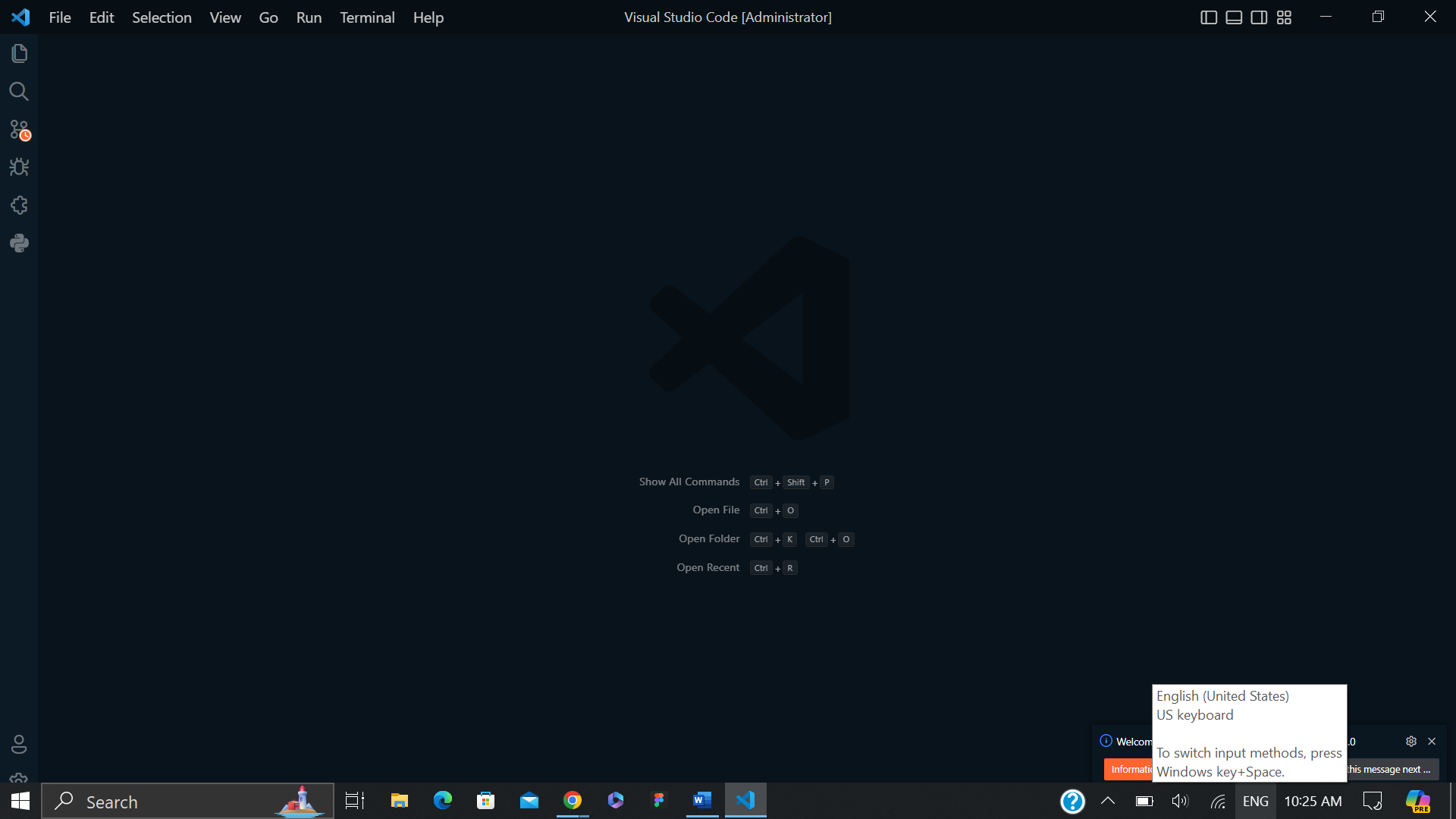
**LOXANA BEATRICE**

**sSE-Assignment-5**

**Installation and Navigation of Visual Studio Code (VS Code) Instructions:**

**Steps to Download and Install Visual Studio Code on Windows 11 though in my case I was using windows 10 pro**

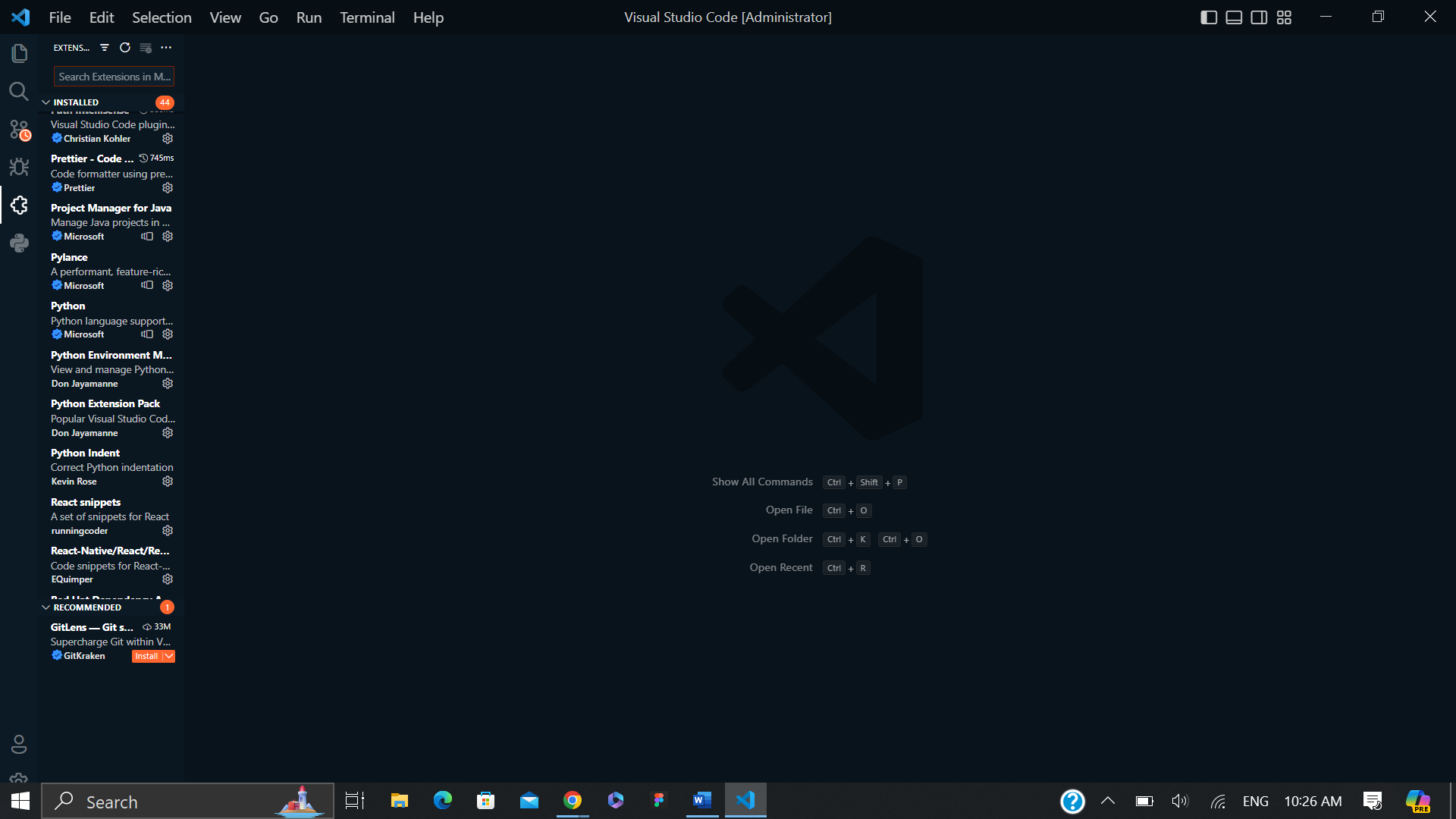
1. **Visit the Visual Studio Code Website:**
   * Open your web browser and go to the official [Visual Studio Code website](https://code.visualstudio.com/).
2. **Download the Installer:**
   * Click on the "Download" button. The website should automatically detect that you are using Windows and offer you the Windows installer (.exe file).
3. **Run the Installer:**
   * Once the download is complete, open the downloaded file. You might see a security warning; click "Run" to continue.
4. **Start the Installation Process:**
   * The Visual Studio Code Setup Wizard will open. Click "Next" to proceed through the installation steps.
5. **Accept the License Agreement:**
   * Read through the license agreement, check the box to accept the terms, and click "Next".
6. **Select Installation Location:**
   * Choose the destination folder for the installation. The default location is usually fine. Click "Next".
7. **Select Additional Tasks:**
   * You will be prompted to select additional tasks such as creating a desktop icon, adding "Open with Code" actions to Windows Explorer, and more. Select the tasks you want and click "Next".
8. **Install:**
   * Click "Install" to begin the installation. The setup wizard will install Visual Studio Code on your system.
9. **Finish Installation:**
   * Once the installation is complete, you can choose to launch Visual Studio Code immediately by checking the box "Launch Visual Studio Code" and clicking "Finish".
10. **First Launch:**
    * If you chose to launch Visual Studio Code, it will open immediately. Otherwise, you can start it from the Start menu or the desktop icon if you created on.



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.**

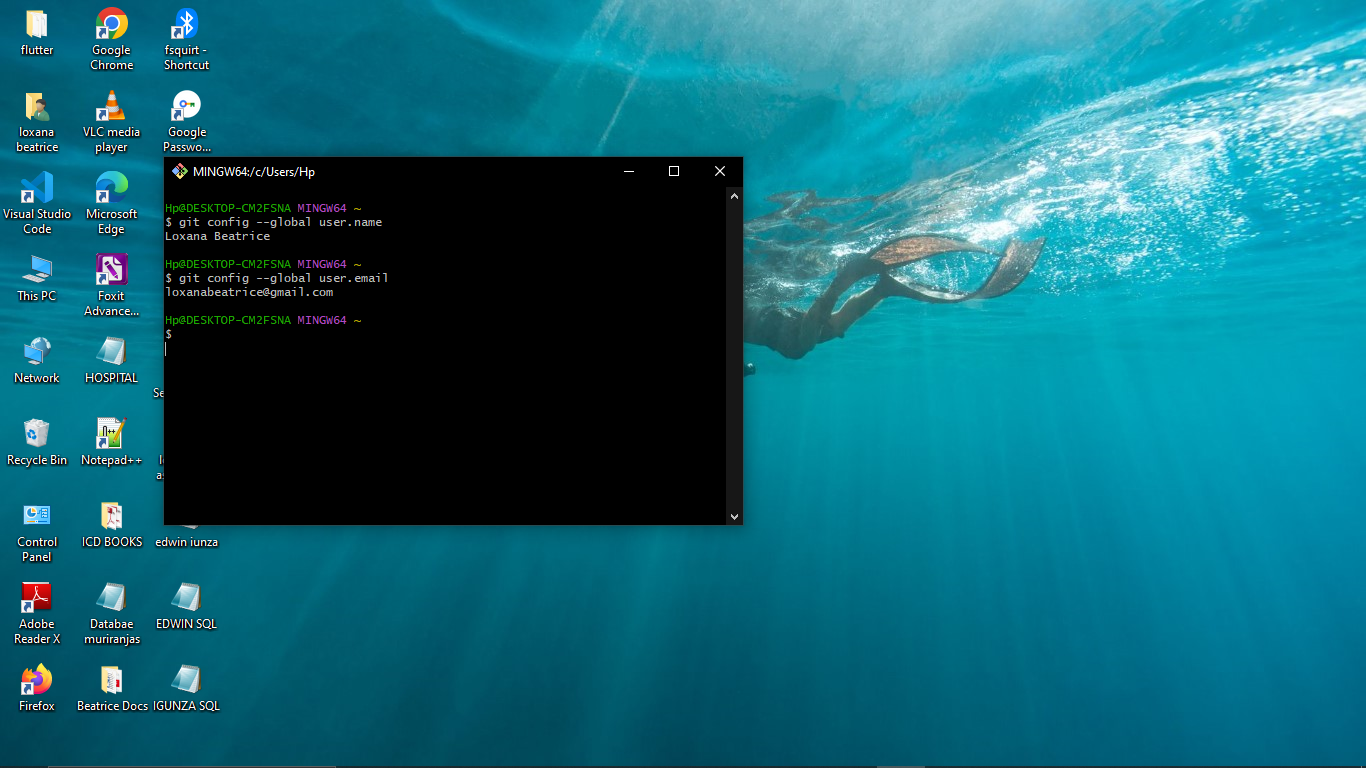
**Extensions:**

* **Install Essential Extensions:**
  + Click on the Extensions icon on the sidebar or press Ctrl+Shift+X and search for the following recommended extensions:
    - **Python** (by Microsoft): If you plan to write Python code.
    - **ESLint**: For JavaScript and Typescript linting.
    - **Prettier - Code formatter**: For consistent code formatting.
    - **GitLens**: Enhances Git capabilities in VS Code.
    - **Live Server**: Launch a local development server with live reload feature for static and dynamic pages.
    - **IntelliCode**: AI-assisted code completions.



**Version Control:**

* **Configure Git:**
  + If you use Git, ensure it is configured correctly. Install Git if it is not already installed. Set your user name and email:



### **Use the Source Control tab in VS Code to manage your repositories. Optimizing the Development Environment**

1. **Workspace Settings:**
   * **Create Workspace:**
     + Create a workspace for your project by going to File > Add Folder to Workspace. This helps in managing project-specific settings.
2. **Snippet Management:**
   * **Configure User Snippets:**
     + Create and manage code snippets by going to File > Preferences > User Snippets. This can help in boosting productivity by automating repetitive code patterns.
3. **Code Navigation:**
   * **Enable Breadcrumbs:**
     + Enable breadcrumbs for easier code navigation by searching for "breadcrumbs" in the settings and enabling it.
4. **Error Checking and Debugging:**
   * **Configure Debugging:**
     + Set up debugging configurations for your language or framework by going to the Debug icon on the sidebar and clicking on create a launch .json file

3**.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.**

The main components of the VS Code user interface are the Activity Bar, Side Bar, Editor Group, and Status Bar. The Activity Bar, located on the far left, provides quick access to core features like Explorer, Search, Source Control, Run and Debug, and Extensions. The Side Bar, next to the Activity Bar, displays detailed views and controls for the selected activity, such as a file tree in Explorer or commit details in Source Control. The Editor Group is the central area where you write and view code, supporting multiple open files in tabs and allowing for split views. The Status Bar at the bottom provides real-time information about the current file, Git branch, notifications, and quick access to settings. Together, these components create an efficient and customizable coding environment in VS Code.

**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.**

### **Extensions for Web Development**

1. **HTML, CSS, and JavaScript Support:**
   * **HTML Snippets:** Provides a set of code snippets for HTML development.
   * **CSS Peek:** Allows you to see CSS definitions by peeking into CSS files or within style blocks.
2. **JavaScript and Frameworks:**
   * **ESLint:** Integrates ESLint into VS Code for identifying and fixing issues in JavaScript code.
   * **Prettier - Code formatter:** An opinionated code formatter that supports various languages and integrates well with ESLint.
   * **npm:** Helps manage npm scripts and dependencies directly from within VS Code.
3. **Version Control:**
   * **GitLens:** Enhances the built-in Git capabilities, providing insights into code changes and history.
4. **Productivity Tools:**
   * **Live Server:** Launches a local development server with live reload for static and dynamic pages.
   * **Debugger for Chrome:** Allows debugging of JavaScript code in the Google Chrome browser.
   * **Path Intellisence:** Autocompletes filenames as you type, providing suggestions for file paths.
5. **Framework-Specific Extensions:**
   * **React Native Tools:** Provides a range of tools for developing with React Native.
   * **Angular Language Service:** Offers rich editing support for Angular templates and Typescript.

### **Examples of Popular Extensions for Web Development**

1. **HTML, CSS, and JavaScript Support:**
   * **HTML Snippets:** Provides code snippets for HTML development.
   * **CSS Peek:** Allows viewing of CSS definitions directly from HTML files.
2. **JavaScript and Frameworks:**
   * **ESLint:** Integrates ESLint into VS Code for identifying and fixing issues in JavaScript code.
   * **Prettier - Code Formatter:** An opinionated code formatter that supports various languages and integrates well with ESLint.
   * **npm:** Helps manage npm scripts and dependencies directly from within VS Code.
3. **Version Control:**
   * **GitLens:** Enhances the built-in Git capabilities, providing insights into code changes and history.
4. **Productivity Tools:**
   * **Live Server:** Launches a local development server with live reload for static and dynamic pages.
   * **Debugger for Chrome:** Allows debugging of JavaScript code in the Google Chrome browser.
   * **Path Intellisense:** Autocompletes filenames as you type, providing suggestions for file paths.
5. **Framework-Specific Extensions:**
   * **React Native Tools:** Provides a range of tools for developing with React Native.
   * **Angular Language Service:** Offers rich editing support for Angular templates and TypeScript.

**6.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? Opening and Using the Integrated Terminal in VS Code**

1. **Opening the Terminal:**
   * Press Ctrl+`` or go to View>Terminal` to open the integrated terminal.
   * You can also open multiple terminals by clicking the + icon in the terminal tab.
2. **Using the Terminal:**
   * Execute shell commands, run scripts, and manage tasks directly within VS Code.
   * Use the drop-down menu to switch between different terminal instances or select different shell types (e.g., Command Prompt, PowerShell, Git Bash).

**Advantages of Using the Integrated Terminal**

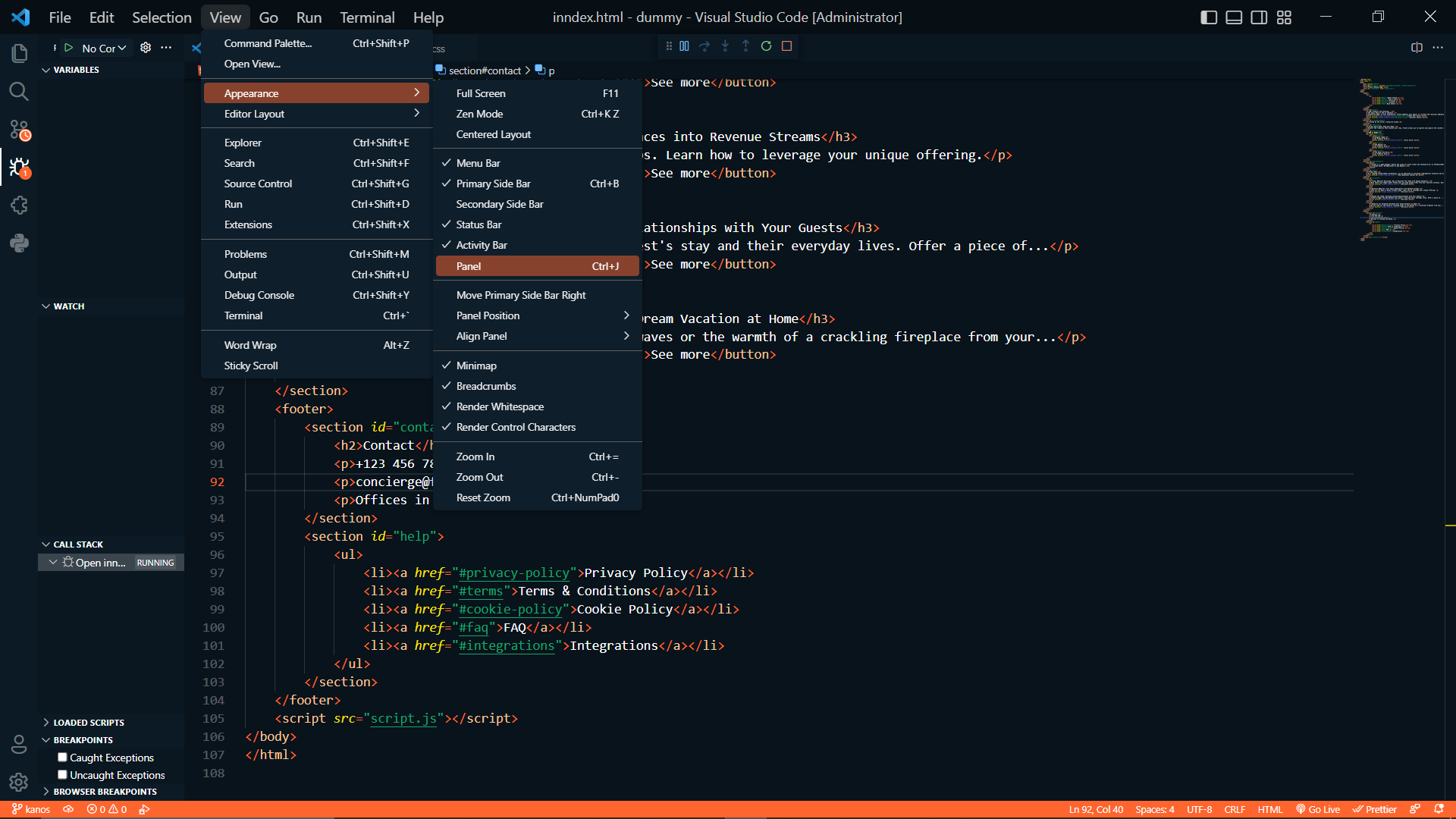
1. **Seamless Workflow:**
   * Code and terminal commands coexist in one window, eliminating the need to switch between applications.
2. **Context Awareness:**
   * Automatically opens in the project’s root directory, providing direct access to project files and environments.
3. **Synchronization:**
   * Share environment variables and paths between the editor and terminal, ensuring consistent behavior.
4. **Customization:**
   * Customize the appearance and behavior of the terminal through VS Code settings.
5. **Task Integration:**
   * Easily run and manage tasks defined in tasks.json, streamlining build and deployment processes.

**7.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?**

### **Efficient Navigation**

1. **Quick Open:**
   * Press Ctrl+P to quickly open files by typing their names.
2. **Explorer Sidebar:**
   * Navigate the folder structure and open files by clicking on them in the Explorer sidebar.
3. **Breadcrumbs:**
   * Enable breadcrumbs (View > Appearance > Show Breadcrumbs) to navigate between directories and files at the top of the editor.
4. **Keyboard Shortcuts:**
   * Use Ctrl+Tab to switch between open files and Ctrl+Shift+E to focus on the Explorer sidebar.
5. **Go to Definition:**
   * Right-click on a symbol and select Go to Definition or press F12 to navigate to the symbol's definition.

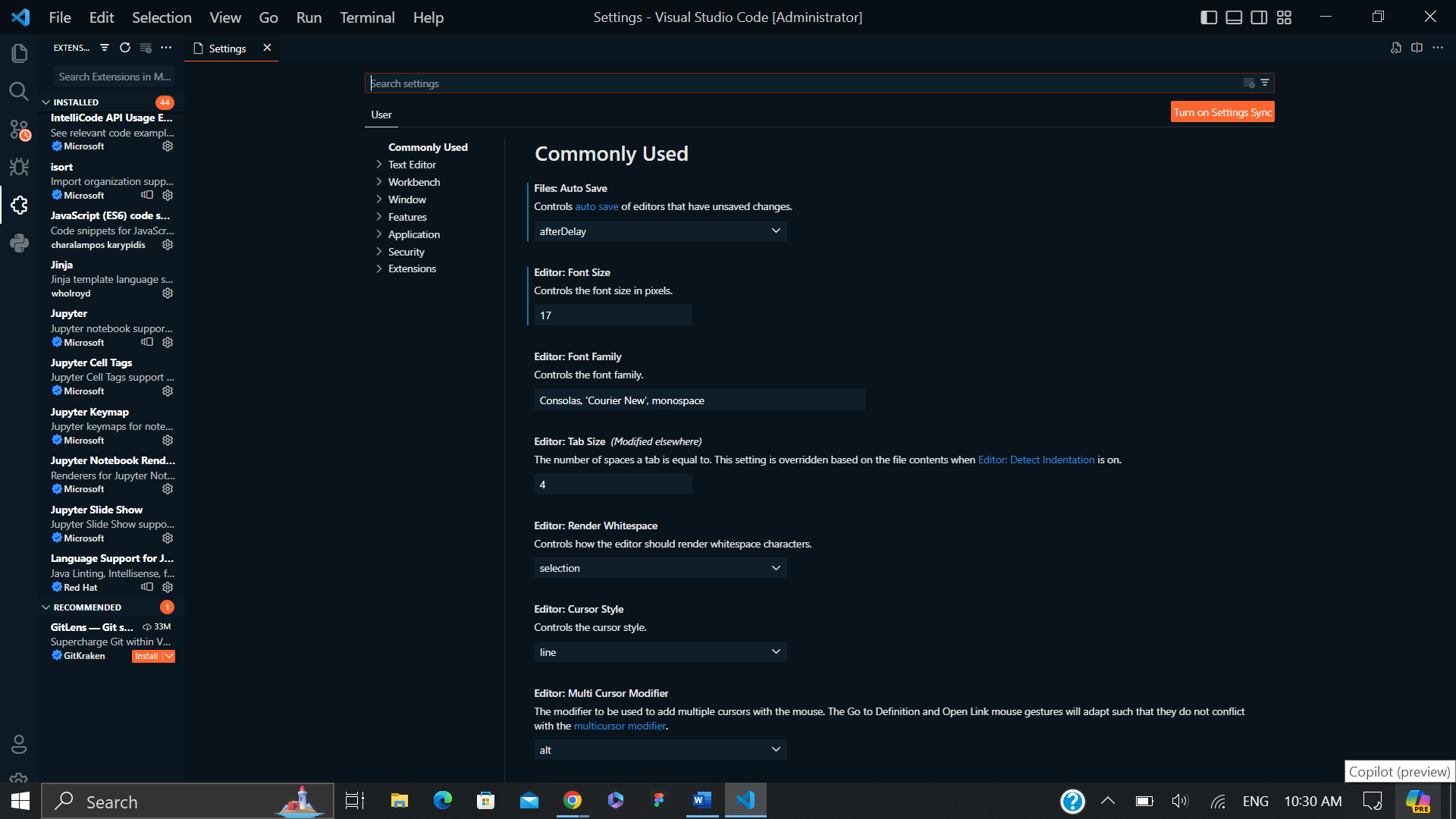


**8.Settings and Preferences:**

**Where can users find and customize settings in VS Code? Provide examples of how to change the theme, font size, and keybindings.**

### **Finding and Customizing Settings in VS Code**

1. **Accessing Settings:**
   * Open settings by going to File > Preferences > Settings or pressing Ctrl+,.
2. **Changing the Theme:**
   * Search for "Color Theme" in the settings and select a theme from the dropdown menu.
   * Alternatively, press Ctrl+K Ctrl+T to open the theme selection menu.
3. **Changing Font Size:**
   * In the settings, search for "Font Size" and adjust the value to your preferred size.
4. **Customizing Keybindings:**
   * Open File > Preferences > Keyboard Shortcuts or press Ctrl+K Ctrl+S.
   * Find the command you want to change, click the pencil icon, and press the desired key combination to set a new keybinding.

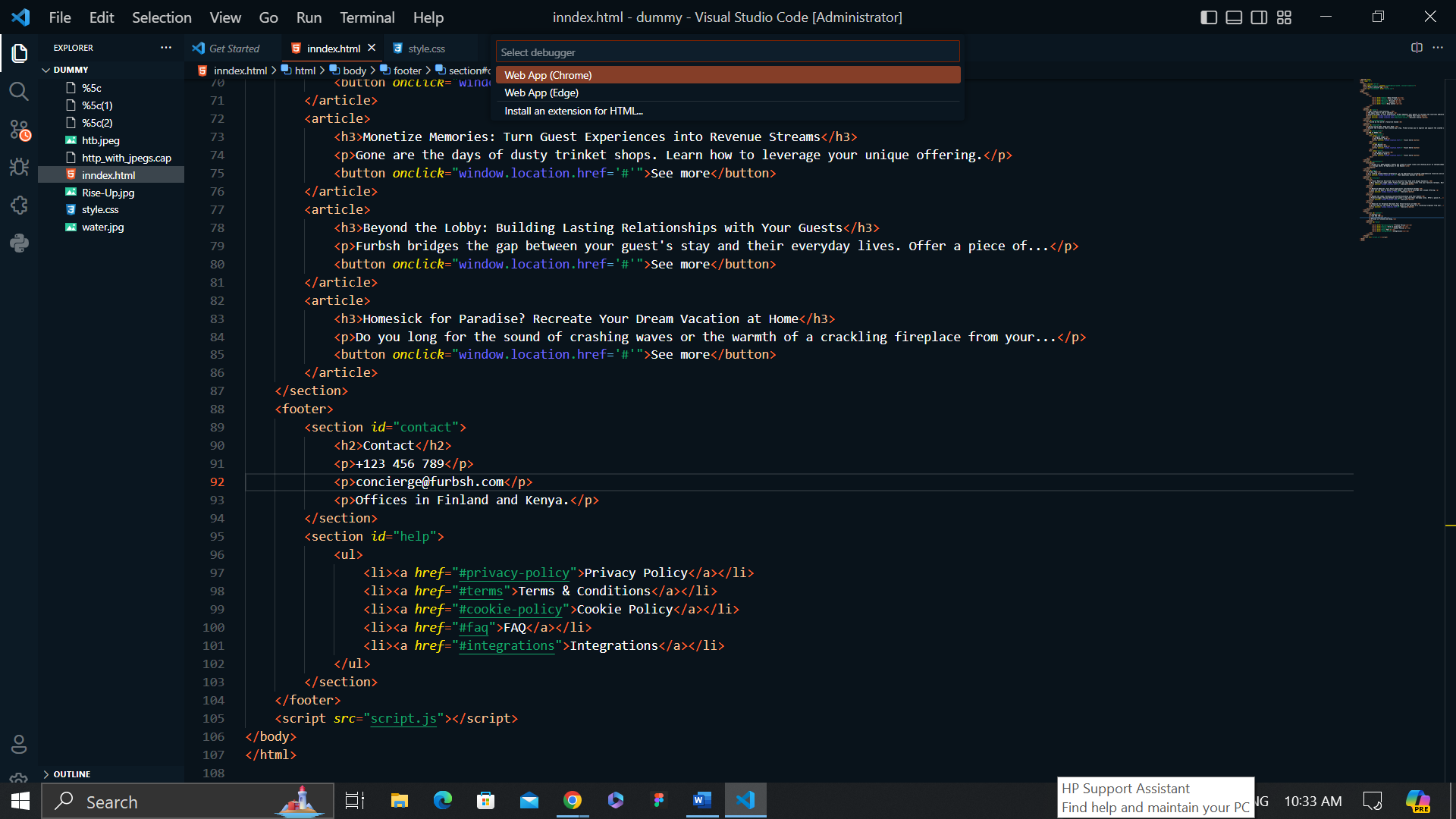


**Debugging in VS Code:**

**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 in VS Code**

1. **Open Your Project:**
   * Open your project folder in VS Code.
2. **Set Up the Debug Configuration:**
   * Go to the Run and Debug view by clicking the Run icon in the Activity Bar or pressing Ctrl+Shift+D.
   * Click create a launch.json file and select the environment (e.g., Node.js, Python).
3. **Set Breakpoints:**
   * Click in the gutter to the left of the line numbers in your code to set breakpoints.
4. **Start Debugging:**
   * Press the green play button in the Run and Debug view or press F5.



### Key Debugging Features in VS Code

1. **Breakpoints:**
   * Set breakpoints to pause code execution at specific lines.
2. **Watch Expressions:**
   * Monitor the value of expressions and variables.
3. **Call Stack:**
   * View the call stack to understand the sequence of function calls.
4. **Variable Inspection:**
   * Inspect variables' current values in the Variables panel.
5. **Step Controls:**
   * Use step over, step into, and step out controls to navigate through code execution.
6. **Debug Console:**
   * Execute commands and evaluate expressions in the context of the debugging session.

**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 VS Code for Version Control**

1. **Initialize a Repository:**
   * Open your project folder in VS Code.
   * Click the Source Control icon in the Activity Bar or press Ctrl+Shift+G.
   * Click Initialize Repository in the Source Control view.
2. **Make Changes and Stage Files:**
   * Edit your files as needed.
   * In the Source Control view, click the + icon next to each changed file to stage it.
3. **Commit Changes:**
   * Enter a commit message in the input box at the top of the Source Control view.
   * Click the checkmark icon to commit the staged changes.
4. **Push Changes to GitHub:**
   * Open the integrated terminal by pressing `Ctrl+``.
   * Add a remote repository: git remote add origin https://github.com/yourusername/your-repo.git.
   * Push the changes: git push -u origin master.

How can users integrate Git with VS Code for version control? Describe the process of initializing a repository, making commits, and pushing changes to GitHubTop of Form

Bottom of Form

Top of Form

Bottom of Fo

Top of Form

Bottom of Form

**Top of Form**

**Bottom of Form**