

Set up a Python development environment and write your first Python script

Table of Contents

- [Introduction](#)
 - [Problem Statement](#)
 - [Prerequisites](#)
 - [SoftwareRequirement](#)
 - [Hardware Requirement](#)
 - [Setting up a Python Environment](#)
 - [Step 1: Download the Python Installer](#)
 - [Step 2: Run the Installer](#)
 - [Step 3: Choose the Optional Installation Features](#)
 - [Step 4: Choosing Advanced Options](#)
 - [Step 5: Final Setup](#)
 - [Step 6: Add Python to Path \(Optional\)](#)
 - [Step 7: Verify Python Was Installed on Windows](#)
 - [Step 8: Connect VS Code with Python](#)
 - [Installing Visual Studio Code](#)
 - [Install the Python Extension for VS Code](#)
 - [Create a Python File and Write the Script](#)
 - [Run Your Python Script](#)
 - [References](#)
-

Introduction

This guide will walk you through setting up a Python development environment on a Windows machine and writing your first Python script. The aim is to ensure that you have the essential tools installed and know how to run Python programs effectively.

Problem Statement

Setting up a Python environment on Windows might seem challenging for beginners. This guide simplifies the process, covering installation, folder structure creation, package installation, and running a basic script.

Prerequisites

Software Requirement

Before proceeding, make sure you have the following software installed:

- **Python 3.13.0**
[Download Python](#)

- **Code Editor**

You can choose any code editor, but **Visual Studio Code (VS Code)** is recommended for its ease of use and Python support.

[Download VS Code](#)

- **Python packages (installed via `pip`)**

We will cover installation of necessary packages in a later step.

Hardware Requirement

You don't need high-end hardware to run Python, but make sure your machine meets the following basic requirements:

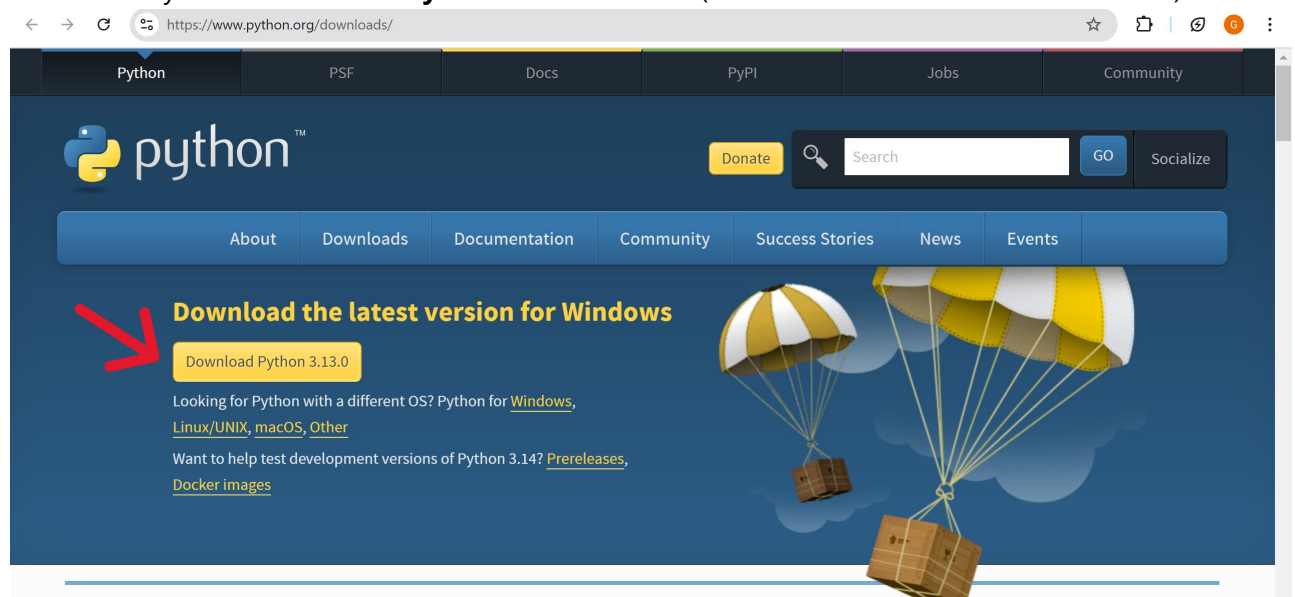
- **Processor:** 1 GHz or faster, 2-core CPU.
- **Memory:** 4 GB (8 GB recommended for better performance)
- **Storage:** 1 GB of free space for Python installation (additional space for projects and dependencies).
- **Operating System:** Windows 10 or higher.

Setting up a Python Environment

Step 1: Download the Python Installer:



- Visit the [official Python website](#).
- Click on the yellow "**Download Python 3.13.0**" button (the latest stable version will be shown).

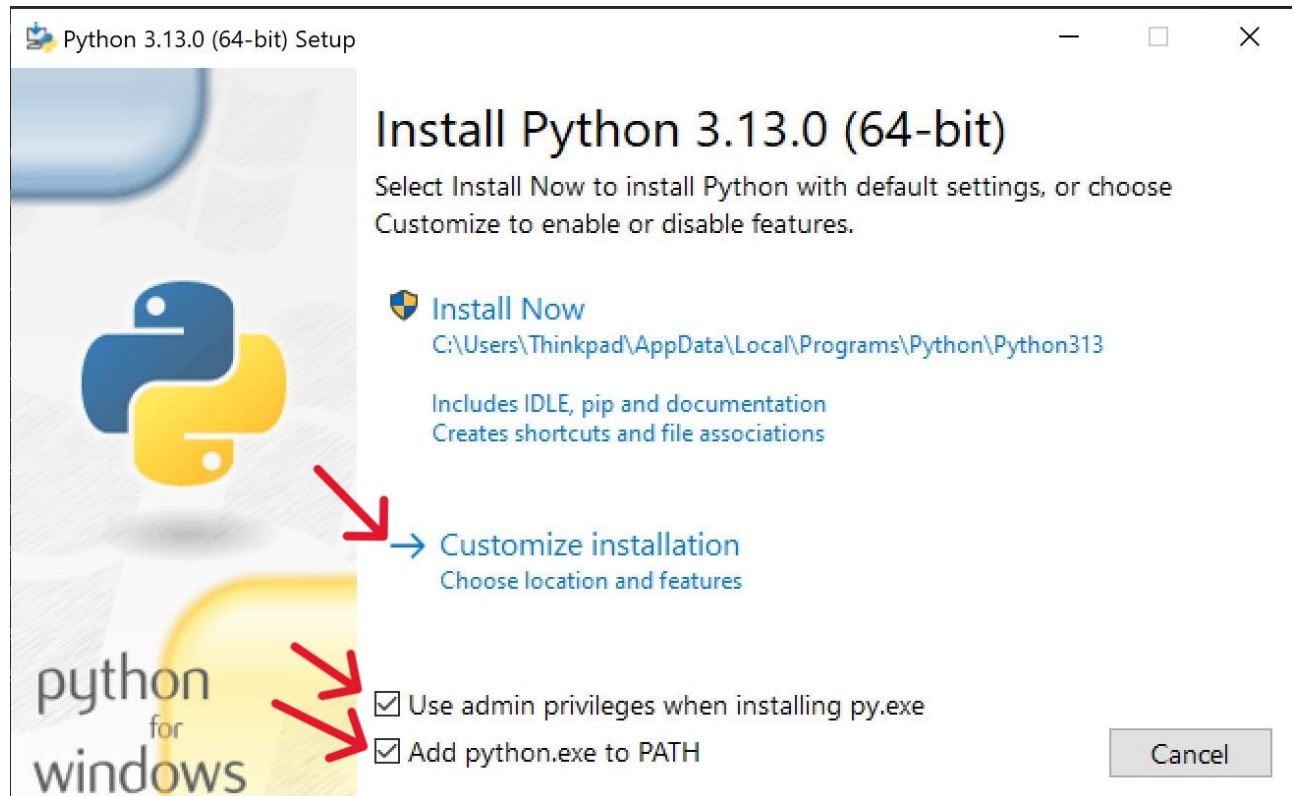


Step 2: Run the Installer:

1. Run the downloaded Python Installer.

2. The installation window shows two checkboxes:

- **Admin privileges:** The parameter controls whether to install Python for the current or all system users. This option allows you to change the installation folder for Python.
- **Add Python to PATH:** The second option places the executable in the PATH variable after installation. You can also add Python to the PATH environment variable manually later.



3. Select the **Install Now** option for the recommended installation (in that case, skip the next two steps).

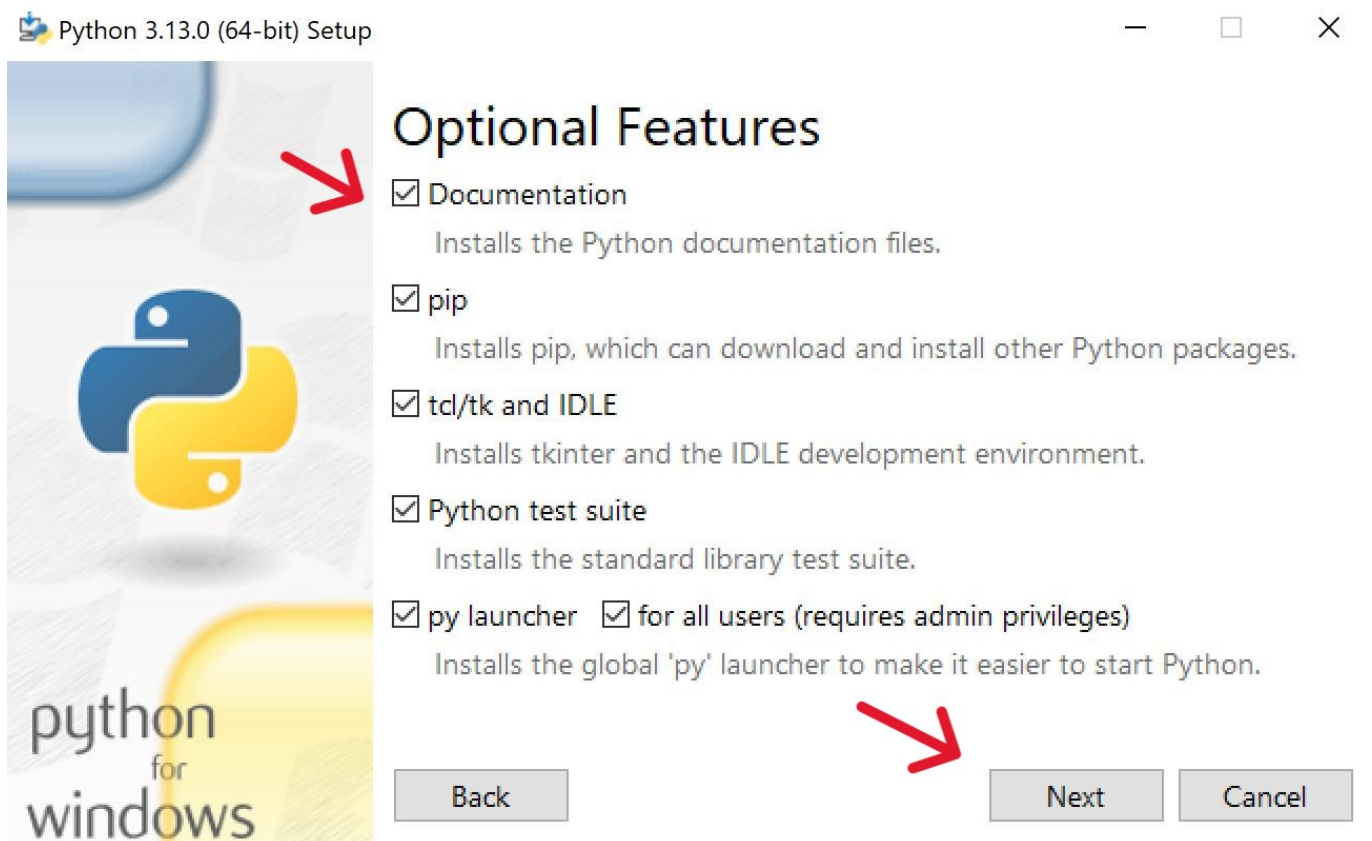
4. To adjust the default installation options, choose **Customize installation** instead and proceed to the following step.

- **Installation Directory:** `C:\Users\[user]\AppData\Local\Programs\Python\Python[version]`
- **Included Components:**
 - **IDLE** (the default Python Integrated Development and Learning Environment).
 - **PIP** (Python's package installer).
 - **Additional Documentation.**
- The installer also creates:
 - Required shortcuts.
 - File associations for `.py` files.

If you choose the "**Customize Installation**" option during setup, you can modify the default configurations, such as the installation location, optional features, and advanced settings. This flexibility allows you to tailor the setup to your specific project requirements or environment.

Step 3: Choose the optional installation features

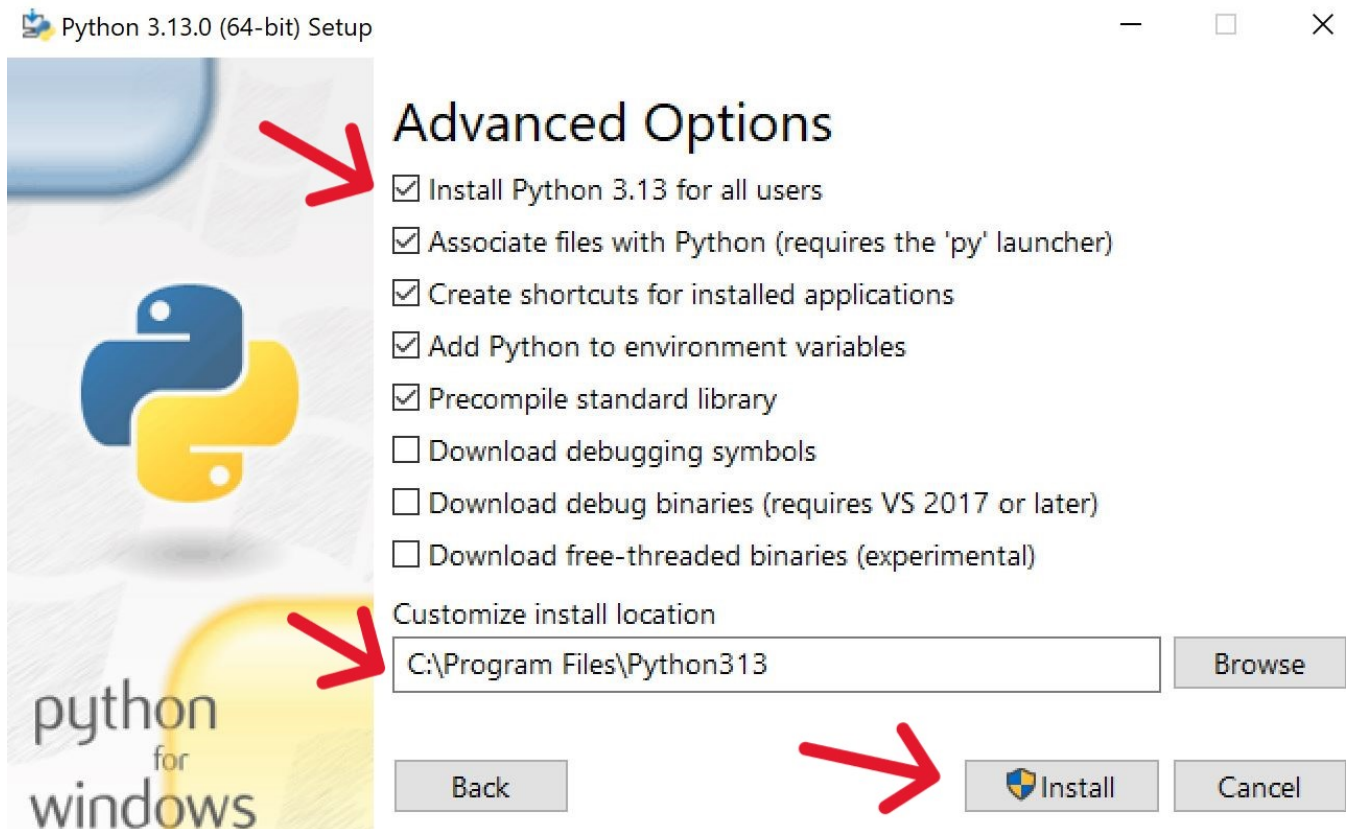
- Python works without these features, but adding them improves the program's usability.



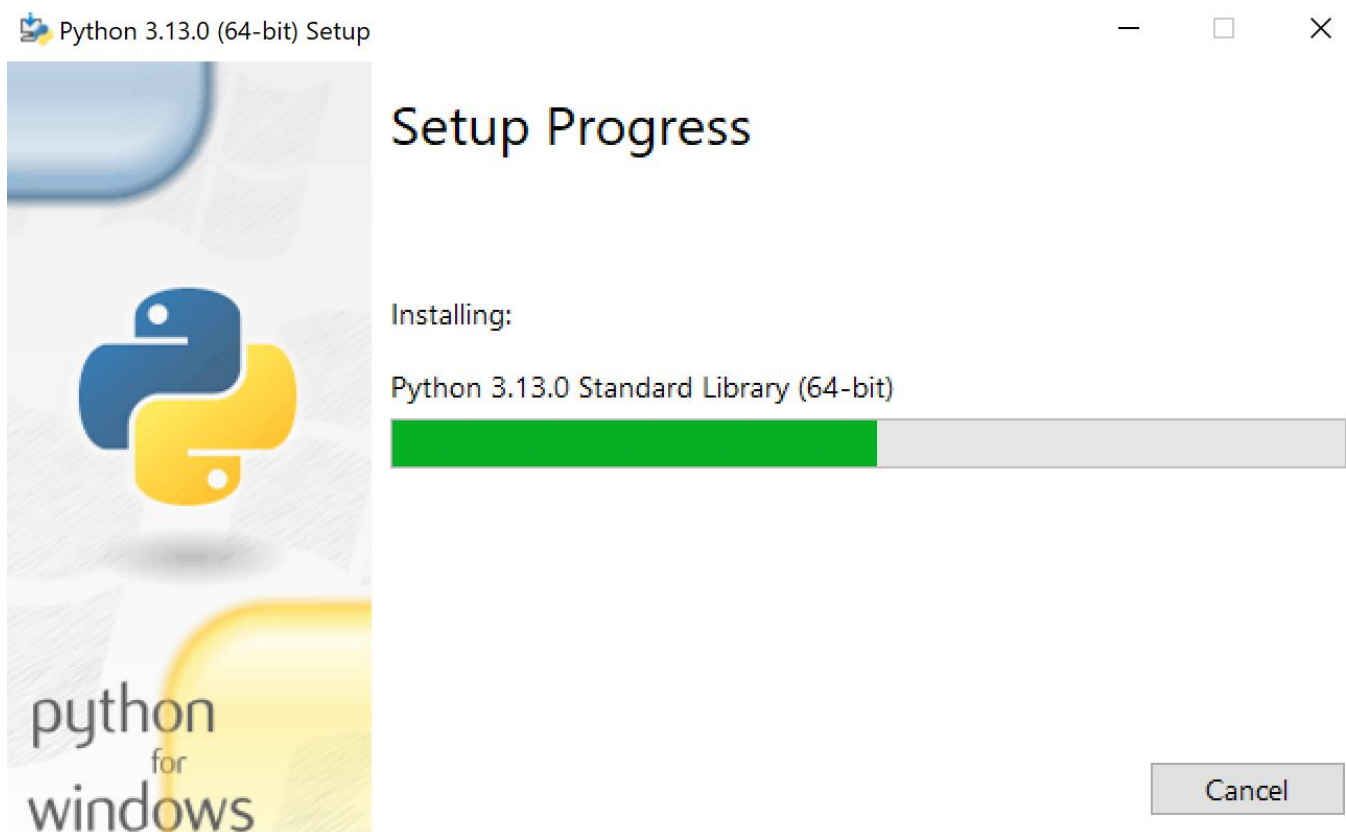
- Click Next to proceed to the Advanced Options screen.

Step 4: Choosing advanced options

- Choose whether to install Python for all users. The option changes the install location to C:\Program Files\Python[version].
- If selecting the location manually, a common choice is C:\Python[version] because it avoids spaces in the path, and all users can access it. Due to administrative rights, both paths may cause issues during package installation.

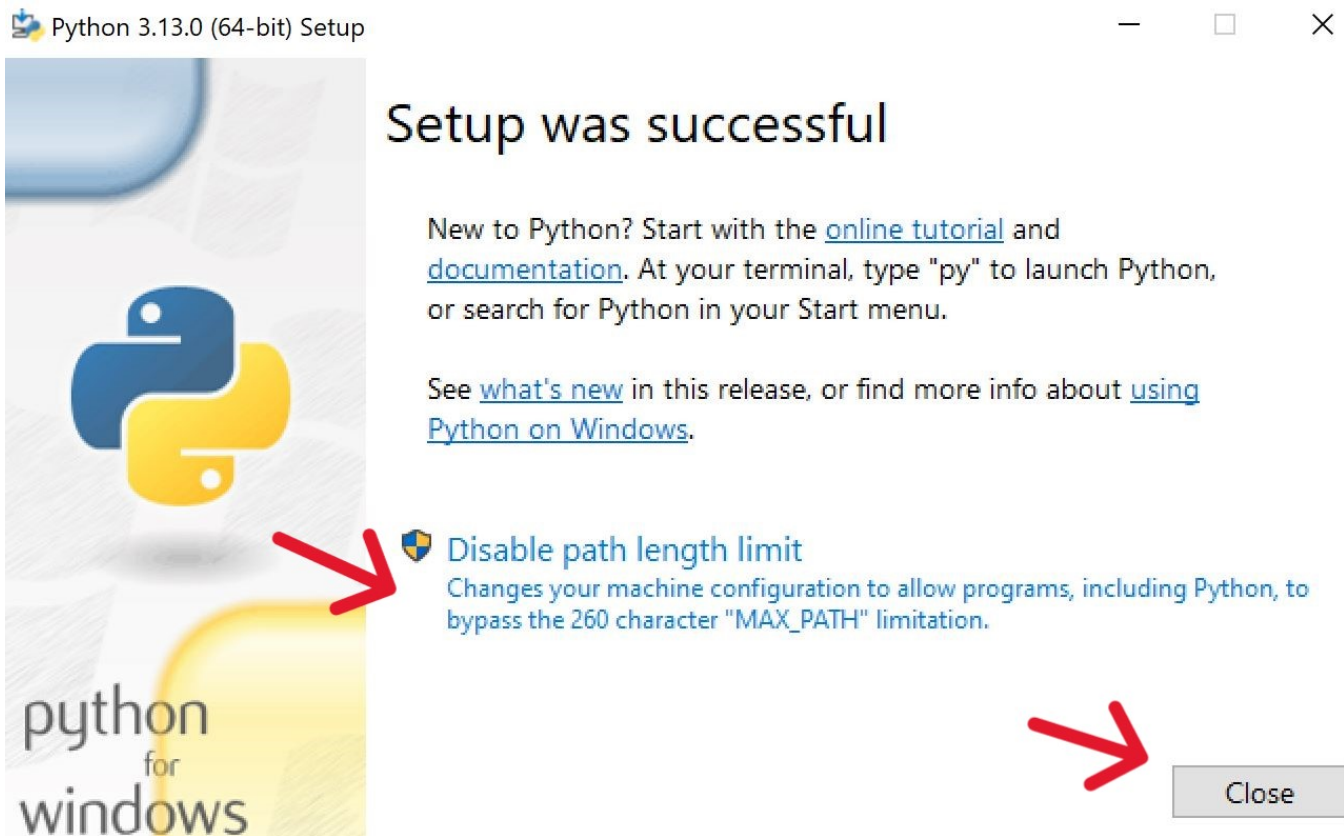


After picking the appropriate options, click Install to start the installation.



Step 5: Final Setup

- Select whether to disable the path length limit. Choosing this option will allow Python to bypass the 260-character MAX_PATH limit.



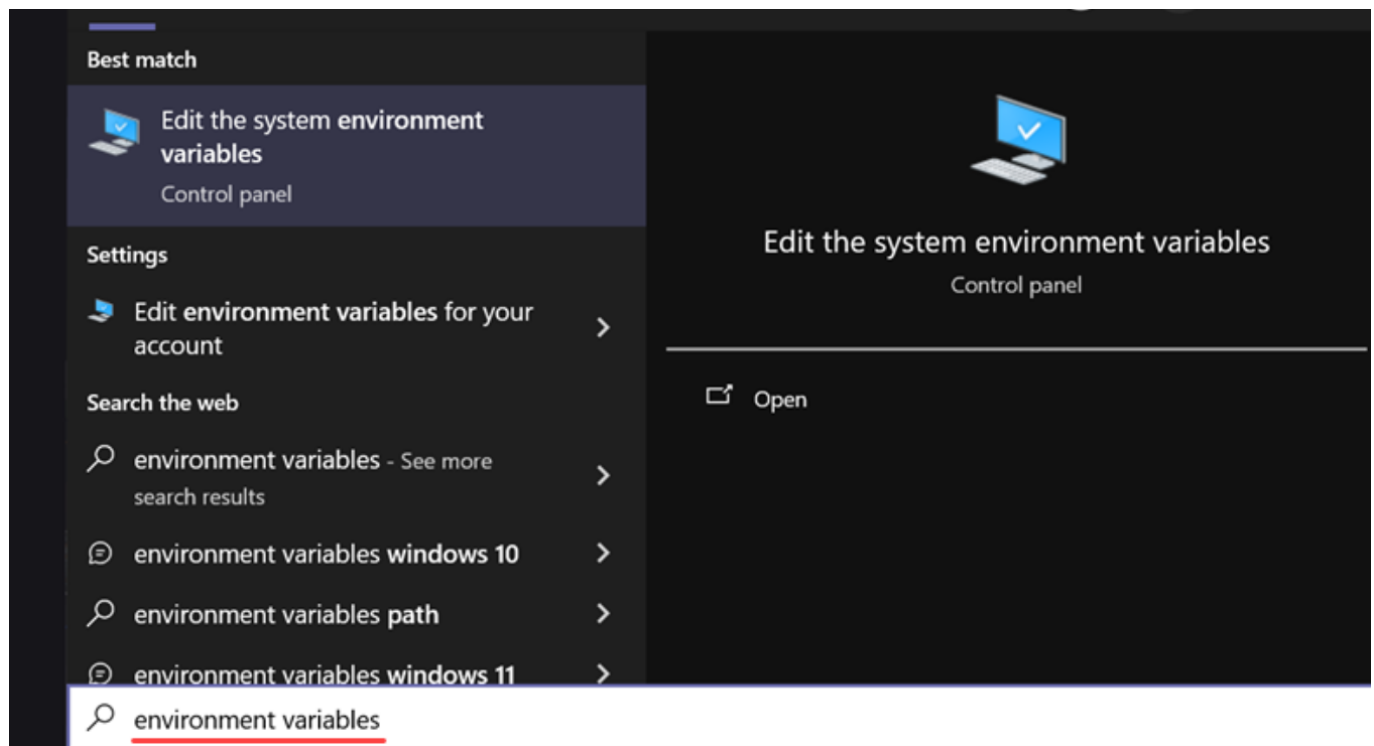
- The option will not affect any other system settings, and disabling it resolves potential name-length issues. We recommend selecting the option and closing the setup.

Step 6: Add Python to Path (Optional)

If the Python installer does not include the Add Python to PATH checkbox or you have not selected that option, continue in this step. Otherwise, skip to the next step.

To add Python to PATH, do the following:

1. In the Start menu, search for Environment Variables and press Enter.



2. Click Environment Variables to open the overview screen.

System Properties



Computer Name Hardware **Advanced** System Protection Remote

You must be logged on as an Administrator to make most of these changes.

Performance
Visual effects, processor scheduling, memory usage, and virtual memory
[Settings...](#)

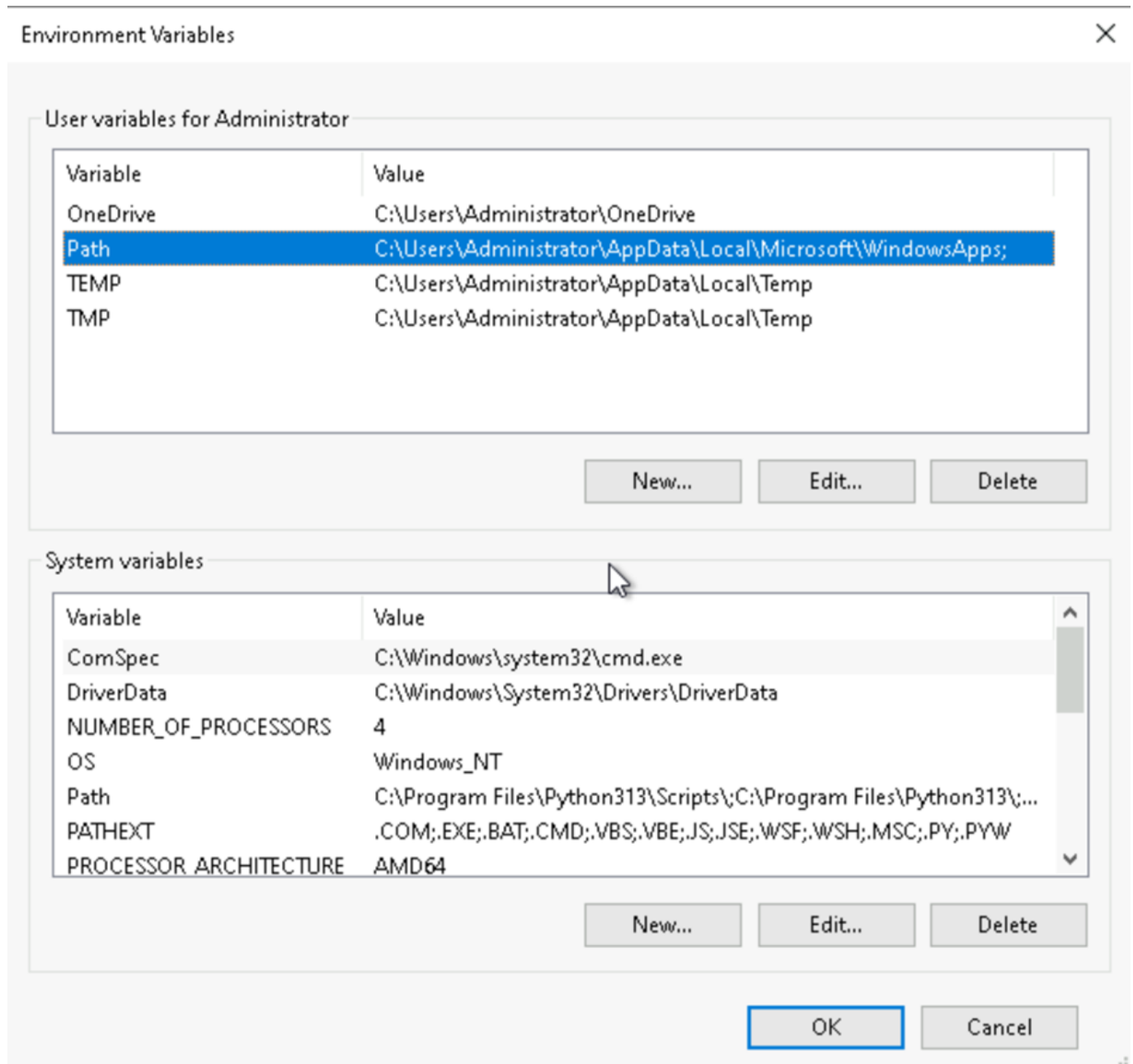
User Profiles
Desktop settings related to your sign-in
[Settings...](#)

Startup and Recovery
System startup, system failure, and debugging information
[Settings...](#)

[Environment Variables...](#)

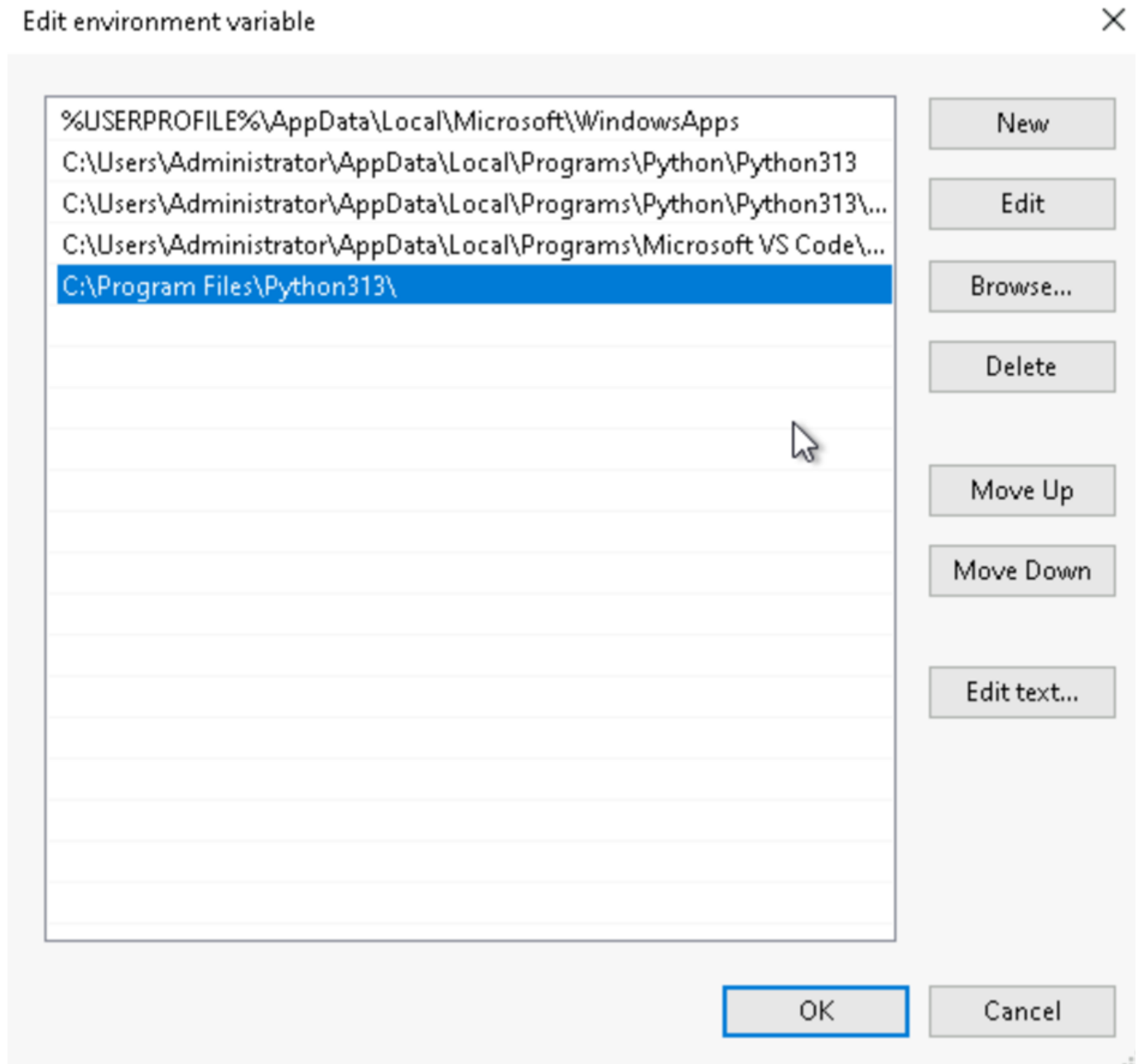
OK Cancel Apply

3. Double-click Path on the list to edit it.



Alternatively, select the variable and click the Edit button.

4. Double-click the first empty field and paste the Python installation folder path for both system and user environmental variables for Administration.



Alternatively, click the **New button** instead and paste the path. Click **OK** to save the changes.

Step 7: Verify Python Was Installed on Windows

The first way to verify that Python was installed successfully is through the command line. Open the command prompt and run the following command:

```
python --version
```

```
C:\Users\Administrator>python --version
Python 3.13.0
```

The output shows the installed Python version.

Verify PIP Was Installed

To verify whether PIP was installed, enter the following command in the command prompt:

```
pip --version
```

If it was installed successfully, you should see the PIP version number, the executable path, and the Python version:

```
C:\Users\Administrator>pip --version
pip 24.2 from C:\Program Files\Python313\Lib\site-packages\pip (python 3.13)
```

PIP has not been installed yet if you get the following output:

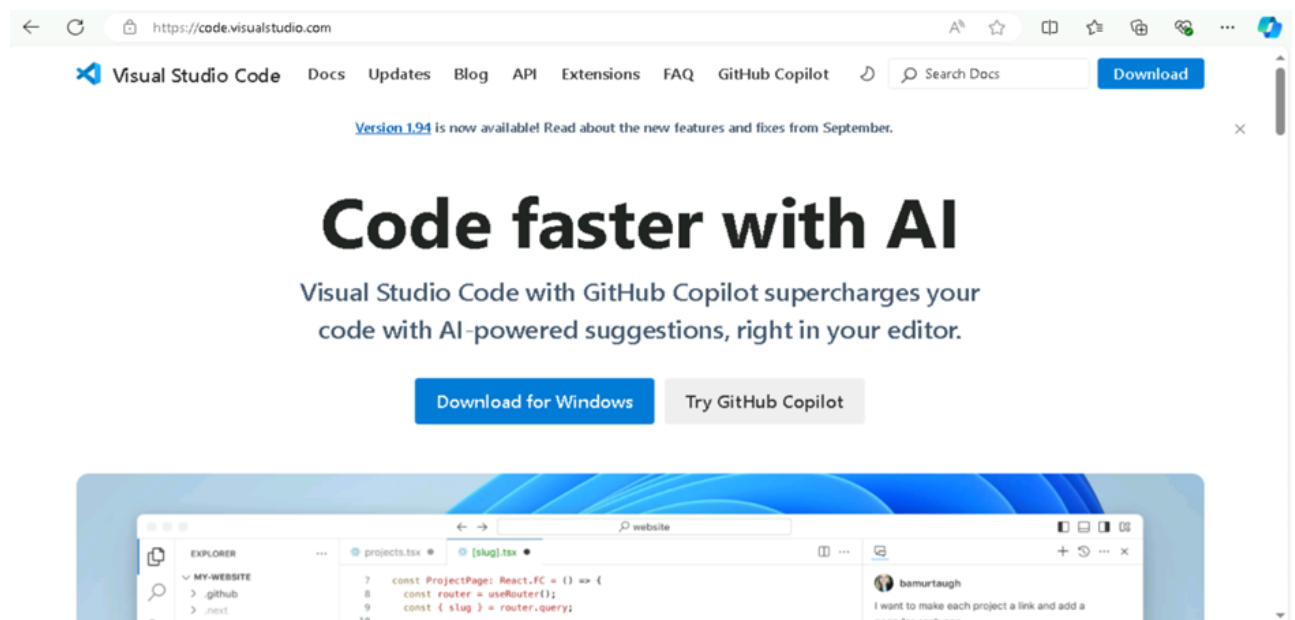
```
'pip' is not recognized as an internal or external command,
Operable program or batch file.
```

Step 8: Connect VSCode with Python

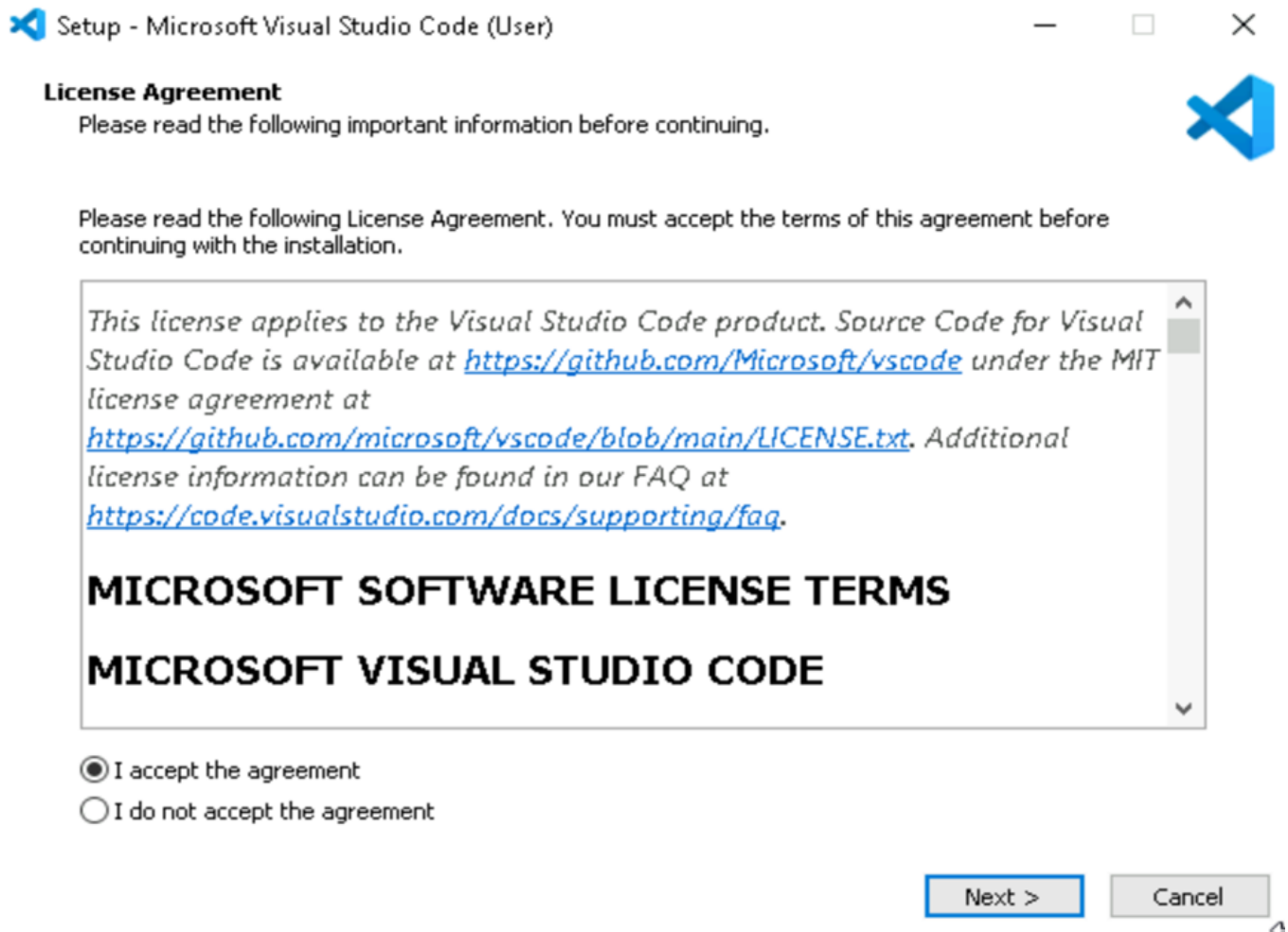
To set up Visual Studio Code (VS Code) with Python, follow these steps:

Installing Visual Studio Code

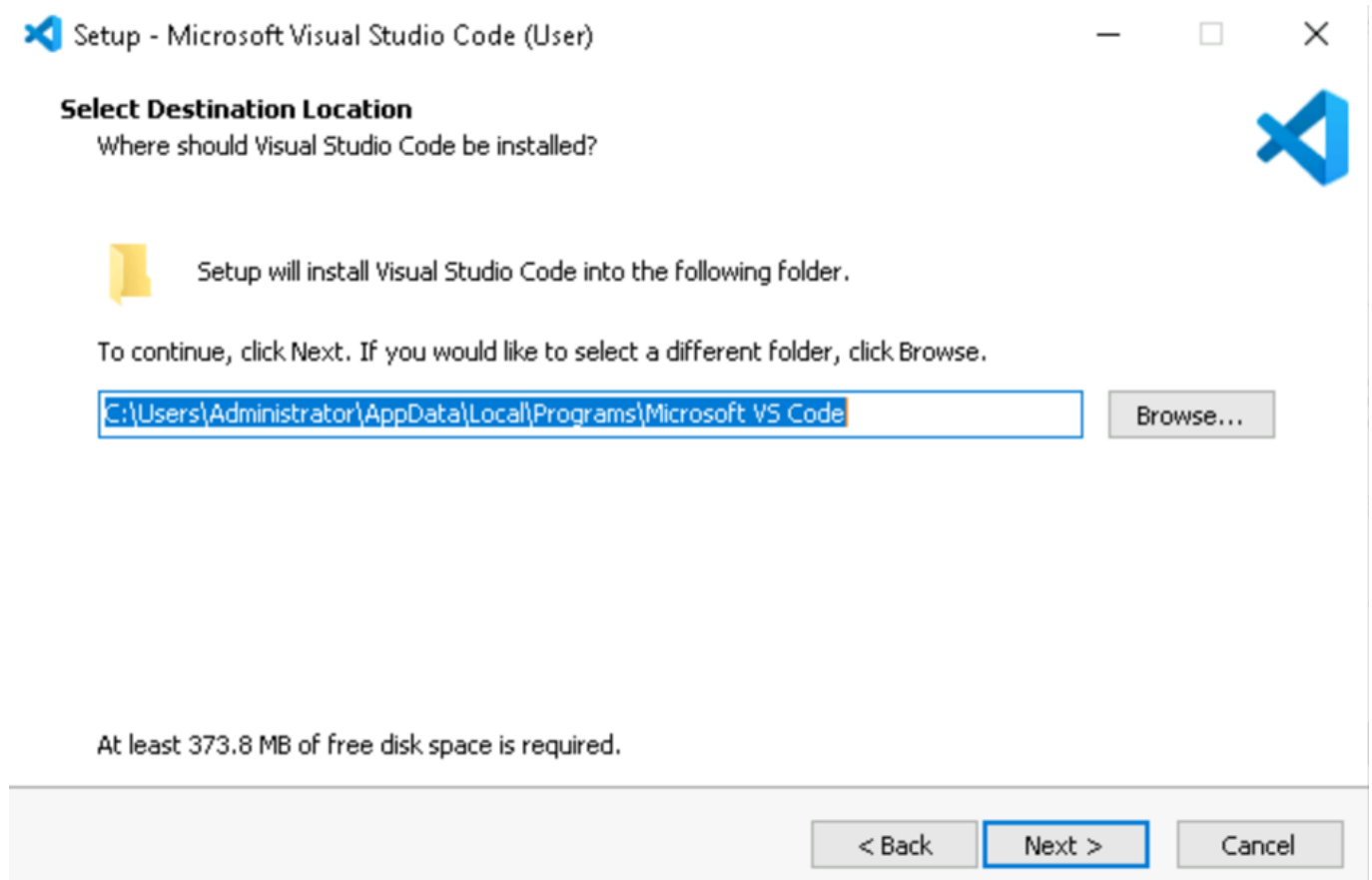
- **Install Visual Studio Code:**
 - Download and install the latest version of [Visual Studio Code](https://code.visualstudio.com) for your operating system.
 - The file will be downloaded onto your system. Open the file and then click on Install.



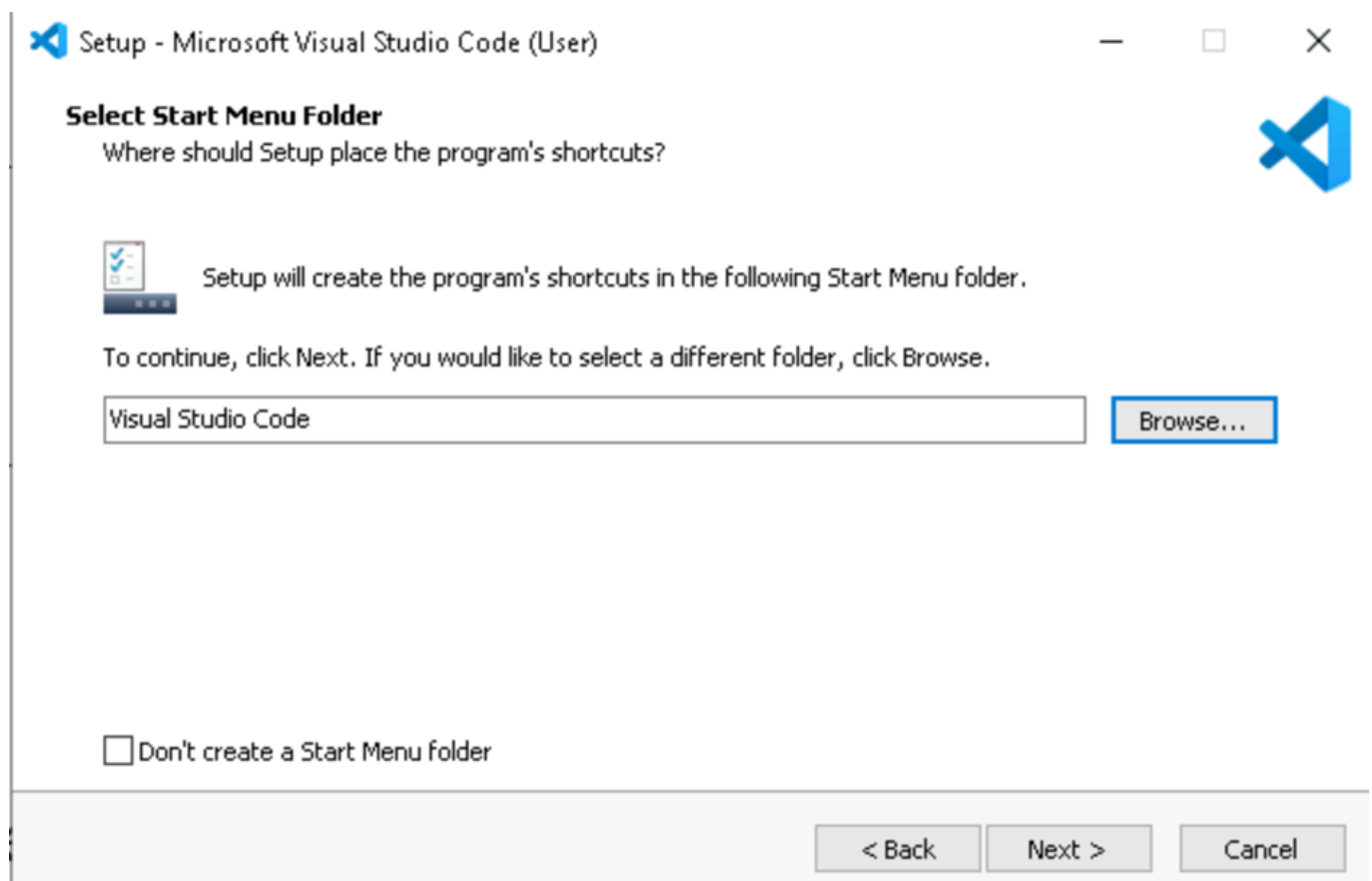
- **Now accept the license agreement.**



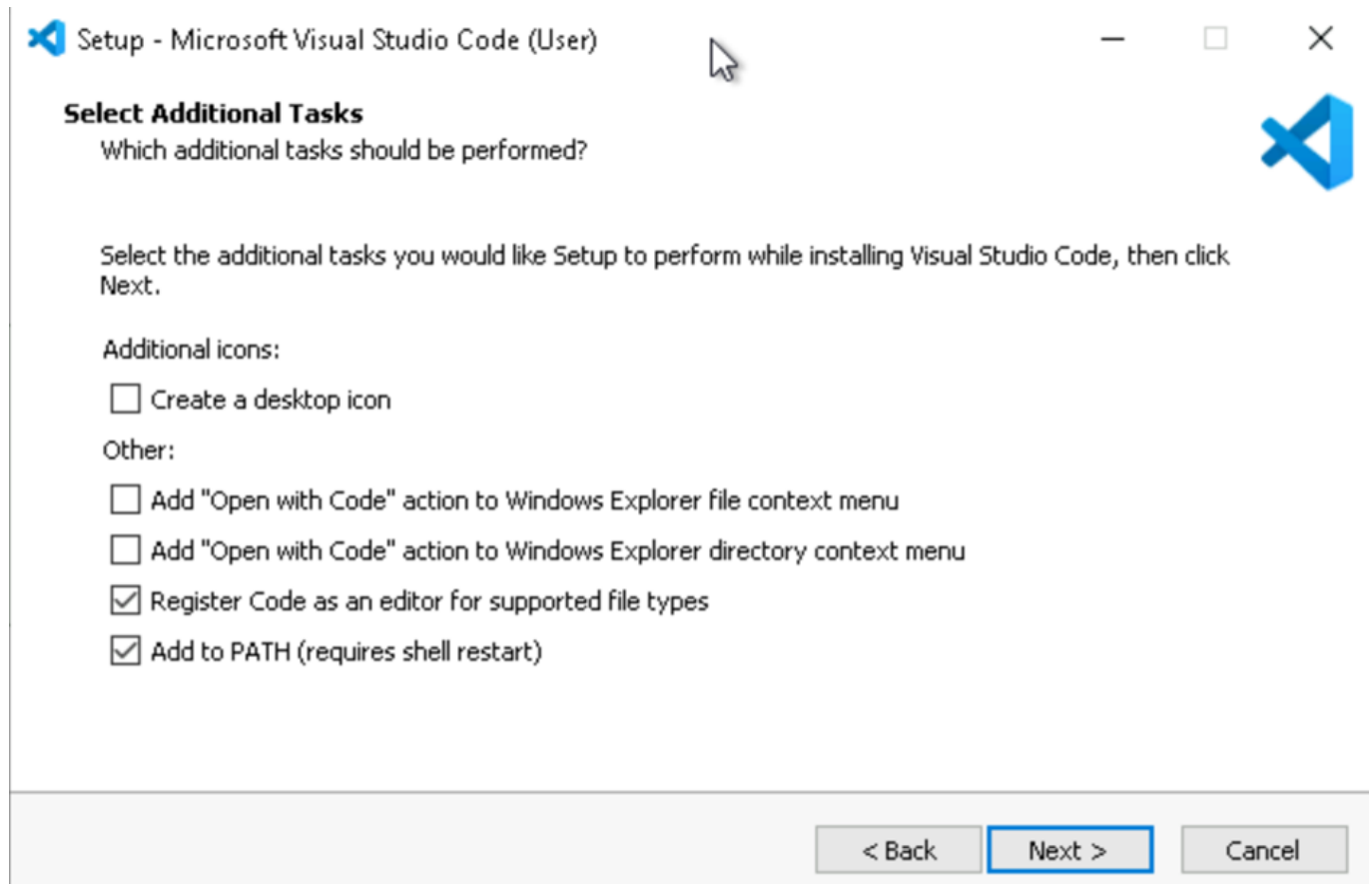
- **Then it prompts for the file location, where you want to save the VS Code file. Browse the location and then click on Next.**



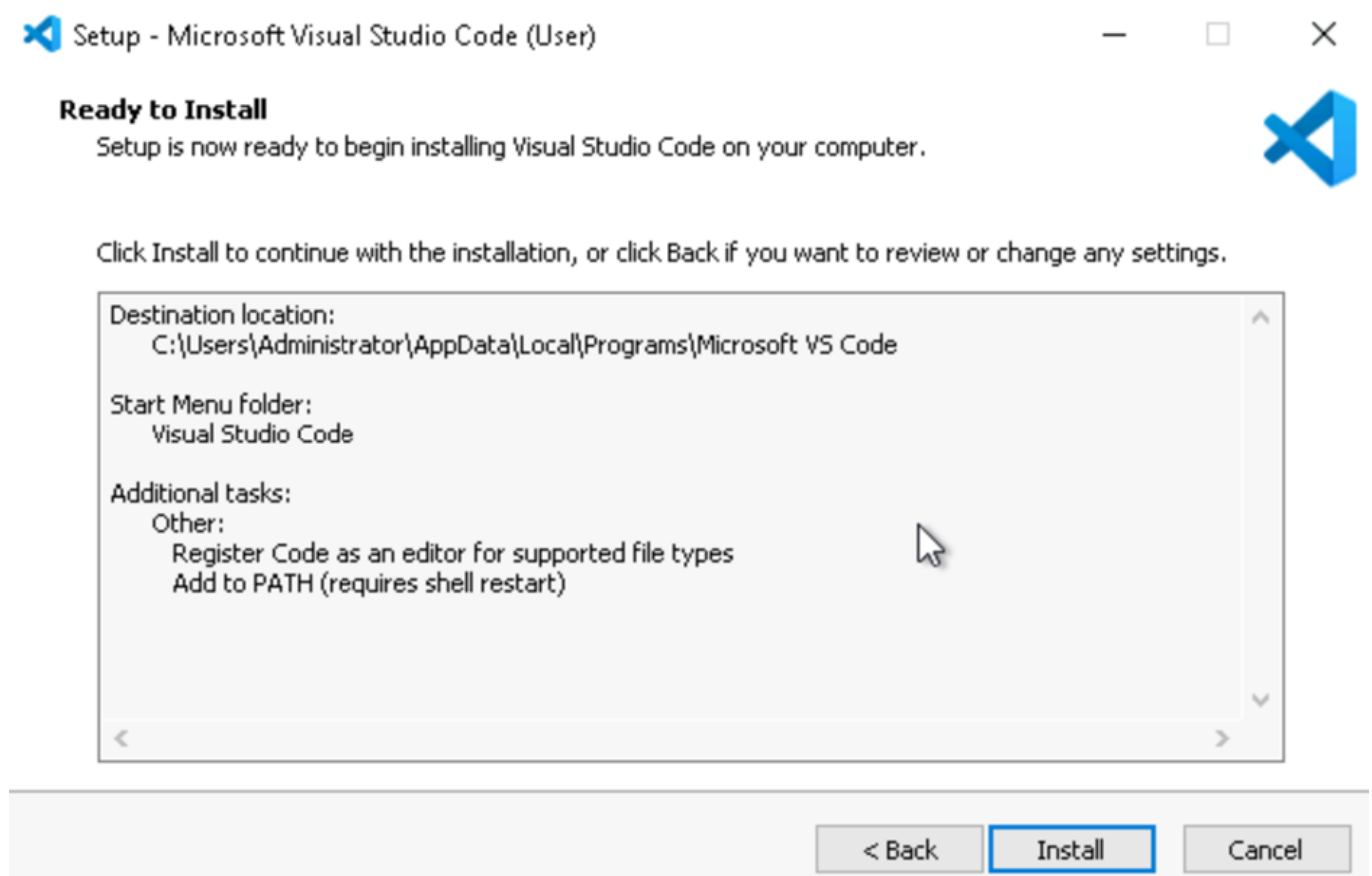
- The next prompt is how you want the VS Code on your startup. Change according to your convenience and click on Next.

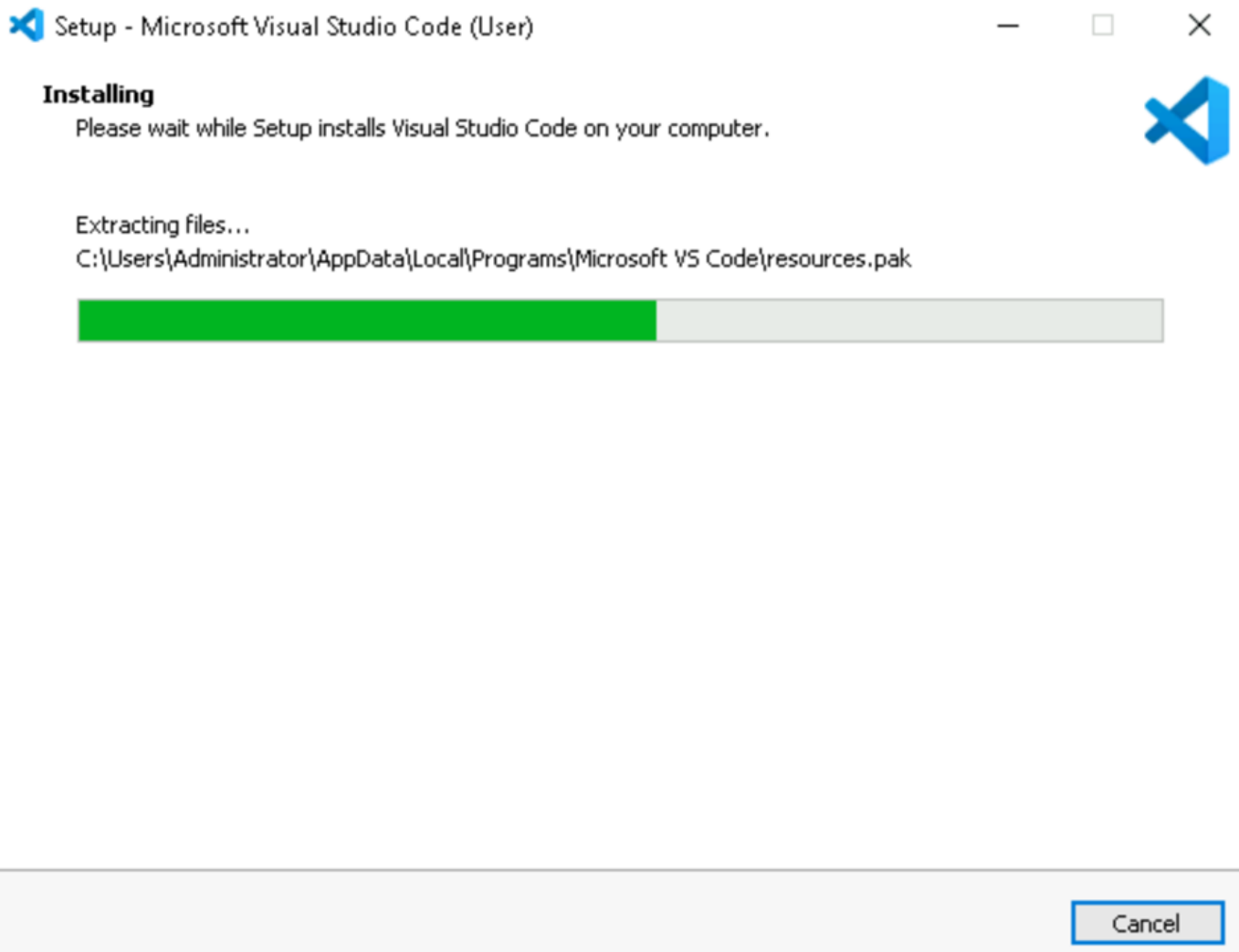


- Next, you see the prompt for the additional task which we want the VS Code to perform. At this step, choose the default settings and then click on next.

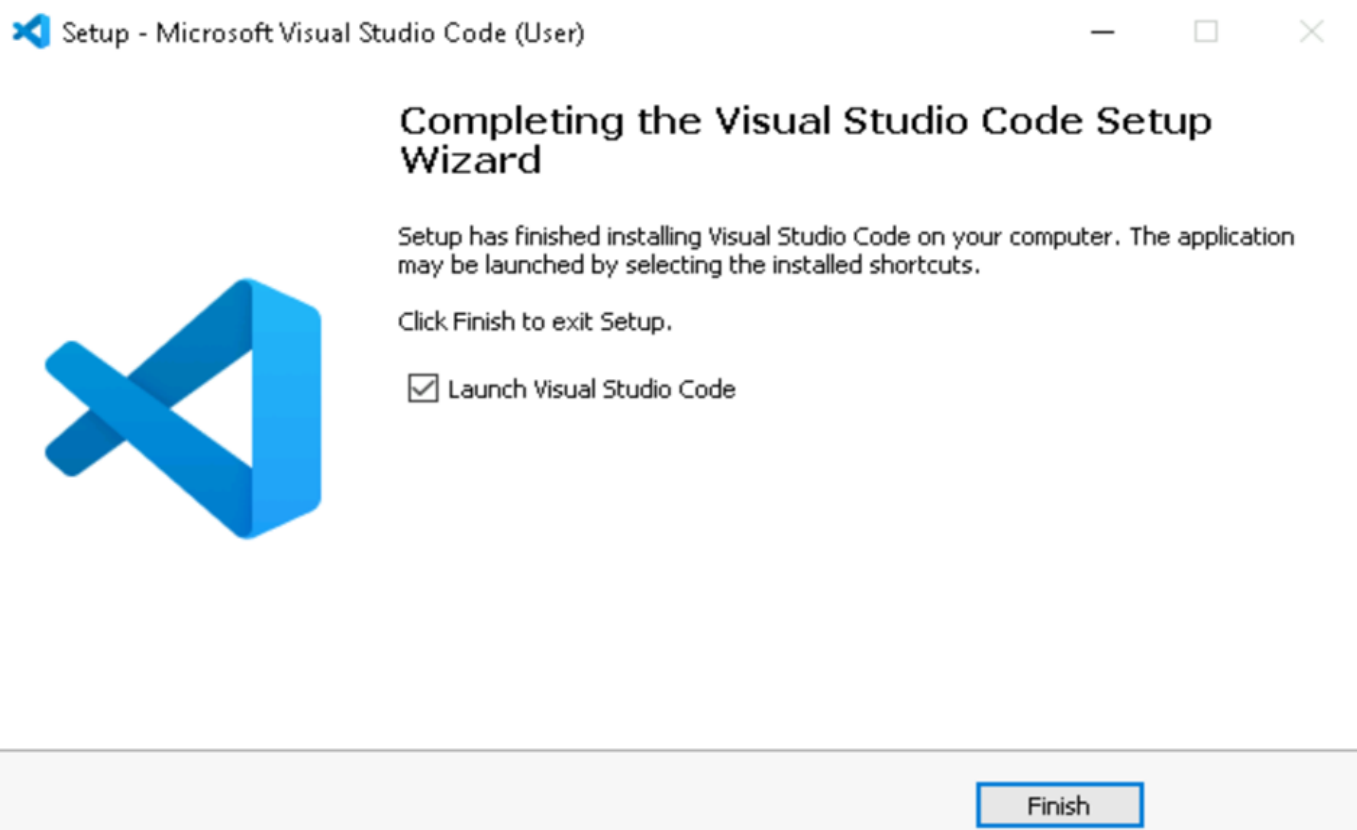


- Click on install , the installation of VS Code will now begin.

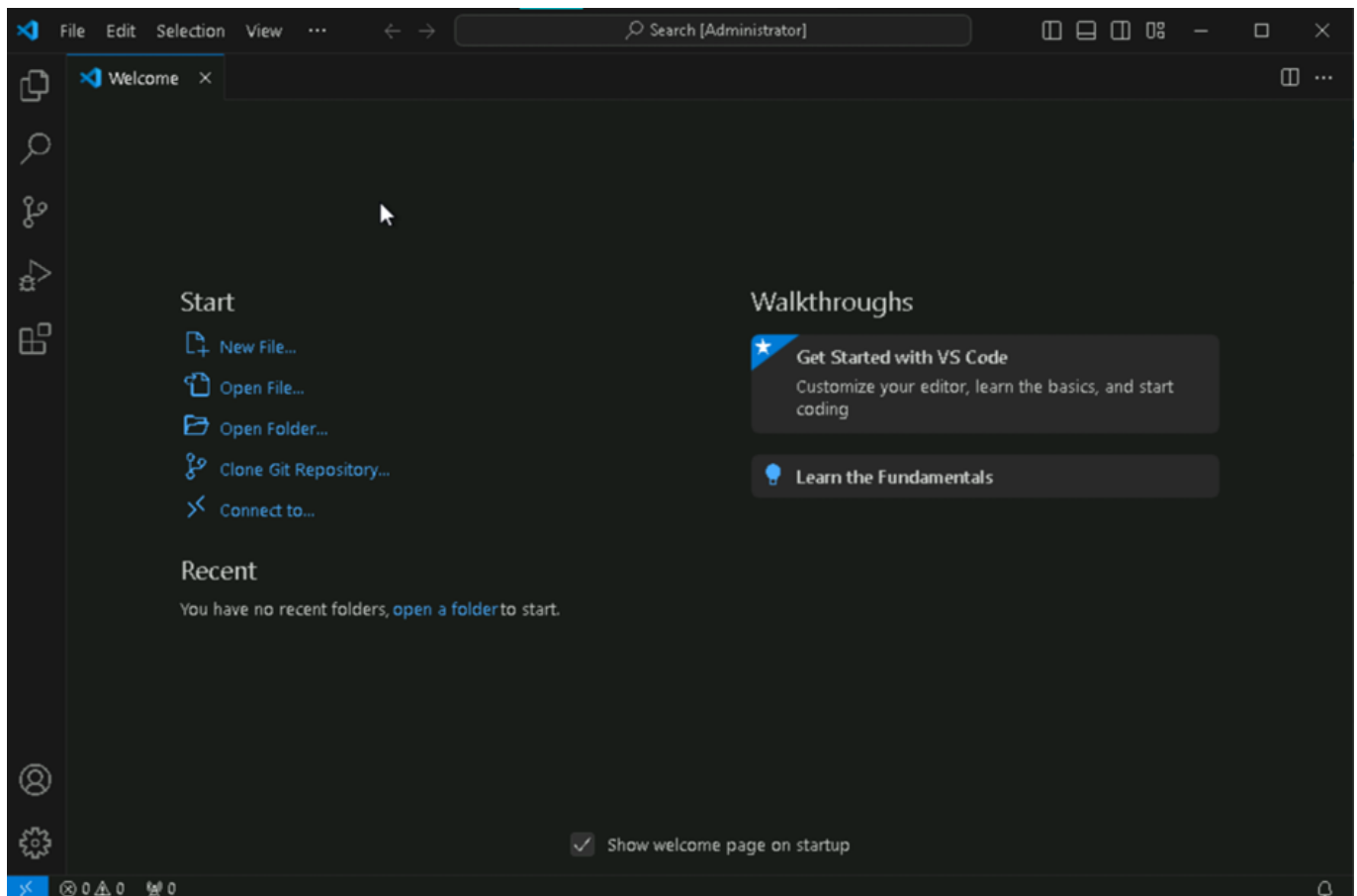




- At this step, we have completed installing VS Code, click on Finish.

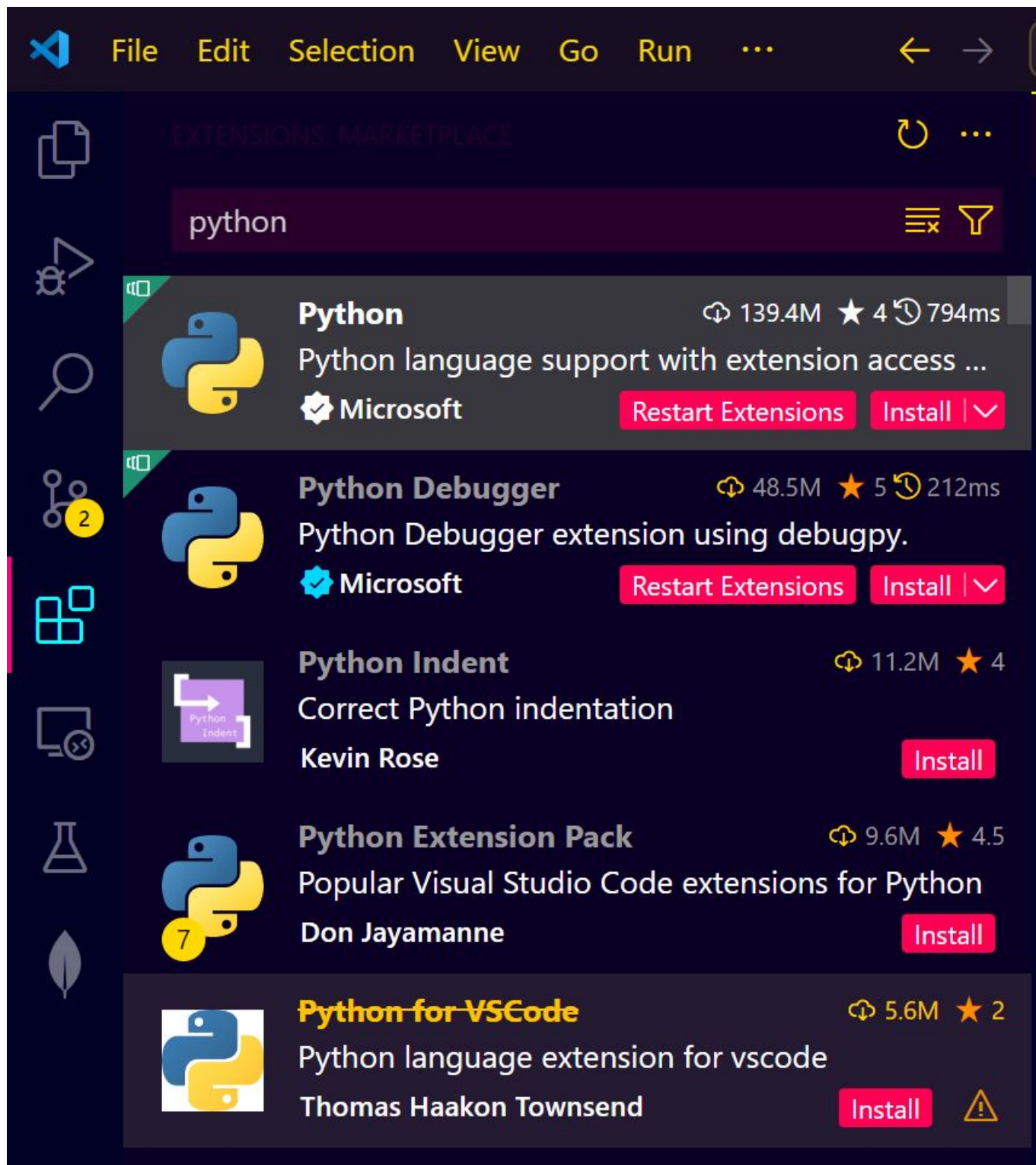


- Now that VS Code installation is successful, the page appears as below:



Install the Python Extension for VS Code:

- Open VS Code.
- Go to the **Extensions** view by clicking on the square icon in the left sidebar or pressing **Ctrl+Shift+X**.



- Search for the **Python** extension by Microsoft.



Extension: Python X

Python v2024.16.0

Microsoft microsoft.com | 139,410,149 | ★★★★★ (599)

Python language support with extension access points for IntelliSense (Pylance...)

Restart Extensions Install Auto Update

DETAILS FEATURES CHANGELOG EXTENSION PACK

Python extension for Visual Studio Code

A Visual Studio Code extension with rich support for the Python language (for all actively supported Python versions), providing access points for extensions to seamlessly integrate and offer support for IntelliSense (Pylance), debugging (Python Debugger), formatting, linting, code navigation, refactoring, variable explorer, test explorer, and more!

Support for vscod.dev

Categories

- Programming Languages
- Debuggers
- Other
- Data Science
- Machine Learning

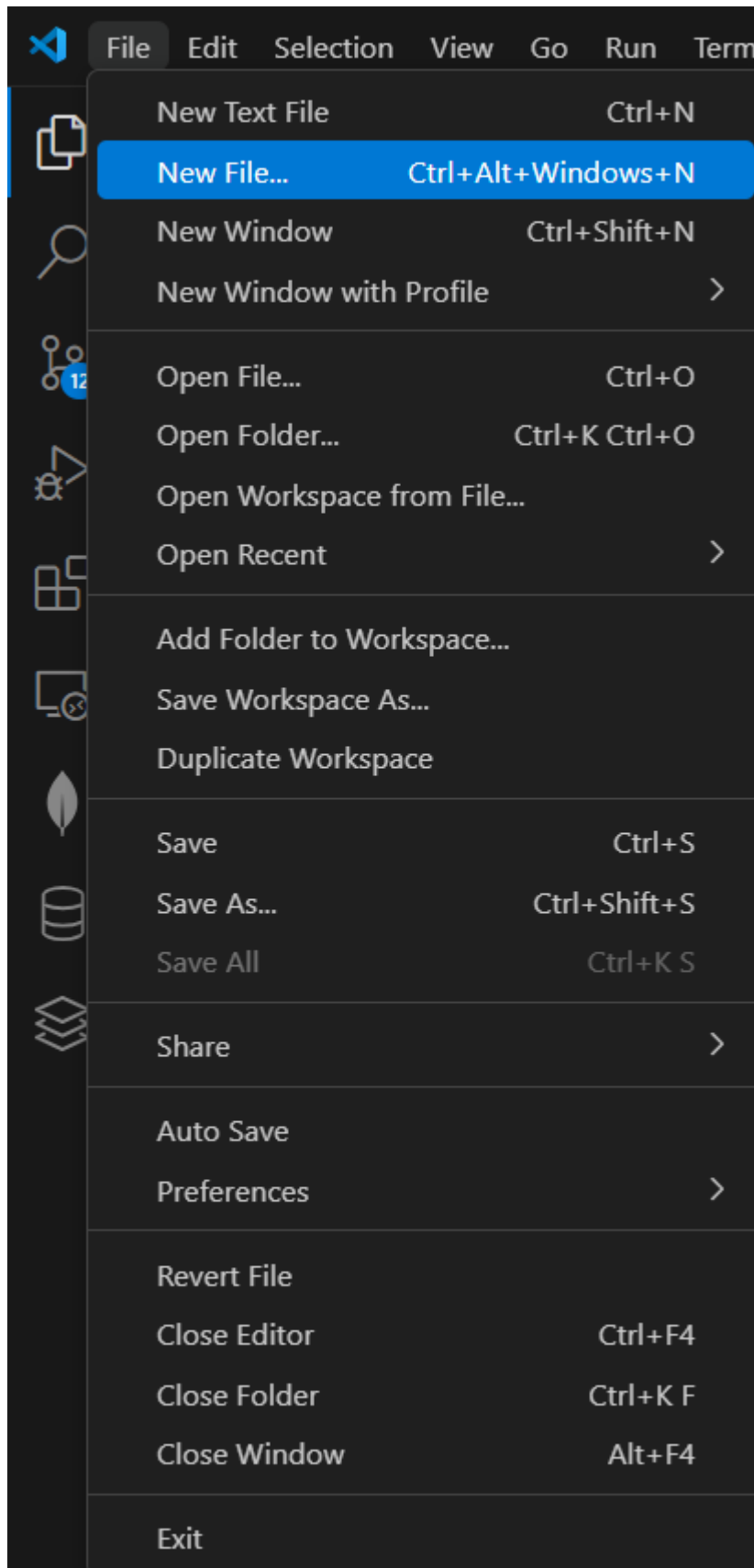
Resources

- [Marketplace](#)
- [Issues](#)

- Click **Install** to add the extension to your VS Code.

Create a Python File and Write the Script

- **Create a Python File:**
 - Open a new file in your workspace.
 - Save it with the `.py` extension (e.g., `modal.py`).



- **Write the Script:**

- In the file, write the following simple script(e.g., `modal.py`):

```
# First Python script
print("Hello, World!")
```

-
- **Save the File:**
 - Save the file (**Ctrl + S**).
-

Run Your Python Script

- **Using VS Code:**
 - Press **F5** or right-click the code and select **Run Python File in Terminal**.
 - The terminal at the bottom will execute the script, and you should see the output:

Output:

```
PS C:\Users\Administrator\Desktop\python> python modal.py
Hello, World!
```

References

- [Python Official Documentation](#)
 - [VS Code Python Extension](#)
-