Assignment On Installation and Navigation of Visual Studio Code (VS Code)

Installation of VS Code

Steps to Download and Install Visual Studio Code on Windows 11:

Prerequisites:

* I ensured the system meets the minimum requirements, including having Windows 11 with internet access.
* I verified that necessary runtimes or environments (like Python or Node.js) were already installed, as required for certain coding tasks.

Download:

* I visited the official Visual Studio Code website.
* I clicked on the 'Download for Windows' button.

Installation:

* After downloading, I opened the installer (VSCodeSetup.exe).
* I followed the installation wizard:
* Accepted the license agreement.
* Chose the installation location.
* Selected additional tasks such as creating a desktop icon and adding VS Code to the PATH.
* I clicked 'Install' and waited for the process to complete.
* Finally, I clicked 'Finish' to launch VS Code.
* First-time Setup

Initial Configurations and Settings:

Theme and Appearance:

I navigated to File > Preferences > Color Theme or pressed Ctrl+K Ctrl+T.

I selected a theme, opting for a Dark+ theme for better visibility.

Font Size and Family:

I went to File > Preferences > Settings or pressed Ctrl+,.

I searched for Font Size and adjusted it to a comfortable level.

I set Editor: Font Family to my preferred coding font, Fira Code.

Extensions:

I clicked on the Extensions icon in the Activity Bar or pressed Ctrl+Shift+X.

I installed essential extensions such as:

Python (for Python development)

ESLint (for JavaScript linting)

Prettier (for code formatting)

Live Server (for web development)

Workspace Settings:

* I created a folder for my workspace and opened it in VS Code.
* I adjusted workspace-specific settings by navigating to File > Preferences > Settings and selecting the Workspace tab.

User Interface Overview

Main Components of the VS Code User Interface:

* Activity Bar:Located on the far left, the Activity Bar provides access to Views and View Containers such as Explorer, Search, Source Control, Run and Debug, and Extensions.
* Side Bar:Adjacent to the Activity Bar, the Side Bar displays different views like File Explorer, Source Control, and extensions, depending on the selected icon in the Activity Bar.
* Editor Group:The central area where files are opened and edited. It supports multiple tabs and split views for editing multiple files side by side.
* Status Bar:Located at the bottom, the Status Bar displays information about the current workspace, such as branch, file encoding, line/column number, and problems in the code.

**Command Palette**

The Command Palette is a powerful tool to access and execute all VS Code commands.

It can be accessed by pressing Ctrl+Shift+P or F1.

Examples of Common Tasks:

Changing the color theme: > Preferences: Color Theme.

Installing extensions: > Extensions: Install Extensions.

Opening settings: > Preferences: Open Settings.

Extensions in VS Code

Role of Extensions:

Extensions enhance the functionality of VS Code by adding support for languages, debuggers, and tools.

Finding, Installing, and Managing Extensions:

I clicked the Extensions icon in the Activity Bar or pressed Ctrl+Shift+X.

I searched for the desired extension.

I clicked 'Install' to add it to my VS Code.

I managed installed extensions through the Extensions view.

Essential Extensions for Web Development:

HTML Snippets

JavaScript (ES6) code snippets

Integrated Terminal

Opening and Using the Integrated Terminal:

I opened the terminal by navigating to View > Terminal or pressing `Ctrl+`` (backtick).

**Advantages of Using the Integrated Terminal:**

* It is seamlessly integrated with the editor.
* It allows running scripts, commands, and viewing output without leaving the editor.
* It supports multiple terminal sessions.

File and Folder Management

Creating, Opening, and Managing Files and Folders:

Creating Files/Folders:

I right-clicked in the Explorer view and selected 'New File' or 'New Folder'.

I used the keyboard shortcut Ctrl+N to create a new file.

Opening Files/Folders:

I used File > Open Folder or Ctrl+K Ctrl+O to open a folder.

I dragged and dropped files/folders into the editor.

Navigating Between Files:

I used Ctrl+P to quickly open files by typing their names.

I used Ctrl+Tab to switch between open files.

Settings and Preferences

Customizing Settings in VS Code:

I accessed settings through File > Preferences > Settings or Ctrl+,.

Examples of Customizations:

Theme: I searched for Color Theme and chose one.

Font Size: I searched for Font Size and set the desired size.

Keybindings: I went to File > Preferences > Keyboard Shortcuts or pressed Ctrl+K Ctrl+S to customize shortcuts.

Debugging in VS Code

Setting Up and Starting Debugging:

Configure Debugger:

I clicked on the Run and Debug icon in the Activity Bar.

I clicked on 'create a launch.json file' to configure the debugger for my project.

Set Breakpoints:

I clicked in the gutter next to the line numbers in my code to set breakpoints.

Start Debugging:

I clicked the green play button in the Run and Debug view or pressed F5.

Key Debugging Features:

Step over, step into, and step out.

Watch variables and view call stack.

Inspect and modify variables.

Using Source Control

Integrating Git with VS Code:

Initialize Repository:

I opened my project folder in VS Code.

I clicked on the Source Control icon in the Activity Bar or pressed Ctrl+Shift+G.

I clicked 'Initialize Repository'.

Making Commits:

I staged changes by clicking the + next to the files.

I entered a commit message and clicked the checkmark to commit.

Pushing Changes to GitHub:

I set up the remote repository by using the terminal or Source Control view.

I used commands git remote add origin <URL> and git push -u origin master.