



Network Automation Tapas

Bite-sized talks to give the audience a
little something to chew on



Network Automation Tapas

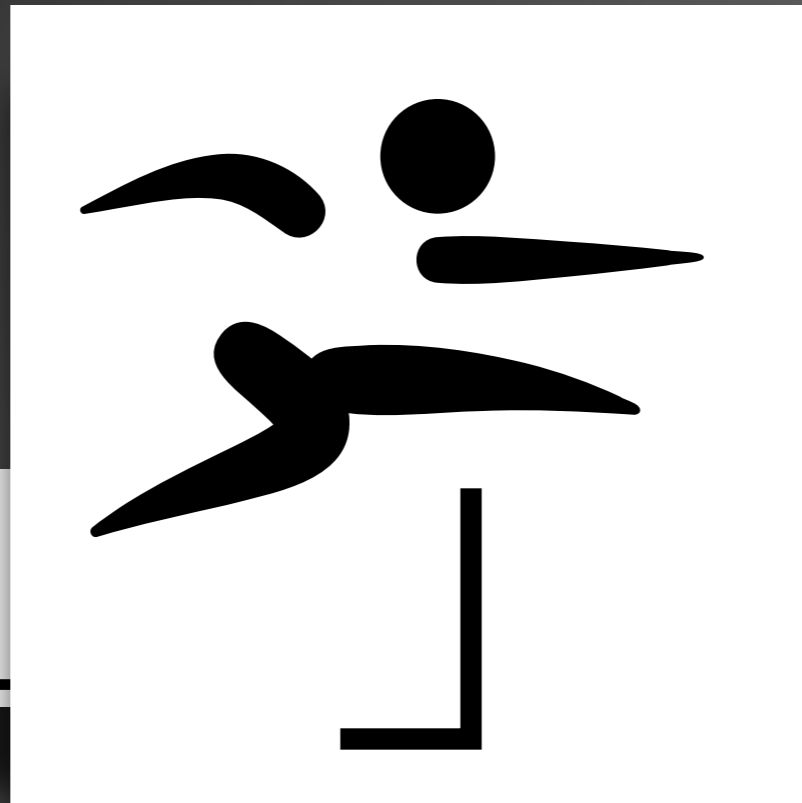
- **Frank Seesink**, Senior Network Engineer
UNC Chapel Hill
- **Maria Isabel Gandia**
CSUC/RedIRIS (GÉANT project)
- **Amy Liebowitz**
University of Michigan
- **AJ Ragusa**
GlobalNOC
- **James Harr**
Internet2
- **Shannon Byrnes**, NetDevOps Engineer
Internet2



Why this session?



Network Automation Tapas



**YA
ML**



Amul



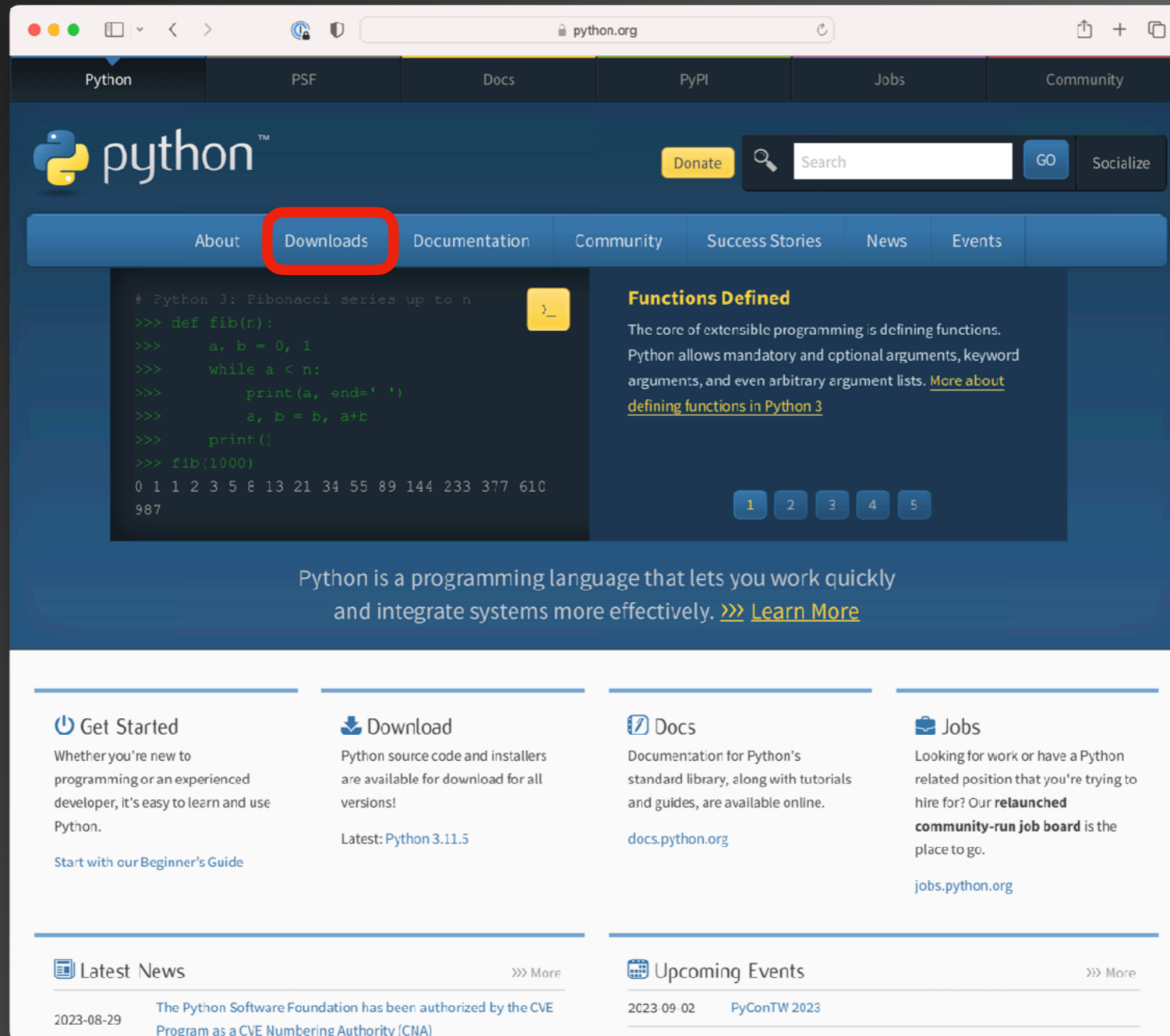
Yum!

Getting Started with Python



Python Software Foundation

Option #1



The screenshot shows the Python.org website. The 'Downloads' menu item is highlighted with a red box. The main content area features a code editor with a Fibonacci function and its output, a 'Functions Defined' section, and a navigation bar with 'About', 'Downloads', 'Documentation', 'Community', 'Success Stories', 'News', and 'Events'. The footer contains sections for 'Get Started', 'Download', 'Docs', 'Jobs', 'Latest News', and 'Upcoming Events'.

```
# Python 3: Fibonacci series up to n
>>> def fib(n):
>>>     a, b = 0, 1
>>>     while a < n:
>>>         print(a, end=' ')
>>>         a, b = b, a+b
>>>     print()
>>> fib(1000)
0 1 1 2 3 5 8 13 21 34 55 89 144 233 377 610
987
```

Functions Defined
The core of extensible programming is defining functions. Python allows mandatory and optional arguments, keyword arguments, and even arbitrary argument lists. [More about defining functions in Python 3](#)

Python is a programming language that lets you work quickly and integrate systems more effectively. >>> [Learn More](#)

Get Started
Whether you're new to programming or an experienced developer, it's easy to learn and use Python.
[Start with our Beginner's Guide](#)

Download
Python source code and installers are available for download for all versions!
Latest: [Python 3.11.5](#)

Docs
Documentation for Python's standard library, along with tutorials and guides, are available online.
[docs.python.org](#)

Jobs
Looking for work or have a Python related position that you're trying to hire for? Our **relaunched community-run job board** is the place to go.
[jobs.python.org](#)

Latest News >>> [More](#)
2023-08-29 [The Python Software Foundation has been authorized by the CVE Program as a CVE Numbering Authority \(CNA\)](#)

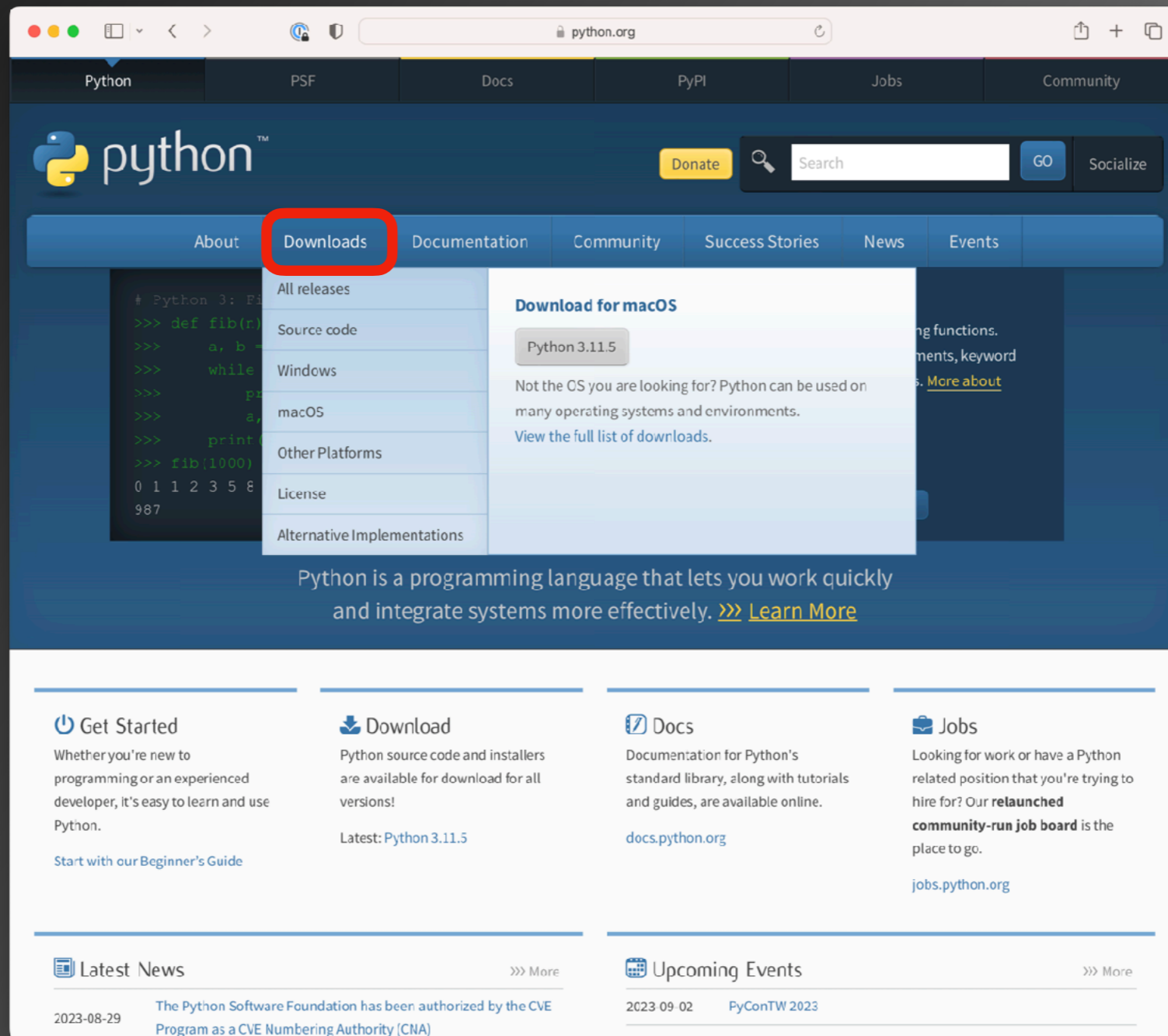
Upcoming Events >>> [More](#)
2023-09-02 [PyConTW 2023](#)

<https://www.python.org/>

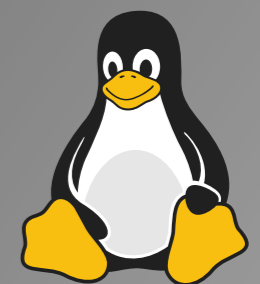


Python Software Foundation

Option #1



The screenshot shows the Python.org website with the 'Downloads' menu item highlighted in a red box. The dropdown menu includes options for 'All releases', 'Source code', 'Windows', 'macOS', 'Other Platforms', 'License', and 'Alternative Implementations'. The 'macOS' option is selected, displaying a 'Download for macOS' section for Python 3.11.5. The page also features a navigation bar with links to 'Python', 'PSF', 'Docs', 'PyPI', 'Jobs', and 'Community', and a footer with sections for 'Get Started', 'Download', 'Docs', 'Jobs', 'Latest News', and 'Upcoming Events'.



<https://www.python.org/>

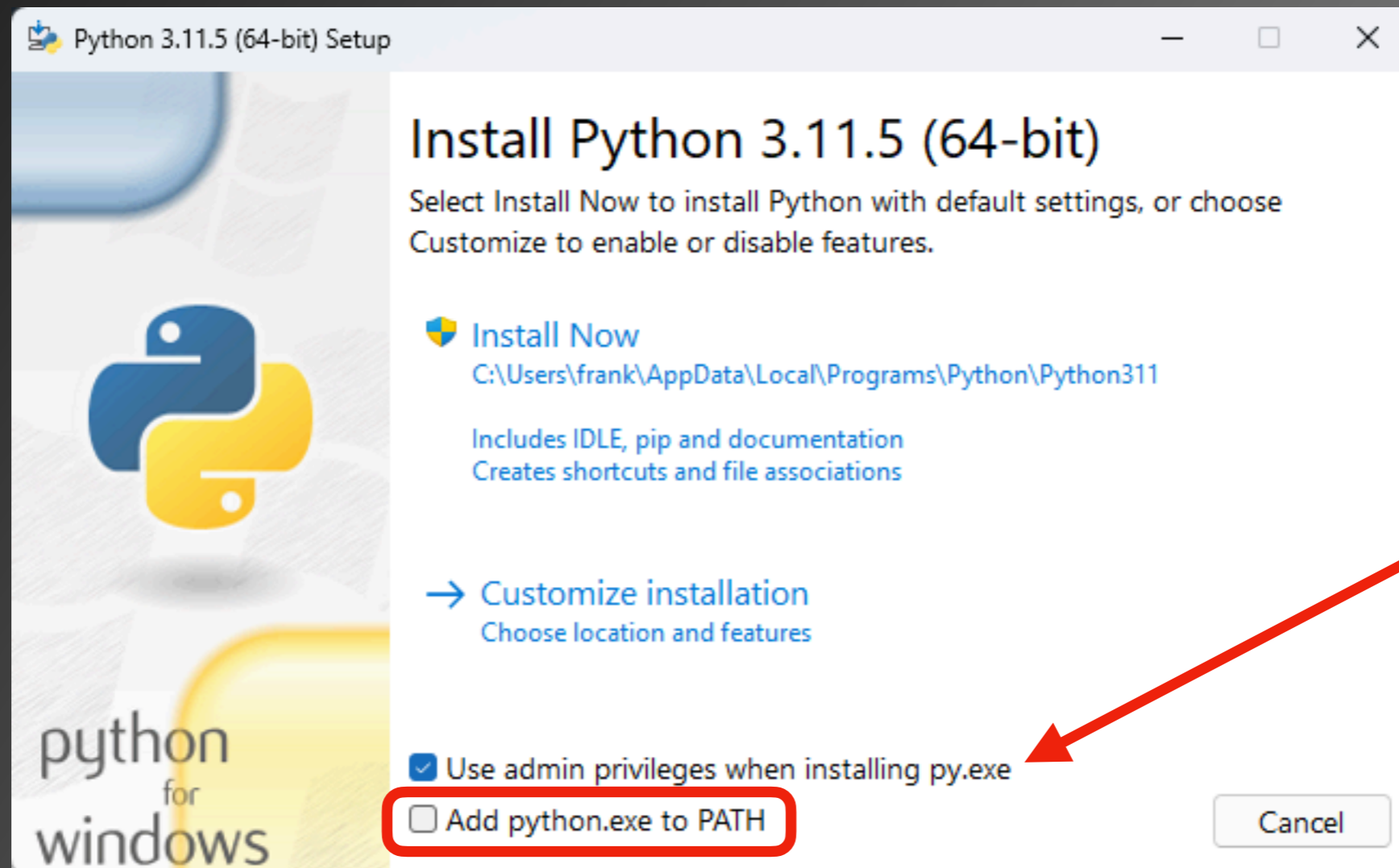


Installing Python

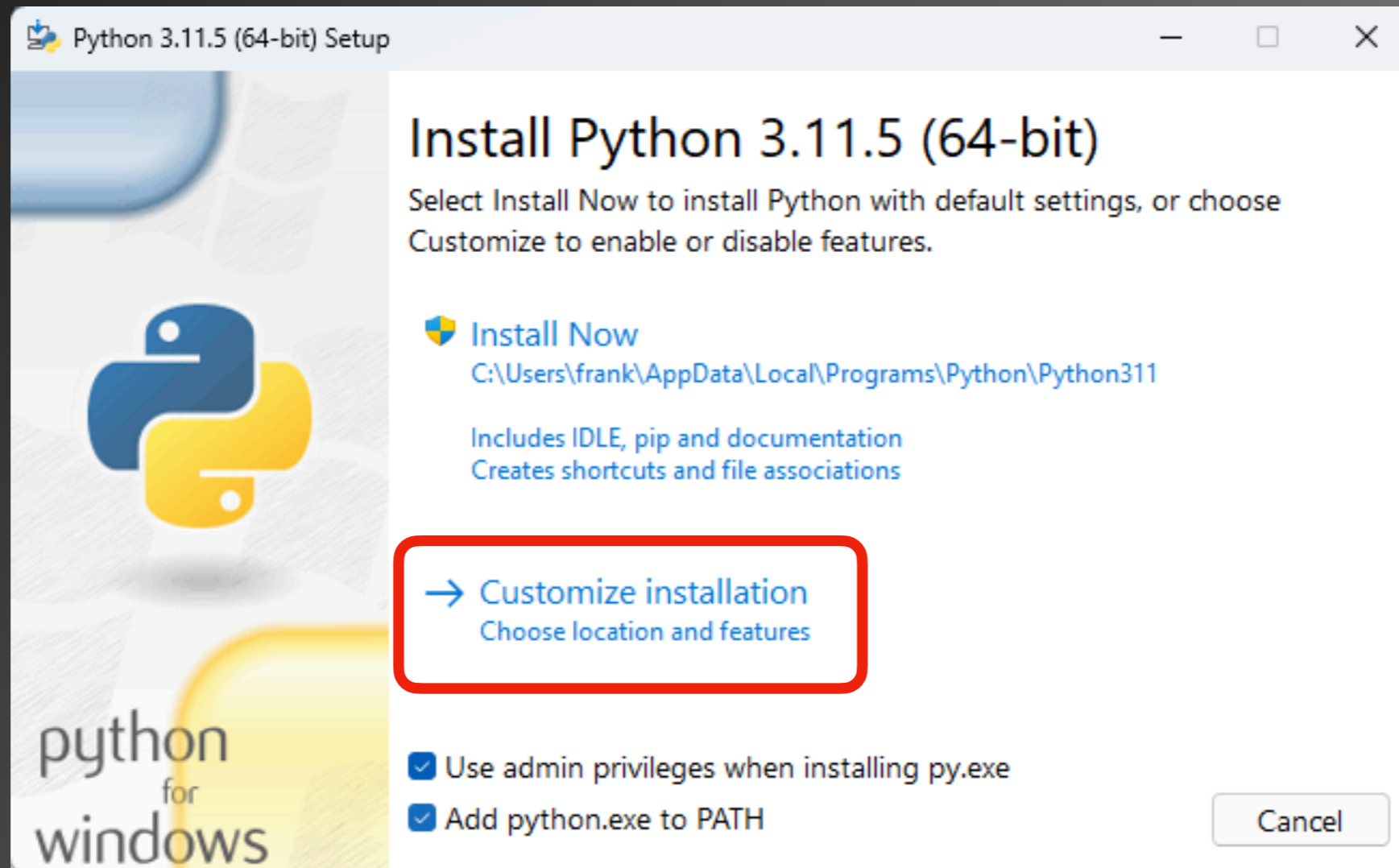
for Windows



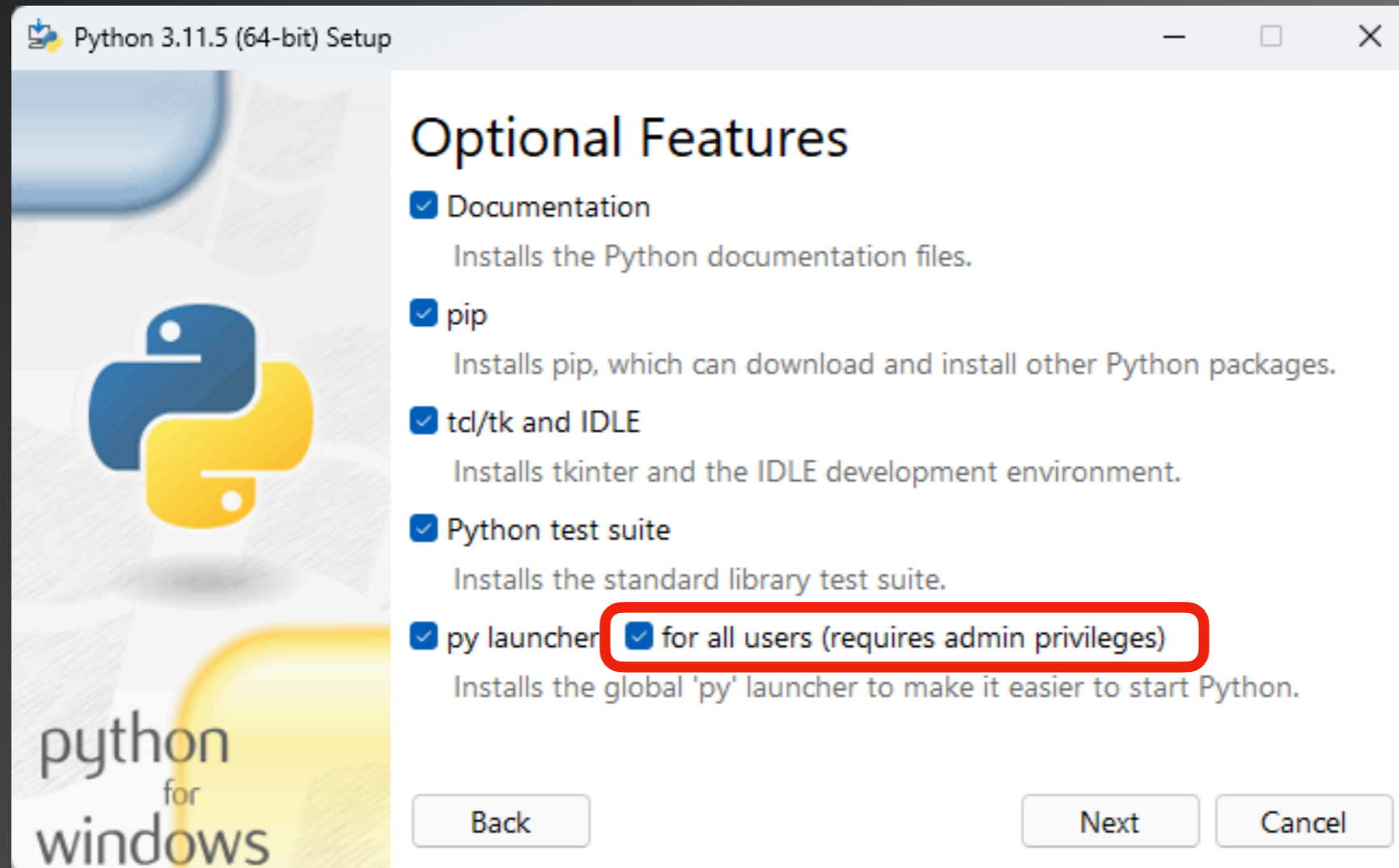
Install Python - Windows



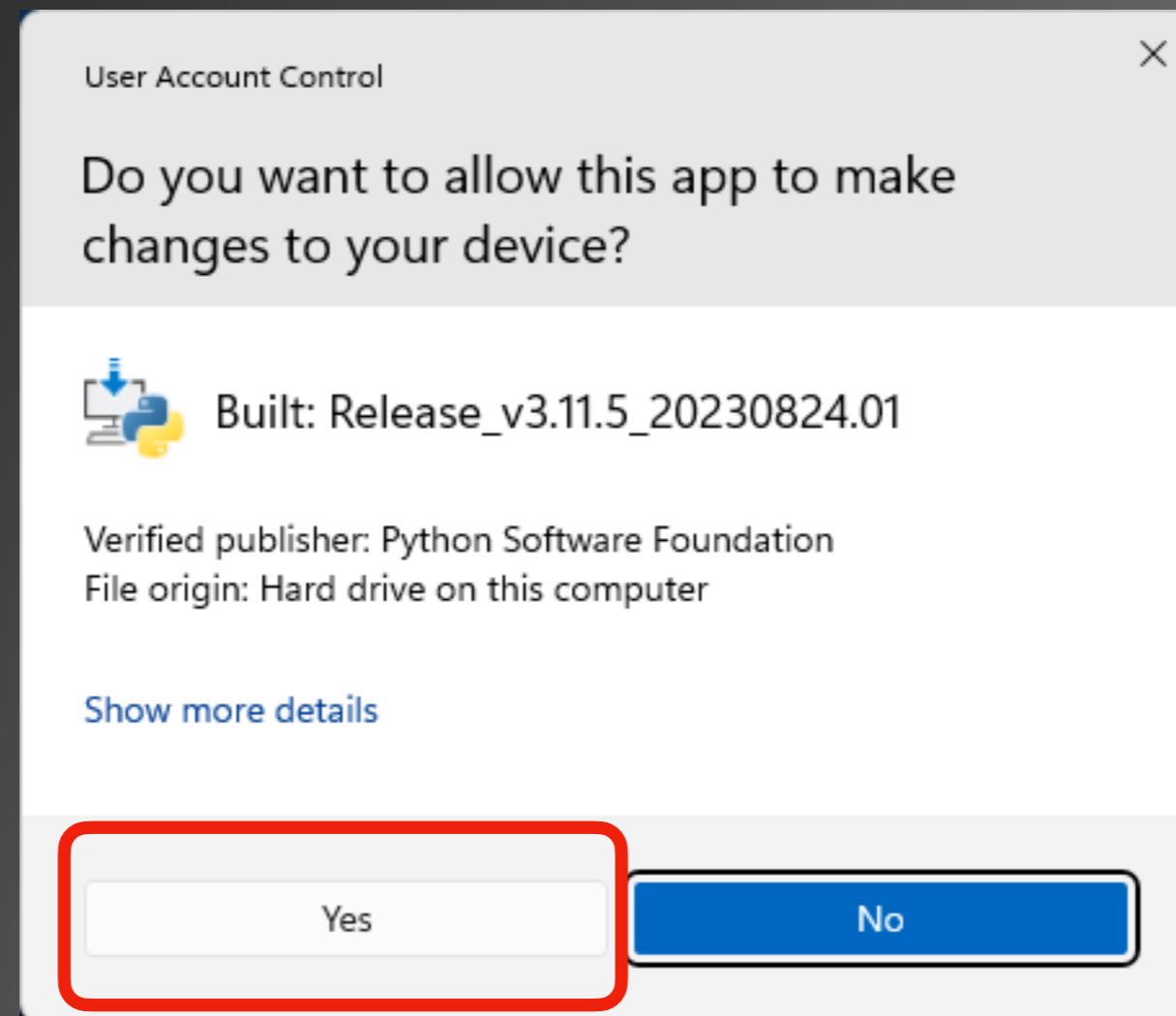
Install Python - Windows



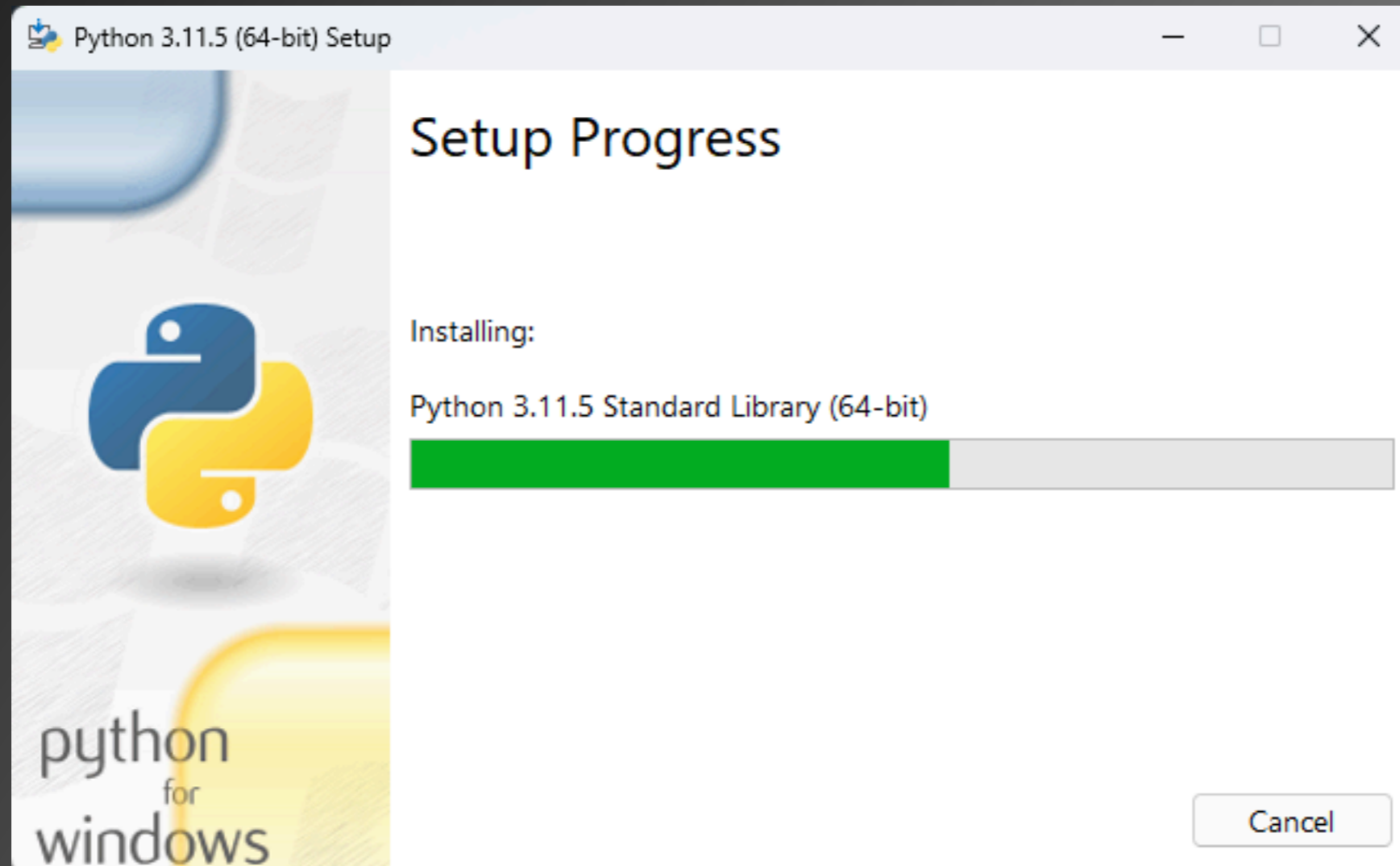
Install Python - Windows



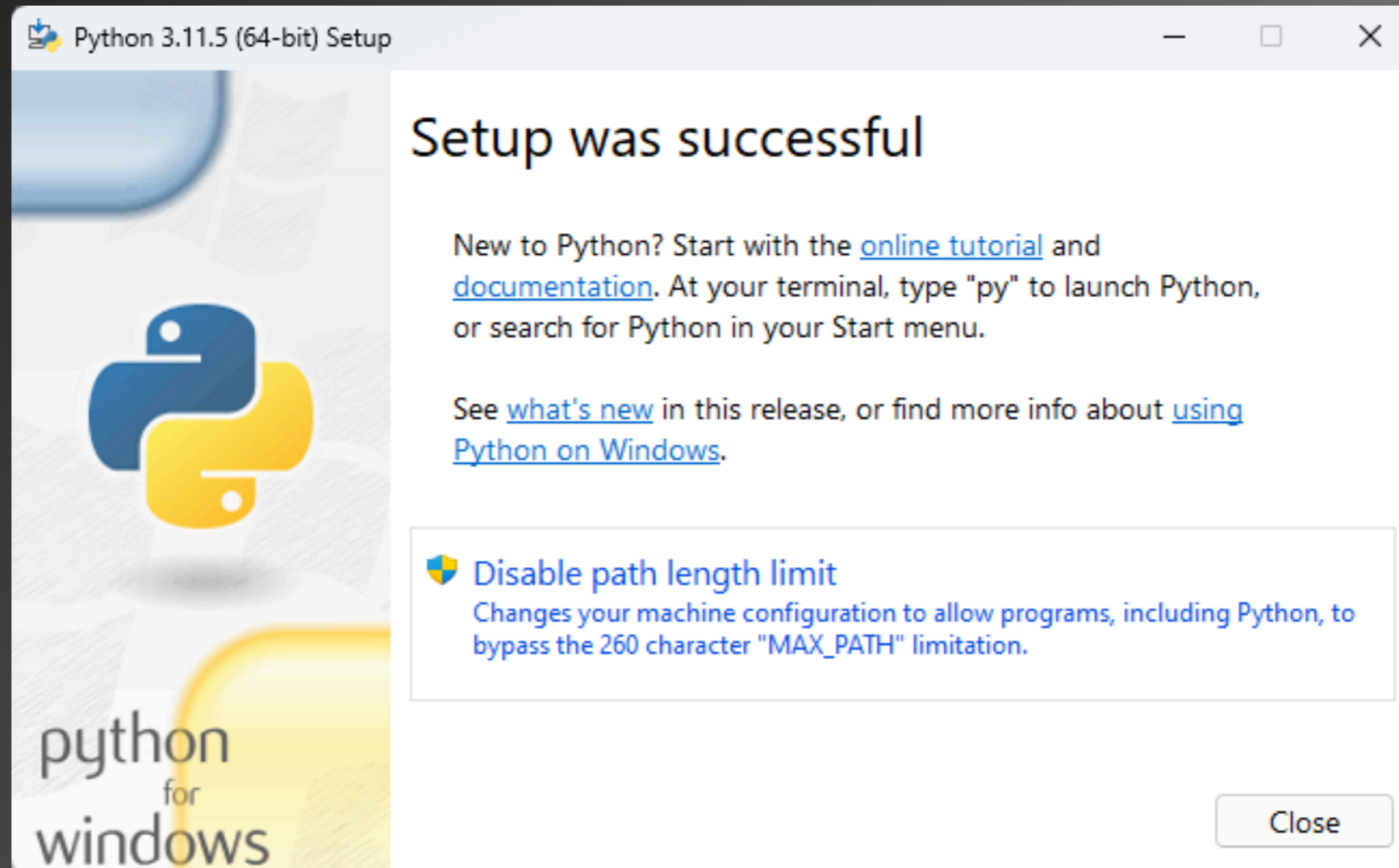
Install Python - Windows



Install Python - Windows



Install Python - Windows



Install Python - Windows

Python.org Windows Installer installs Python in

```
C:\Users\\AppData\Local\Programs\Python\Python311\
```

Python modules (e.g., seen using **pip list -v**) are located in

```
C:\Users\\AppData\Local\Programs\Python\Python311\
Lib\site-packages\
```



Install Python - Windows

Option #2: Microsoft Store



Simply

1. open the Microsoft Store and search for “python”,
or
2. open PowerShell/Command Prompt and just type **python** to bring up the Store.




Install Python - Windows

Microsoft Store

Search apps, games, movies, and more

Home
Apps
Gaming
Movies & TV



Python 3.11

Python Software Foundation

Get

4.3 ★ Average | 356 Ratings

The Python 3.11 interpreter and runtime

Developer tools

ESRB EVERYONE

Screenshots

```
Command Prompt - python3.11
Microsoft Windows [Version 10.0.22621.2134]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Pythonista>python3.11
Python 3.11.5 (tags/v3.11.5:cce6ba9, Aug 24 2023, 14:38:34) [MSC v.1936 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more.
>>> import sys
>>> print(f"Welcome, to Python {sys.version_info.major}.{sys.version_info.minor}")
Welcome, to Python 3.11
>>> |
```

Description

Python is an easy to learn, powerful programming language. It has efficient high-level data structures and a simple but effective approach to object-oriented programming. Python's elegant syntax and dynamic typing, together with its interpreted nature, make it an ideal language for scripting and rapid application development in many areas on most platforms.


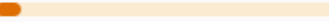
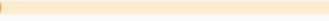

The Python interpreter and the extensive standard library are freely available in source or binary form for all major platforms from the Python web site, <https://www.python.org/>, and may be freely distributed. The same site also contains distributions of and pointers to many free third party Python modules, programs and tools, and additional documentation.

...

[Show more](#)

Ratings and reviews

4.3

5 ★ 
4 ★ 
3 ★ 
2 ★ 



Install Python - Windows

Microsoft Store installs Python in

```
C:\Users\\AppData\Local\Microsoft\WindowsApps\
```

Python modules (e.g., seen using **pip list -v**) are located in

```
C:\Users\\AppData\Local\Packages\  
PythonSoftwareFoundation.Python.3.11_... \LocalCache\  
local-packages\Python311\site-packages\
```



Install Python - Windows

Option #3: Chocolatey

The Package Manager for Windows

<https://chocolatey.org/>



Simply open PowerShell as an administrative shell (i.e., “Run as Administrator”) and enter

```
choco install python
```



Install Python - Windows

```
Administrator: Windows PowerShell
1 packages installed.
PS C:\Users\frank> dir /

Directory: C:\

Mode                LastWriteTime         Length Name
----                -
d-----           5/6/2022  10:24 PM             PerfLogs
d-r---           9/2/2023   3:43 PM             Program Files
d-r---           5/6/2022  11:10 PM             Program Files (x86)
d-r---           3/23/2023   3:30 PM             Users
d-----           9/2/2023   3:52 PM             Windows

PS C:\Users\frank> choco install python
Chocolatey v2.2.2
Installing the following packages:
python
By installing, you accept licenses for the packages.
Progress: Downloading chocolatey-compatibility.extension 1.0.0... 100%

chocolatey-compatibility.extension v1.0.0 [Approved]
chocolatey-compatibility.extension package files install completed. Performing other installation steps.
Installed/updated chocolatey-compatibility extensions.
The install of chocolatey-compatibility.extension was successful.
Software installed to 'C:\ProgramData\chocolatey\extensions\chocolatey-compatibility'
Progress: Downloading chocolatey-core.extension 1.4.0... 100%

chocolatey-core.extension v1.4.0 [Approved]
chocolatey-core.extension package files install completed. Performing other installation steps.
Installed/updated chocolatey-core extensions.
The install of chocolatey-core.extension was successful.
Software installed to 'C:\ProgramData\chocolatey\extensions\chocolatey-core'
Progress: Downloading chocolatey-windowsupdate.extension 1.0.5... 100%

chocolatey-windowsupdate.extension v1.0.5 [Approved]
chocolatey-windowsupdate.extension package files install completed. Performing other installation steps.
Installed/updated chocolatey-windowsupdate extensions.
The install of chocolatey-windowsupdate.extension was successful.
Software installed to 'C:\ProgramData\chocolatey\extensions\chocolatey-windowsupdate'
Progress: Downloading KB2919442 1.0.20160915... 100%

KB2919442 v1.0.20160915 [Approved]
KB2919442 package files install completed. Performing other installation steps.
The package KB2919442 wants to run 'ChocolateyInstall.ps1'.
Note: If you don't run this script, the installation will fail.
Note: To confirm automatically next time, use '-y' or consider:
choco feature enable -n allowGlobalConfirmation
Do you want to run the script?([Y]es/[A]ll - yes to all/[N]o/[P]rint):
```



Install Python - Windows

```
Administrator: Windows PowerShell
Progress: Downloading vcredist2015 14.0.24215.20170201... 100%

vcredist2015 v14.0.24215.20170201 [Approved]
vcredist2015 package files install completed. Performing other installation steps.
The install of vcredist2015 was successful.
Software installed to 'C:\ProgramData\chocolatey\lib\vcredist2015'
Progress: Downloading python311 3.11.5... 100%

python311 v3.11.5 [Approved]
python311 package files install completed. Performing other installation steps.
Installing 64-bit python311...
python311 has been installed.
Added C:\ProgramData\chocolatey\bin\python3.11.exe shim pointed to 'c:\python311\python.exe'.
Python installed to: 'C:\Python311'
Restricting write permissions to Administrators
python311 can be automatically uninstalled.
Environment Vars (like PATH) have changed. Close/reopen your shell to
see the changes (or in powershell/cmd.exe just type `refreshenv`).
The install of python311 was successful.
Software installed as 'exe', install location is likely default.
Progress: Downloading python3 3.11.5... 100%

python3 v3.11.5 [Approved]
python3 package files install completed. Performing other installation steps.
The install of python3 was successful.
Software installed to 'C:\ProgramData\chocolatey\lib\python3'
Progress: Downloading python 3.11.5... 100%

python v3.11.5 [Approved]
python package files install completed. Performing other installation steps.
The install of python was successful.
Software installed to 'C:\ProgramData\chocolatey\lib\python'

Chocolatey installed 13/13 packages.
See the log for details (C:\ProgramData\chocolatey\logs\chocolatey.log).

Installed:
- chocolatey-compatibility.extension v1.0.0
- chocolatey-core.extension v1.4.0
- chocolatey-windowsupdate.extension v1.0.5
- KB2919355 v1.0.20160915
- KB2919442 v1.0.20160915
- KB2999226 v1.0.20181019
- KB3033929 v1.0.5
- KB3035131 v1.0.3
- python v3.11.5
- python3 v3.11.5
- python311 v3.11.5
- vcredist140 v14.36.32532
- vcredist2015 v14.0.24215.20170201
PS C:\Users\frank>
```



Install Python - Windows

Microsoft Store installs Python in

```
C:\Python311\
```

Python modules (e.g., seen using **pip list -v**) are located in

```
C:\Python311\Lib\site-packages\
```



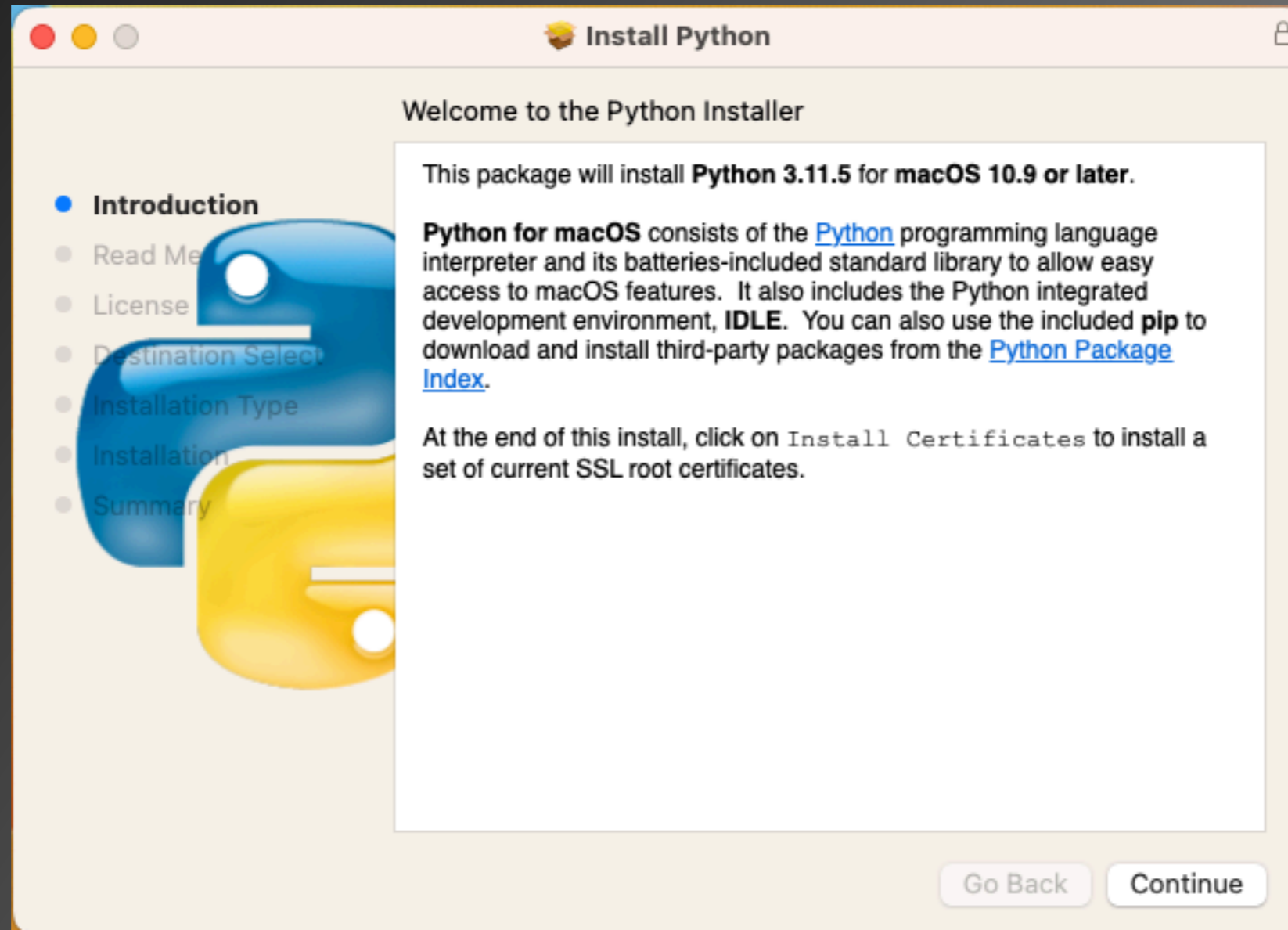


Installing Python

for macOS



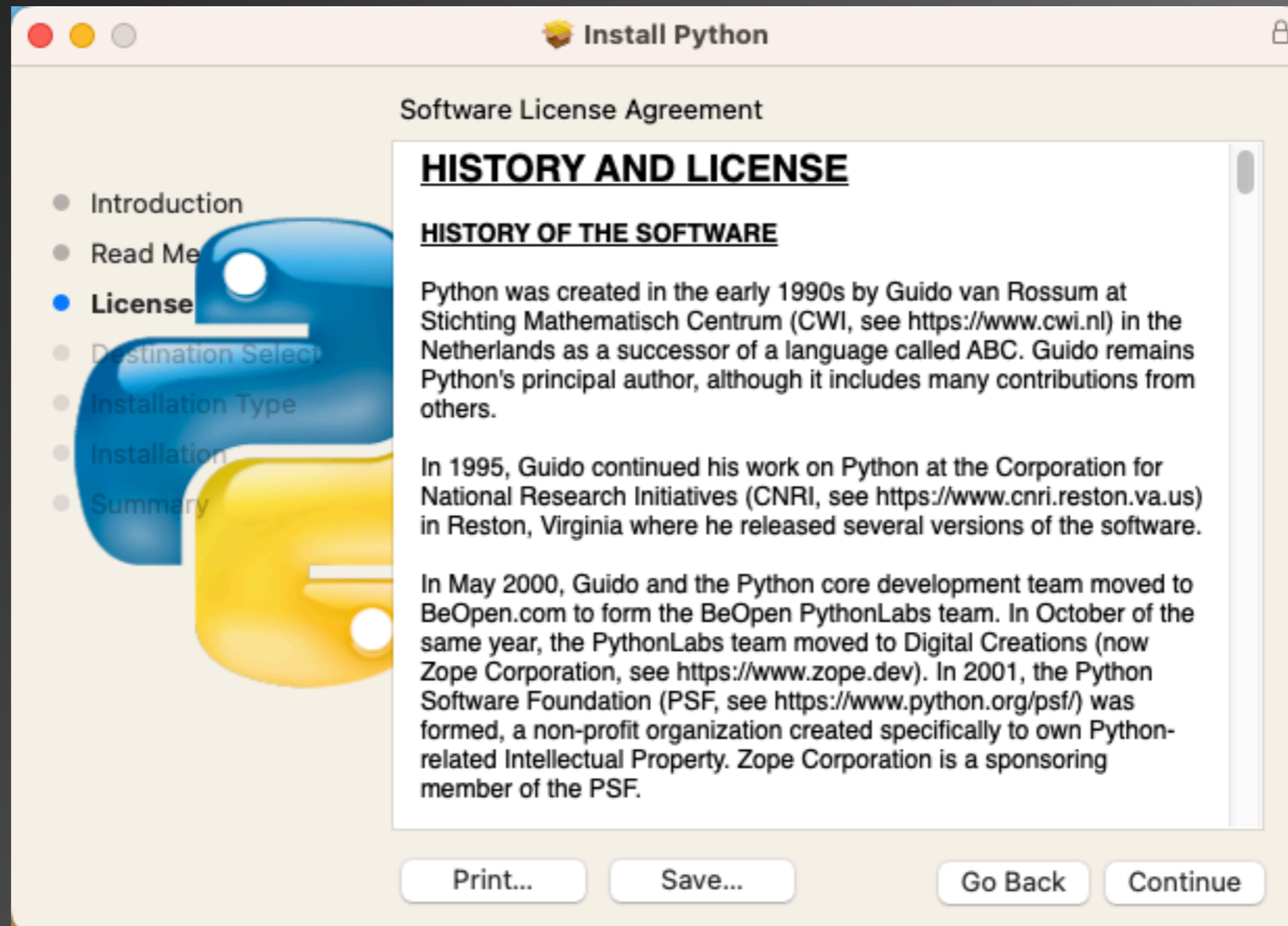
Install Python - macOS



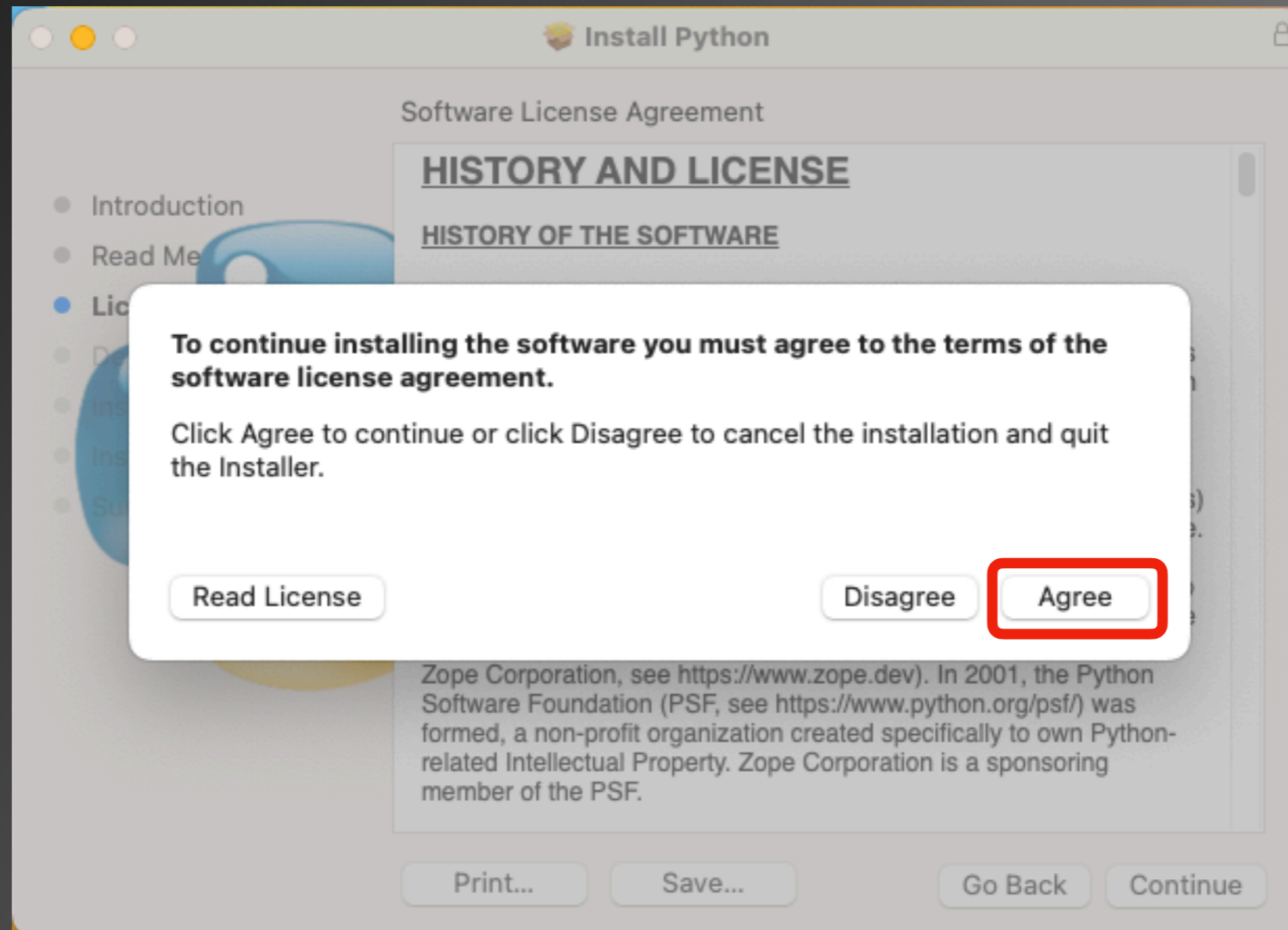
Install Python - macOS



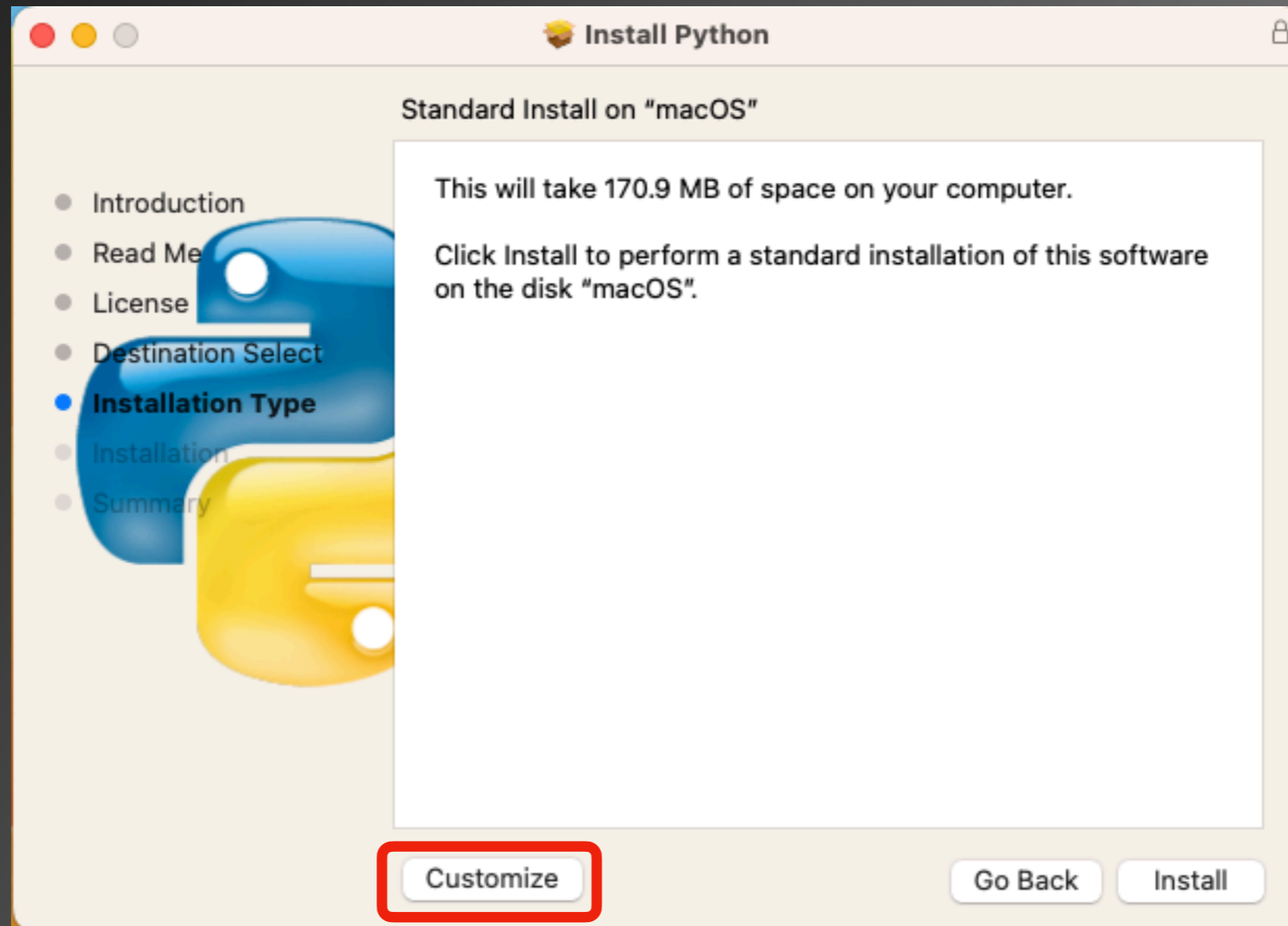
Install Python - macOS



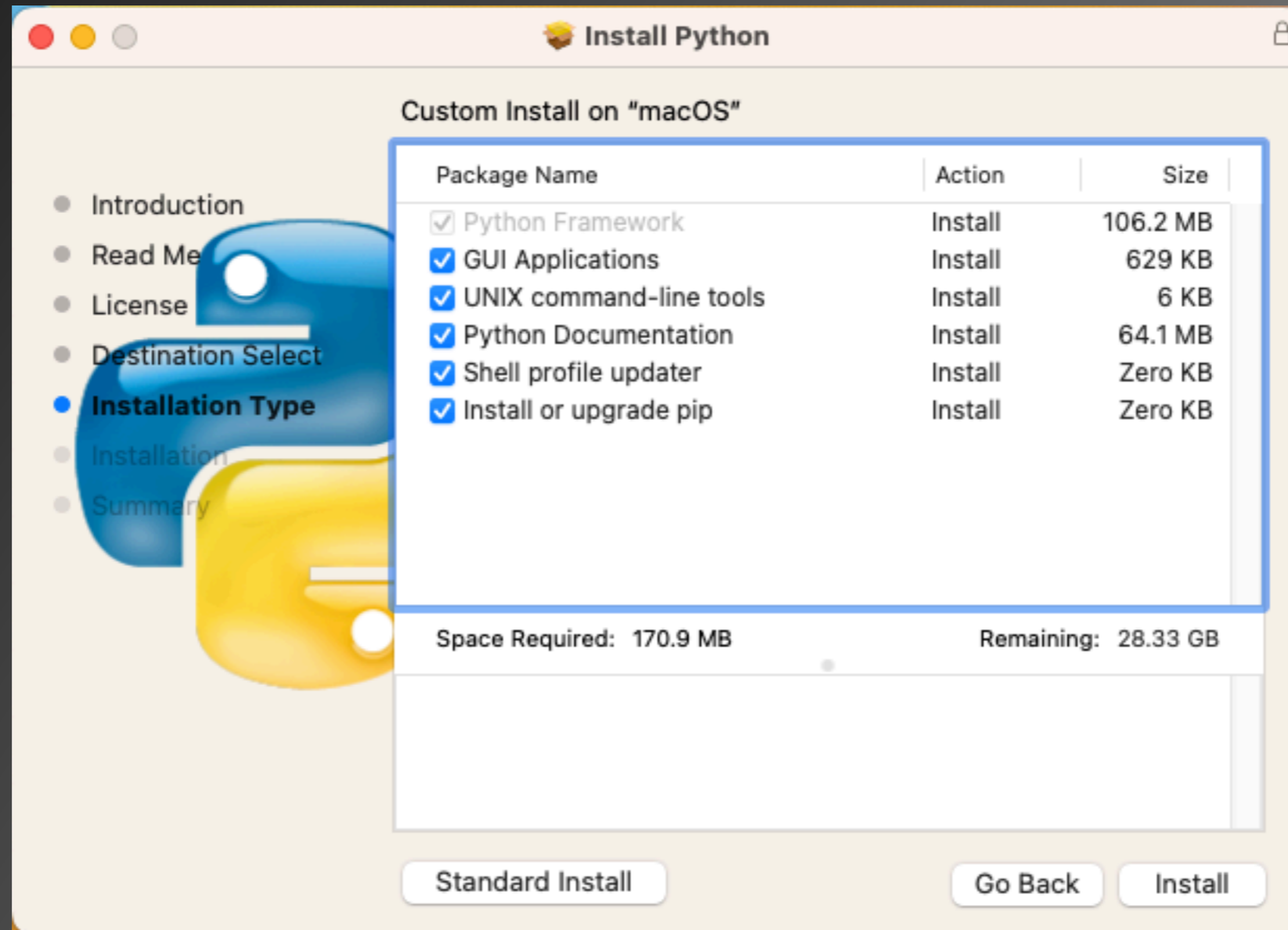
Install Python - macOS



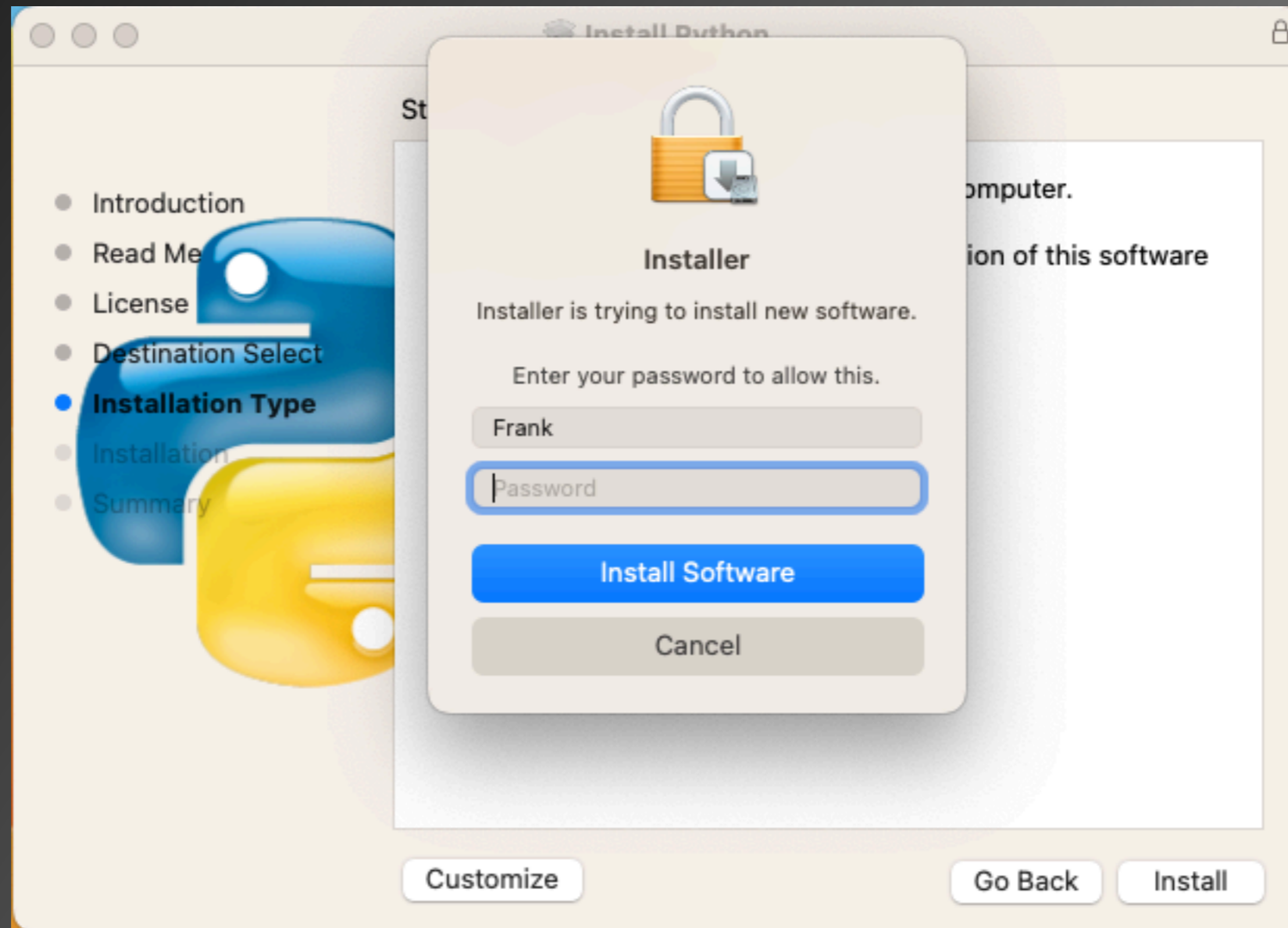
Install Python - macOS



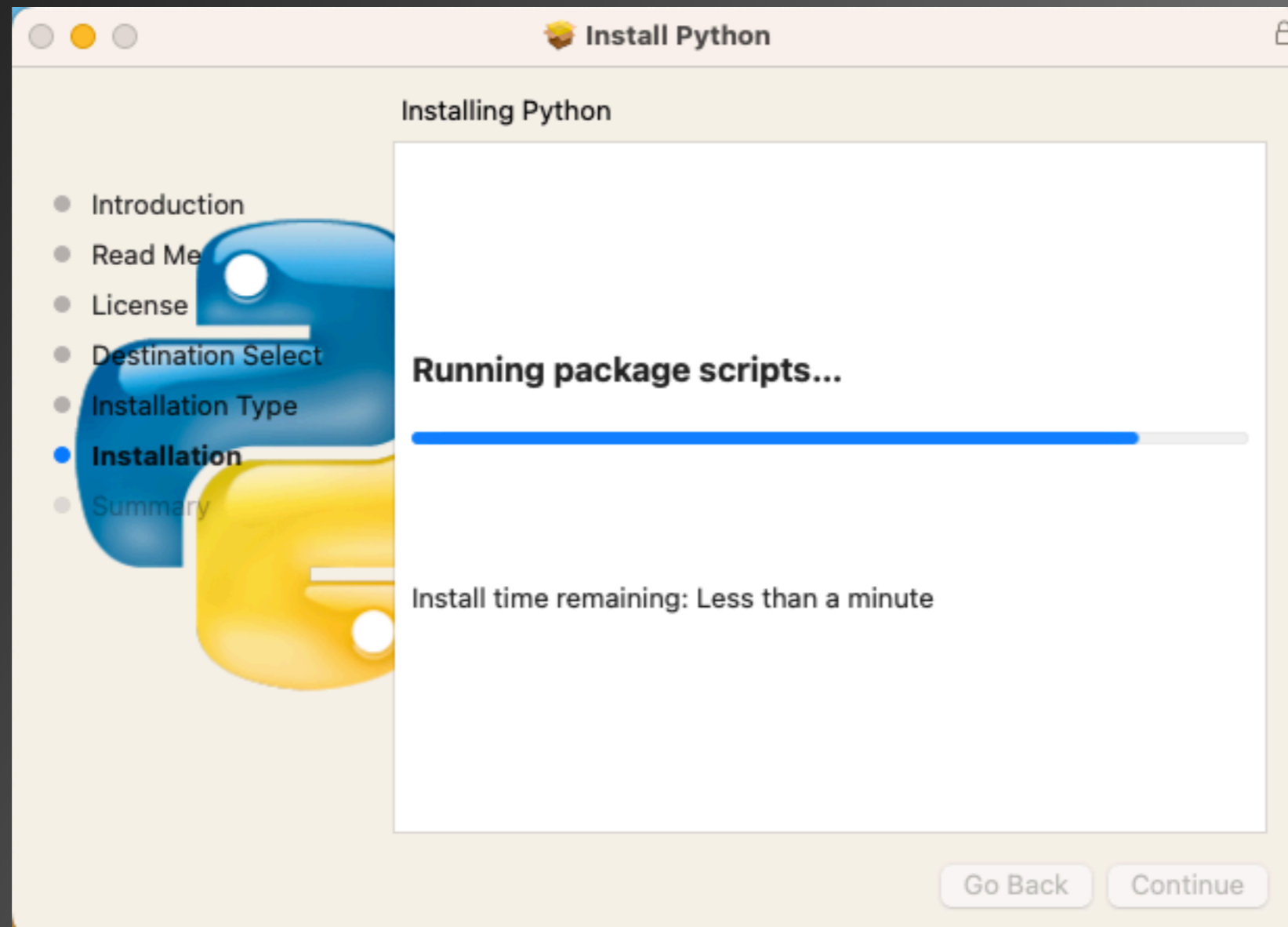
Install Python - macOS



Install Python - macOS



Install Python - macOS



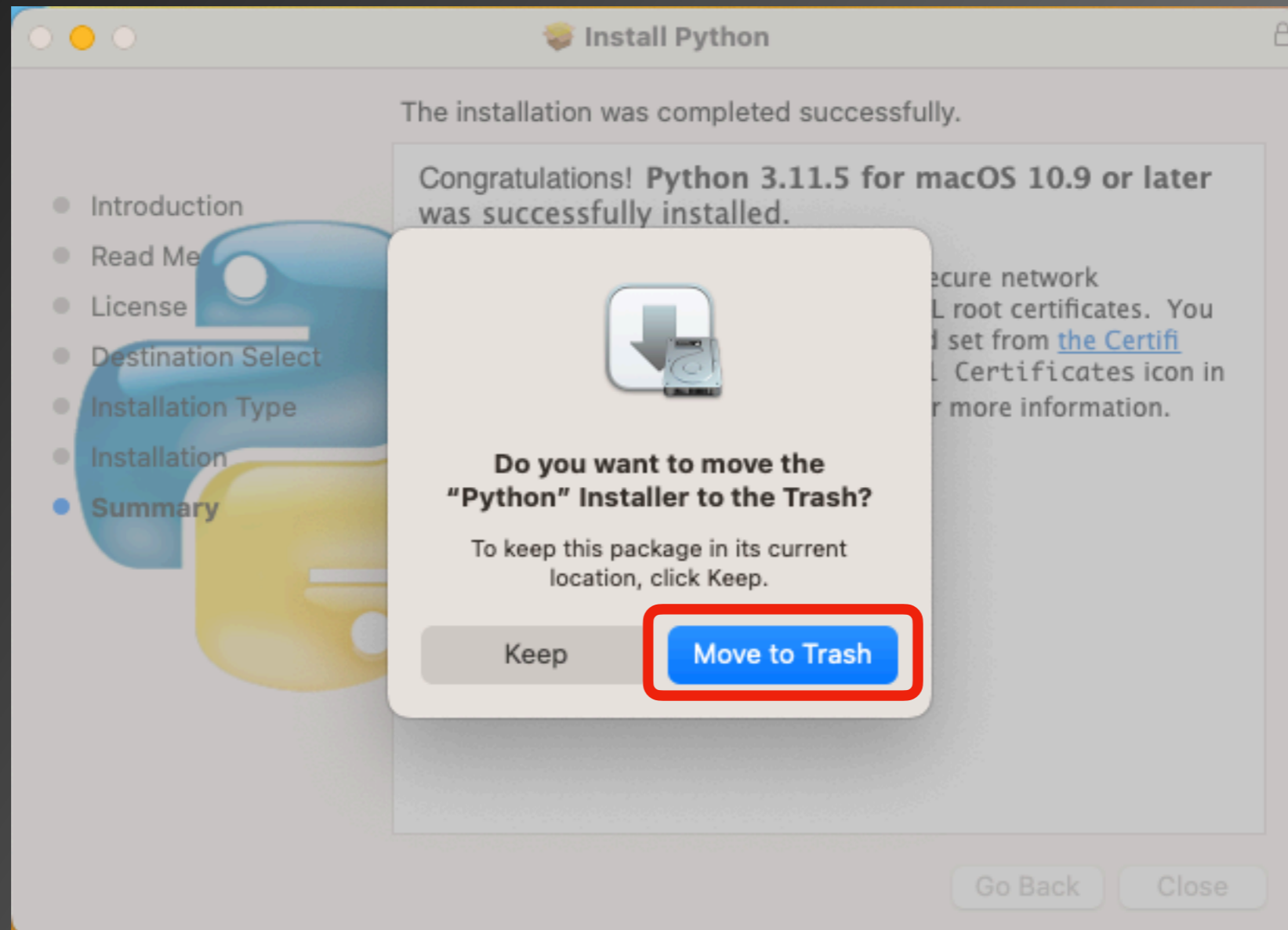
Install Python - macOS



Install Python - macOS



Install Python - macOS



Install Python - macOS

Python.org macOS Installer installs Python in

```
/Library/Frameworks/Python.framework/Versions/3.11/
```

Python modules (e.g., seen using **pip3 list -v**) are located in

```
/Library/Frameworks/Python.framework/Versions/3.11/lib/python3.11/site-packages/
```



Install Python - macOS



Option #2: Homebrew

The Missing Package Manager for macOS (or Linux)

<https://brew.sh/>

Simply having Homebrew installed provides you with a version of Python3 (it comes with the XCode Command Line Tools that Homebrew installs). However, it is not the latest. To update to the current version, simply open Terminal and enter

```
brew install python
```



Install Python - macOS

```
bin — ruby -W1 --disable=gems,rubyopt /usr/local/Homebrew/Library/Homebrew/brew.rb install python — 122x46
[frank@Franks-Mac bin % brew install python
==> Downloading https://formulae.brew.sh/api/formula.json
[#=#- #
==> Downloading https://formulae.brew.sh/api/cask.json

==> Fetching dependencies for python@3.11: mpdecimal, ca-certificates, openssl@3, readline, sqlite and xz
==> Fetching mpdecimal
==> Downloading https://ghcr.io/v2/homebrew/core/mpdecimal/manifests/2.5.1
##### 100.0%
==> Downloading https://ghcr.io/v2/homebrew/core/mpdecimal/blobs/sha256:91f795d74747bf8723022ac813f3f81d71fefb77
##### 100.0%
==> Fetching ca-certificates
==> Downloading https://ghcr.io/v2/homebrew/core/ca-certificates/manifests/2023-08-22
##### 100.0%
==> Downloading https://ghcr.io/v2/homebrew/core/ca-certificates/blobs/sha256:a331e92e7a759571296581f029e5cc2ec7
##### 100.0%
==> Fetching openssl@3
==> Downloading https://ghcr.io/v2/homebrew/core/openssl/3/manifests/3.1.2-1
##### 100.0%
==> Downloading https://ghcr.io/v2/homebrew/core/openssl/3/blobs/sha256:2bea791e9eacc59e0a9099065f3229afaf2b68a9
##### 100.0%
==> Fetching readline
==> Downloading https://ghcr.io/v2/homebrew/core/readline/manifests/8.2.1
##### 100.0%
==> Downloading https://ghcr.io/v2/homebrew/core/readline/blobs/sha256:abe9d3f3eec3ba2339860faa6a978b9909194c65c
##### 100.0%
==> Fetching sqlite
==> Downloading https://ghcr.io/v2/homebrew/core/sqlite/manifests/3.43.0_1
##### 100.0%
==> Downloading https://ghcr.io/v2/homebrew/core/sqlite/blobs/sha256:273c47c1769f04c5f5ff3ac5cb9b4d6ac8b29284029
##### 100.0%
==> Fetching xz
==> Downloading https://ghcr.io/v2/homebrew/core/xz/manifests/5.4.4
##### 100.0%
==> Downloading https://ghcr.io/v2/homebrew/core/xz/blobs/sha256:4c25f68798c0b4c9b869e78fd9cd7f8f723c51ea56d6
##### 100.0%
==> Fetching python@3.11
==> Downloading https://ghcr.io/v2/homebrew/core/python/3.11/manifests/3.11.5
##### 100.0%
==> Downloading https://ghcr.io/v2/homebrew/core/python/3.11/blobs/sha256:c87f0729bff2c3ab0cb3a66f7187ff0c621eed
##### 100.0%
==> Installing dependencies for python@3.11: mpdecimal, ca-certificates, openssl@3, readline, sqlite and xz
==> Installing python@3.11 dependency: mpdecimal
==> Pouring mpdecimal--2.5.1.ventura.bottle.tar.gz
🍺 /usr/local/Cellar/mpdecimal/2.5.1: 71 files, 2MB
==> Installing python@3.11 dependency: ca-certificates
```



Install Python - macOS

```
bin — -zsh — 122x46

pip3 install <package>
They will install into the site-package directory
/usr/local/lib/python3.11/site-packages

tkinter is no longer included with this formula, but it is available separately:
brew install python-tk@3.11

gdbm (`dbm.gnu`) is no longer included in this formula, but it is available separately:
brew install python-gdbm@3.11
`dbm.ndbm` changed database backends in Homebrew Python 3.11.
If you need to read a database from a previous Homebrew Python created via `dbm.ndbm`,
you'll need to read your database using the older version of Homebrew Python and convert to another format.
`dbm` still defaults to `dbm.gnu` when it is installed.

For more information about Homebrew and Python, see: https://docs.brew.sh/Homebrew-and-Python
==> Summary
📦 /usr/local/Cellar/python@3.11/3.11.5: 3,287 files, 61MB
==> Running `brew cleanup python@3.11`...
Disable this behaviour by setting HOMEBREW_NO_INSTALL_CLEANUP.
Hide these hints with HOMEBREW_NO_ENV_HINTS (see `man brew`).
==> Caveats
==> python@3.11
Python has been installed as
/usr/local/bin/python3

Unversioned symlinks `python`, `python-config`, `pip` etc. pointing to
`python3`, `python3-config`, `pip3` etc., respectively, have been installed into
/usr/local/opt/python@3.11/libexec/bin

You can install Python packages with
pip3 install <package>
They will install into the site-package directory
/usr/local/lib/python3.11/site-packages

tkinter is no longer included with this formula, but it is available separately:
brew install python-tk@3.11

gdbm (`dbm.gnu`) is no longer included in this formula, but it is available separately:
brew install python-gdbm@3.11
`dbm.ndbm` changed database backends in Homebrew Python 3.11.
If you need to read a database from a previous Homebrew Python created via `dbm.ndbm`,
you'll need to read your database using the older version of Homebrew Python and convert to another format.
`dbm` still defaults to `dbm.gnu` when it is installed.

For more information about Homebrew and Python, see: https://docs.brew.sh/Homebrew-and-Python
frank@Franks-Mac bin %
```

Install Python - macOS

Homebrew macOS Installer installs Python in

```
/usr/local/bin/
```

Python modules (e.g., seen using **pip3 list -v**) are located in

```
/usr/local/lib/python3.11/site-packages/
```



Install Python - macOS

Option #3: MacPorts

Mac Ports

An open-source community initiative to design an easy-to-use system for compiling, installing, and upgrading either command-line, X11 or Aqua based open-source software on the Mac operating system

<https://www.macports.org/>

To install Python, simply open Terminal and enter

```
sudo port install python311 py311-pip
```



Install Python - macOS

```
frank — tcsh8.6 — sudo — 142x46
Last login: Sun Sep  3 18:40:11 on ttys000
[frank@Franks-Mac ~ % sudo port install python311 py311-pip
Password:
----> Computing dependencies for python311
The following dependencies will be installed:
bzip2
expat
gettext-runtime
libedit
libffi
libiconv
ncurses
openssl
openssl3
python3_select
python_select
sqlite3
xz
zlib
Continue? [Y/n]:
----> Fetching archive for bzip2
Warning: Your DNS servers incorrectly claim to know the address of nonexistent hosts. This may cause checksum mismatches for some ports. See t
his page for more information: <https://trac.macports.org/wiki/MisbehavingServers>
----> Attempting to fetch bzip2-1.0.8_0.darwin_22.x86_64.tbz2 from https://packages.macports.org/bzip2
----> Attempting to fetch bzip2-1.0.8_0.darwin_22.x86_64.tbz2.rmd160 from https://packages.macports.org/bzip2
----> Installing bzip2 @1.0.8_0
----> Activating bzip2 @1.0.8_0
----> Cleaning bzip2
----> Fetching archive for expat
----> Attempting to fetch expat-2.5.0_0.darwin_22.x86_64.tbz2 from https://packages.macports.org/expat
----> Attempting to fetch expat-2.5.0_0.darwin_22.x86_64.tbz2.rmd160 from https://packages.macports.org/expat
----> Installing expat @2.5.0_0
----> Activating expat @2.5.0_0
----> Cleaning expat
----> Fetching archive for libiconv
----> Attempting to fetch libiconv-1.17_0.darwin_22.x86_64.tbz2 from https://packages.macports.org/libiconv
----> Attempting to fetch libiconv-1.17_0.darwin_22.x86_64.tbz2.rmd160 from https://packages.macports.org/libiconv
----> Installing libiconv @1.17_0
----> Activating libiconv @1.17_0
----> Cleaning libiconv
----> Fetching archive for gettext-runtime
----> Attempting to fetch gettext-runtime-0.21.1_0.darwin_22.x86_64.tbz2 from https://packages.macports.org/gettext-runtime
----> Attempting to fetch gettext-runtime-0.21.1_0.darwin_22.x86_64.tbz2.rmd160 from https://packages.macports.org/gettext-runtime
----> Installing gettext-runtime @0.21.1_0
----> Activating gettext-runtime @0.21.1_0
----> Cleaning gettext-runtime
```

Install Python - macOS

```
frank — -zsh — 142x46
----> Activating xz @5.4.4_0
----> Cleaning xz
----> Fetching archive for python311
----> Attempting to fetch python311-3.11.5_0+lto+optimizations.darwin_22.x86_64.tbz2 from https://packages.macports.org/python311
----> Attempting to fetch python311-3.11.5_0+lto+optimizations.darwin_22.x86_64.tbz2.rmd160 from https://packages.macports.org/python311
----> Installing python311 @3.11.5_0+lto+optimizations
----> Activating python311 @3.11.5_0+lto+optimizations
----> Cleaning python311
----> Computing dependencies for py311-pip
The following dependencies will be installed:
  pip_select
  py311-setuptools
Continue? [Y/n]:
----> Fetching archive for pip_select
----> Attempting to fetch pip_select-0.1_3.darwin_22.noarch.tbz2 from https://packages.macports.org/pip_select
----> Attempting to fetch pip_select-0.1_3.darwin_22.noarch.tbz2.rmd160 from https://packages.macports.org/pip_select
----> Installing pip_select @0.1_3
----> Activating pip_select @0.1_3
----> Cleaning pip_select
----> Fetching archive for py311-setuptools
----> Attempting to fetch py311-setuptools-68.1.2_0.darwin_any.noarch.tbz2 from https://packages.macports.org/py311-setuptools
----> Attempting to fetch py311-setuptools-68.1.2_0.darwin_any.noarch.tbz2.rmd160 from https://packages.macports.org/py311-setuptools
----> Installing py311-setuptools @68.1.2_0
----> Activating py311-setuptools @68.1.2_0
----> Cleaning py311-setuptools
----> Fetching archive for py311-pip
----> Attempting to fetch py311-pip-23.2.1_0.darwin_any.noarch.tbz2 from https://packages.macports.org/py311-pip
----> Attempting to fetch py311-pip-23.2.1_0.darwin_any.noarch.tbz2.rmd160 from https://packages.macports.org/py311-pip
----> Installing py311-pip @23.2.1_0
----> Activating py311-pip @23.2.1_0
----> Cleaning py311-pip
----> Updating database of binaries
----> Scanning binaries for linking errors
----> No broken files found.
----> No broken ports found.
----> Some of the ports you installed have notes:
  py311-pip has the following notes:
    To make the Python 3.11 version of pip the one that is run when you execute the commands without a version suffix, e.g. 'pip', run:
    sudo port select --set pip pip311
    sudo port select --set pip3 pip311
  python311 has the following notes:
    To make this the default Python or Python 3 (i.e., the version run by the 'python' or 'python3' commands), run one or both of:

    sudo port select --set python python311
    sudo port select --set python3 python311
frank@Franks-Mac ~ %
```



Install Python - macOS

MacPorts macOS Installer installs Python in

```
/opt/local/bin/
```

Python modules (e.g., seen using **pip3 list -v**) are located in

```
/opt/local/Library/Frameworks/Python.framework/Versions/  
3.11/lib/python3.11/site-packages/
```





Installing Python

for Linux



Install Python - Linux

RHEL/CENTOS/Rocky/Alma Linux

```
rpm/yum/dnf install python3
```

Ubuntu/Debian Linux

```
apt install python3
```



WSL



Python Basics



Python REPL

REPL = Read, Evaluate, Print, and Loop

```
$ python3
Python 3.11.5 (v3.11.5:cce6ba91b3, Aug 24
2023, 10:50:31) [Clang 13.0.0
(clang-1300.0.29.30)] on darwin
Type "help", "copyright", "credits" or
"license" for more information.
>>> print("Hello world")
Hello world
>>>
```

To exit the REPL, hit [CTRL][D] or type `exit()`.



First Python Script

1. In a text editor write

```
#!/usr/bin/python3  
print("Hello world!")
```

2. Save this to **myfirst.py**
3. Open a terminal, navigate to where this file is located, and run

```
python3 myfirst.py
```



pip

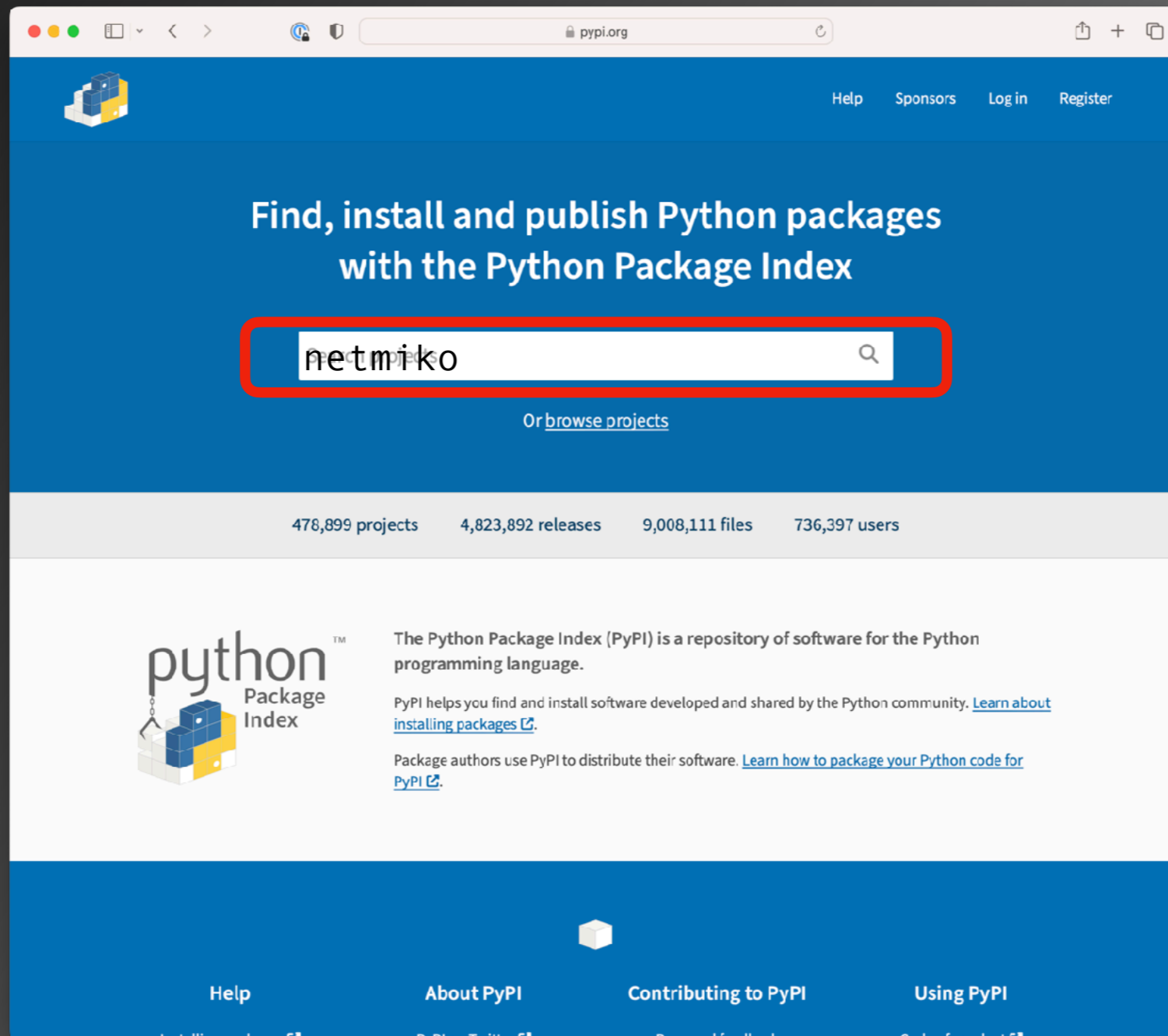
pip is the package installer for Python. You can use pip to install packages from the Python Package Index and other indexes.

e.g.,

```
pip install requests  
pip install netmiko  
pip install gspread
```



Python Package Index (PyPI)

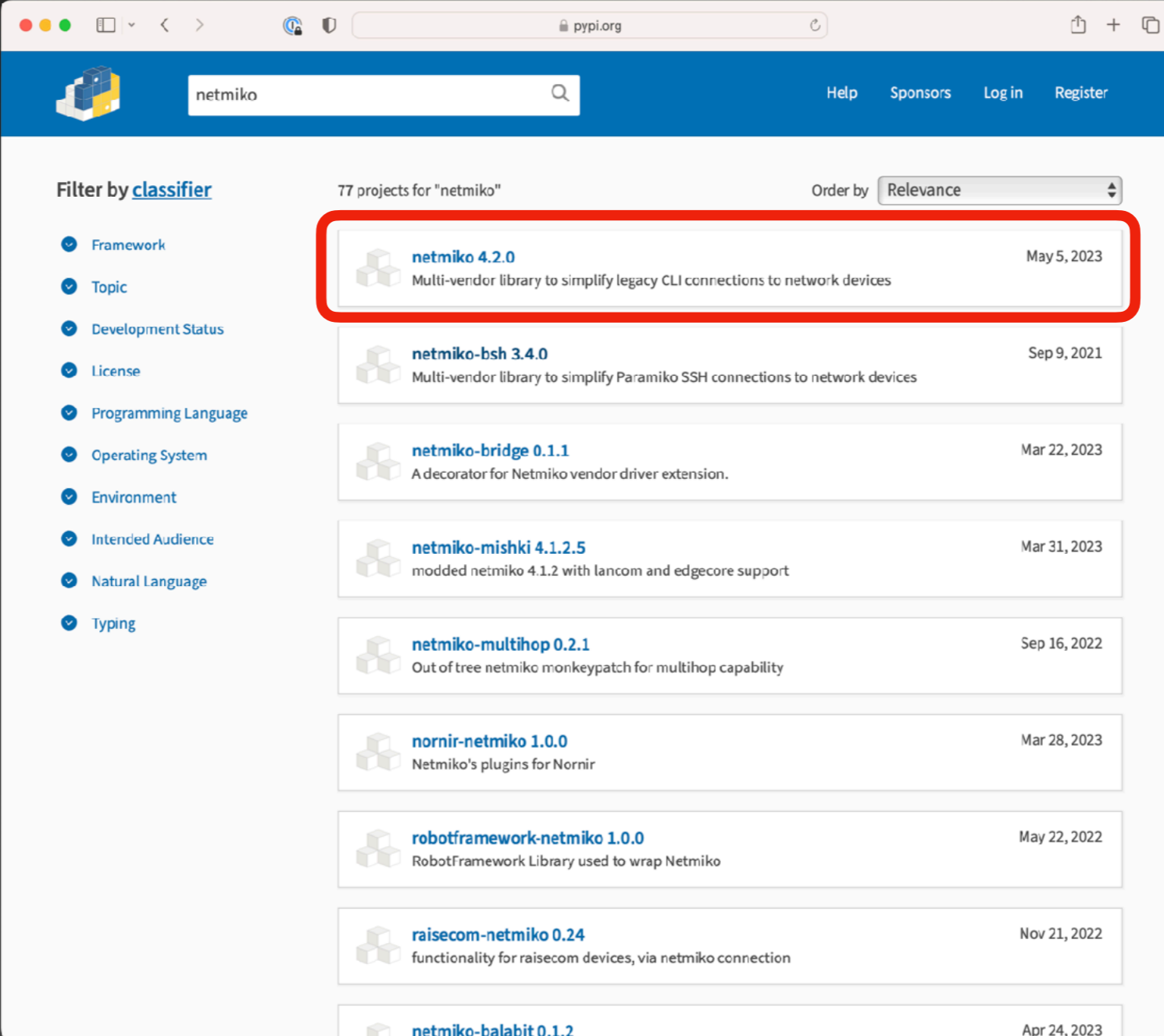


The screenshot shows the PyPI website interface. At the top, there is a navigation bar with links for Help, Sponsors, Log in, and Register. The main heading reads "Find, install and publish Python packages with the Python Package Index". Below this is a search bar containing the text "netmiko", which is highlighted with a red rectangular border. Underneath the search bar is a link that says "Or browse projects". A statistics bar displays: 478,899 projects, 4,823,892 releases, 9,008,111 files, and 736,397 users. The footer contains the Python Package Index logo and a description: "The Python Package Index (PyPI) is a repository of software for the Python programming language." It also includes links for "Learn about installing packages" and "Learn how to package your Python code for PyPI". At the very bottom, there are links for Help, About PyPI, Contributing to PyPI, and Using PyPI.

<https://www.pypi.org/>



Python Package Index (PyPI)



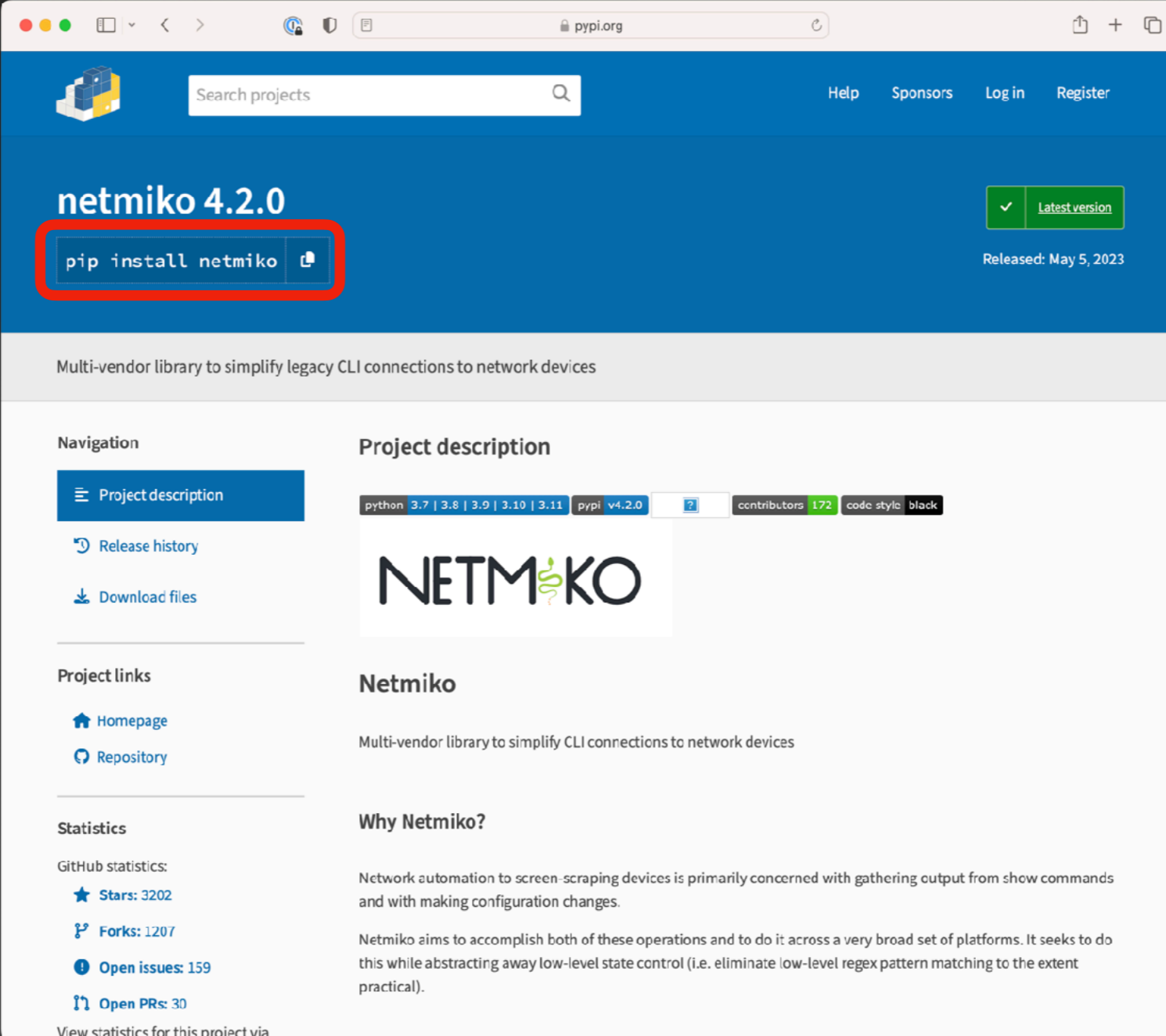
The screenshot shows the PyPI website interface. At the top, there is a search bar with the text "netmiko" and a search icon. To the right of the search bar are links for "Help", "Sponsors", "Log in", and "Register". Below the search bar, there is a "Filter by classifier" section on the left with various filters like Framework, Topic, Development Status, License, Programming Language, Operating System, Environment, Intended Audience, Natural Language, and Typing. The main content area shows "77 projects for 'netmiko'" and a dropdown menu for "Order by" set to "Relevance". A red box highlights the first result, "netmiko 4.2.0", which is described as a "Multi-vendor library to simplify legacy CLI connections to network devices" and was released on "May 5, 2023". Other results include "netmiko-bsh 3.4.0", "netmiko-bridge 0.1.1", "netmiko-mishki 4.1.2.5", "netmiko-multihop 0.2.1", "nornir-netmiko 1.0.0", "robotframework-netmiko 1.0.0", "raisecom-netmiko 0.24", and "netmiko-balabit 0.1.2".

Package Name	Description	Release Date
netmiko 4.2.0	Multi-vendor library to simplify legacy CLI connections to network devices	May 5, 2023
netmiko-bsh 3.4.0	Multi-vendor library to simplify Paramiko SSH connections to network devices	Sep 9, 2021
netmiko-bridge 0.1.1	A decorator for Netmiko vendor driver extension.	Mar 22, 2023
netmiko-mishki 4.1.2.5	modded netmiko 4.1.2 with lancom and edgecore support	Mar 31, 2023
netmiko-multihop 0.2.1	Out of tree netmiko monkeypatch for multihop capability	Sep 16, 2022
nornir-netmiko 1.0.0	Netmiko's plugins for Nornir	Mar 28, 2023
robotframework-netmiko 1.0.0	RobotFramework Library used to wrap Netmiko	May 22, 2022
raisecom-netmiko 0.24	functionality for raisecom devices, via netmiko connection	Nov 21, 2022
netmiko-balabit 0.1.2		Apr 24, 2023

<https://www.pypi.org/>



Python Package Index (PyPI)



The screenshot shows the PyPI website for the `netmiko` package, version 4.2.0. The package is highlighted as the "Latest version" and was released on May 5, 2023. A red box highlights the installation command: `pip install netmiko`. The page includes a search bar, navigation links (Help, Sponsors, Log in, Register), and a description of the package as a "Multi-vendor library to simplify legacy CLI connections to network devices". The left sidebar contains navigation options (Project description, Release history, Download files), project links (Homepage, Repository), and statistics (Stars: 3202, Forks: 1207, Open issues: 159, Open PRs: 30). The main content area includes a "Project description" section with a logo and a "Why Netmiko?" section explaining its purpose in network automation.

<https://www.pypi.org/>



venv

The `venv` module supports creating lightweight “virtual environments”, each with their own independent set of Python packages installed in their site directories. A virtual environment is created on top of an existing Python installation, known as the virtual environment’s “base” Python, and may optionally be isolated from the packages in the base environment, so only those explicitly installed in the virtual environment are available.

- <https://docs.python.org/3/library/venv.html>



venv

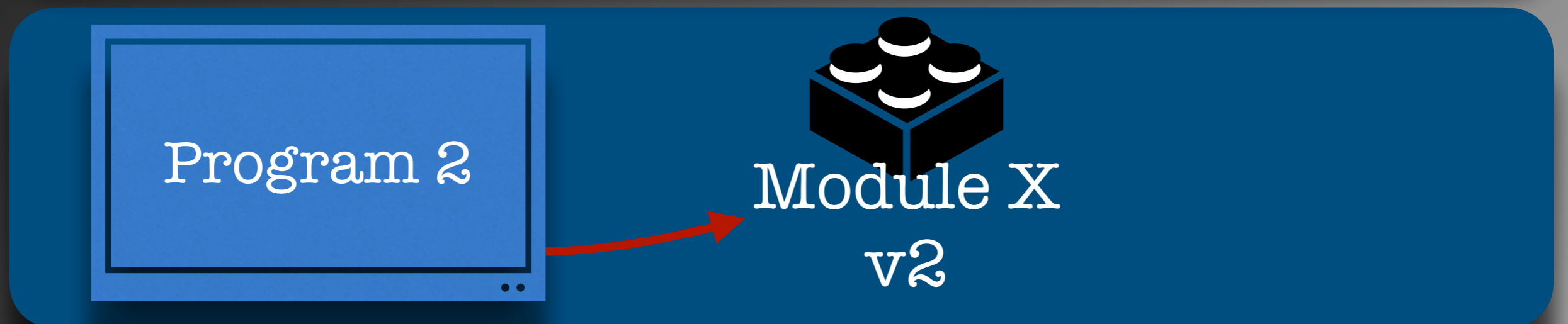
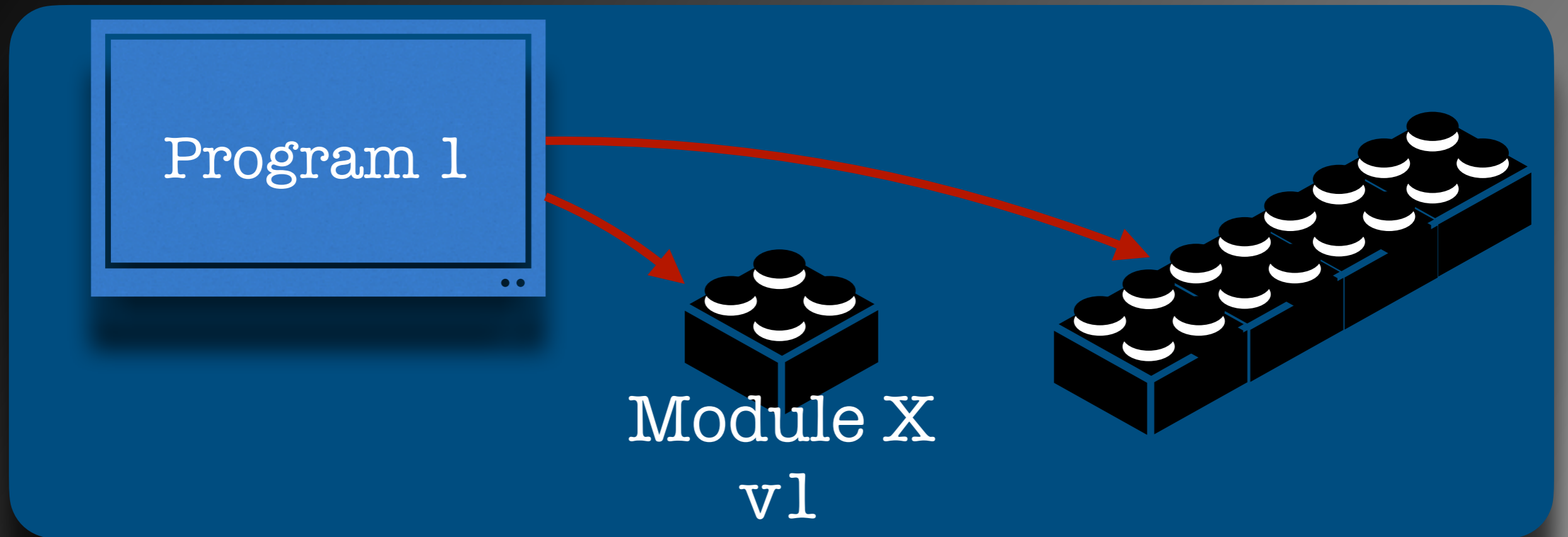
So... why?

Once you begin using Python, you will inevitably encounter situations where one Python program expects a module v1 while another only works with v2. If all Python scripts are in the same environment...
KABOOM!

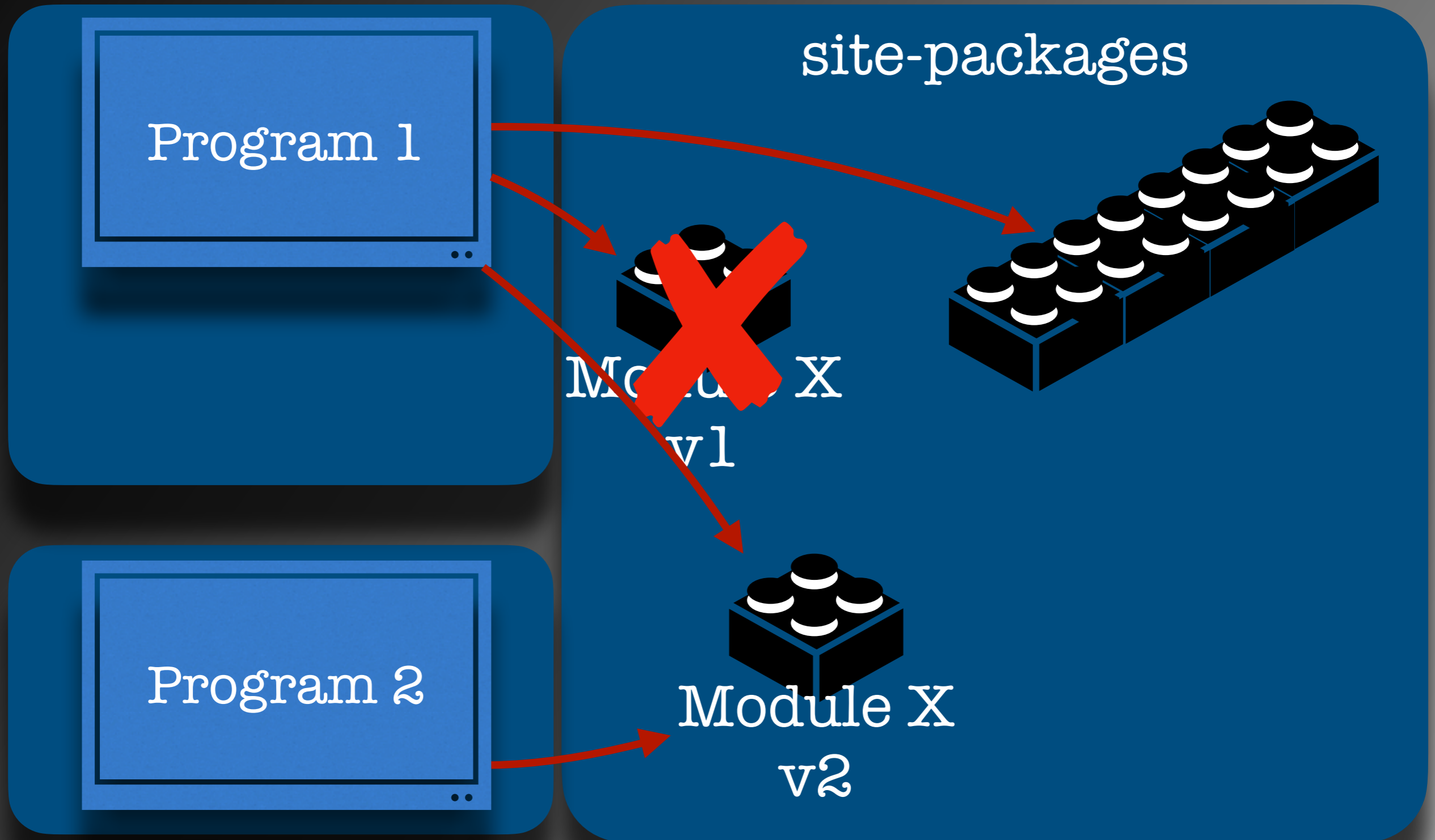
Virtual environments allow you to isolate/separate different Python programs from each other and provide each Python program with the modules and versions it expects.



Why We Need venv



Why We Need venv



venv

For example, you might do the following:

```
$ python3 -m venv venv
$ ls -l venv
bin
include
lib
pyvenv.cfg
$ source venv/bin/activate
(venv) $ pip list
```

This tells the Python interpreter to run module (-m) **venv** and create a new virtual environment in a directory named 'venv' in the current directory. We then activate that virtual environment.



IDE

“An integrated development environment (IDE) is a software application that provides comprehensive facilities for software development. An IDE normally consists of at least a source-code editor, build automation tools, and a debugger.”

- https://en.wikipedia.org/wiki/Integrated_development_environment

Examples:

- IDLE
- Visual Studio Code (VSCode) / VSCodium
- PyCharm



IDLE

```
Python 3.11.5 (tags/v3.11.5:cce6ba9, Aug 24 2023, 14:38:34) [MSC v.1936 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> |
```

Installed with Python!

Visual Studio Code (VSCode)

code.visualstudio.com

Visual Studio Code Docs Updates Blog API Extensions FAQ Learn Search Docs Download

Version 1.81 is now available! Read about the new features and fixes from July.

Code editing. Redefined.

Free. Built on open source. Runs everywhere.

Download Mac Universal Stable Build

Web, Insiders edition, or other platforms

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

EXTENSIONS: MARKETPLACE

- Python 2019.6.24221 4.5
- DitLens - Git su... 4.6
- D/C++ 0.24.0 3.5
- ESLint 1.9.0 4.5
- Debugger for C... 4.11.0 4.4
- Language Sup... 0.47.0 4.5
- Java Linting, Intellisense, formatin... Red Hat
- vscode-icons 3.8.0 4.5
- Vetur 0.21.1 4.5

```
src > components > JS blog-post.js > <function> > @blogPost
1 import React from "react"
2 import React from "react"
3 import Image from "gatsby-image"
4
5 export default ({ data }) => {
6   const blogPost = data.cms.blogPost
7
8   return (
9     <div>
10      <blogPost>
11      <blogPost>
12      <Inac>
13    )
14    <div>
15    <div>
16    </div>
17  )
18  </div>
19
20
21 export const query = graphql`
```

PROBLEMS TERMINAL

Info [vscode]: Compiling...
[Docker] Compiled successfully in 25ms
Info [vscode]: Compiled successfully.

IntelliSense Run and Debug Built-in Git Extensions

Una Kravets @Una
VS @code does so many things right. I'm constantly impressed by the UX, and customizing workspace / user preferences is no exception. 🙌🏻 It just keeps

Jonathan Dunlap @jedlbox
VS Code will have deep remote development. You can connect to a container running a different OS and use any VS Code plugins, linting, debugging for that

Pavithra Kodmad @PKodmad
VS Code is my most used and favorite editor. I love being able to customize the editor - changing the layout, the icons, fonts and color scheme is so

<https://code.visualstudio.com/>



Visual Studio Code (VSCode)

VSCode offers syntax highlighting, auto-completion, integrated Git support, and too many features to list here.

Be sure to check out their extensions which provide almost everything a developer could hope for.

<https://marketplace.visualstudio.com/VSCode>

<https://code.visualstudio.com/>



Thank You



[https://frank.seesink.com/presentations/
Internet2TechEx-Fall2023/](https://frank.seesink.com/presentations/Internet2TechEx-Fall2023/)

Frank Seesink
frank@seesink.com
frank@unc.edu

