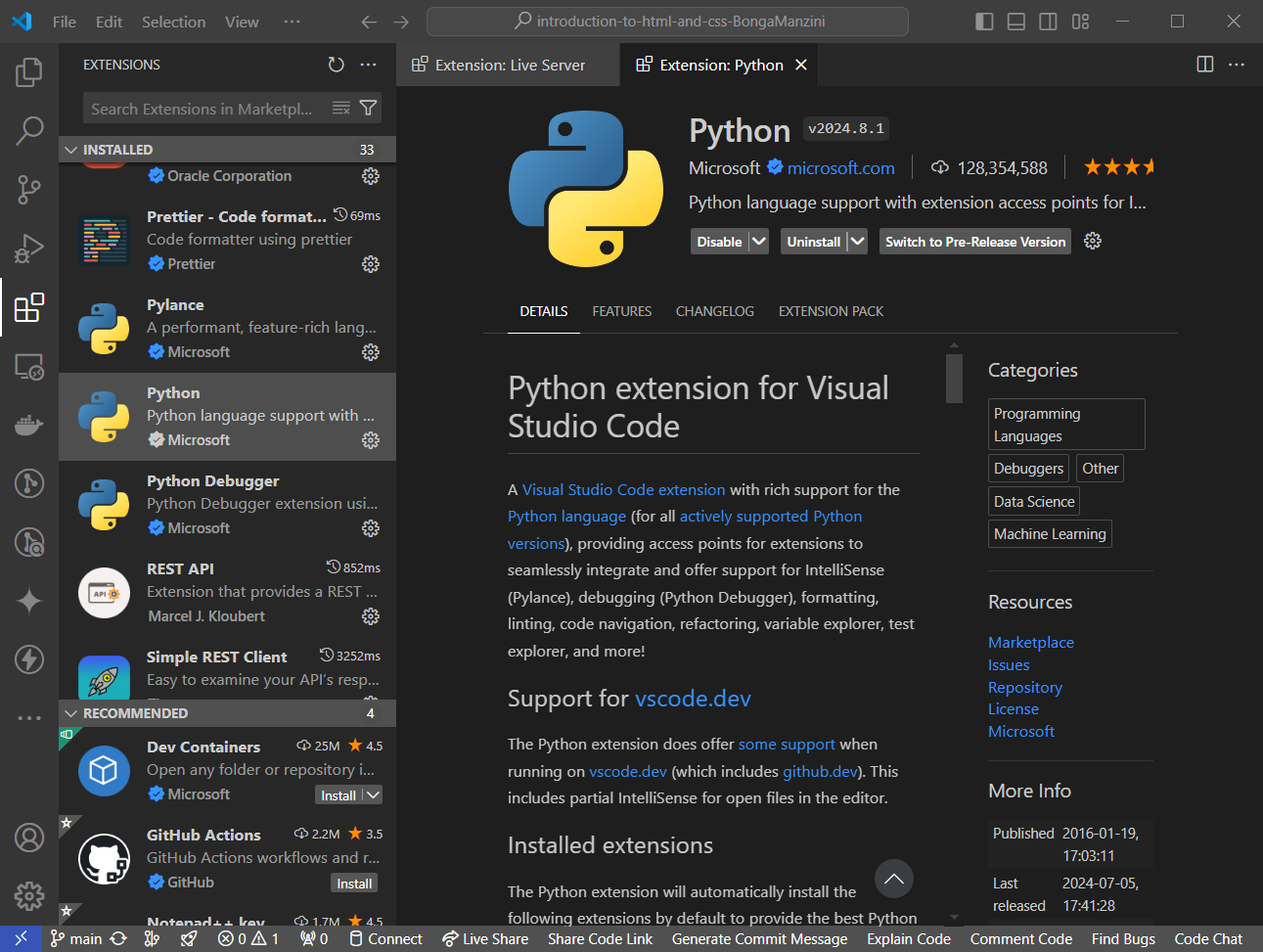


- ESLint: For JavaScript/TypeScript linting.

- Live Server: For a live-reloading local server.

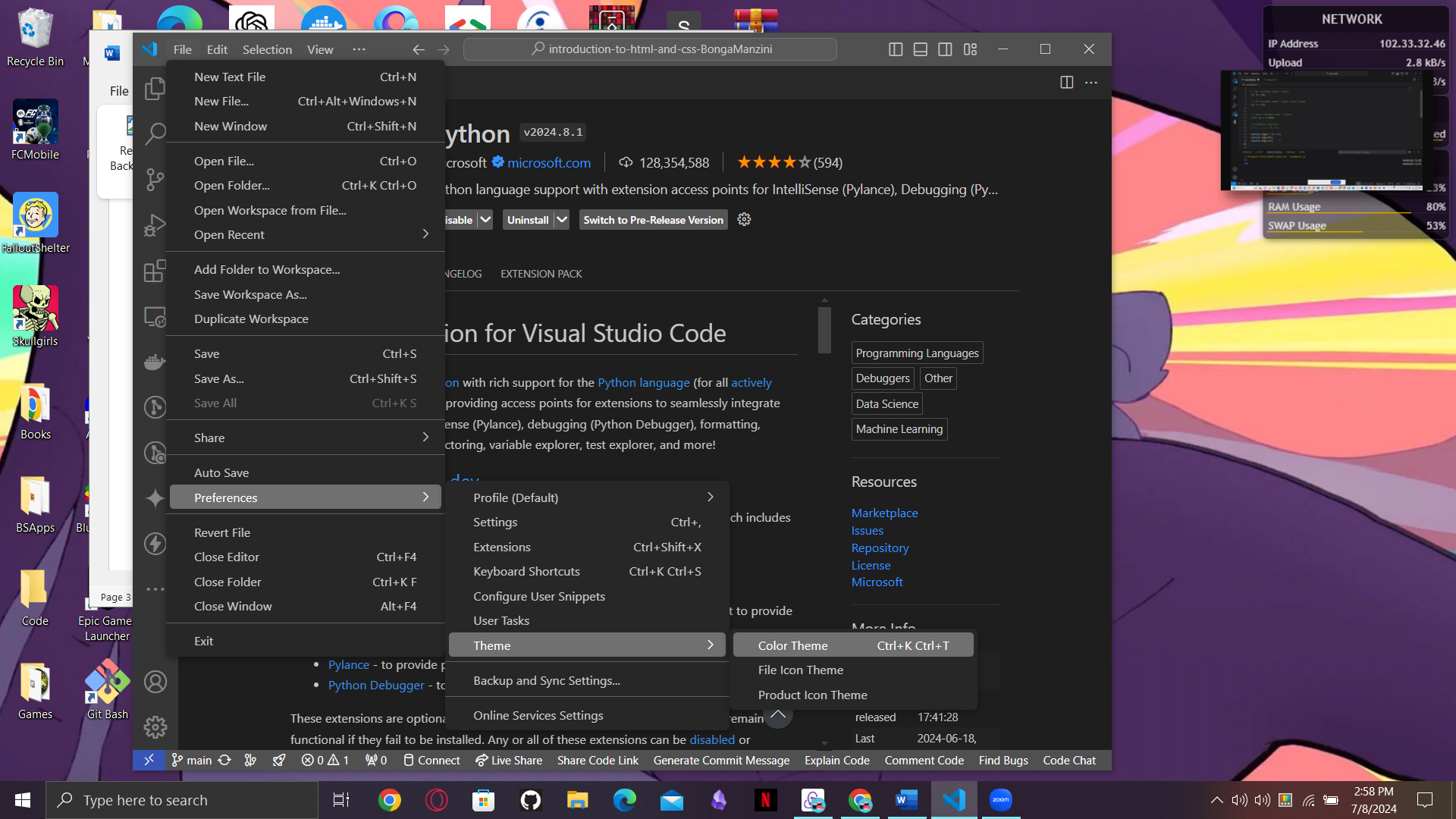


- Python: If you work with Python.



3. Theme and Appearance:

- Go to `File` > `Preferences` > `Color Theme` to select a theme.



- To install new themes, go to the Extensions view (`Ctrl+Shift+X`) and search for themes.

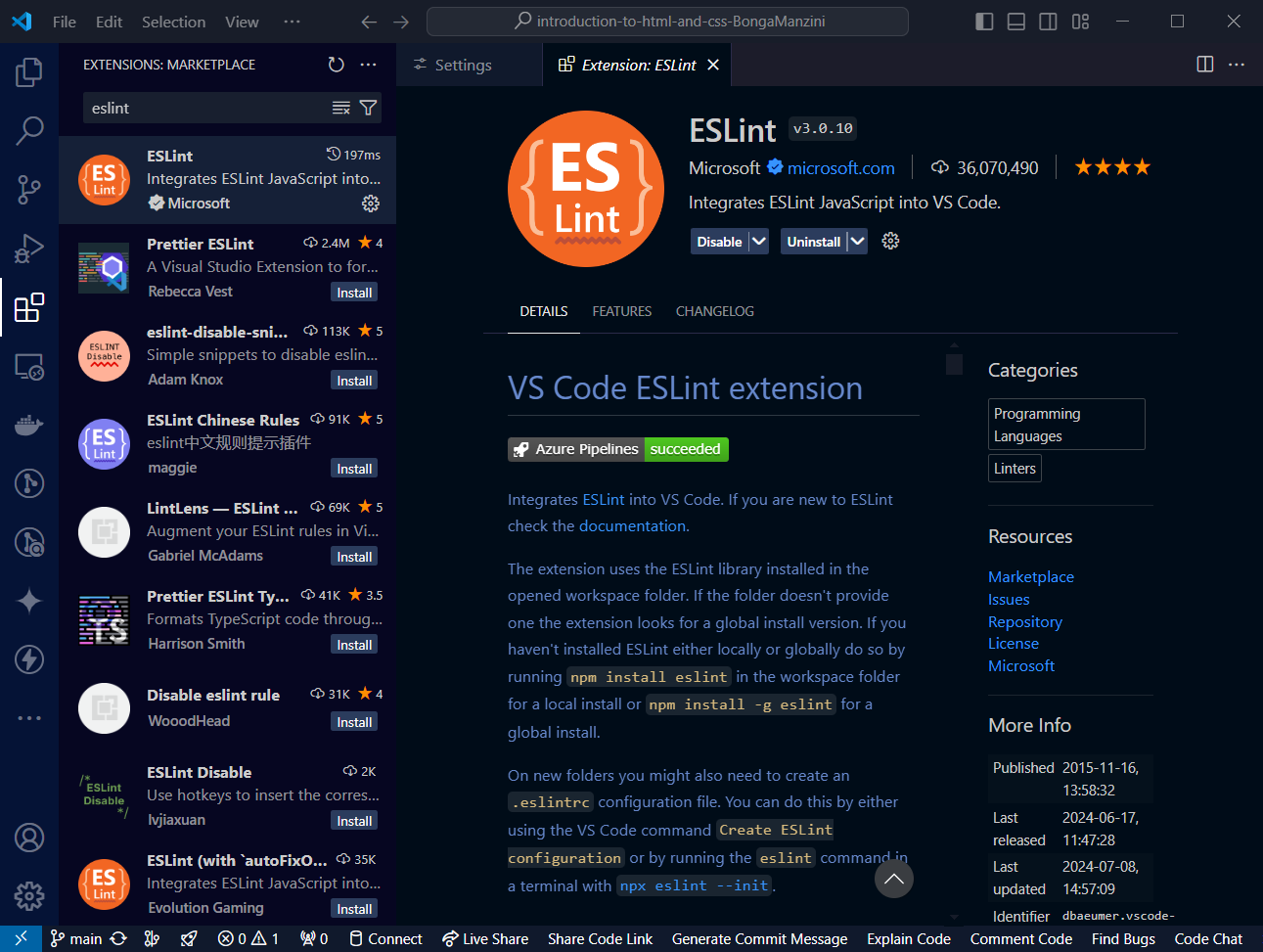
4. Editor Settings:

- Font Size: `File` > `Preferences` > `Settings`, search for "Font Size" and adjust as needed.

- Auto Save: Enable auto-saving files with `File` > `Auto Save`.

5. Configure Linting and Formatting:

- Set up ESLint and Prettier to ensure consistent code style.



- You can configure these in your project’s settings or create a `.eslintrc` and `.prettierrc` file in your project.

# 3. User Interface Overview

Main Components of the VS Code UI

1. Activity Bar: Located on the far left, it allows you to switch between views (Explorer, Search, Source Control, Run and Debug, Extensions).

2. Side Bar: Displays different views like the Explorer, Source Control, and Extensions. It's context-sensitive depending on the selected activity.

3. Editor Group: The central area where you open and edit files. You can open multiple files in tabs and split the editor into multiple groups for side-by-side editing.

4. Status Bar: At the bottom, it provides information about the current file and workspace, such as encoding, line endings, and language mode. It also shows warnings, errors, and version control status.

# 4. Command Palette

The Command Palette is a powerful tool in VS Code that allows you to access and execute commands.

- Accessing the Command Palette: Press **`Ctrl+Shift+P`** or **`F1`**.

- Common Tasks:

- Open files: `> Open File`

- Install extensions: `> Extensions: Install Extensions`

- Change settings: `> Preferences: Open Settings`

- View version control: `> Git: View History`

# 5. Extensions in VS Code

Extensions add functionality to VS Code.

- Finding Extensions: Go to the Extensions view by clicking the Extensions icon in the Activity Bar or pressing **`Ctrl+Shift+X`**.

- Installing Extensions: Search for an extension, click on it, and then click the "Install" button.

- Managing Extensions: You can enable, disable, or uninstall extensions from the Extensions view.

Essential Extensions for Web Development:

- HTML CSS Support

- JavaScript (ES6) code snippets

- Debugger for Chrome

- Path Intellisense

# 6. Integrated Terminal

Opening and Using the Integrated Terminal

- Open Terminal: `View` > `Terminal` or press `` Ctrl+` ``

- Advantages:

- Run commands directly within VS Code without switching context.

- Supports multiple terminals and various shells (cmd, PowerShell, bash).

# 7. File and Folder Management**.**

Creating, Opening, and Managing Files and Folders

- Creating Files/Folders: Right-click in the Explorer view (Side Bar) and select `New File` or `New Folder`.

- Opening Files/Folders: Double-click to open files. To open a folder, use `File` > `Open Folder`.

- Navigation: Use the `Go` menu or `Ctrl+P` to quickly navigate to a file.

# 8. Settings and Preferences

Customizing Settings

- Access Settings: `File` > `Preferences` > `Settings` or press **`Ctrl+,`**.

- Examples:

- Change Theme: `Color Theme` in settings.

- Font Size: Search for "Font Size" in settings.

- Keybindings: `File` > `Preferences` > `Keyboard Shortcuts`.

# 9. Debugging in VS Code

Setting Up and Starting Debugging

1. Open Debug View: Click the Debug icon in the Activity Bar or press **`Ctrl+Shift+D`**.

2. Configure Launch.json: Click the gear icon to create a `launch.json` file if it doesn't exist.

3. Set Breakpoints: Click in the gutter next to the line numbers.

4. Start Debugging: Press `**F5`** to start debugging.

5. Key Debugging Features:

- Watch Variables: Monitor variables.

- Call Stack: View the call stack.

- Breakpoints: Set conditional breakpoints.

# 10. Using Source Control

Integrating Git with VS Code

1. Initialize Repository: Open the command palette and type `> Git: Initialize Repository`.

***git init***

2. Making Commits:

***git add .***

***git commit -m "Your commit message"***

- Make changes to your files.

- Stage changes in the Source Control view.

- Write a commit message and click the checkmark to commit.

3. Pushing Changes to GitHub:

***git push -u origin main***

- Open the command palette and type `> Git: Push` to push changes.

- Follow prompts to authenticate with GitHub if needed.