

Python Setup

Jeremy Mikecz

2026-01-13

Install & Set up Python with Visual Studio Code and UV

This document provides detailed instructions for installing and setting up Visual Studio Code and Python (via uv)

Table of Contents

Step 1: Install Visual Studio Code (10 minutes)

- Go to <https://code.visualstudio.com/>
- Download for your operating system (it auto-detects)
- Install VS Code with default options
- Launch VS Code

Once opened, let's install essential extensions (click Extensions icon - looks like four squares - in sidebar or press Ctrl+Shift+X):

1. Search for **'Python'** - install the one by Microsoft
2. Search for **'Jupyter'** - install the one by Microsoft
3. Search for **'Quarto'** - use to create and work with Quarto notebooks and presentations
4. Search for **'Git History'** (optional but useful)

These extensions will help VS Code work with Python.

Step 2: Install uv (5 minutes)

Now we install uv, which will handle EVERYTHING else for us - including Python itself. This is the modern way to manage Python projects. uv is a fast Python package and project manager that we'll use for managing dependencies.

Dependencies are other software that are needed for a program to run. For Python, this often means

Open the integrated terminal in VS Code:

- **Windows/Linux:** Press `Ctrl + ``` (that's the backtick key, usually above Tab)
- **Mac:** Press `Cmd + ```

Or use the menu: Terminal → New Terminal

Windows (PowerShell)

Windows: Type in a Powershell terminal

```
powershell -c "irm https://astral.sh/uv/install.ps1 | iex"
```

If you get an error like this - `Error: Powershell requires an execution policy...` - then we need to temporarily bypass Powershell's policies by running:

```
powershell -ExecutionPolicy ByPass -c "irm https://astral.sh/uv/install.ps1 | iex"
```

Alternate Installations for Windows

If the above did not work you can also try:

winget option 1. [install winget](#) 2. Run in the terminal from VSC:

```
winget install --id=astral-sh.uv -e
```

scoop option 1. [install scoop](#) 2. Run in the terminal from VSC:

```
scoop install uv
```

Troubleshooting Windows: If uv is not recognized:

Check if `%USERPROFILE%\local\bin` is in your PATH

Add manually: System Properties → Environment Variables → Path → Add the directory
Restart your terminal

macOS/Linux

Type the following in a Mac/Linux terminal:

```
curl -LsSf https://astral.sh/uv/install.sh | sh
```

Alternate Installations for Macs

If the above did not work on your Mac you can also try:

homebrew option

Post-Installation (macOS)

```
# If using the shell script, add to your shell profile
# For zsh (default on modern macOS):
echo 'export PATH="$HOME/.local/bin:$PATH"' >> ~/.zshrc
source ~/.zshrc

# For bash:
echo 'export PATH="$HOME/.local/bin:$PATH"' >> ~/.bashrc
source ~/.bashrc

# Verify
uv --version
```

Linux Installation

```
# Shell script (works on most distributions)
curl -LsSf https://astral.sh/uv/install.sh | sh

# Add to PATH (add to ~/.bashrc or ~/.zshrc)
export PATH="$HOME/.local/bin:$PATH"

# Reload shell
source ~/.bashrc # or source ~/.zshrc

# Verify
uv --version
```

Troubleshooting Linux:

If curl is not installed: `sudo apt install curl` (Ubuntu/Debian)

Permission denied: The script installs to `~/.local/bin`, which shouldn't require `sudo`

Verify Installation

After installation completes:

1. **Close the terminal tab** in VS Code (click the trash can icon)
2. **Open a new terminal** (Ctrl/Cmd + ‘ again)
3. Verify installation by running the following (Mac/Linux or if Windows in Command Prompt or Git Bash):

```
uv --version
```

You should see something like `uv 0.4.x` or higher.

Troubleshooting time: Common issues

- “If ‘uv’ is not recognized: The installation script may need admin rights, or you may need to restart VS Code”
- “Mac users: If you get a security warning, go to System Preferences → Security & Privacy → Allow”

Step 3: Let uv Install Python (3 minutes)

With `uv` we don’t need to install Python separately. When we create our first project, `uv` will automatically install the right Python version for us. But let’s verify `uv` can find or install Python:

Enter the following in the terminal (Mac/Linux or if Windows in Command Prompt or Git Bash):

```
uv python list
```

This shows all Python versions `uv` knows about. Don’t worry if it’s empty - `uv` will download Python when we need it.

We can install the latest version of Python with:

```
uv python install
```

This downloads and installs the latest stable version of Python, managed by uv. No PATH configuration needed, no installers to run, no clicking through installation wizards.

Or we can specify an older version to install:

Enter the following in the terminal (Mac/Linux or if Windows in Command Prompt or Git Bash):

```
uv python install 3.12
```

Verify:

Enter the following in the terminal (Mac/Linux or if Windows in Command Prompt or Git Bash):

```
uv python list
```

Advantages of using uv for Python Installation

1. No need to go to python.org, download an installer
2. don't need to worry about "Add to PATH" (a common error in Python installation, when a Python IDE like VSC cannot find Python or cannot find the version of Python you want to use)
3. don't need to worry about conflicting with other Python installations

Brief Cheatsheet

the brief, no explanation need version:

1. Install [Visual Studio Code](#)
2. Open VSC, install **Python**, **Jupyter**, **Quarto** (optional), and **Git** (optional) extensions.
3. install uv
 1. in Windows Powershell: `powershell -c "irm https://astral.sh/uv/install.ps1 | iex"`
 2. on Mac/Linux terminal: `curl -LsSf https://astral.sh/uv/install.sh | sh`
4. Check installation with `uv --version`.
5. Install Python with `uv python install`.