

# Shipping Python to Mac Clients in 2020

Tim Sutton

Site Reliability Engineer

Mobile Developer Experience / Mac Compute



# Why are we here?

**Whether you're writing & distributing *your own code* or  
project**

**or**

**managing, packaging & deploying the code that was  
*written by others***

# What's necessary to get the code to run

- Runtime (Python itself + its core libraries)
  - Complications: which crypto libraries it uses
- Additional pip / pypi packages
- The actual code you want to run

# We used to have it so good

- `/usr/bin/python` (and `/S/L/Frameworks/Python`) were part of the system
- Apple included PyObjC and a few friends (xattr, six, etc.)
- Crypto libraries weren't getting updates, but SSLv3 and TLS < 1.2 still worked.. for a while
- Python 2 wasn't EOL

# Mac OS X Lion

The power of Mac OS X. The magic of iPad.










OS X Lion (10.7)

1 

\$27.99

[Remove](#)

 1-3 business days via email  
Delivery options for: [J2L 2J2](#) 

 **Pickup:**  
Apple Store Pickup is currently  
unavailable

Subtotal

\$27.99

Shipping

FREE

 Estimated tax for: [J2L 2J2](#) 

\$4.19

**Your Total**

**\$32.18**

Check Out

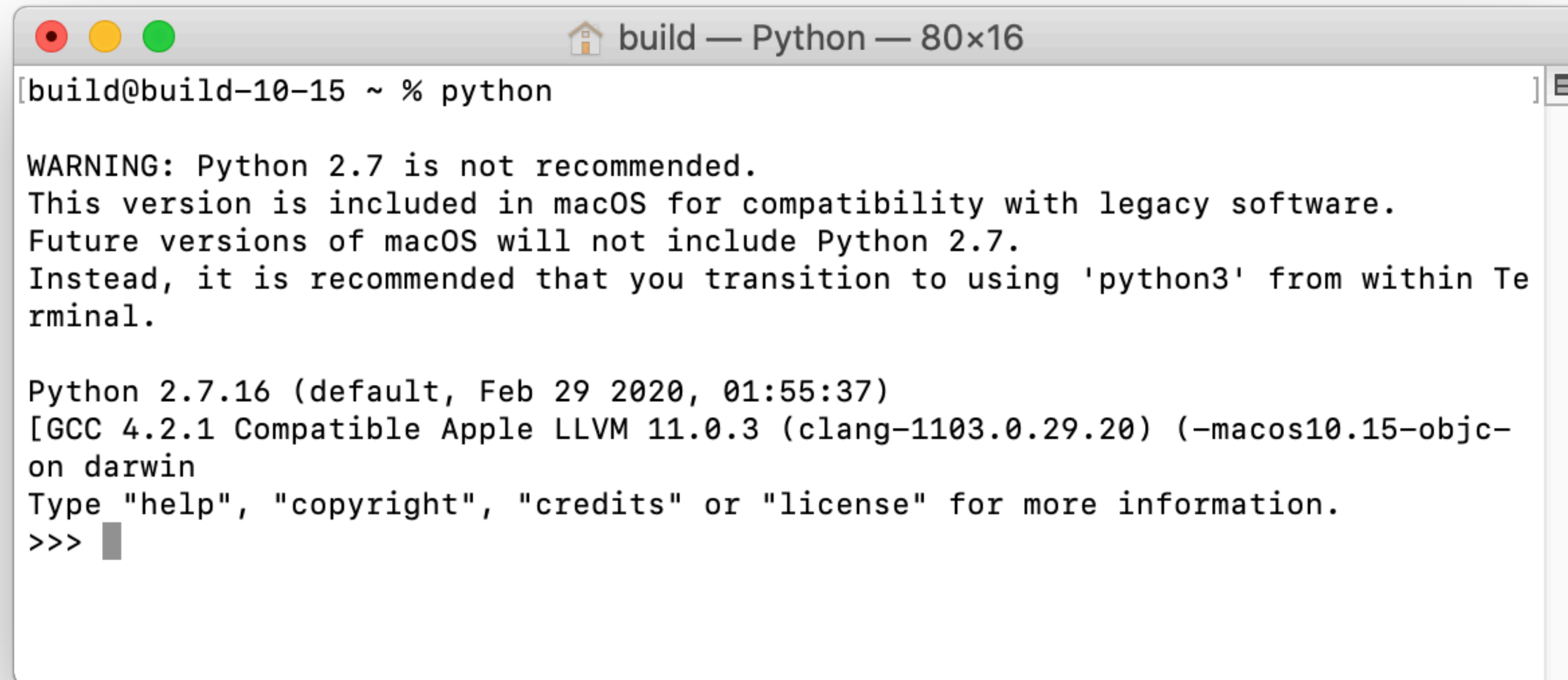
# Scripting Language Runtimes

## Deprecations

- Scripting language runtimes such as Python, Ruby, and Perl are included in macOS for compatibility with legacy software. Future versions of macOS won't include scripting language runtimes by default, and might require you to install additional packages. If your software depends on scripting languages, it's recommended that you bundle the runtime within the app. (49764202)
- Use of Python 2.7 isn't recommended as this version is included in macOS for compatibility with legacy software. Future versions of macOS won't include Python 2.7. Instead, it's recommended that you run `python3` from within Terminal. (51097165)

***[https://developer.apple.com/documentation/macos\\_release\\_notes/macos\\_catalina\\_10\\_15\\_release\\_notes](https://developer.apple.com/documentation/macos_release_notes/macos_catalina_10_15_release_notes)***

# Catalina



A screenshot of a macOS Terminal window. The title bar shows a home icon, the text 'build — Python — 80x16', and standard macOS window controls (red, yellow, green buttons). The terminal content shows the command '[build@build-10-15 ~ % python]' followed by a warning message about Python 2.7, version information for Python 2.7.16, and the prompt '>>>' with a cursor.

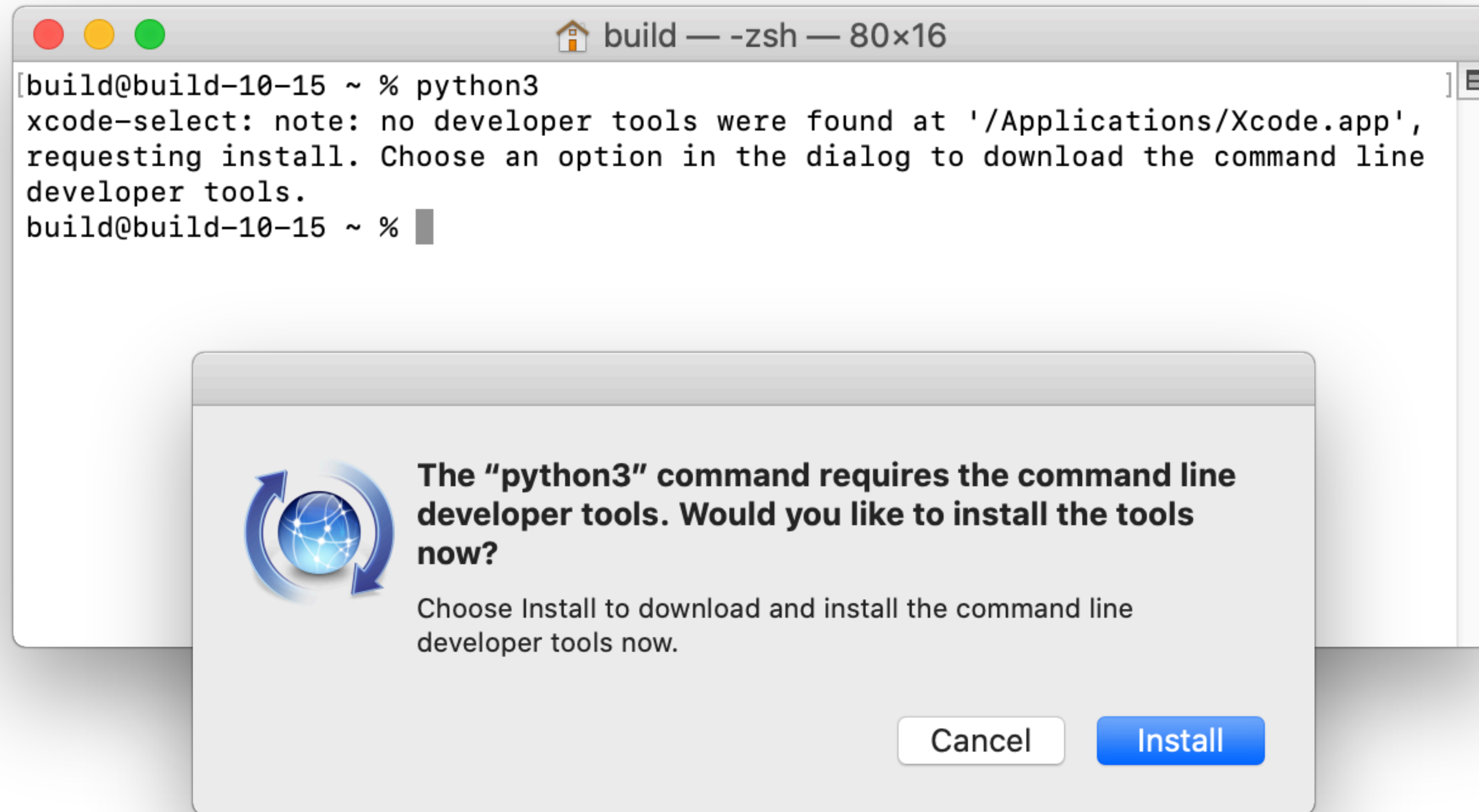
```
[build@build-10-15 ~ % python]

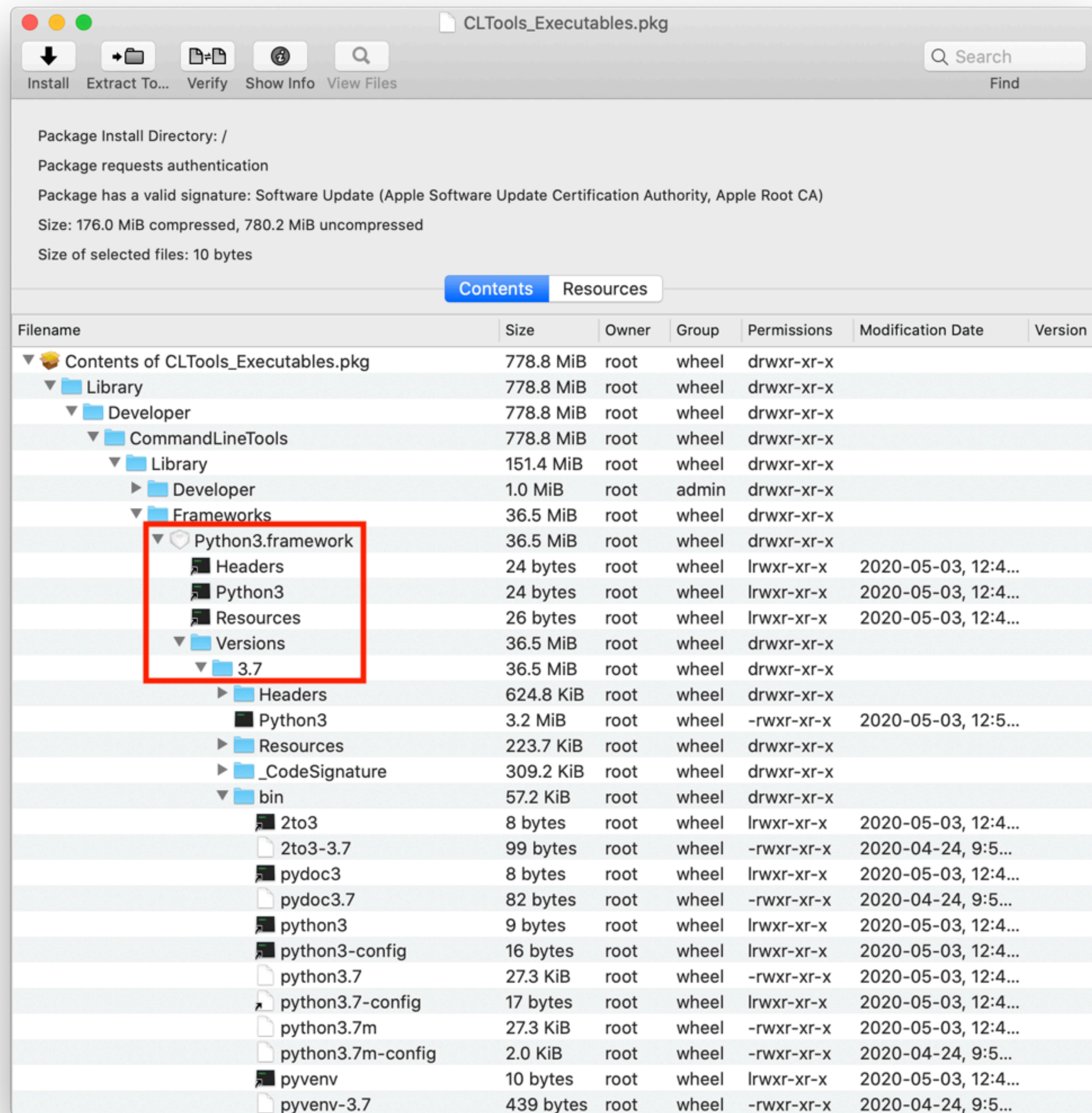
WARNING: Python 2.7 is not recommended.
This version is included in macOS for compatibility with legacy software.
Future versions of macOS will not include Python 2.7.
Instead, it is recommended that you transition to using 'python3' from within Te
rминаl.

Python 2.7.16 (default, Feb 29 2020, 01:55:37)
[GCC 4.2.1 Compatible Apple LLVM 11.0.3 (clang-1103.0.29.20) (-macos10.15-objc-
on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>> █
```



# Catalina







CLTools\_Executables.pkg

Install

Extract To...

Verify

Show Info

View Files

Search

Find

Package Install Directory: /

Package requests authentication

Package has a valid signature: Software Update (Apple Software Update Certification Authority, Apple Root CA)

Size: 176.0 MiB compressed, 780.2 MiB uncompressed

Size of selected files: 88.7 KiB

Contents

Resources


Filename	Size	Owner	Group	Permissions	Modification Date	Version
Contents of CLTools_Executables.pkg	778.8 MiB	root	wheel	drwxr-xr-x		
Library	778.8 MiB	root	wheel	drwxr-xr-x		
Developer	778.8 MiB	root	wheel	drwxr-xr-x		
CommandLineTools	778.8 MiB	root	wheel	drwxr-xr-x		
Library	151.4 MiB	root	wheel	drwxr-xr-x		
Developer	1.0 MiB	root	admin	drwxr-xr-x		
Frameworks	36.5 MiB	root	wheel	drwxr-xr-x		
Python3.framework	36.5 MiB	root	wheel	drwxr-xr-x		
Headers	24 bytes	root	wheel	lrwxr-xr-x	2020-05-03, 12:4...	
Python3	24 bytes	root	wheel	lrwxr-xr-x	2020-05-03, 12:4...	
Resources	26 bytes	root	wheel	lrwxr-xr-x	2020-05-03, 12:4...	
Versions	36.5 MiB	root	wheel	drwxr-xr-x		
3.7	36.5 MiB	root	wheel	drwxr-xr-x		
Headers	624.8 KiB	root	wheel	drwxr-xr-x		
Python3	3.2 MiB	root	wheel	-rwxr-xr-x	2020-05-03, 12:5...	
Resources	223.7 KiB	root	wheel	drwxr-xr-x		
Info.plist	507 bytes	root	wheel	-rw-r--r--	2020-04-24, 9:5...	
Python.app	133.9 KiB	root	wheel	drwxr-xr-x		
patches.tar.bz2	88.7 KiB	root	wheel	-rw-r--r--	2020-04-24, 9:5...	
version.plist	510 bytes	root	wheel	-rw-r--r--	2020-04-24, 9:5...	
_CodeSignature	309.2 KiB	root	wheel	drwxr-xr-x		
bin	57.2 KiB	root	wheel	drwxr-xr-x		
include	10 bytes	root	wheel	drwxr-xr-x		
lib	32.1 MiB	root	wheel	drwxr-xr-x		
share	15.8 KiB	root	wheel	drwxr-xr-x		
Current	3 bytes	root	wheel	lrwxr-xr-x	2020-05-03, 12:4...	
PrivateFrameworks	113.9 MiB	root	wheel	drwxr-xr-x		
usr	627.4 MiB	root	wheel	drwxr-xr-x		
bin	351.4 MiB	root	wheel	drwxr-xr-x		
2to3	64 bytes	root	wheel	lrwxr-xr-x	2020-05-03, 12:4...	

# How else can we install Python?


- Management/deployment:
  - [python.org](https://python.org) installer packages
  - Relocatable Python (<https://github.com/gregneagle/relocatable-python>)
    - (Relies on [python.org](https://python.org) installers)
- Development:
  - <https://github.com/pyenv/pyenv>
  - <https://github.com/indygreg/python-build-standalone>
  - homebrew

# python.org

PythonPSFDocsPyPIJobsCommunity

python™

Donate



GO

Socialize

AboutDownloadsDocumentationCommunitySuccess StoriesNewsEvents

Python 3.8.3

Release Date: May 13, 2020

This is the third maintenance release of the Python 3.8 series.

The Python 3.8 series is the newest stable release of Python.

All releases

Source code

Windows

Mac OS X

Other Platforms

License

Alternative Implementations

Download for Mac OS X

Python 3.8.3

Not the OS you are looking for? Python can be used on many operating systems and environments.

[View the full list of downloads.](#)

Major new features of the 3.8 series, compared to 3.7

- [PEP 572](#), Assignment expressions
- [PEP 570](#), Positional-only arguments
- [PEP 587](#), Python Initialization Configuration (improved embedding)
- [PEP 590](#), Vectorcall: a fast calling protocol for CPython
- [PEP 578](#), Runtime audit hooks



python-3.8.3-macosx10.9.pkg

Package Info

All Files

postinstall

Receipts

Name	Date Modified	Size	Kind
Applications	--	477 KB	Folder
Python 3.8	--	477 KB	Folder
icon	2020-05-13	Zero KB	Document
IDLE.app	--	188 KB	Application
Install Certificates.command	2020-05-13	1 KB	Terminal shell script
License.rtf	2020-05-13	13 KB	Rich text (RTF)
Python Launcher.app	--	269 KB	Application
ReadMe.rtf	2020-05-13	3 KB	Rich text (RTF)
Update Shell Profile.command	2020-05-13	3 KB	Terminal shell script
Library	--	113.9 MB	Folder
Frameworks	--	113.9 MB	Folder
Python.framework	--	113.9 MB	Framework
usr	--	704 bytes	Folder
local	--	704 bytes	Folder
bin	--	704 bytes	Folder
2to3	2020-05-13	66 bytes	Symbolic link
2to3-3.8	2020-05-13	70 bytes	Symbolic link
idle3	2020-05-13	67 bytes	Symbolic link
idle3.8	2020-05-13	69 bytes	Symbolic link
pydoc3	2020-05-13	68 bytes	Symbolic link
pydoc3.8	2020-05-13	70 bytes	Symbolic link
python3	2020-05-13	69 bytes	Symbolic link
python3-config	2020-05-13	76 bytes	Symbolic link
python3.8	2020-05-13	71 bytes	Symbolic link
python3.8-config	2020-05-13	78 bytes	Symbolic link

Python.framework

Framework

113.9 MB

Modified

--

Owner

--

Group

--

Permissions

an unk...

Read & Write

an unk...

Read only

Everyo...

Read only

Version

--

Identifier

--

Package

org.python.Python.PythonFramework-3.8  
org.python.Python.PythonDocumentation-3.8

Bundle

Will overwrite files

All Files > Library > Frameworks > Python.framework

3 items, 114.4 MB installed

# relocatable-python

gregneagle / relocatable-python

Watch

8

Star

65

Fork

9

<> Code

Issues 1

Pull requests 1

Actions

Projects 0

Wiki

Security 0

Insights

A tool for building standalone relocatable Python.framework bundles

18 commits

1 branch

0 packages

0 releases

2 contributors

Apache-2.0

Branch: master








New pull request

Create new file

Upload files

Find file

Clone or download

	gregneagle Python 3 fix for localizing shebangs in Python.framework/Versions/Cur...	Latest commit 8bce58e on Feb 8
	locallibs Python 3 fix for localizing shebangs in Python.framework/Versions/Cur...	4 months ago
	.gitignore First code commit	2 years ago
	LICENSE Initial commit	2 years ago
	README.txt Update README.txt	6 months ago
	make_relocatable_python_framework... Make scripts in Python.framework/Versions/Current/bin relocatable. Fi...	4 months ago
	research_notes.txt First code commit	2 years ago

README.txt

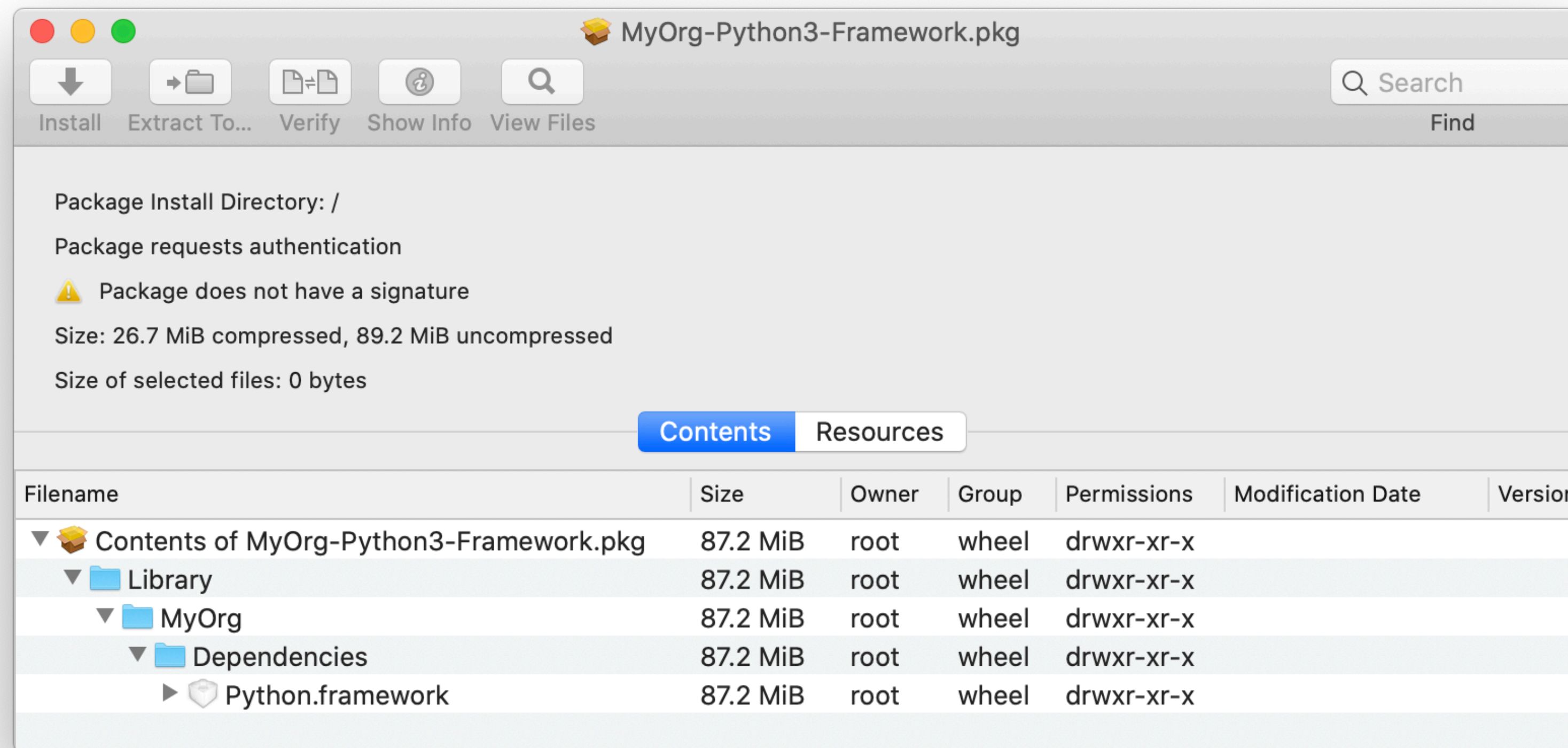
This is a tool to make a relocatable Python framework containing PyObjC.

NOTE: while the resulting frameworks (and interpreters) have been successfully used in several projects (among them Imagr, Munki, and AutoPkg) there is no guarantee it is suitable as a general-purpose Python installation.

A relocatable Python.framework is ideal for embedding into an application's Frameworks directory, and can even be used to get PyObjC-based apps and tools running in the macOS Recovery environment, which does not include Python.



# relocatable-python



```
#!/Library/MyOrg/Dependencies/Python.framework/Versions/Current/bin/python3
```

# Building a single binary

- PyInstaller
  - Easy to use, bundles runtime shared libs and minimum required modules
  - No recompilation, it's more of a packaging solution
- Nuitka
  - Compiler, optimized for performance
- PyOxidizer
  - New and experimental, runs a Python embeddable runtime via a Rust binary

# PyInstaller

```
$ rg pyinstaller osx_server_irc_logs
```

```
##osx-server_20140430.log
```

```
1328:[14:59:49] <frogor> Yeah. My current one is pyinstaller
```

```
##osx-server_20140314.log
```

```
1703:[19:40:13] <frogor> natewalck: pyinstaller is awesome (if you use the development branch) btw.
```

```
##osx-server_20140729.log
```

```
2486:[19:59:02] <frogor> I use pyinstaller to compile them to self-contained .exe
```

```
##osx-server_20140919.log
```

```
2448:[17:33:01] <frogor> Though pyinstaller might be able to wrap that all up into an .app .....
```

```
2449:[17:33:07] <frogor> (I love pyinstaller to death)
```

```
##osx-server_20131030.log
```

```
1259:[14:33:56] <frogor> I'm only doing PowerShell to be polite to my coworkers. pyinstaller makes wonderful Windows .exe files.
```

```
1267:[14:36:10] <frogor> jaharmi: Mostly because without something like pyinstaller, you can't natively run python on Windows without installing the interpreter.
```

```
1269:[14:36:24] <frogor> But pyinstaller will let you do self-contained re-distributable .exe files
```



# Demo



***PyInstaller***

*#!/bin/env python*

*import os*

*import sys*

*11110000111000100011100011100*

*111111111110100100011110111010*

*11111111111000111110000111100011*

# Caveats with PyInstaller

- Needs Python's shared library support
  - pyenv: install using the `--enable-shared` configure option
  - May be possible to use with `.framework` builds of Python
- Not really optimized for performance or load speed
  - Single binary is cooler than the default "directory" distribution build, but adds to initial execution time in order to unpack the compressed archive

**Learn about your Python**

# sysconfig module: learn about the runtime configuration

(From Apple's Python 3)

```
>>> import sysconfig
```

```
>>> sysconfig.get_config_vars()
```

```
{'OPENSSL_LIBS': '-lssl -lcrypto -framework TrustEvaluationAgent',
```

```
  'OPENSSL_INCLUDES': '-I/BuildRoot/Applications/Xcode.app/Contents/Developer/  
Platforms/MacOSX.platform/Developer/SDKs/MacOSX10.15.Internal.sdk/usr/local/  
libressl-2.8/include',
```

```
  'OPENSSL_LDFLAGS': '-L/BuildRoot/Applications/Xcode.app/Contents/Developer/  
Platforms/MacOSX.platform/Developer/SDKs/MacOSX10.15.Internal.sdk/usr/local/  
libressl-2.8/lib/static -F/BuildRoot/Applications/Xcode.app/Contents/Developer/  
Platforms/MacOSX.platform/Developer/SDKs/MacOSX10.15.Internal.sdk/System/Library/  
PrivateFrameworks']}
```

# SSL config

```
# From python.org and relocatable-python Python 3 packages
>>> import ssl
>>> ssl.OPENSSL_VERSION
'OpenSSL 1.1.1d 10 Sep 2019'
```

```
$ otool -L ./Versions/3.8/lib/python3.8/lib-dynload/_ssl.cpython-38-darwin.so
```

```
./Versions/3.8/lib/python3.8/lib-dynload/_ssl.cpython-38-darwin.so:
    /Library/Frameworks/Python.framework/Versions/3.8/lib/libssl.1.1.dylib (compatibility version 1.1.0,
current version 1.1.0)
    /Library/Frameworks/Python.framework/Versions/3.8/lib/libcrypto.1.1.dylib (compatibility version
1.1.0, current version 1.1.0)
    /usr/lib/libSystem.B.dylib (compatibility version 1.0.0, current version 1197.1.1)
```



# SSL config

```
# From Apple's Python
>>> import ssl
>>> ssl.OPENSSL_VERSION
'LibreSSL 2.8.3'
```

```
$ otool -L /Library/Developer/CommandLineTools/Library/Frameworks/Python3.framework/Versions/Current/lib/
python3.7/lib-dynload/_ssl.cpython-37m-darwin.so
```

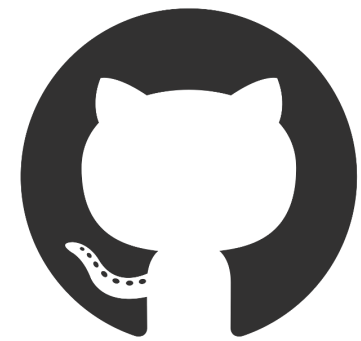
```
/Library/Developer/CommandLineTools/Library/Frameworks/Python3.framework/Versions/Current/lib/python3.7/
lib-dynload/_ssl.cpython-37m-darwin.so:
```

```
  /System/Library/PrivateFrameworks/TrustEvaluationAgent.framework/Versions/A/TrustEvaluationAgent
  (compatibility version 1.0.0, current version 33.0.0)
  /usr/lib/libSystem.B.dylib (compatibility version 1.0.0, current version 1281.0.0)
```

# SSL config - certs

- Anything involving internally-trusted roots, or cert auth, you'll need to know which trust store is being used by your Python
- Check out certifi package, and `certifi.where()` - Python requests package will use it out of the box
  - May need to occasionally update bundled certifi package in your app
  - <https://requests.readthedocs.io/en/master/user/advanced/#ssl-cert-verification>
- See homebrew openssl formula code for an example of how to port certificates from a keychain into a .pem file
  - <https://github.com/Homebrew/homebrew-core/blob/master/Formula/openssl@1.1.rb>

# Thank you!



@timsutton



@tvsutton

<https://macops.ca/macdevopsyvr-2020>