SE-Assignment-5: Installation and Navigation of Visual Studio Code (VS Code)

Installation of VS Code:

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

Prerequisites:

Ensure that your Windows 11 system meets the minimum requirements for running Visual Studio Code.

Make sure you have a stable internet connection.

Download VS Code:

Visit the official Visual Studio Code website (https://code.visualstudio.com/Download).

Click on the "Windows" button to download the installer for Windows.

Once the download is complete, double-click the installer file to start the installation process.

Install VS Code:

Follow the on-screen instructions in the installer wizard.

Choose the destination folder where you want to install VS Code.

Click "Next" and then "Install" to begin the installation process.

Once the installation is complete, click "Finish" to exit the installer.

First-time Setup:

After installing VS Code, adjust the following initial configurations and settings for an optimal coding environment:

Extensions: Install essential extensions for your programming languages and workflow, such as language support, debugging tools, and code formatting extensions.

Settings: Customize VS Code settings according to your preferences, including theme selection, font size, and keybindings.

Integrated Terminal: Set up the integrated terminal preferences, such as the default shell and terminal font size.

Version Control: Configure Git settings and integrate with VS Code for version control operations.

User Interface Overview:

The main components of the VS Code user interface include:

Activity Bar: Provides quick access to different views and features such as Explorer, Search, Source Control, Debugging, and Extensions.

Side Bar: Contains various panels like Explorer (file browser), Search, Source Control (Git), and Extensions.

Editor Group: Displays open files and their contents. Multiple editor groups can be created for side-by-side editing.

Status Bar: Shows information about the current file, Git branch status, and other features like language mode and indentation.

Command Palette:

The Command Palette in VS Code is a powerful tool that allows users to execute commands and perform various tasks. It can be accessed by pressing Ctrl+Shift+P or Cmd+Shift+P on macOS. Common tasks that can be performed using the Command Palette include:

Opening files and folders

Running tasks and commands

Installing and managing extensions

Changing settings and preferences

Extensions in VS Code:

Extensions play a crucial role in extending the functionality of VS Code. Users can find, install, and manage extensions through the Extensions view in the Activity Bar. Essential extensions for web development include:

IntelliSense for code completion

Debugger for debugging code

ESLint for code linting

Live Server for live preview of web pages

GitLens for enhanced Git integration

Integrated Terminal:

To open and use the integrated terminal in VS Code, press Ctrl+` or go to View -> Terminal from the menu. The integrated terminal provides advantages such as:

Seamless integration with the editor workflow

Access to powerful command-line tools without leaving VS Code

Ability to run tasks, compile code, and execute commands directly from the terminal

File and Folder Management:

Users can create, open, and manage files and folders in VS Code using the Explorer view in the Side Bar. To navigate between different files and directories efficiently, users can use keyboard shortcuts like Ctrl+P to open files by name and Ctrl+Shift+E to focus on the Explorer view.

Settings and Preferences:

Settings in VS Code can be found and customized through the Settings view or by editing the settings.json file. Examples of settings customization include:

Changing the theme: File -> Preferences -> Color Theme

Adjusting font size: File -> Preferences -> Settings -> Text Editor -> Font Size

Configuring keybindings: File -> Preferences -> Keyboard Shortcuts

Debugging in VS Code:

To set up and start debugging a simple program in VS Code, follow these steps:

Install the necessary debugging extensions for your programming language.

Open the file you want to debug and set breakpoints by clicking in the gutter next to the line numbers.

Launch the debugger by pressing F5 or selecting the debug configuration from the Debug view.

Use the debugging features such as stepping through code, inspecting variables, and viewing call stacks to troubleshoot issues.

Using Source Control:

To integrate Git with VS Code for version control, follow these steps:

Install Git on your system if not already installed.

Initialize a Git repository for your project using the command line or the Source Control view in VS Code.

Make commits by staging changes and adding commit messages.