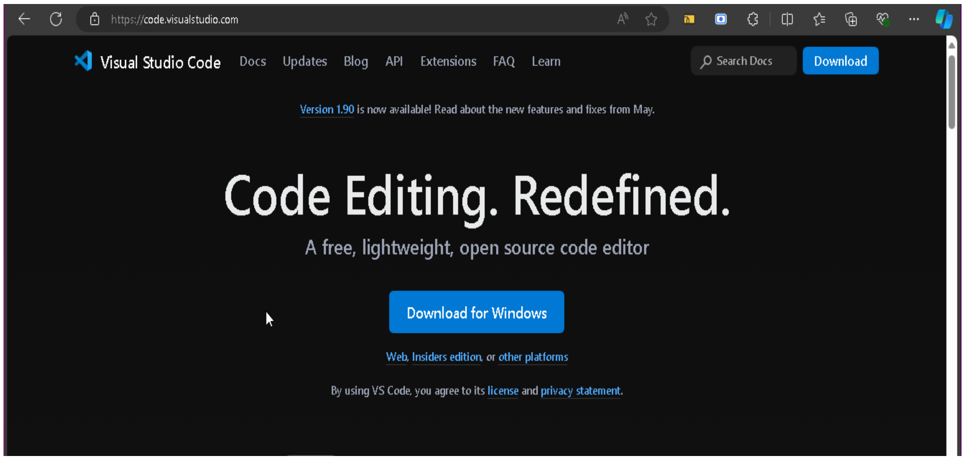
Installation and Navigation of Visual Studio Code (VS Code)

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

1. **Downloading the Installer:**
   a) Visit the Visual Studio Code website https://code.visualstudio.com/



Version 1.90 is now available! Read about the new features and fixes from May.

Code Editing. Redefined.

A free, lightweight, open source code editor

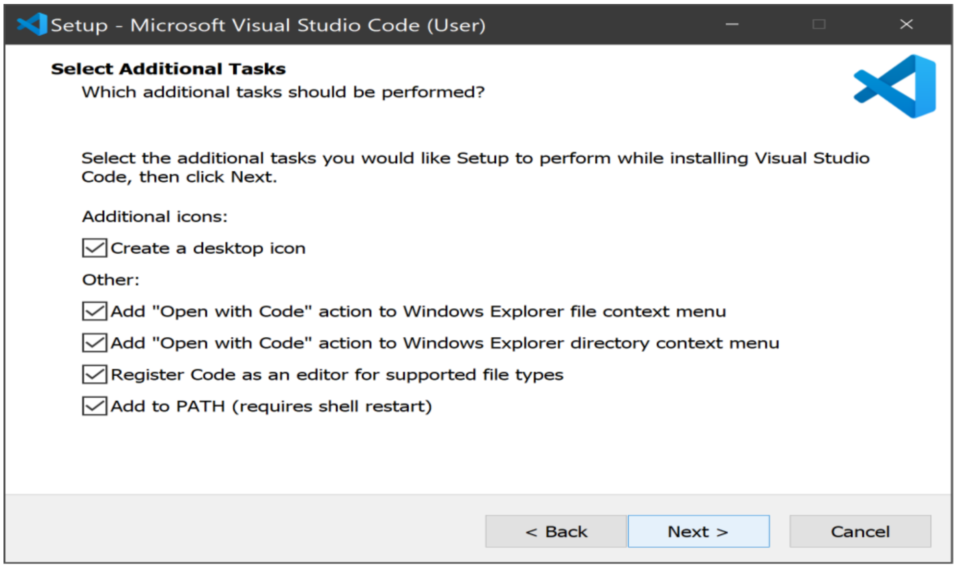
Web, Insiders edition, or other platforms

By using VS Code, you agree to its license and privacy statement.

[image 1]

   b) Click on the '**Download for Windows**' or click '**Other platforms**' to download VSCode compatible with your device Operating System.

2. **Run the Installer:**
   a) Open the downloaded installer file from the Downloads folder, if you don't find it, check in the Programs folder inside the downloads folder.
   b) Follow the setup wizard, accepting the license agreement and selecting the installation location.
   c) **Select Additional Tasks:**
      - Choose to **add VS Code to your PATH** for command line access.
      - Optionally, create a desktop icon and **associate VS Code with supported file types**.



Setup - Microsoft Visual Studio Code (User)

Select Additional Tasks

Which additional tasks should be performed?

Select the additional tasks you would like Setup to perform while installing Visual Studio Code, then click Next.

Additional icons:

☑ Create a desktop icon

Other:

☑ Add "Open with Code" action to Windows Explorer file context menu

☑ Add "Open with Code" action to Windows Explorer directory context menu

☑ Register Code as an editor for supported file types

☑ Add to PATH (requires shell restart)

[image 2]

d) **Complete the Installation:**

- Click 'Install' and wait for the installation to finish.
- Launch VS Code directly from Desktop or via the Start menu.

Prerequisites:

- **Administrator Privileges for Installation**: You need to have administrator privileges on your computer to install VSCode. Administrator privileges allow you to make changes to the system, such as installing software, which might require modifying system files or settings. Without these privileges, you may not be able to complete the installation process.
- **Preferably use windows 11, for better support, and functions.**

2. After installing VS Code, what initial configurations and settings should be adjusted for an optimal coding environment? Mention any important settings or extensions.

Initial Configurations and Settings:

1. **Theme and Appearance:**

a) Go to File > Preferences > Color Theme to select a preferred theme (e.g., Dark+, Light+). I would advise you to choose dark+ theme in order to avoid exposing your eyes to too much of bright light.

2. **Extensions:**

a) Access the Extensions view by clicking the Extensions icon on the Activity Bar or pressing Ctrl+Shift+X.

b) **Essential extensions for web development include:**

- GitHub Pull Requests – makes it easy to pull and push file to GitHub repository.
- ESLint for JavaScript and TypeScript linting.
- HTML CSS Support – CSS Intellisense for HTML.
- Prettier - Code formatter for code formatting.
- Live Server for live-reloading local server.

NB: The Extensions you will install will be determined by the programming languages you will be working with.

3. **Settings Sync:**

a) Enable Settings Sync to synchronize your settings across multiple devices by going to File > Preferences > Settings Sync and signing in.

4. **Editor Configurations:**

a) Adjust editor settings by navigating to File > Preferences > Settings.

b) Common settings include changing font size, enabling line numbers, turn off detect indentation, and configuring auto-save.

3. 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. **Activity Bar:**
   - Located on the far left side, it provides quick access to different views such as Explorer, Search, Source Control, Run and Debug, Extensions, and some installed Extensions are displayed on the Activity Bar if they are enabled e.g. GitHub Pull Requests.
2. **Side Bar:**
   - Adjacent to the Activity Bar, it displays the contents of the selected view. E.g. the Explorer view shows the folder and file structure.
3. **Editor Group:**
   - The central area where files are opened and edited. Multiple editor groups can be created for side-by-side editing.
4. **Status Bar:**

- It’s Located at the bottom, it provides information about the current file and workspace, such as encoding, format, the Venv selected for python, line an column the typing cursor is at, and the active Git branch, E.T.C.

4. 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 is a tool for executing commands and navigating within VS Code. It is access by pressing Ctrl+Shift+P (or F1).

Examples of Common Tasks:

- Clear command history: Type in the input box [clear command history]
- Changing the color theme: Type in the input box [Color Theme] and select a theme.
- Opening settings: Type in the input box [Preferences: Open Settings].
- Installing extensions: Type in the input box [Extensions: Install Extensions].

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

1. **Role:**
   - Extensions enhance the functionality of VS Code by adding support for additional languages, tools, debuggers, and features.
2. **Finding and Installing Extensions:**
   - Click the Extensions icon on the Activity Bar or press Ctrl+Shift+X.
   - Search for the desired extension and click 'Install'.
3. **Managing Extensions:**
   - Disable or uninstall extensions from the Extensions view.
4. **Essential Extensions for Web Development:**
   - HTML Snippets.
   - Live Server.
   - Better Comments.
   - Live Sass Compiler.

6. 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?

1. **Opening the Terminal:**
   - Go to View > Terminal or press Ctrl+`.
2. **Using the Terminal:**
   - The terminal supports multiple instances and types such as PowerShell, Command Prompt, and Git Bash.
3. **Advantages:**
   - Seamless workflow within the editor.
   - Integrated environment with access to project files and version control.

7. 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 File > Open File or File > Open Folder.
   - Drag and drop files/folders into the editor.
3. **Navigating Between Files:**
   - Use the Explorer view or the Ctrl+P shortcut to quickly open files by name.

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

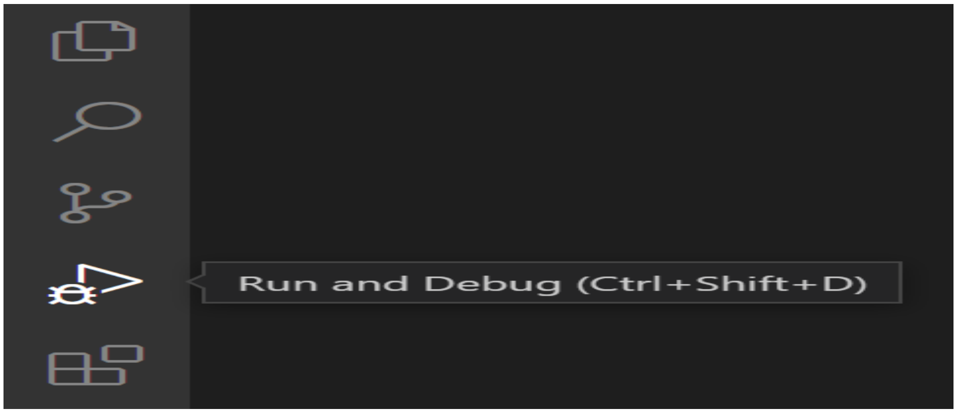
1. **Finding Settings:**
   - Go to File > Preferences > Settings or press Ctrl+.
   - From the Command Palette (Ctrl+Shift+P) and Type [Preferences: Open Settings.
2. **Examples of Customizations:**
   - Theme: Click File at the top of the Editor > Preferences > Color Theme.

- AutoSave: Click File at the top of the Editor > Click Auto Save
- Font Size: Adjust in Settings under 'Editor: Font Size'.
- Keybindings: Click File at the top of the Editor > Preferences > Keyboard Shortcuts.

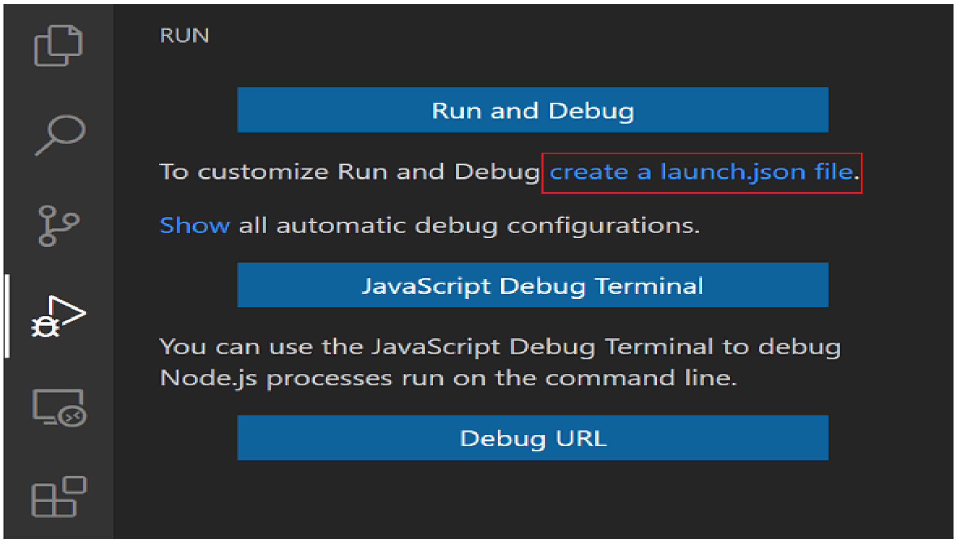
9. Outline the steps to set up and start debugging a simple program in VS Code. What are some key debugging features available in VS Code?

Setting Up and Starting Debugging:

1. **Open the Debug View:**
    - Click the Run and Debug icon on the Activity Bar or press Ctrl+Shift+D.



2. **Configure Debugging:**
    - Click 'create a launch.json file' in the Run Start view to configure the debugger for your project.



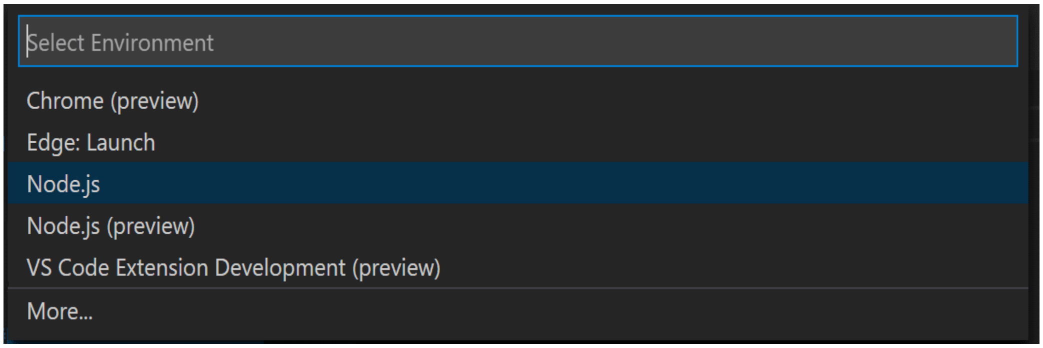
To customize Run and Debug create a launch.json file.

Show all automatic debug configurations.

You can use the JavaScript Debug Terminal to debug Node.js processes run on the command line.

3. **Start Debugging:**

- Set breakpoints by clicking in the gutter next to the line numbers.
- Select Run and Debug on the Debug start view or Click the play button or press F5 to start debugging.
- VSCode will try to automatically detect your debug environment, but if it doesn't you will have to choose manually.



Key Debugging Features:

- Breakpoints, multi-threaded debugging, expression evaluation, memory dump debugging, step over/into, watch expressions, and variable inspection.

10. 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. **Initializing a Repository:**
   - Open your project folder in VSCode. If it's a new project, you can create the folder first.
     a) Look for the "Source Control" tab (it appears like a fork in the road icon) on the left Activities bar.
     b) If VSCode doesn't detect an existing Git repository, it will display an "Initialize Repository" button. Click on it.

   NB: This action is similar to running git init in the command line and creates the necessary files for Git to track your project's history.

1. **Making Commits:**
   - Edit your files as usual. VS Code will visually indicate changes (modified lines, new files, etc.) in the Source Control tab.
   - Stage specific changes you want to include in your commit using the "+" button next to the file or using the "Stage Changes" option in the context menu.
   - Once you're ready to commit, open the Source Control tab again.
   - Click on the checkmark icon or use the "Commit" option.

    - Add a descriptive commit message explaining the changes you made.
    - Hit Enter to create the commit.

2. **Pushing Changes to GitHub:**
    - Make sure you have a GitHub account and a repository created for your project.
    - You can sign into VS Code with your GitHub account for a more streamlined workflow (optional).
    - In the Source Control tab, look for the branch name displayed on the bottom left corner of the status bar.
    - Click on the button next to the branch name and select "Publish branch".
    - Alternatively, you can use the "Git: Push" command in the Command Palette (Ctrl+Shift+P).

    NB: This will initiate the process of pushing your committed changes to the remote repository on GitHub. You might need to provide your GitHub credentials during the first push.

Referrences:

- VS Code documentation on Git: Visual Studio Code Tips and Tricks
- VSCode Debugging: Debugging in Visual Studio Code, Debug C++ in Visual Studio Code
- Video tutorial on using Git with VS Code: (1057) How to setup and use GitHub with Visual Studio Code [2024] - YouTube