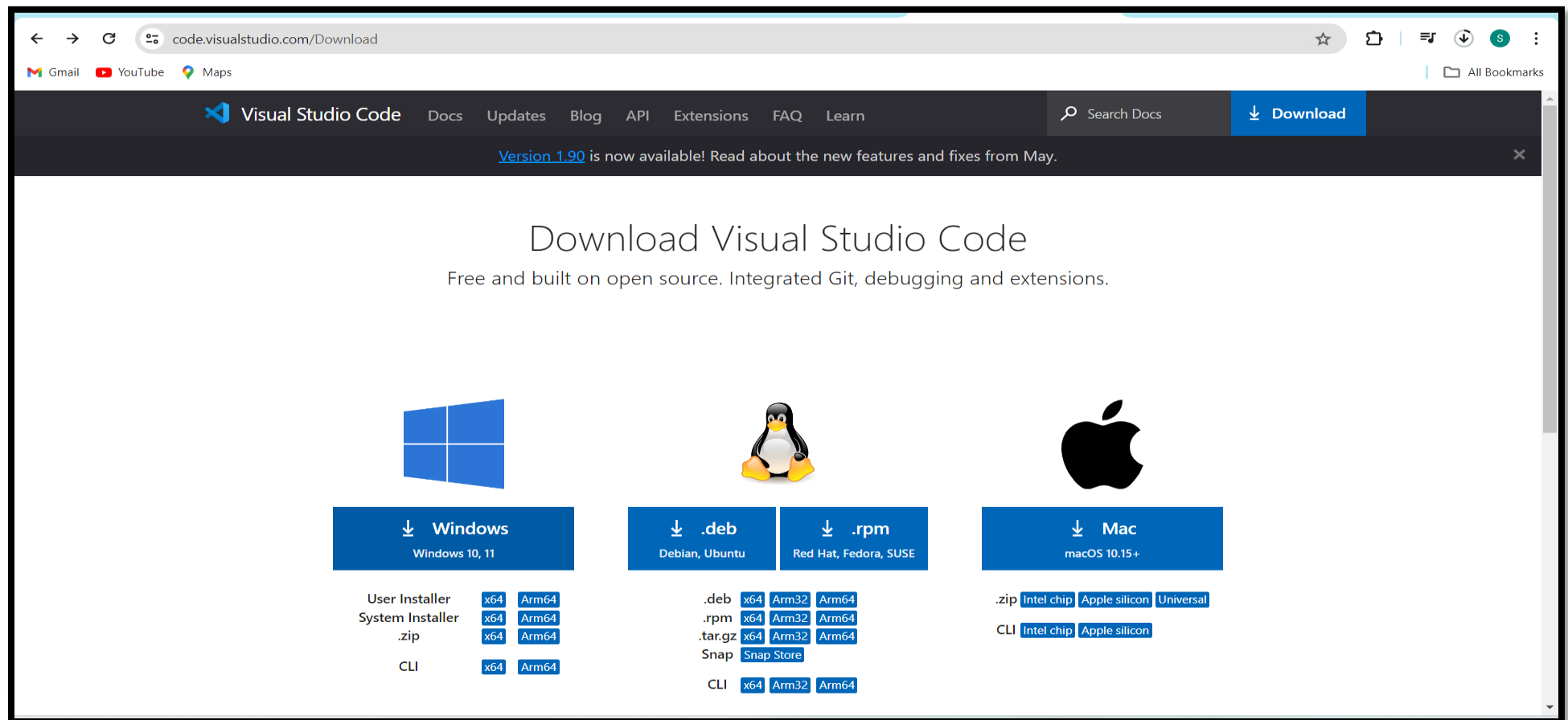
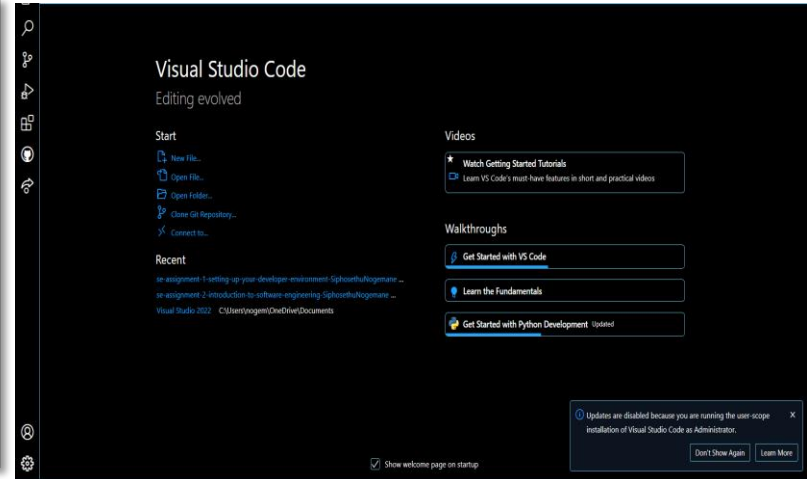
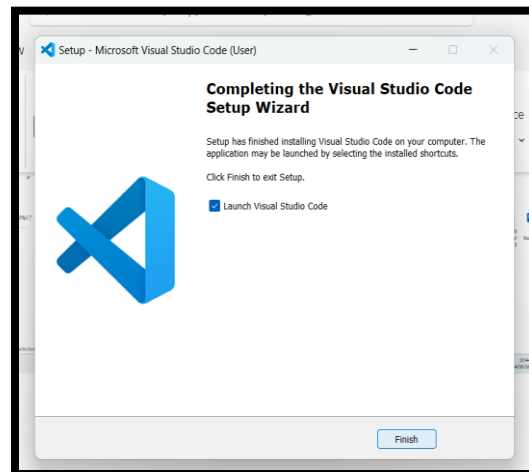
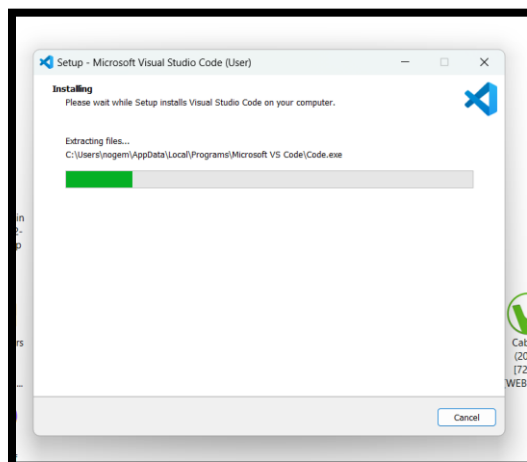
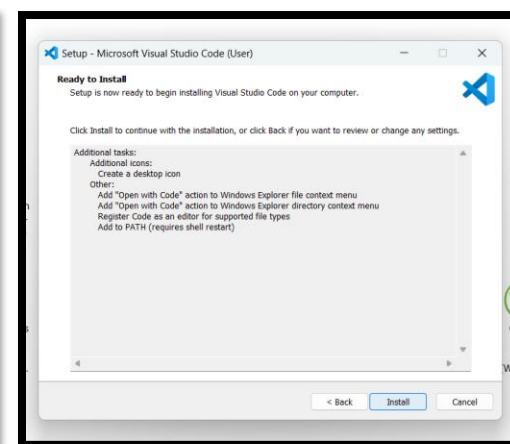
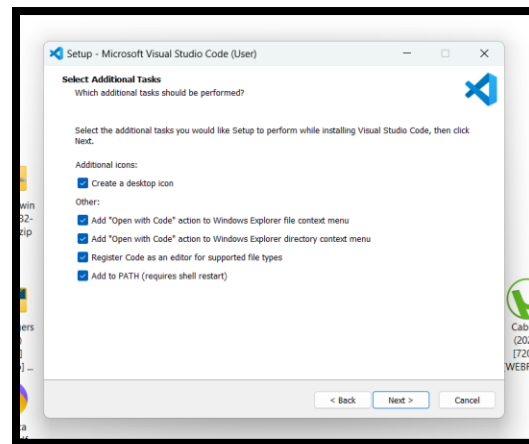
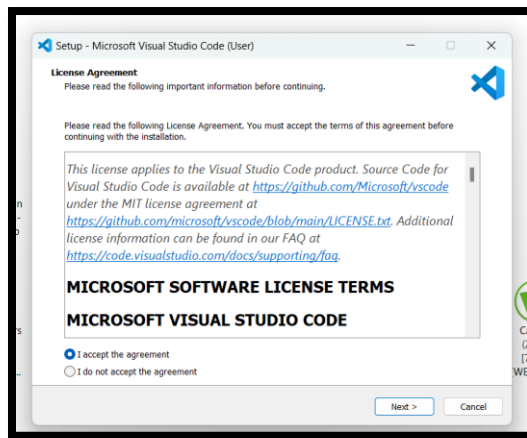


## 1. Installation of VS Code:

- **Visit the Official Website:** Open Google Chrome and go to the official Visual Studio Code website.
- **Download Visual Studio Code:** Click the “Download for Windows” button on the website to start downloading the Visual Studio Code setup file.
- **Locate the Downloaded File:** Once the download is complete, find the file in your "Downloads" folder. The file name will typically be something like VSCodeSetup.exe.



- **Run the Installer:** Double-click on the `VSCodeSetup.exe` file to start the installation process.
- **Accept the License Agreement:** In the setup wizard, read the license agreement. If you agree, check the box to accept the agreement and click "Next".
- **Select Additional Tasks:** In the setup options:
  - Check the box for "Create a desktop icon".
  - Check the box for "Add 'Open with Code' action to Windows Explorer file context menu".
  - Check the box for "Add 'Open with Code' action to Windows Explorer directory context menu".
  - Check the box for "Register Code as an editor for supported file types".
  - Check the box for "Add to PATH (requires shell restart)".
- **Install the Application:** Click "Next" to proceed. The installation process will begin.
- **Finish Installation:** Once the installation is complete, click on "Finish". Visual Studio Code will launch automatically.



## 2. First-time Setup

You begin by choosing a theme that matches your visual preferences from **'File > Preferences > Color Theme'**. This option can have a major impact on readability and comfort during long coding sessions. Next, use **'View > Extensions'** or the shortcut **'Ctrl+Shift+X'** to discover and install necessary extensions for your individual development requirements. These expansions can include language support, debugging tools, and productivity boosts. To adjust the font size; go to **'File > Preferences > Settings'** and search for **"editor.font"** to select a comfortable size for coding and viewing code. Streamline your workflow and conform to your preferred coding habits by personalising keyboard shortcuts using **'File >**

**Preferences > Keyboard Shortcuts’.**

### **3. User Interface Overview:**

**Activity Bar:** Allows easy access to several views such as Explorer, Search, Source Control, and Extensions.

**Side Bar:** Displays views such as Explorer (file management), Search, Source Control (Git), and Extensions.

**Editor Group:** Where files are opened for editing.

**Status Bar:** Displays information about the current file, the Git branch state, and other notifications.

### **4. Command Palette:**

One of the key components of Visual Studio Code for carrying out operations and rapidly completing tasks inside the editor is the Command Palette. It helps users swiftly navigate and manipulate many features of their coding environment. It may be accessed by pressing {Ctrl+Shift+P}. For example, you can use "Open File" and the filename to open a specific file, or you can use "Switch Editor" to go between editor tabs. These commands improve workflow by reducing the need for manual menu navigation and sophisticated keyboard shortcut memorisation, making routine activities in VS Code more accessible and straightforward.

### **5. Extensions in VS Code:**

Visual Studio Code (VS Code) extensions are essential for expanding its capabilities, meeting particular development needs, and increasing efficiency. Through the VS Code Marketplace, users can search for and install a variety of extensions by utilising the Extensions view ({Ctrl+Shift+X}). These extensions offer features like additional editing capabilities, debugging tools, and language compatibility, and they integrate seamlessly into the editor once installed. Extension management is simple for users, who may disable, uninstall, or update extensions as needed. Essential extensions for web development include Live Server, which provides instantaneous live previews of web pages, HTML CSS Support, which enhances HTML/CSS autocompletion, and ESLint, which improves JavaScript code quality within VS Code. All of these extensions speed up development processes.

### **6. Integrated Terminal:**

Visual Studio Code integrates an effective, built-in terminal right into the editor environment, increasing developer efficiency. The integrated terminal, which can be accessed by using the keyboard shortcut {Ctrl+` (backtick)}, enables developers to run scripts, execute shell commands, and

use command-line tools without having to switch between separate terminal windows. By removing the need for context switching between apps, this smooth connection improves efficiency and streamlines workflow. Through the use of the integrated terminal, developers can run build scripts, use Git commands to manage version control, and debug applications—all inside the comfort of the VS Code interface. This feature is a useful tool for software development projects of various sizes and complexity since it not only saves time but also offers a unified environment for developing, testing, and debugging.

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

File and folder management is simplified in Visual Studio Code (VS Code) to facilitate productive project organisation and development workflows. You can quickly create new files or folders by using the keyboard keys Ctrl+N for a new file and Ctrl+Shift+N for a new folder, or by right-clicking in the Explorer window. Using Ctrl+P to rapidly locate and open files by name, or double-clicking on files in the Explorer window, are two simple ways to open files. Using the Explorer view to explore directories for more extensive file management chores and Ctrl+Tab to switch between open files make navigating between files within the editor easier.

## **8. Settings and Preferences:**

Tailoring VS Code's settings and preferences to individual preferences and project requirements improves the coding environment. Navigate to File > Preferences > Settings to access settings. From there, users may change a number of things, like the font size (by searching for "editor font size"), color theme (by searching for "workbench color theme"), and keyboard shortcuts (by selecting "keyboard shortcuts"). With the aid of these adjustments, the editor's functionality and user interface can be tailored to maximise comfort and productivity during coding sessions.

## **9. Debugging in VS Code:**

When it comes to debugging, Visual Studio Code offers powerful capabilities for easily setting up and carrying out debugging sessions. Open your project folder in VS Code after installing any required debuggers or extensions for your programming language. You can use the editor margin to set breakpoints, Run > Add Configuration to define debugging launch configurations, and F5 or Run > Start Debugging to initiate debugging. During development, debugging capabilities like call stack navigation, variable inspection for real-time data analysis, step-through execution to trace code flow, and more can help find and fix problems more quickly.

## 10. Using Source Control:

When it comes to debugging, Visual Studio Code offers robust features for easily setting up and carrying out debugging sessions. Open your project folder in VS Code after installing any required debuggers or extensions for your programming language. You can use the editor margin to set breakpoints, Run > Add Configuration to define debugging launch configurations, and F5 or Run > Start Debugging to initiate debugging. During development, debugging capabilities like call stack navigation, variable inspection for real-time data analysis, step-through execution to trace code flow, and more can help find and fix problems more quickly.

## References

<https://docs.github.com/en/codespaces/developing-in-a-codespace/developing-in-a-codespace>

<https://developer.salesforce.com/tools/vscode/en/sf-dev-env/vscode-commands>

<https://code.visualstudio.com/docs/editor/extension-marketplace>