1.Installation of Vs code

- Visit https://code.visualstudio.com/download

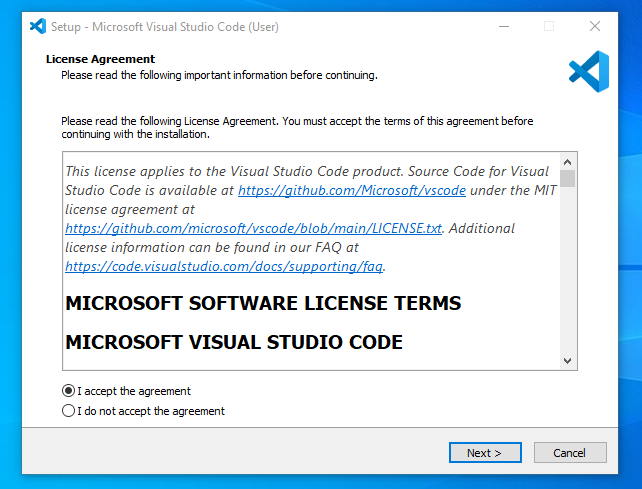
-Press the “Download for Windows” button on the website to start the download of the Visual Studio Code Application of your system type



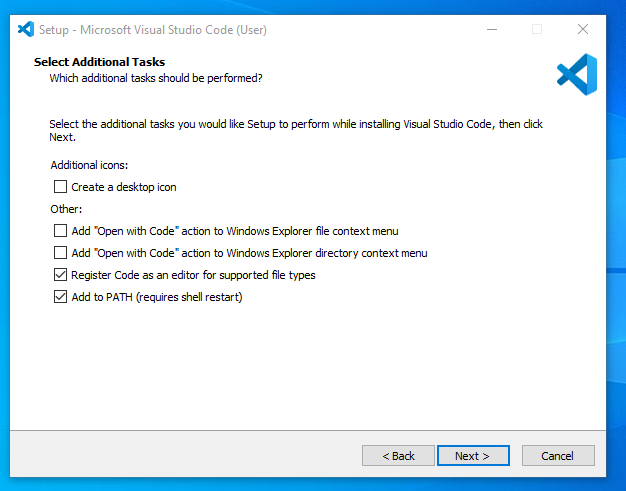
-When the download finishes, then the Visual Studio Code Icon setup file appears in the downloads folder in your desktop

- right click on it to run as administrator

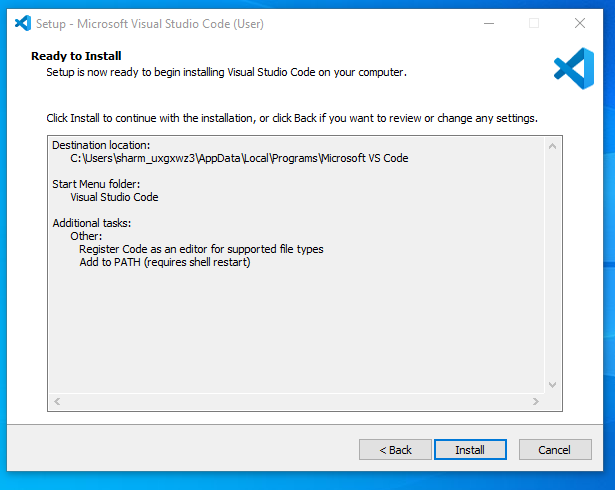
-After the Installer opens, it will ask you to accept the terms and conditions of the Visual Studio Code. Click on I accept the agreement and then click the Next button.



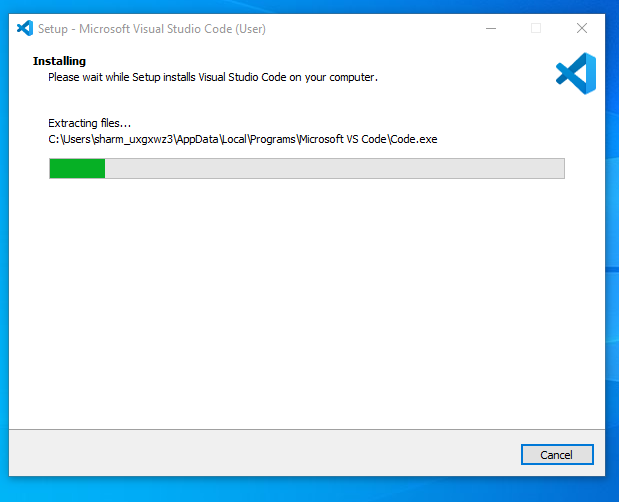
-Choose the location data for running the Visual Studio Code. It will then ask you to browse the location. Then click on the Next button



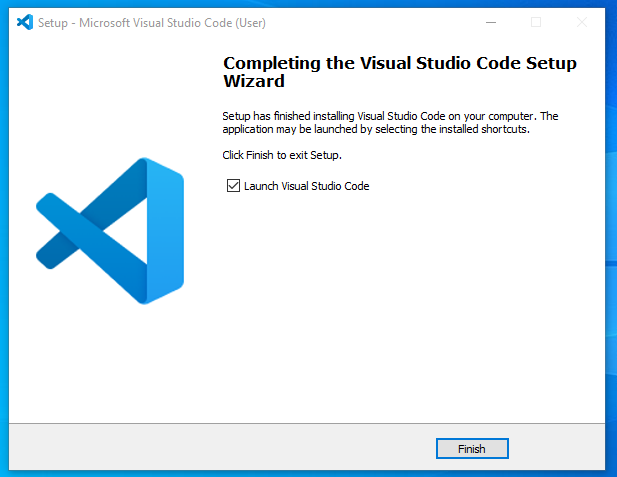
-Then it will ask to begin the installation setup. Click on the Install button.



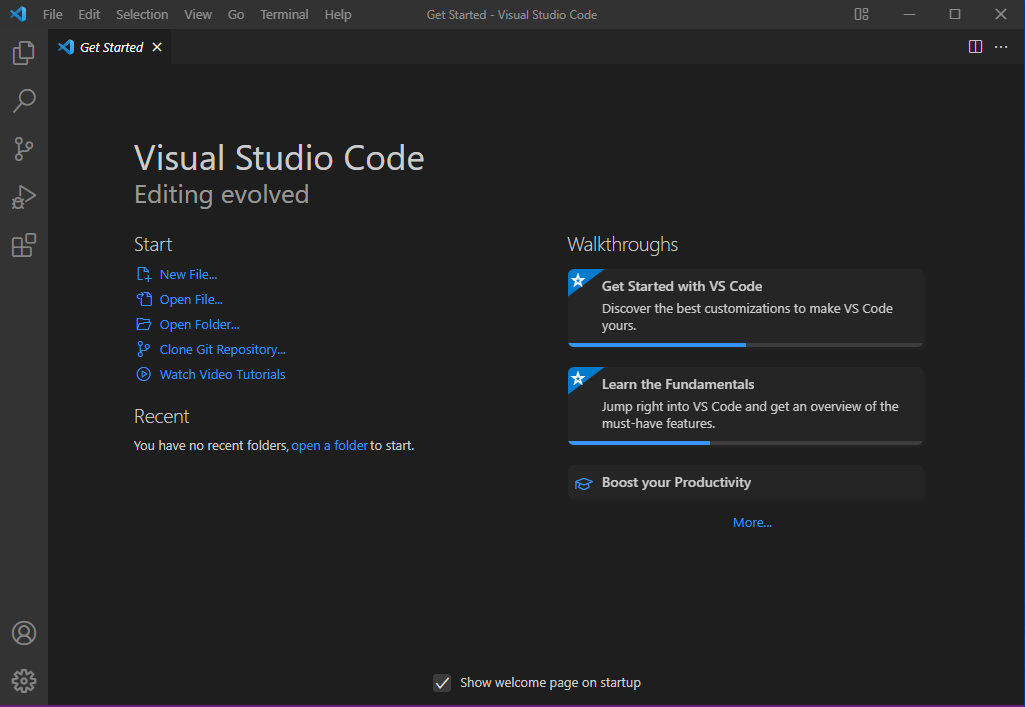
-After clicking on Install, it will take about 1 minute to install the Visual Studio Code on your device.



-After the Installation setup for Visual Studio Code is finished, it will show a window like this below. Tick the “Launch Visual Studio Code” checkbox and then click Next.



-After the previous step, the Visual Studio Code window opens successfully. Now you can create a new file in the Visual Studio Code window and choose a language of yours to begin your programming journey!



Reference - https://www.geeksforgeeks.org/how-to-install-visual-studio-code-on-windows/

2.Configurations and settings for Optimal coding

- Language support for languages e.g python, CSS intelligence, etc

-Linter or formatter e.g prettier, Eslint

-Version control: git lens for git etc

-debuggers- for debugging specific languages

-themes and icons: their are variety of themes and icons you can choose from

\*on the settings icon

-activate auto-save

-enabe word wrap

-Open your terminal and set your default terminal control as gitbash

\*set up your version control

Configure user information

Open your git bash

-git config --global user.name

git config --global user.email

3.User interface Overview

-1. Activity BarThe Activity Bar is located on the far left side of the window. It provides quick access to different views like Explorer, Search, Source Control, Run and Debug, Extensions, and other installed views

2. Side BarThe Side Bar is next to the Activity Bar, typically on the left side of the window. It displays the contents of the selected view from the Activity Bar. For example, if you click on the Explorer icon in the Activity Bar, the Side Bar will show the file explorer. It can also be moved to the right side if preferred.

3. Editor GroupThe Editor Group is the central part of the window where you open and edit your files. VS Code supports multiple editor groups, allowing you to split the editor into multiple sections to view and edit files side-by-side

4. Status BarThe Status Bar is located at the bottom of the window. It provides information about the current file and workspace, such as the file's encoding, line endings, language mode, and more. It also shows the status of background tasks and can include information from extensions.

4.Command pallete

The Command Palette in Visual Studio Code is not a static part of the UI like the Activity Bar or Status Bar. Instead, it's a pop-up interface that you can invoke from anywhere in the editor. Here's how you can access it:Opening the Command PaletteWindows/Linux: Press Ctrl + Shift + P.macOS: Press Cmd + Shift + P.

FunctionalityThe Command Palette allows you to access and execute a wide range of commands in VS Code. You can start typing the name of the command you want to use, and it will filter the results dynamically. It's a powerful tool for quickly performing tasks without having to navigate through menus.Visual RepresentationWhen you open the Command Palette, it appears as a text input field at the top center of the editor window:

-The Command Palette is an essential tool in VS Code, offering quick and easy access to virtually all of the editor's functionality.

5.Extension in Vs Code

Roles of extension

Extensions in VS Code significantly expand its capabilities, making it a versatile and powerful tool for developers. They provide support for a wide range of programming languages, tools, and frameworks, enhance productivity, improve code quality, and enable customization to suit individual workflows and preferences.

-the extension is located at the right of the side pair, it often represented by a four blocks with one of the block stretch away from the other three stacked together

-Users can find suitable extension for their language by click on the extension icon and searching for the desired extension, when found install it. Extension can be managed when they are navigating retyping the extension name again in the extension search box

\* examples of web developement extensions

-html css support

-live server

-css peek

-html snippets

-Eslint

-prettier

-react code snippets

-javascript snippets

-git lens

6.How to access Integrated terminal in Vs code

-the integrated terminal cab be accessed in two ways

- first through the top right side menu bar

-select view

-under view, select terminal

-a new terminal is opened beneath the editor space

- the second way is by right clicking at the right hand side of editor space where the icons are displayed, it more easier if a folder or file is already opened just right click below the folder and

a pop up having open integrated terminal will appear

-select the integrated terminal and a new terminal is opened beneath the editor space

Advantages of using integrated terminal over external terminal

1. Seamless Workflow Integration

- Reduces context switching.

- Starts in the project directory.

2. Enhanced Productivity

- Supports multiple split terminals.

- Allows task automation.

3.Unified Environment

- Consistent experience across OSes.

- Customizable environment.

4. Better Visibility and Management

- Managed within a single window.

- Quick toggling with keyboard shortcuts.

5. Improved Debugging and Error Handling

- Integrated with debugging tools.

- Inline terminal outputs related to code.

7.File and Folder Management

Creating a file in Vs code

- Click on the File on the left upper hand side, navigate to files or folder, choose create file or folder and choose your preferred directory on your desktop.

Opening a file or folder in Vs code

-Open a File: Click on File > Open File... and select the file you want to open.

-Open a Folder: Click on File > Open Folder... and choose the folder you want to open. This will open the folder in the Explorer view on the sidebar.

Opening and Managing Files and Folders

Opening Files and Folders:

Open a File: Click on File > Open File... and select the file you want to open.

Open a Folder: Click on File > Open Folder... and choose the folder you want to open. This will open the folder in the Explorer view on the sidebar.

Explorer View:

Navigating: The Explorer view on the left sidebar shows your project's files and folders. You can expand and collapse folders to navigate through your project.

Creating Files and Folders: Right-click in the Explorer view and select New File or New Folder to create new files or folders.

File Tabs:

Opening Files in Tabs: Double-click on a file in the Explorer view to open it in a new tab. Single-click to preview the file.

Managing Tabs: You can drag and drop tabs to reorder them. Right-click on a tab to see options like Close, Close Others, and Close All.

Managing File and Folder Operations

Renaming:Right-click on a file or folder in the Explorer view and select Rename. You can also press F2 after selecting the file or folder.

Moving:Drag and drop files or folders within the Explorer view to move them to a different location.

Deleting:Right-click on a file or folder in the Explorer view and select Delete.

Copying and Pasting:Right-click on a file or folder and select Copy. Then, right-click where you want to paste it and select Paste.

Explorer View:

-Navigating: The Explorer view on the left sidebar shows your project's files and folders. You can expand and collapse folders to navigate through your project.

-Creating Files and Folders: Right-click in the Explorer view and select New File or New Folder to create new files or folders.

8.Settings and Presence

Accessing Settings

Settings UI:

Open the Command Palette (Ctrl+Shift+P or Cmd+Shift+P on Mac) and type Preferences: Open Settings (UI). Alternatively, you can click on the gear icon in the lower-left corner of the window and select Settings.

Settings JSON:

For advanced users, settings can also be edited directly in the settings.json file. Open the Command Palette and type Preferences: Open Settings (JSON).

Changing the Theme

Using Settings UI:

Open the Settings UI.

In the search bar, type Color Theme.

Click on Color Theme under Preferences.

Choose your preferred theme from the list. You can also access the theme picker quickly by opening the Command Palette and typing Preferences: Color Theme.

Using Command Palette:

Open the Command Palette (Ctrl+Shift+P or Cmd+Shift+P on Mac).

Type Preferences: Color Theme and press Enter.

Use the arrow keys to browse the available themes and press Enter to select one.

Changing the Font Size

Using Settings UI:

Open the Settings UI.

In the search bar, type Font Size.

Under Editor: Font Size, change the value to your desired font size.

Using Settings JSON:

Open the settings.json file.

Add or modify the following line:

json

Copy code

"editor.fontSize": 16

Replace 16 with your desired font size.

Changing Keybindings

Using Keyboard Shortcuts UI:

Open the Command Palette (Ctrl+Shift+P or Cmd+Shift+P on Mac) and type Preferences: Open Keyboard Shortcuts.

This opens the Keyboard Shortcuts editor where you can search for commands and change their keybindings.

Editing Keybindings JSON:

Open the Command Palette and type Preferences: Open Keyboard Shortcuts (JSON).

This opens the keybindings.json file where you can manually add or modify keybindings. For example, to change the keybinding for saving a file to Ctrl+S, you would add:

json

Copy code

[

{

"key": "ctrl+s",

"command": "workbench.action.files.save"

}

]

Examples

Change Theme to Dark+ (default dark):

Open Command Palette and type Preferences: Color Theme.

Select Dark+ (default dark) from the list.

Change Font Size to 14:

Open Settings UI.

Search for Font Size and set the value of Editor: Font Size to 14.

Change Keybinding for Copy:

Open Command Palette and type Preferences: Open Keyboard Shortcuts.

Find the Copy command and click on the pencil icon next to it.

Press the new key combination you want to use (e.g., Ctrl+Shift+C).

By using these methods, you can easily customize VS Code to suit your preferences and improve your coding experience.

9. Debugging in Vs code

Open Your Project:

Launch VS Code and open your project folder via File > Open Folder.

Create/Open Your Program:

Open an existing file or create a new one, like main.py for Python.

Add Debug Configuration:

Go to the Run view (Ctrl+Shift+D), click on create a launch.json file or the gear icon.

Select the appropriate environment (e.g., Python, Node.js), which generates a launch.json file.

Modify launch.json if needed. Example for Python:

json

Copy code

{

"version": "0.2.0",

"configurations": [

{

"name": "Python: Current File",

"type": "python",

"request": "launch",

"program": "${file}",

"console": "integratedTerminal"

}

]

}

Set Breakpoints:

Click next to the line numbers in your code to set breakpoints.

Start Debugging:

Press the green play button in the Run view or F5 to begin debugging. Your program pauses at breakpoints.

Key Debugging Features

Breakpoints:

Set and manage breakpoints by clicking in the gutter. Conditional breakpoints can be set by right-clicking.

Watch Expressions:

Add expressions to watch in the Run view to monitor their values during debugging.

Call Stack:

View the sequence of function calls in the Call Stack section to understand program flow.

Variables:

Inspect variable values in the VARIABLES section. Expand objects and arrays to see details.

Debug Console:

Evaluate expressions and run commands in the Debug Console for real-time inspection.

Step Controls:

Navigate your code with step controls: Continue (F5), Step Over (F10), Step Into (F11), Step Out (Shift+F11), Restart (Ctrl+Shift+F5), and Stop (Shift+F5).

Exception Handling:

Configure the debugger to break on exceptions via the launch.json file or the Debug view settings.

Using these steps and features, you can efficiently set up and debug programs in VS Code

12.Using Source Codes

Setting Up Git in VS Code

Install Git:

Ensure Git is installed on your system. You can download it from git-scm.com and follow the installation instructions.

Verify Git Installation:

Open the terminal in VS Code (Ctrl+ or Cmd+ on Mac) and type git --version to verify Git is installed correctly.

Configure Git:

Set your username and email for Git by running the following commands in the terminal:

bash

git config --global user.name

git config --global user.email

Using Git in Vscode

First open an repository in your GitHub account, name It Expense Tracker

-Open your local folder containing your project(Expense tracker )in Vs code stored in local device

-open a terminal either by right-clicking immediately below your project at the left hand side or navigate to view on the topmost left hand side of Vs code and click on terminal

-make gitbash your default terminal

-cd in your folder

-git init (to initializs your project to GitHub)

-git add . (To effect any changes made on your local repo to remote repository)

-git commit -m "my first commit" ( this save your changes and flags it with a message)

-git branch -M branch Main(to choose your desired branch)

-git remote add origin URl( URL of your remote repository)

-git push -u origin main(this pushes your local project to your remote repo)

References:

-https://www.geeksforgeeks.org/how-to-install-visual-studio-code-on-windows/

-chatgpt