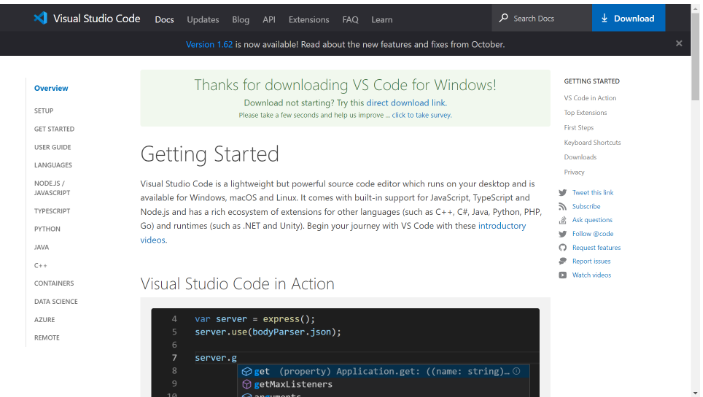
Installation of VS Code

Steps to Download and Install Visual Studio Code on Windows 1

1. Download Visual Studio Code

- Open your web browser and navigate to the [Visual Studio Code website](https://code.visualstudio.com/).

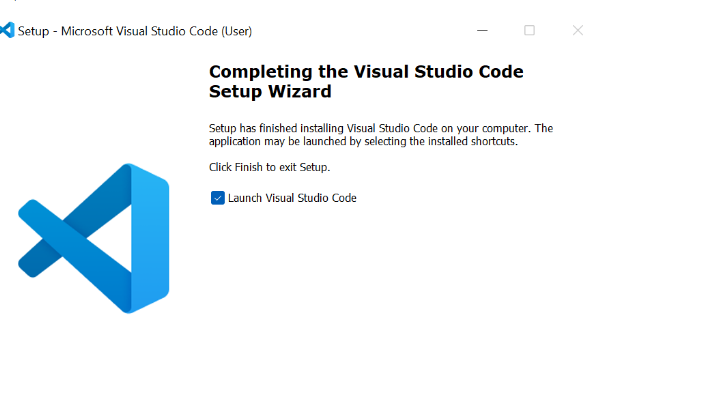
- Click on the "Download for Windows" button to download the VS Code installer for Windows 11.



2. Run the Installer:

- Once the download is complete, open the downloaded `.exe` file.

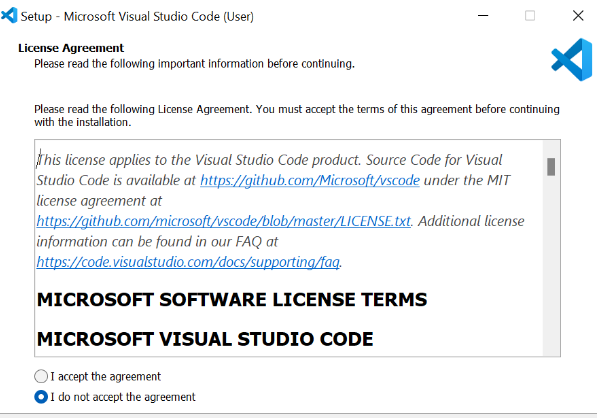
- The Visual Studio Code Setup Wizard will open. Click "Next" to continue.



3. Accept License Agreement:

- Read the license agreement and click "I accept the agreement" to proceed.

- Click "Next."



4. Choose Installation Location:

- Select the destination folder where you want to install VS Code or leave it as the default.

- Click "Next."

5. Select Additional Tasks:

- Check the boxes for additional tasks such as creating a desktop icon, adding to the PATH, and registering code as an editor for supported file types.

- Click "Next."

6. Install:

- Click "Install" to begin the installation process.

- Once the installation is complete, click "Finish" to exit the setup wizard. Optionally, you can launch VS Code immediately by checking the "Launch Visual Studio Code" box.

Prerequisites:

- Windows 11 operating system.

- Administrator privileges to install software.

- An internet connection to download the installer.

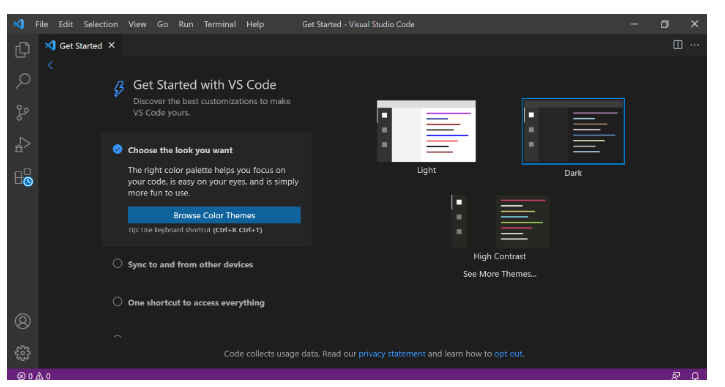
First-time Setup

Initial Configurations and Settings for an Optimal Coding Environment:

1. Theme and Appearance:

- Open VS Code.

- Go to `File > Preferences > Color Theme` or use the shortcut `Ctrl+K Ctrl+T` to choose a preferred color theme.



2. Font Size and Family:

- Go to `File > Preferences > Settings` or use `Ctrl+,`.

- Search for `font size` and adjust the font size to your preference.

- Search for `font family` to change the font style if needed.

3. Extensions:

- Click on the Extensions icon in the Activity Bar on the side of the window or use `Ctrl+Shift+X`.

- Install essential extensions like:

- `Prettier - Code Formatter` for consistent code formatting.

- `ESLint` for JavaScript linting.

- `Live Server` for a local development server with live reload feature.

- `Python` for Python development.

- `GitLens` for enhanced Git capabilities.

User Interface Overview

Main Components of the VS Code User Interface:

1. Activity Bar:

- Located on the far left side of the window.

- Provides access to different views such as Explorer, Search, Source Control, Run and Debug, and Extensions.

2. Side Bar:

- Adjacent to the Activity Bar.

- Displays different panels based on the selected view from the Activity Bar, such as the file explorer, search results, or extensions list.

3. Editor Group:

- Central area where you open and edit files.

- Supports multiple tabs and split views to work on different files simultaneously.

4. Status Bar:

- Located at the bottom of the window.

- Displays information about the current workspace, such as the current Git branch, line and column numbers, and language mode.

Command Palette

Command Palette in VS Code:

- Accessed by pressing `Ctrl+Shift+P` or `F1`.

- Allows you to execute any command without navigating through menus.

- Examples of common tasks:

- `>Open File` to open files quickly.

- `>Git: Clone` to clone a repository.

- `>View: Toggle Terminal` to open/close the integrated terminal.

Extensions in VS Code

Role of Extensions in VS Code:

- Extensions enhance the functionality of VS Code.

- Users can find and install extensions from the Extensions view (`Ctrl+Shift+X`).

- Manage installed extensions from the same view.

Examples of Essential Extensions for Web Development:

- `Prettier - Code Formatter`

- `ESLint`

- `Live Server`

- `Debugger for Chrome`

- `HTML CSS Support`

Integrated Terminal

Opening and Using the Integrated Terminal:

1. Open Integrated Terminal:

- Use the shortcut `Ctrl+`` (backtick) or navigate to `View > Terminal`.

2. Advantages:

- Directly run commands, scripts, and version control commands within the same window.

- Supports multiple terminal instances and split views.

File and Folder Management

Creating, Opening, and Managing Files and Folders:

1. Creating Files and Folders:

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

- Use `File > New File` or `File > New Folder`.

2. Opening Files:

- Double-click a file in the Explorer view.

- Use `Ctrl+O` to open a file dialog.

3. Navigating Between Files and Directories

- Use `Ctrl+P` to quickly open files by name.

- Use the breadcrumbs at the top of the Editor Group for navigation.

Settings and Preferences

Customizing Settings in VS Code:

1. Accessing Settings:

- Go to `File > Preferences > Settings` or use `Ctrl+,`.

2. Examples of Customizations:

- Change Theme

- Search for `Color Theme` and select from the available options.

-Change Font Size:

- Search for `font size` and adjust the value.

- Change Keybindings:

- Go to `File > Preferences > Keyboard Shortcuts` or use `Ctrl+K Ctrl+S`.

Debugging in VS Code

Setting Up and Starting Debugging:

1. Open a Program File:

- Open the file you want to debug.

2. Set Breakpoints:

- Click in the gutter next to the line number to set breakpoints.

3. Configure Debugger:

- Go to `Run > Add Configuration...` and select the appropriate environment (e.g., Node.js, Python).

4. Start Debugging:

- Press `F5` or go to `Run > Start Debugging`.

Key Debugging Features:

- Watch expressions.

- Call stack.

- Variables.

- Debug console.

Using Source Control

Integrating Git with VS Code:

1. Initialize a Repository:

- Open the Source Control view from the Activity Bar.

- Click `Initialize Repository` in the Source Control panel.

2. Making Commits:

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

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

3. Pushing Changes to GitHub:

- Open the Command Palette (`Ctrl+Shift+P`), type `Push`, and select `Git: Push`.

- If it's the first time, you'll need to set up the remote repository URL.