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:

a)Installation of VS Code

•Describe the Steps to Download and install Visual studio code on windows 11 operating system. Include and prerequisites that may be needed

steps to download and install:

1. Download the Installer:

- Go to the official Visual Studio Code website: [https://code.visualstudio.com/](https://code.visualstudio.com/).

- Click on the download button for Windows to get the installer.

2. Run the Installer:

- Locate the downloaded installer (`VSCodeSetup.exe`) and double-click to run it.

3. Setup Wizard:

- Follow the prompts in the setup wizard. Agree to the license terms, choose the installation location, and select any additional tasks (such as creating a desktop icon).

4. Install:

- Click on the install button and wait for the process to complete. Once done, you can choose to launch VS Code immediately.

Prerequisites:

- Windows 11.

- Administrator privileges to install software.

- Internet connection to download the installer.

b) First-time Setup:

•After installing VS Code, what initial configuration and settings should be adjusted for an optimal coding environment?

1. Theme and Appearance:

- Open the command palette (`Ctrl+Shift+P`), type "Preferences: Color Theme," and choose a theme that suits your preference.

2. Font Size and Family:

- Go to `File > Preferences > Settings`, then navigate to `Text Editor > Font`. Adjust the font size and family as needed.

3. Extensions:

- Essential extensions for an optimal coding environment can be installed from the Extensions view (`Ctrl+Shift+X`). Examples include:

-Prettier: Code formatter.

- ESLint: Linting for JavaScript.

- GitLens: Enhances Git capabilities.

- Debugger for Chrome: Debugging for web development.

4. Workspace Settings:

- Configure workspace-specific settings via the `.vscode/settings.json` file for project-specific configurations.

c)User Interface Overview:

**Main Components of the VS Code User Interface**

1. Activity Bar:

- Located on the far left, it provides access to various views like Explorer, Search, Source Control, Run and Debug, and Extensions.

2. Side Bar:

- Displays the content of the selected view from the Activity Bar. For example, the Explorer view shows the file and folder structure of the workspace.

3. Editor Group:

- The central area where you open and edit your files. You can have multiple editor groups open in a grid layout for side-by-side editing.

4. Status Bar:

- Located at the bottom, it provides information about the current project, such as branch name, line endings, encoding, and any background tasks like build or test results.

d) Command Palette:

What is the Command Palette in the vs code and it can be accessed?:

- The Command Palette is a powerful tool in VS Code that allows you to execute commands and perform tasks quickly without navigating through menus.

Accessing the Command Palette:

- Press `Ctrl+Shift+P` or `F1`.

Provide Examples of Common Tasks That can be performed using the command palette:

- Changing color themes: "Preferences: Color Theme."

- Formatting code: "Format Document."

- Opening settings: "Preferences: Open Settings (JSON)."

e) Extensions in VS Code:

•Describe the Role of Extensions in VS code. How can users find, install, and manage extensions? Provides examples of essential extensions for web development:

- Extensions enhance the functionality of VS Code, providing additional tools and features tailored to specific development needs.

Finding, Installing, and Managing Extensions:

1. Finding Extensions:

- Open the Extensions view (`Ctrl+Shift+X`) and search for the desired extension.

2. Installing Extensions:

- Click on the Install button for the selected extension.

3. Managing Extensions:

- You can enable, disable, or uninstall extensions from the Extensions view.

Essential Extensions for Web Development:

- Live Server: Launch a local development server with a live reload feature.

- ESLint: Integrates ESLint into VS Code.

- Prettier: Code formatter.

- Debugger for Chrome: Debug your JavaScript code in the Chrome browser.

- Emmet: Provides shortcuts for writing HTML and CSS.

f) Integrated Terminal:

•Describe how to open and use the integrated terminal in VS Code. What are the advantages of using the integrated terminal compaired and external terminal? File and Folder Management:

-Opening and Using the Integrated Terminal:

- Open the terminal with `Ctrl+` or `View > Terminal`.

**What are the Advantages of Using the Integrated Terminal Compaired to an external terminal?**

- Directly interact with your project within the same window.

- Run commands, scripts, and version control without switching contexts.

- Supports multiple terminal sessions.

g)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?

1. Creating Files/Folders.

- Right-click in the Explorer view and select "New File" or "New Folder."

2. Opening Files/Folders.

- Use `Ctrl+O` to open a file and `Ctrl+K Ctrl+O` to open a folder.

**Navigate Between Different Files and Directories Efficiently?**

- Use the Explorer view or `Ctrl+P` to quickly open files by name.

- Navigate between opened files using `Ctrl+Tab` or Ctrl+Shift+Tab ‘.

h)Settings and Preferences:

•Where can users find and customize settings in VS Code? Provide examples of how to change the theme, front size, and key bindings.

- Access settings through `File > Preferences > Settings` or `Ctrl+,`.

**Provide Examples of how to change the theme front size and keybindings**.

1. Changing Theme

- Search for "Theme" in settings and choose your preferred theme.

2. Adjusting Font Size:

- Navigate to `Text Editor > Font` and set the font size.

3. Customizing Keybindings:

- Go to `File > Preferences > Keyboard Shortcuts` or `Ctrl+K Ctrl+S` and modify keybindings as needed.

I) 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?

Setting Up and Starting Debugging:

1. Open the Run and Debug View:

- Click on the Run and Debug icon in the Activity Bar or press `Ctrl+Shift+D`.

2. Configure Debugger:

- Click on the gear icon to create a `launch.json` file for your project with the appropriate configuration.

3. Set Breakpoints:

- Click in the gutter next to the line numbers in the editor to set breakpoints.

4. Start Debugging:

- Click the green play button or press `F5`.

* **what are some key debugging Features available in vs code**?

- Breakpoints: Pause execution at specific lines.

- Call Stack: View the call stack at any point during execution.

- Watch Expressions: Monitor variable values.

- Variables View: Inspect variables in the current scope.

j)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**.

1. Initialize a Repository:

- Open the Source Control view with `Ctrl+Shift+G` and click on "Initialize Repository."

2. Making Commits:

- Stage changes by clicking the "+" icon next to modified files.

- Enter a commit message and click the checkmark icon to commit.

3. Pushing Changes to GitHub:

- Ensure your repository is connected to a remote (e.g., GitHub).

- Use the integrated terminal to run `git push` or use the Source Control view's push button.

