1 **Installation of VS Code**

To download and install Visual Studio Code (VS Code) on a Windows 11 operating system, follow these steps:

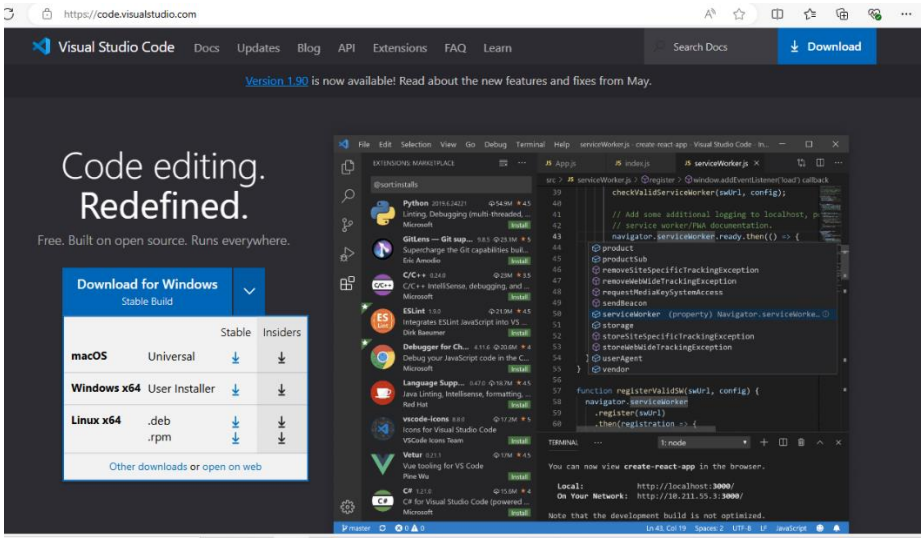
Prerequisites

- Ensure you have administrator access on your Windows 11 PC.
- A stable internet connection is required to download the installation files.

Steps to Download and Install Visual Studio Code

Download Visual Studio Code

- Open your web browser and go to the Visual Studio Code website.
- Click on the "Download for Windows" button. This will download the installer.



1. Run the Installer:

- Once the download is complete, navigate to your Downloads folder or the location where the installer was saved.
- Double-click on the VSCodeSetup-x.x.x.exe file to run the installer.

2. Begin Installation:

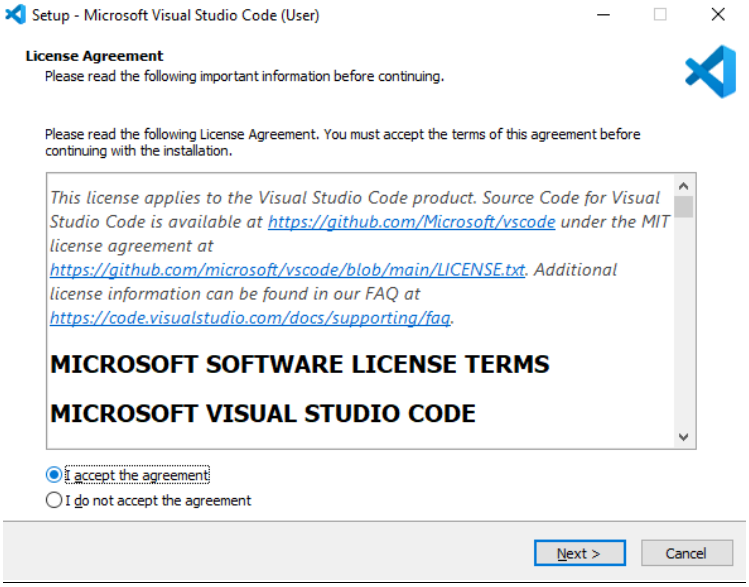
- A User Account Control prompt may appear asking for permission to allow the installer to make changes to your device. Click "Yes" to continue.

3. Setup Wizard:

- The Visual Studio Code Setup wizard will open. Click "Next" to proceed with the installation.

4. License Agreement:

- Read the license agreement and click "I accept the agreement" if you agree to the terms. Then, click "Next".



5. Select Destination Location:

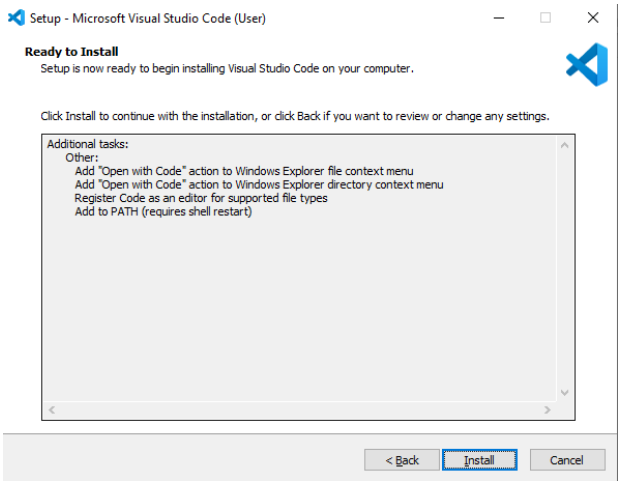
- Choose the installation location or leave it as the default. Click "Next".

6. Select Additional Tasks:

- You can select additional tasks such as creating a desktop icon or adding VS Code to the PATH. It's recommended to select "Add to PATH" to make it easier to run VS Code from the command line. Click "Next".

7. Ready to Install:

- Review your selections and click "Install" to begin the installation.



8. Complete the Installation:

- The installation process will take a few moments. Once it's complete, you can choose to launch Visual Studio Code immediately by checking the "Launch Visual Studio Code" checkbox. Click "Finish".

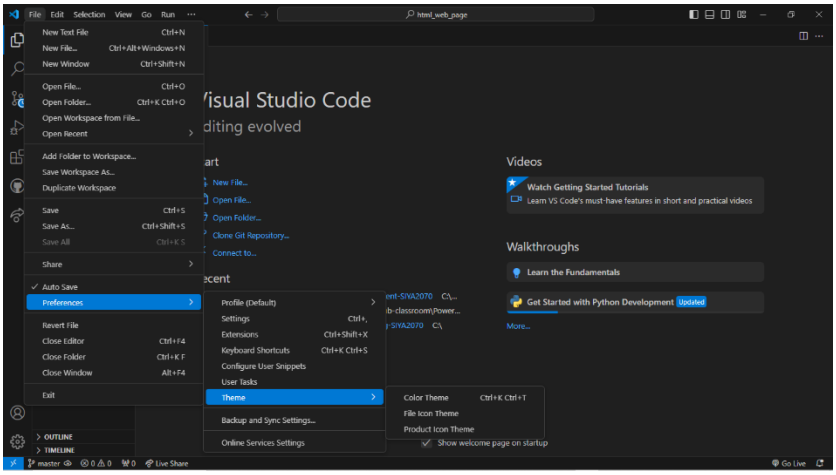
2 First time Setup

After installing Visual Studio Code (VS Code) you can configure it for an optimal coding environment

General Settings includes:

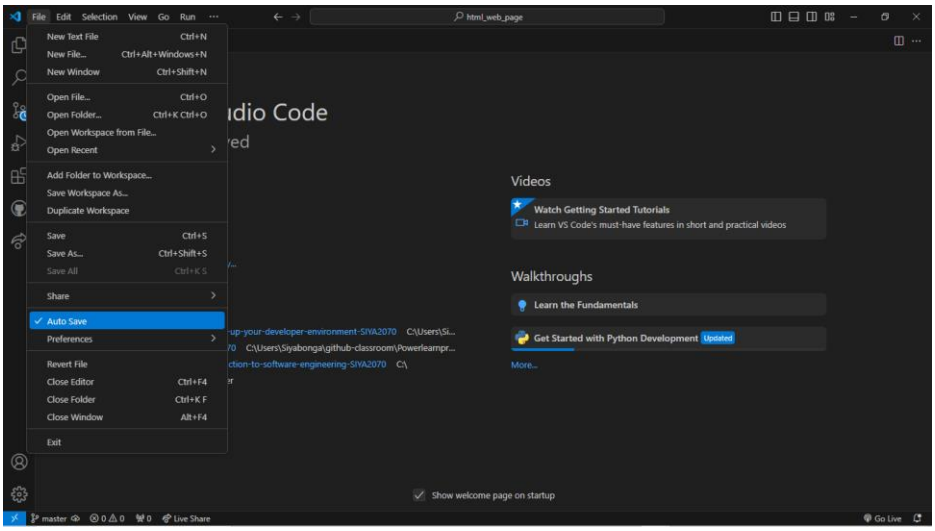
1. Theme:

- Go to File > Preferences > Color Theme and choose a theme that suits your preference. Popular themes include "Dark+", "Light+", and "Dracula".



2. Auto Save:

- Enable auto-save to avoid losing work.
  Go to File > Preferences > Settings, search for "auto save" make sure there is a tick next to auto sae



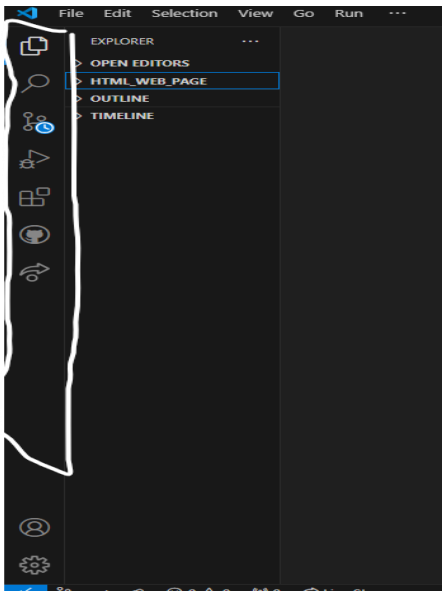
Extensions

1. Python:
    - Install from the Extensions view by searching for "Python". (the square icon with four squares inside)

3 User Interface Overview:

1. Activity Bar
It is on the left side of the window. It provides quick access to different views and features. The default icons include:

- Explorer, Search, Source Control, Run and Debug, Extensions, Additional Views.

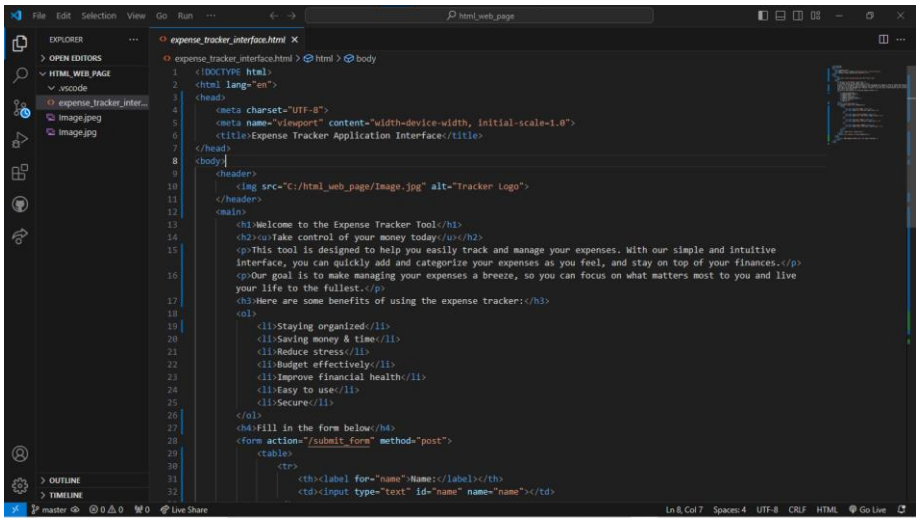


2. Side Bar

- To the right of the Activity Bar. It displays different panels depending on the selected view in the Activity Bar. Which may include**:**
  Explorer Panel, Search Panel, Source Control Panel, Debug Panel, Extensions Panel:

3 Editor Group

This is the main area of the interface, taking up the central part of the window. It is the primary area for editing code. Tabs, Split Editor, File Preview



4. Status Bar

The horizontal bar at the bottom of the window. It provides information about the current state of the editor and the active file. It can display:
Current Branch, Line and Column Numbers, Language Mode & Notifications

4 Command Palette

The Command Palette is a feature of VS Code that allows you to access all available commands based on your current context. You can use the Command Palette to run commands, open files, search for symbols, and more.

To access the Command Palette

- Press Ctrl+Shift+P.

Alternatively, you can open the Command Palette by clicking on the "View" menu at the top of the window and selecting "Command Palette...".

Common Tasks Using the Command Palette include but not limited to

1. Open a File:
   - Type >Open File and select a file from your workspace.
2. Toggle Sidebar Visibility:

- Type >View: Toggle Side Bar Visibility to show or hide the sidebar.

3. Open Terminal:
    - Type >Terminal: Create New Integrated Terminal to open a new terminal instance.
4. Change Theme:
    - Type >Preferences: Color Theme and choose a different theme for the editor.
5. Install Extensions:
    - Type >Extensions: Install Extensions to open the Extensions view and install new extensions.
6. Run Tasks:
    - Type >Tasks: Run Task to run a predefined task.
7. Git Commands:
    - Type >Git: Commit to commit changes.
    - Type >Git: Pull to pull changes from a remote repository.
    - Type >Git: Push to push changes to a remote repository.

5) Extensions in VS Code

Extensions in VS Code enhance its functionality and customize the development environment to suit various needs. They add features like language support, debuggers, themes, and more, making VS Code a versatile and powerful tool for developers.

Click the Extensions icon on the Activity Bar (the square icon with four squares inside) or press Ctrl+Shift+X. Type keywords in the search bar to find extensions related to your needs.

1. **Installing Extensions**:

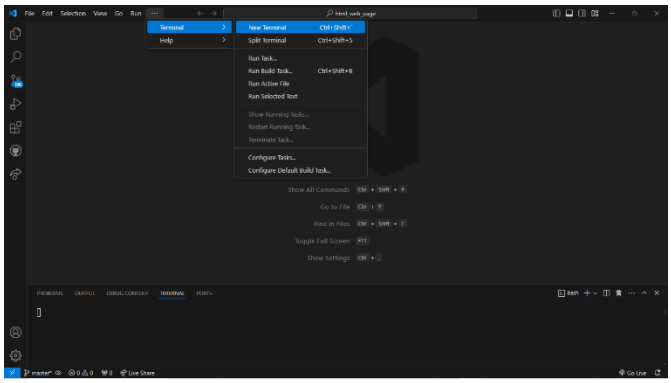
- **From Extensions View**: Once you find an extension, click the "Install" button.

6 Integrated Terminal

How to Open and Use the Integrated Terminal in VS Code

1. **Opening the Terminal**:

- Go to View > Terminal from the top menu.



2. **Using the Terminal**:

- **Running Commands**: You can type and execute commands just like in any other terminal.

Advantages

1. Convenience:

- Keep your development environment within a single application, reducing the need to switch contexts.
- Easily open and manage terminals with keyboard shortcuts and commands within VS Code.

2. Enhanced Integration:

    - **T**Use the terminal to run build tasks, scripts, and other automated processes defined in your project.
    - Run your debugging sessions directly from the terminal, keeping all outputs and interactions in one place.

3. Customization:

- Customize the appearance and behavior of the integrated terminal through VS Code settings (File > Preferences > Settings).
- Choose your preferred shell (e.g., PowerShell, Command Prompt, Git Bash) by setting it in the terminal configuration.

7 File and folder management:

Using the Explorer View:

- **Create a File**: In the Explorer view (accessible by clicking the first icon in the Activity Bar or pressing Ctrl+Shift+E), right-click on the folder where you want to create a new file and select "New File". Type the name of the file and press Enter.
- **Create a Folder**: Right-click in the Explorer view and select "New Folder". Type the name of the folder and press Enter.

Managing Files and Folders

1. **Rename**:

- Right-click on the file or folder in the Explorer view and select "Rename". Type the new name and press Enter.

2. **Move**:

- Drag and drop the file or folder to the desired location in the Explorer view.

3. **Delete**:

- Right-click on the file or folder and select "Delete". Confirm the deletion when prompted.

4. **Copy and Paste**:

- Right-click on the file or folder and select "Copy". Right-click in the desired location and select "Paste".

Navigating Between Different Files and Directories Efficiently

1. **Quick Open**:

- Press Ctrl+P to open the Quick Open dialog. Type the name of the file you want to open and select it from the list.

2. **Explorer View**:

- Use the Explorer view to navigate through your project files and folders by expanding and collapsing directories.

3. **File Tabs**:

- Open files are displayed in tabs at the top of the editor. Click on a tab to switch to that file. Use Ctrl+Tab (Windows/Linux) or Cmd+Tab (Mac) to switch between open files.

4. **Terminal**:

- Use the integrated terminal to navigate the file system with commands like cd, ls (Unix), or dir (Windows). This is especially useful for performing tasks not easily done through the GUI.

8 Settings and preference

Users can find and customize settings in VS Code through the Settings interface.

Accessing Settings

1. **Settings UI**:

- Go to File > Preferences > Settings
- Shortcut: Press Ctrl+

1. Changing the Theme

1. **Settings UI**:

- Open the Settings UI.
- In the search bar at the top, type "theme".
- Click on "Color Theme".
- Select a theme from the list that appears.

2. Changing the Font Size

1. **Settings UI**:

- Open the Settings UI.
- In the search bar at the top, type "font size".
- Locate the "Editor: Font Size" setting.
- Adjust the font size by entering a new value.

3. Changing Keybindings

1. **Keybindings UI**:

- Go to File > Preferences > Keyboard Shortcuts
- Shortcut: Press Ctrl+K Ctrl+S.

2. **Search and Modify Keybindings**:

- In the Keyboard Shortcuts editor, you can search for specific commands using the search bar at the top.

- Click on the pencil icon next to the command you want to change.
- Press the new key combination you want to assign to that command and press Enter.

9 Debugging in VS code

Using Python program as an example.

1. Open the folder containing your project files in VS Code (Ctrl+K Ctrl+O).

2. Create a Simple Program

3. Create a new Python file (e.g., my_project.py)

4. Set Up the Debug Configuration

4.1 Open the Debug View:

4.1.1 Click the Debug icon in the Activity Bar (left sidebar) or press Ctrl+Shift+D.

4.2 Create a Debug Configuration:

4.2 .1 Click on the gear icon to open the launch.json file.

4.2.2 If prompted to select an environment, choose Python.

5. Set Breakpoints

- Open your Python file (app.py).
- Click in the gutter next to the line numbers where you want to set breakpoints. A red dot will appear to indicate a breakpoint.

6. Start Debugging

- With the file open and breakpoints set, click the green play button in the Debug view or press F5 to start debugging.

7. Key Debugging Features in VS Code

1. Breakpoints:
2. Watch:
3. Call Stack:
4. Variables:
5. Step Controls:
6. Debug Console:
7. Integrated Terminal:

10 Using source code

1. Initializing a Git Repository

1. Open Your Project Folder:

- Open the folder containing your project files in VS Code (File > Open Folder or Ctrl+K Ctrl+O).

2. Initialize Git:

- You can open the integrated terminal (Ctrl+) and run the command: *"git init"*

3. Making Commits

- you can use the integrated terminal to commit changes and run the command:
  git commit -m "Your commit message"

4. Pushing Changes to GitHub

- you can use the integrated terminal to commit changes and run the command:
  git push -u origin master
  git push -u origin main

Reference

Documentation for Visual Studio Code
Visual Studio Code Tips and Tricks

Visual Studio Code User Interface

Visual Studio Code Themes

Visual Studio Code Key Bindings

Integrated Terminal in Visual Studio Code

Debugging configurations for Python apps in Visual Studio Code