

PyCon Korea 2019

How did I get here?

Pradyun Gedam

CONNECT
THE
PYTHONISTAS



Who?

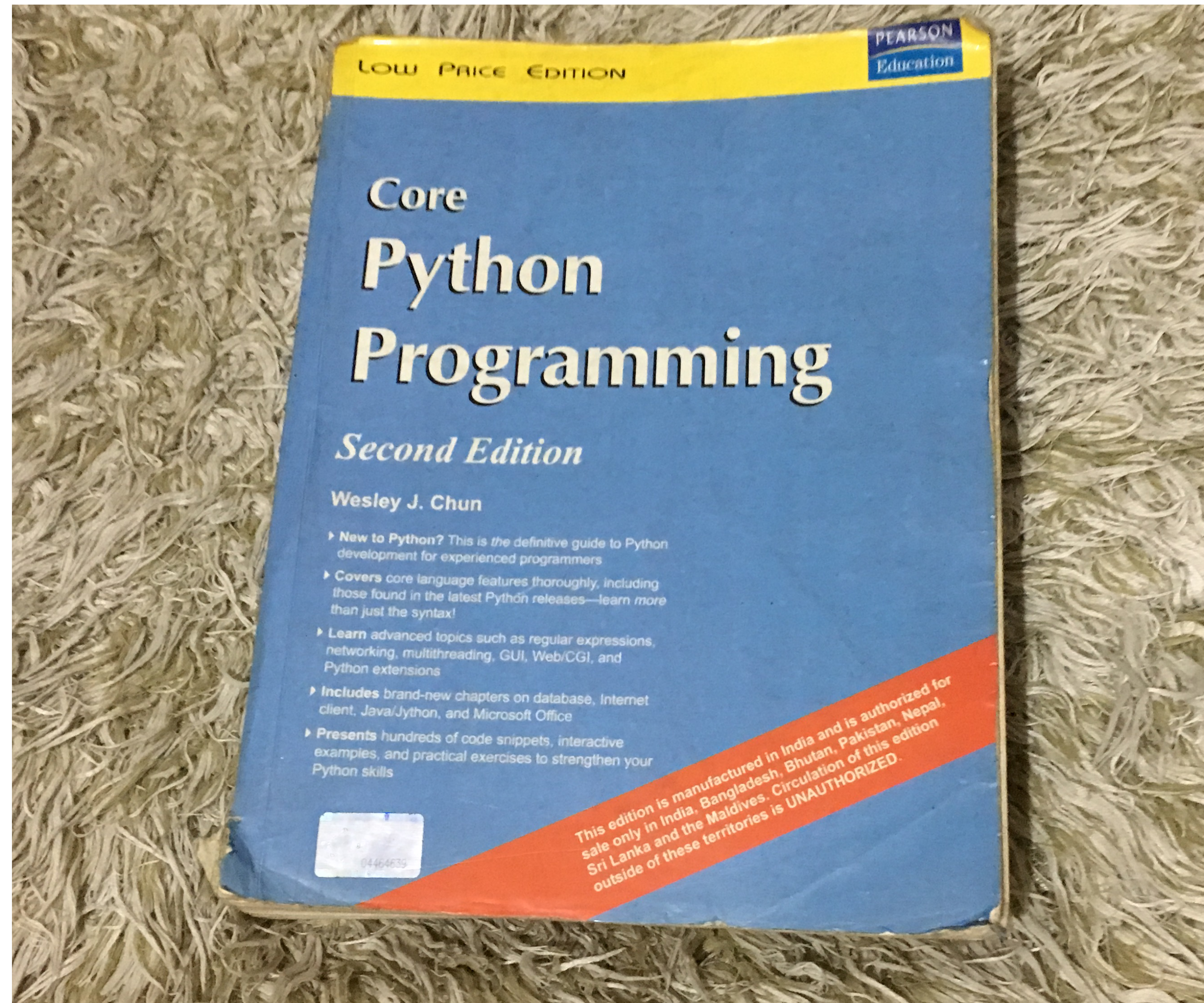
- College Student
- Python Packaging Authority Member
- TOML Core Team
- PSF Fellow



My Journey

Learning






```
>>> print "Hello World!"
```



```
>>> print "Hello World!"  
File "<stdin>", line 1  
    print "Hello World!"  
          ^
```

SyntaxError: invalid syntax



Quick Links (3.4.0)

- » [Documentation](#)
- » [Windows Installer](#)
- » [Mac OS X Installer](#)
- » [Source Distribution](#)

Quick Links (2.7.6)

- » [Documentation](#)
- » [Windows Installer](#)
- » [Mac OS X Installer](#)
- » [Source Distribution](#)




```
>>> print "Hello World!"  
Hello World!
```



```
>>> print "Hello World!"  
Hello World!
```

```
>>> print("Hello World!")  
Hello World!
```



```
>>> print "Hello World!"
```

```
SyntaxError: Missing parentheses in call to  
'print'. Did you mean print("Hello World")?
```



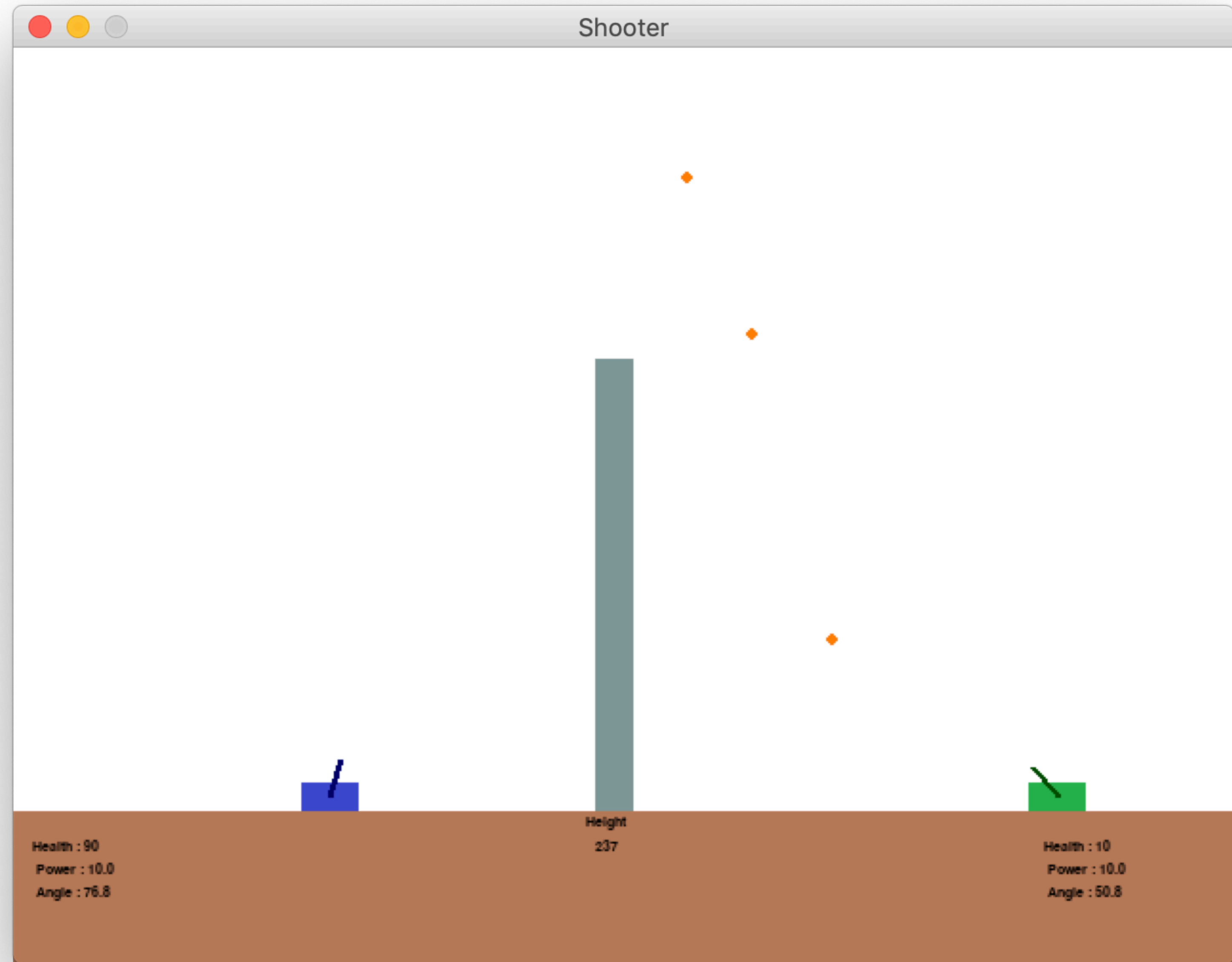
Exploring





Source: http://www.pygame.org/docs/pygame_logo.gif







Source: https://raw.githubusercontent.com/viblo/pymunk/master/docs/src/_static/pymunk_logo_animation.gif



Source: https://raw.githubusercontent.com/viblo/pymunk/master/docs/src/_static/pymunk_logo_animation.gif



Python on Android

I couldn't find a good image. :(



Source: <https://commons.wikimedia.org/wiki/File:Enthought-Logo-Horizontal-2018-Blue-2500x354.png>



 pradyun / **Py2C** 

 Code

 Issues **0**

 Pull requests **0**

More ▼

 Settings

Py2C — a Python to C++ converter

Source: <https://github.com/pradyun/Py2C>



	enthought/traitsui Python 3 compatibility and Travis integration ✓ #165 by pradyunsg was merged on Aug 19, 2015  updated on Jul 4, 2017
	enthought/pyface Python 3 compatibility and Travis integration #105 by pradyunsg was merged on Aug 19, 2015  updated on Jul 4, 2017  4.6.0 version
	enthought/enable Python 3 compatibility and Travis integration for it. ✓ #137 by pradyunsg was merged on May 17, 2016  updated on May 17, 2016
	enthought/apptools Python 3 compatibility and Travis integration #18 by pradyunsg was closed on Feb 25, 2016  updated on Feb 25, 2016
	enthought/envisage Python 3 compatibility and Travis integration for it. ✓ #48 by pradyunsg was closed on Jan 27, 2016  updated on Jan 27, 2016

Source: <https://github.com/pulls?q=is%3Apr+author%3Apradyunsg+archived%3Afalse+sort%3Aupdated-desc+user%3Aenthought+is%3Aclosed>



Contributing



Add "upgrade" and "upgrade-all" commands #59

 Closed

vbabiy opened this issue on Mar 15, 2011 · 251 comments

Source: <https://github.com/pypa/pip/issues/59>





pradyunsg commented on Oct 5, 2013

Member



Any Updates? Is there any plan to add this, it's 3 years old now..

Source: <https://github.com/pypa/pip/issues/59>



Abstract

This document discusses the issues with pip's upgrade behaviour and proposes a resolution for the various issues.

Motivation

There has been a lot of discussion and half-done work related to modifying pip to add an upgrade command to pip. Some of these discussions are spread across various Github Issues and Pull Request and even buried in Mailing List Archives. This document is the result of an attempt to assimilate all the ideas, proposals and problems discussed in all these places into one place.

Source: <https://gist.github.com/pradyunsg/4c9db6a212239fee69b429c96cdc3d73>





pradyunsg commented on Jun 7, 2016

Member



@pradyunsg As you can see, there's still a lot of unresolved issues here. Are you still interested in taking this forward? I'm reluctant to fire off another long debate if it's just going to stall again...

I expected there to be. I am interested in taking this forward. Let's do this! 😊

Source: <https://github.com/pypa/pip/issues/59>



Add an upgrade-strategy option #3972

 Merged as 807f879 dstufft merged 7 commits into `pypa:master` from `pradyunsg:non-eager-upgrades`  on Sep 19, 2016

Source: <https://github.com/pypa/pip/pull/3972>



[GSoC-general] GSoC 2017 under PSF

Pradyun Gedam [pradyunsg at gmail.com](mailto:pradyunsg@gmail.com)

Tue Feb 7 23:15:04 EST 2017

- Previous message (by thread): [\[GSoC-general\] Pytest as a Suborg](#)
- Next message (by thread): [\[GSoC-general\] GSoC 2017 under PSF](#)
- **Messages sorted by:** [\[date \]](#) [\[thread \]](#) [\[subject \]](#) [\[author \]](#)

Hello Everyone!

My name is Pradyun Gedam. I'm currently a first year student VIT University in India.

I would like to apply for GSoC 2017 under PSF.

I currently have a project in mind - the "pip needs a dependency resolver" issue [1]. I would like to take on this specific project but am willing to do some other project as well.

For some background, around mid 2016, I started contributing to pip. The first issue I tackled was #59 [2] - a request for upgrade command and an upgrade-all command that has been open for over 5.5 years. Over the months following that, I've have had the opportunity to work with and understand multiple parts of pip's codebase while working on this issue and a few others. This search on GitHub issues [3] also provides a good summary of what work I've done on pip.

[2]: <https://github.com/pypa/pip/issues/988>

[2]: <https://github.com/pypa/pip/issues/59>

[3]: <https://github.com/pypa/pip/issues?q=author%3Apradyunsg>

Eagerly-waiting-for-a-response-ly,
Pradyun Gedam

Source: <https://mail.python.org/pipermail/gsoc-general/2017/000111.html>



Pradyun Gedam

Adding Proper Dependency Resolution to pip

Organization: Python Software Foundation

pip does not resolve dependencies correctly when there are conflicting requirements. The lack of dependency resolution has caused hard-to-diagnose...

Source: <https://summerofcode.withgoogle.com/archive/2017/projects/>

Wrapping up GSoC 2017

Aug 24, 2017 • Pradyun Gedam • [#final-submission](#)

This is my final post as a part of GSoC 2017 and will serve as my “Work Product Submission”.

Source: <https://pradyunsg.me/gsoc-2017/08/24/wrapping-up-gsoc/>



pip needs a dependency resolver #988

🔔 Open

cboylan opened this issue on Jun 11, 2013 · 121 comments



(2019-06-23)

This is being worked on by **@pradyunsg**, in continuation of his GSoC 2017 project. A substantial amount of code cleanup has been done, and is ongoing, to make it tractable replace the current resolver, in a reasonable manner. This work enabled pip ≥ 10 to warn when it is going to make an installation that breaks the dependency graph. (The installations are not aborted in such scenarios, for backwards compatibility.)

A more substantial update, detailing the current state of affairs and the plan moving forward, is available [here](#).

Source: <https://github.com/pypa/pip/issues/988>





Community



“I came for the language but I stay for the community”

Brett Cannon



Thank You!

CONNECT
THE ★
PYTHONISTAS

