Packaging and Distributing Your Code

Reindert-Jan Ekker
nl.linkedin.com/in/rjekker/
@rjekker
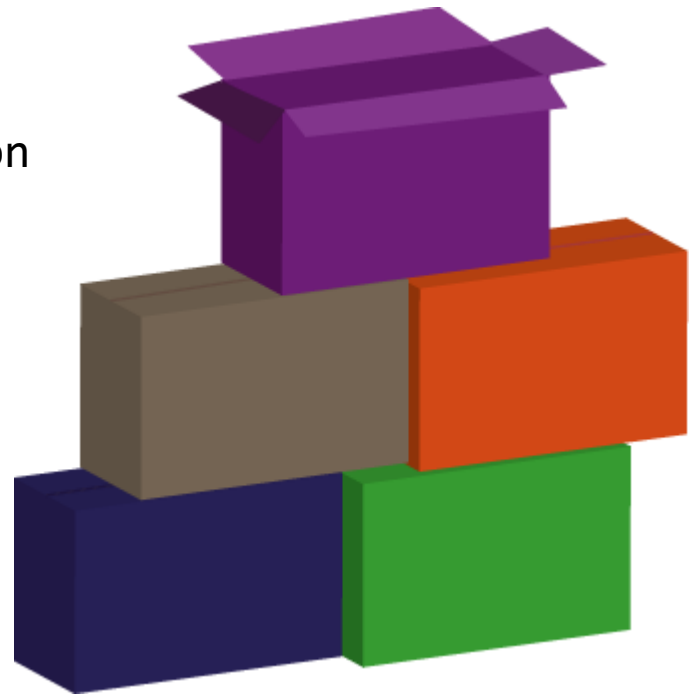
pluralsight
hardcore dev and IT training

Overview

- **Packaging with setuptools**

- **Uploading to PiPy**

- **Distributing as a Standalone Executable**

Preparing Your Project for Packaging

- **`setup.py`**
  - Is a Python script
  - Call setup with at least: `name, version`
  - use `setuptools` (not distutils)
  - Dependencies: `install_requires`

- **`MANIFEST.in`**
  - If you need to explicitly exclude or include files

Packaging

- **`python setup.py sdist`**
  - Source distribution

- **`python setup.py bdist_wheel`**
  - Need to install package wheel
  - Supports C extensions and platform-specific distribution
  - Faster installation

- **`python setup.py register`**

- **`python setup.py sdist upload`**

Distributing Executables

- **Automatic script creation**
  - ☐ entry_points in setup.py

- **Py2exe**
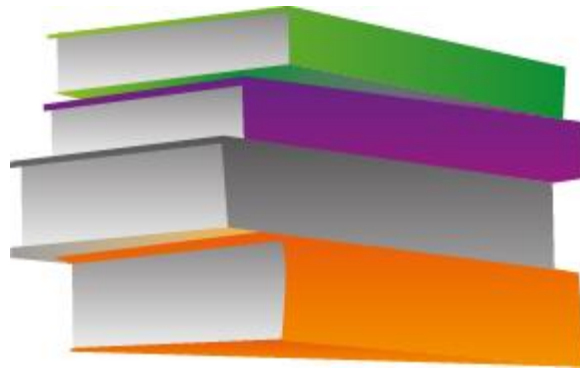 - ☐ Creates a Windows executable (includes Python interpreter)
  - ☐ http://www.py2exe.org/

- **Py2app**
  - ☐ Creates a standalone Mac OS X application (includes Python interpreter)
  - ☐ http://pythonhosted.org/py2app/

- **PyInstaller**
  - ☐ Supports multiple platforms
  - ☐ http://www.pyinstaller.org/

Resources

- **Setuptools Documentation (http://goo.gl/JA6AAO)**
  - https://pythonhosted.org/setuptools/setuptools.html#developer-s-guide

- **A Sample Project for Packaging (http://goo.gl/yFgGkM)**
  - https://github.com/pypa/sampleproject/

- **Python Packaging Authority Documentation (http://goo.gl/Rx4kv1)**
  - http://python-packaging-user-guide.readthedocs.org/en/latest/tutorial.html

Summary

- **Packaging with setuptools**
  - Project setup
  - source-dist
  - wheel

- **Uploading to PiPy**

- **Distributing as a Standalone Executable**

Visuals by Britt van de Pol