**SE-Assignment-5**

**Installation and Navigation of Visual Studio Code (VS Code) Instructions: Answer the following questions based on your understanding of the installation and navigation of Visual Studio Code (VS Code). Provide detailed explanations and examples where appropriate.**

**Questions:**

1. **Installation of VS Code:**
   * **Describe the steps to download and install Visual Studio Code on Windows 11 operating system. Include any prerequisites that might be needed.**

### Prerequisites:

Ensure your system meets the following requirements:

1. **Operating System**: Windows 11 (32-bit or 64-bit).
2. **Disk Space**: At least 200 MB of free disk space.
3. **Internet Connection**: Required for downloading VS Code and its extensions.

### Steps to Download and Install Visual Studio Code:

1. **Download VS Code**:
   * Open your web browser and go to the official Visual Studio Code website: <https://code.visualstudio.com/>.
   * Click on the "Download for Windows" button. The website should automatically detect your operating system as Windows and offer the correct version for download.
2. **Run the Installer**:
   * Once the download is complete, locate the downloaded installer file (VSCodeSetup-x64-xxx.exe where xxx represents the version number).
   * Double-click on the installer file to run it. You may be prompted by User Account Control (UAC); click Yes to proceed.
3. **Accept License Agreement**:
   * In the VS Code Setup window, review the license agreement.
   * Check the box that indicates you accept the terms and conditions, then click on Next.
4. **Select Destination Location** (Optional):
   * Choose the destination folder where you want to install Visual Studio Code. The default location (C:\Program Files\Microsoft VS Code) is usually fine for most users.
   * Click on Next to continue.
5. **Select Start Menu Folder** (Optional):
   * Choose a Start Menu folder for creating shortcuts. The default folder (Microsoft Visual Studio Code) is typically sufficient.
   * Click on Next.
6. **Create Desktop Icon** (Optional):
   * Check the box if you want to create a desktop icon for easier access to VS Code.
   * Click on Next.
7. **Choose Additional Tasks** (Optional):
   * Optionally, choose any additional tasks such as creating a Quick Launch icon or adding VS Code to the PATH environment variable.
   * Click on Next.
8. **Install**:
   * Click on the Install button to start the installation process. This may take a few moments depending on your system's performance.
9. **Complete Installation**:
   * Once the installation is complete, you should see a "Completing the Visual Studio Code Setup Wizard" screen.
   * Ensure the checkbox for "Launch Visual Studio Code" is checked.
   * Click on Finish to exit the installer and launch Visual Studio Code.
10. **Launch Visual Studio Code**:
    * VS Code should now be installed on your system. It will automatically launch after the installation completes.

The system setup requires elevation to Administrator privileges to run and will place the installation under the system's Program Files. The in-product update flow will also require elevation, making it less streamlined than the user setup. On the other hand, installing VS Code using the system setup means that it will be available to all users in the system.

1. **First-time Setup:**
   * **After installing VS Code, what initial configurations and settings should be adjusted for an optimal coding environment? Mention any important settings or extensions.**

 **Extensions**: Explore and install extensions from the VS Code Marketplace to enhance functionality for your development environment.

 **Settings**: Customize VS Code settings (Ctrl+, or Cmd+,) to tailor the editor to your preferences (e.g., theme, keybindings).

User and workspace settings

Editor settings through the preferences options or from the commands palette.

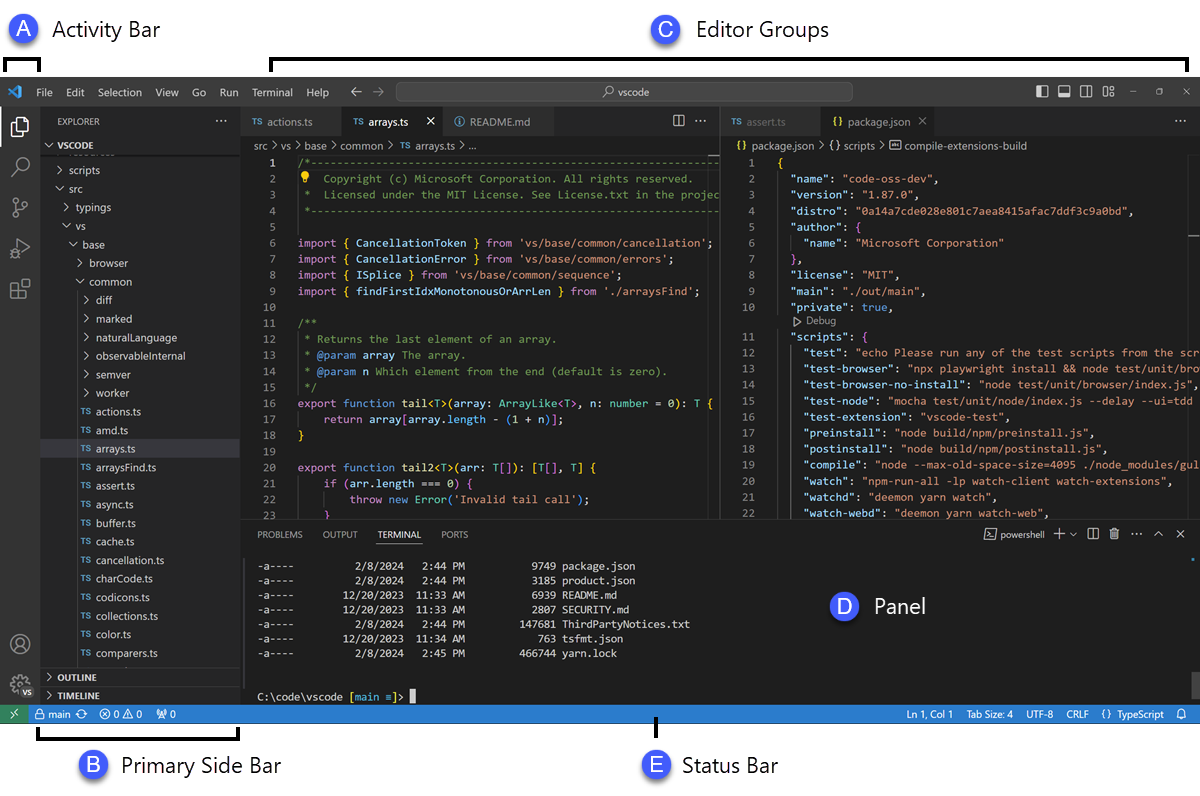
Installing extensions

Source control settings

Edit settings setting can be edited by either a checkbox, an text input field, or a dropdown. Edit the text or select the option you want to change to the desired settings.

Setup will add Visual Studio Code to your %PATH%, so from the console you can type 'code .' to open VS Code on that folder. You will need to restart your console after the installation for the change to the %PATH% environmental variable to take effect.

1. **User Interface Overview:**
   * **Explain the main components of the VS Code user interface. Identify and describe the purpose of the Activity Bar, Side Bar, Editor Group, and Status Bar.**



1. **Editor** - The main area to edit your files. You can open as many editors as you like side by side vertically and horizontally.
2. **Primary Side Bar** - Contains different views like the Explorer to assist you while working on your project.
3. **Status Bar** - Information about the opened project and the files you edit.
4. **Activity Bar** - Located on the far left-hand side. Lets you switch between views and gives you additional context-specific indicators, like the number of outgoing changes when Git is enabled. You can change the position of the Activity Bar.
5. **Panel** - An additional space for views below the editor region. By default, it contains output, debug information, errors and warnings, and an integrated terminal. The Panel can also be moved to the left or right for more vertical space.
6. **Command Palette:**
   * **What is the Command Palette in VS Code, and how can it be accessed? Provide examples of common tasks that can be performed using the Command Palette.**

The Command Palette provides access to many commands. You can run editor commands, open files, search for symbols, and see a quick outline of a file, all using the same interactive window

Accessed using the command **Ctrl+Shift+P**

1. **Extensions in VS Code:**
   * **Discuss the role of extensions in VS Code. How can users find, install, and manage extensions? Provide examples of essential extensions for web development.**

VS Code extensions let you add languages, debuggers, and tools to your installation to support your development workflow.

### Finding Extensions

1. **Open the Extensions View**:
   * Click on the Extensions icon in the Activity Bar on the side of the VS Code window (it looks like four squares).
   * Alternatively, use the keyboard shortcut Ctrl+Shift+X (Windows/Linux) or Cmd+Shift+X (Mac) to open the Extensions view directly.
2. **Search for Extensions**:
   * In the Extensions view, you can search for extensions using the search bar at the top.
   * Type keywords related to the functionality you are looking for (e.g., Python, GitLens, Markdown, etc.).
3. **Explore Recommended and Popular Extensions**:
   * VS Code provides tabs such as Recommended, Popular, and Installed to help you discover extensions.
   * Browse through different categories (e.g., Programming Languages, Debuggers, Themes) using the sidebar on the left.
4. **Read Reviews and Ratings**:
   * Click on an extension to view its details, including description, version history, reviews, ratings, and installation statistics.
   * Reviews and ratings can provide insights into the extension's usefulness and reliability.

### Installing Extensions

1. **Install from the Extensions View**:
   * Once you've found an extension you want to install, click on the Install button next to its name.
   * Wait for the installation to complete. VS Code will notify you when the installation finishes.
2. **Install from the Command Palette**:
   * Open the Command Palette (Ctrl+Shift+P or Cmd+Shift+P) and type Extensions: Install Extensions.
   * Enter the name of the extension you want to install and press Enter.

### Managing Extensions

1. **Disable or Uninstall Extensions**:
   * In the Extensions view, click on the gear icon next to an installed extension.
   * Choose Disable to temporarily disable the extension or Uninstall to remove it completely.
2. **Enable Disabled Extensions**:
   * In the Extensions view, switch to the Disabled tab to see extensions that are currently disabled.
   * Click on Enable next to an extension to re-enable it.
3. **Update Extensions**:
   * VS Code automatically checks for updates to installed extensions.
   * If updates are available, you'll see an Update button next to the extension in the Extensions view.
   * Click on Update to install the latest version.
4. **Manage Extension Settings**:
   * Some extensions have configurable settings that you can manage in the Settings view (Ctrl+, or Cmd+,).
   * Search for the extension name or category to find specific settings related to an extension.

For web development we can have the following extensions: Liveserver, javascript booster, css peek, intelisense, intellicode

1. **Integrated Terminal:**
   * **Describe how to open and use the integrated terminal in VS Code. What are the advantages of using the integrated terminal compared to an external terminal?**

Click on view in the menu bar then select terminal. This opens the terminal window

Terminal window has the advantage of providing integration with the editor to support features such as links to workspace files and error detection. These are:

**Seamless Integration**:

* The integrated terminal is built directly into VS Code, providing a seamless and cohesive development environment. You don't need to switch between multiple applications, keeping your workflow focused within VS Code itself.

**Contextual Awareness**:

* The integrated terminal automatically opens in the root directory of your current workspace. This context-awareness allows for easier navigation and execution of commands related to your project without having to manually navigate to the project directory in an external terminal.

1. **File and Folder Management:**
   * **Explain how to create, open, and manage files and folders in VS Code. How can users navigate between different files and directories efficiently?**

**Creating Files**:

* To create a new file, you can use either the Explorer view or the Command Palette:
  + **Explorer View**: Right-click on a folder in the Explorer sidebar (left-hand side) and select New File.
  + **Command Palette**: Use the shortcut Ctrl+Shift+P (Windows/Linux) or Cmd+Shift+P (Mac) to open the Command Palette, then type New File and press Enter.

**Opening Files**:

* There are several ways to open files in VS Code:
  + **Explorer View**: Double-click on a file in the Explorer sidebar to open it.
  + **Command Palette**: Use the Command Palette (Ctrl+P or Cmd+P), type the name of the file, and press Enter to open it.
  + **Keyboard Shortcut**: Use Ctrl+P (Windows/Linux) or Cmd+P (Mac) directly to open the Command Palette and then type the file name.

**Renaming Files and Folders**:

* **Explorer View**: Right-click on a file or folder in the Explorer sidebar and choose Rename, or press F2 with the item selected.
* **Command Palette**: Use Ctrl+P or Cmd+P to open the Command Palette, type Rename File or Rename Folder, and press Enter.

**Deleting Files and Folders**:

* **Explorer View**: Right-click on a file or folder in the Explorer sidebar and select Delete, or press Delete or Backspace after selecting the item.
* **Command Palette**: Use the Command Palette (Ctrl+Shift+P or Cmd+Shift+P), type Delete File or Delete Folder, and press Enter.

### Navigating Between Files and Directories

1. **Explorer Sidebar Navigation**:
   * Use the Explorer sidebar (left-hand side) to navigate between files and folders by clicking on them.
2. **File Navigation**:
   * **Switching Between Open Files**: Use Ctrl+Tab (Windows/Linux) or Cmd+Tab (Mac) to cycle through recently opened files.
   * **Go to File**: Use Ctrl+P (Windows/Linux) or Cmd+P (Mac), then start typing the file name to quickly jump to a specific file.
3. **Directory Navigation**:
   * **Explorer View**: Use the Explorer sidebar to navigate through directories by expanding and collapsing folders.
   * **Command Palette**: Use the Command Palette to quickly switch between opened folders or open a new one (Ctrl+P or Cmd+P, then type Open Folder).
4. **Settings and Preferences:**
   * **Where can users find and customize settings in VS Code? Provide examples of how to change the theme, font size, and keybindings.**

In Visual Studio Code (VS Code), users can find and customize settings through the **Settings** interface, accessible via the Settings icon on the Activity Bar or by using the Command Palette.

#### Changing the Theme

1. **Changing the Theme**:
   * Go to **Settings** (Ctrl+, or Cmd+,).
   * Search for Color Theme.
   * Click on the dropdown under "Color Theme" and select a new theme from the list.
   * Example: Changing to the "Dark+ (default dark)" theme.

#### Adjusting Font Size

1. **Changing Font Size**:
   * Go to **Settings** (Ctrl+, or Cmd+,).
   * Search for Font Size.
   * Find the setting Editor: Font Size.
   * Adjust the value (e.g., change from default 14 to 16) to increase or decrease the font size.
   * Example: Changing the font size from 14 to 16.

#### Customizing Keybindings

1. **Customizing Keybindings**:
   * Go to **Settings** (Ctrl+, or Cmd+,).
   * Search for Keybindings.
   * Click on Open Keyboard Shortcuts (JSON) to open the keybindings.json file.
   * Here, you can customize keybindings by editing the JSON file directly or by using the graphical interface provided.
   * Example: Adding a custom keybinding for a specific command or changing an existing keybinding.
2. **Debugging in VS Code:**
   * **Outline the steps to set up and start debugging a simple program in VS Code. What are some key debugging features available in VS Code?**

 **Install Required Extensions**:

* Ensure you have the necessary extensions installed for the programming language you're working with. For example, if you're programming in JavaScript, you might need the Debugger for Chrome extension.

 **Open Your Project**:

* Open the folder containing your project in VS Code.

 **Create or Open a Debug Configuration**:

* Click on the **Run** icon in the Activity Bar on the side of the window (it looks like a play button inside a bug).
* Click on the **Add Configuration** button or edit the launch.json file directly to configure your debugger.

 **Configure Launch Settings**:

* VS Code provides templates for various debug configurations (e.g., for Node.js, Python, C++, etc.). Select the appropriate template or customize your own configuration.
* Specify details such as the program to debug, arguments, environment variables, etc.

 **Start Debugging**:

* Once your launch configuration is set up:
  + Click on the green play button (Start Debugging) next to the configuration dropdown in the Debug view.
  + Alternatively, use the F5 key as a shortcut to start debugging.

 **Debugging Controls**:

* Use the debugging controls provided in the Debug toolbar (visible when debugging is active):
  + **Pause/Resume**: Pause or resume the execution of your program.
  + **Step Over/Into/Out**: Step through your code line-by-line, stepping over function calls or into them.
  + **Restart**: Restart the debugging session.
  + **Stop**: Stop debugging and terminate the program.
  + **Variables**: View and inspect the values of variables in your code.
  + **Watch**: Add specific variables or expressions to monitor their values during debugging.

1. **Using Source Control:**
   * **How can users integrate Git with VS Code for version control? Describe the process of initializing a repository, making commits, and pushing changes to GitHub.**

Integrating Git with Visual Studio Code (VS Code) is straightforward and enhances version control capabilities within your development workflow. Here’s a step-by-step guide on how to initialize a Git repository, make commits, and push changes to GitHub:

### Initializing a Git Repository

1. **Open Your Project in VS Code**:
   * Open the folder containing your project in VS Code.
2. **Open the Source Control View**:
   * Click on the **Source Control** icon in the Activity Bar on the side (it looks like a square with a branch symbol).
3. **Initialize Git Repository**:
   * Click on the **Initialize Repository** button (it looks like a plus + icon) or use the Command Palette (Ctrl+Shift+P or Cmd+Shift+P) and type Git: Initialize Repository.
   * Choose the root directory of your project as the location for the Git repository.

### Making Commits

1. **Stage Changes**:
   * In the Source Control view, you'll see a list of changes detected by Git.
   * Click on the + icon next to each file you want to include in the commit, or use the Stage All Changes button (checkmark icon) to stage all changes.
2. **Commit Changes**:
   * Enter a commit message in the text box labeled Message (Ctrl+Enter to commit, Ctrl+Escape to cancel).
   * Press Ctrl+Enter (Windows/Linux) or Cmd+Enter (Mac) to commit your changes.

### Pushing Changes to GitHub

1. **Configure Remote Repository (GitHub)**:
   * If you haven’t done so already, create a repository on GitHub.
   * Obtain the HTTPS or SSH URL of your GitHub repository.
2. **Add Remote Repository**:
   * Use the Command Palette (Ctrl+Shift+P or Cmd+Shift+P) and type Git: Add Remote.
   * Paste the URL of your GitHub repository when prompted.
3. **Push Changes**:
   * After committing your changes locally, click on the ellipsis (...) in the Source Control view and select Push.
   * Alternatively, use the Command Palette (Ctrl+Shift+P or Cmd+Shift+P) and type Git: Push.
4. **Authenticate (if necessary)**:
   * If it's your first time pushing to GitHub from VS Code, you may need to authenticate. Follow the prompts in the VS Code integrated terminal or authentication dialog.
5. **Monitor Push Progress**:
   * VS Code will display the progress of the push operation in the lower right corner (Git indicator)

**References:**

Google.com

Visual studio code documentations

Chat GPT prompts

Submission Guidelines:

* Your answers should be well-structured, concise, and to the point.
* Provide screenshots or step-by-step instructions where applicable.
* Cite any references or sources you use in your answers.
* Submit your completed assignment by 1st July