

INTRO TO PYTHON: CONCLUSION

Stdlib, PyPI, and where to go from here

DON'T REINVENT THE WHEEL

You don't have to just reuse your own code.

You can also find and reuse code from:

- Python standard library (stdlib)
- Python Package Index (PyPI)

We'll go into these next.

PYTHON STANDARD LIBRARY

Also called the **stdlib**, it:

- Comes pre-installed with Python.
- Has modules for math, path handling, etc.

STDLIB HAS MODULES LIKE RANDOM

random module:

- Has pseudo-random number generators.

You can import it as if it were any other .py file:

```
import random
```

EXAMPLE: USING RANDOM

Create a file called **odds.py** and open it in your text editor.

Type in the code on the next slide...

EXAMPLE: USING RANDOM



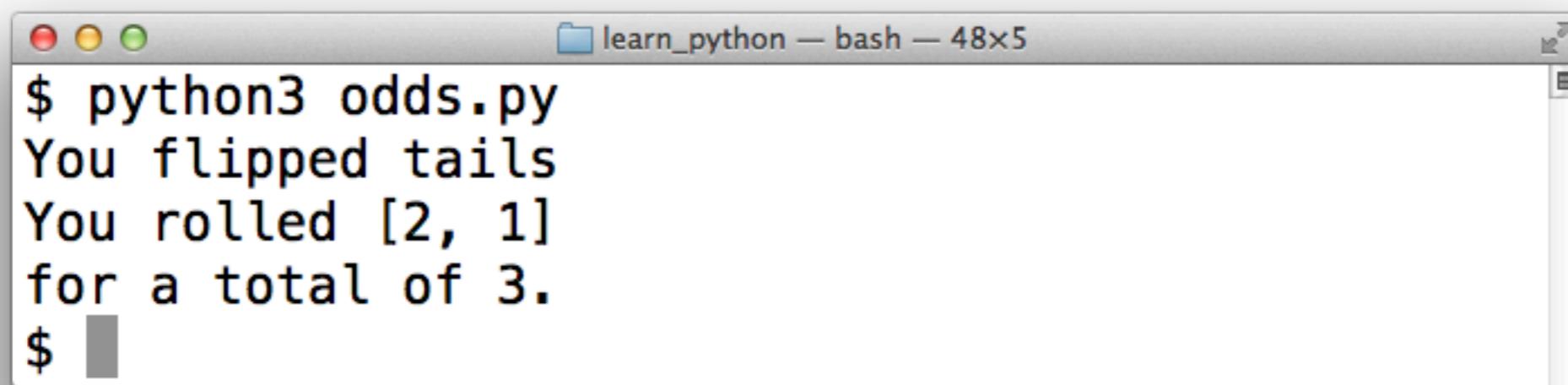
A screenshot of a Python code editor window titled "odds.py — learn_python". The code in the editor is as follows:

```
1 import random
2
3 def flip_coin():
4     return random.choice(["heads", "tails"])
5
6 print("You flipped {}".format(flip_coin()))
7
8 def roll_dice():
9     die_1 = random.randint(1, 6)
10    die_2 = random.randint(1, 6)
11    return [die_1, die_2]
12
13 dice = roll_dice()
14 print("You rolled {}".format(dice))
15 print("for a total of {}".format(sum(dice)))
```

The status bar at the bottom shows "Line: 6:44 | Python" and "Soft Tabs: 4". A tooltip "flip_coin()" is visible near the end of the 15th line.

EXAMPLE: USING RANDOM

Run the program, as usual:



```
$ python3 odds.py
You flipped tails
You rolled [2, 1]
for a total of 3.
$
```

STDLIB IS PRE-INSTALLED

Just import a module and use it! No installation necessary.

The screenshot shows a web browser window with the title "The Python Standard Library". The address bar displays "Python Software Foundation [US] https://docs.python.org/3.4/library/". The page content is a hierarchical list of Python standard library modules. On the left, there's a sidebar with links to "Previous topic" (10. Full Grammar specification), "Next topic" (1. Introduction), "This Page" (Report a Bug, Show Source), and a "Quick search" field with a "Go" button. The main content area lists the following sections and their sub-modules:

- 5.4. exception hierarchy
- 6. Text Processing Services
 - 6.1. string — Common string operations
 - 6.2. re — Regular expression operations
 - 6.3. difflib — Helpers for computing deltas
 - 6.4. textwrap — Text wrapping and filling
 - 6.5. unicodedata — Unicode Database
 - 6.6. stringprep — Internet String Preparation
 - 6.7. readline — GNU readline interface
 - 6.8. rlcompleter — Completion function for GNU readline
- 7. Binary Data Services
 - 7.1. struct — Interpret bytes as packed binary data
 - 7.2. codecs — Codec registry and base classes
- 8. Data Types
 - 8.1. datetime — Basic date and time types
 - 8.2. calendar — General calendar-related functions
 - 8.3. collections — Container datatypes
 - 8.4. collections.abc — Abstract Base Classes for Containers
 - 8.5. heapq — Heap queue algorithm
 - 8.6. bisect — Array bisection algorithm
 - 8.7. array — Efficient arrays of numeric values
 - 8.8. weakref — Weak references
 - 8.9. types — Dynamic type creation and names for built-in types
 - 8.10. copy — Shallow and deep copy operations
 - 8.11. pprint — Data pretty printer
 - 8.12. reprlib — Alternate `repr()` implementation
 - 8.13. enum — Support for enumerations
- 9. Numeric and Mathematical Modules
 - 9.1. numbers — Numeric abstract base classes
 - 9.2. math — Mathematical functions
 - 9.3. cmath — Mathematical functions for complex numbers
 - 9.4. decimal — Decimal fixed point and floating point arithmetic

PYTHON PACKAGE INDEX

Some packages in here are so much better than the stdlib.

The screenshot shows the homepage of the Python Package Index (PyPI). The page has a header with the PyPI logo and navigation links for "Python Software Foundation [US]" and "https://pypi.python.org/pypi". Below the header is a search bar and a "search" button. The main content area features a large Python logo and the text "» Package Index". A sidebar on the left contains links for "PACKAGE INDEX", "ABOUT", "NEWS", "DOCUMENTATION", "DOWNLOAD", "COMMUNITY", "FOUNDATION", and "CORE DEVELOPMENT". The central content area includes sections for "PyPI - the Python Package Index", "Get Packages", "Package Authors", and "Infrastructure". The "PyPI - the Python Package Index" section states there are 48963 packages. The "Get Packages" section provides instructions for using pip. The "Package Authors" section explains how to submit packages. The "Infrastructure" section details how to interoperate with the index. At the bottom, a table lists recent package updates:

Updated	Package	Description
2014-09-19	induction 0.1	A simple web framework based on asyncio.
2014-09-19	unicore-cms 0.2.1	JSON based CMS for Universal Core
2014-09-19	PyUPC-EAN 2.6.2	A barcode library/module for python.
2014-09-19	plonetheme.INTKmodern 0.1.0	An installable Diazo theme for Plone 4
2014-09-19	mad2 0.1.12	file metadata tagger
2014-09-19	rainbowstream 1.0.5	A smart and nice Twitter client on terminal.

PACKAGE VS. MODULE

A package is a folder containing:

- Useful modules
- Sometimes other packaging cruft
 - e.g. files to help release your package on PyPI

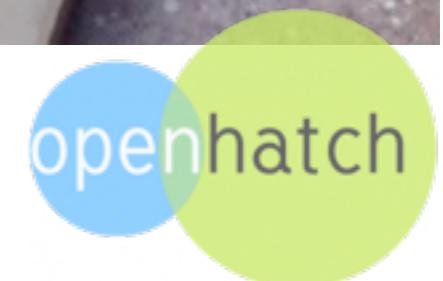
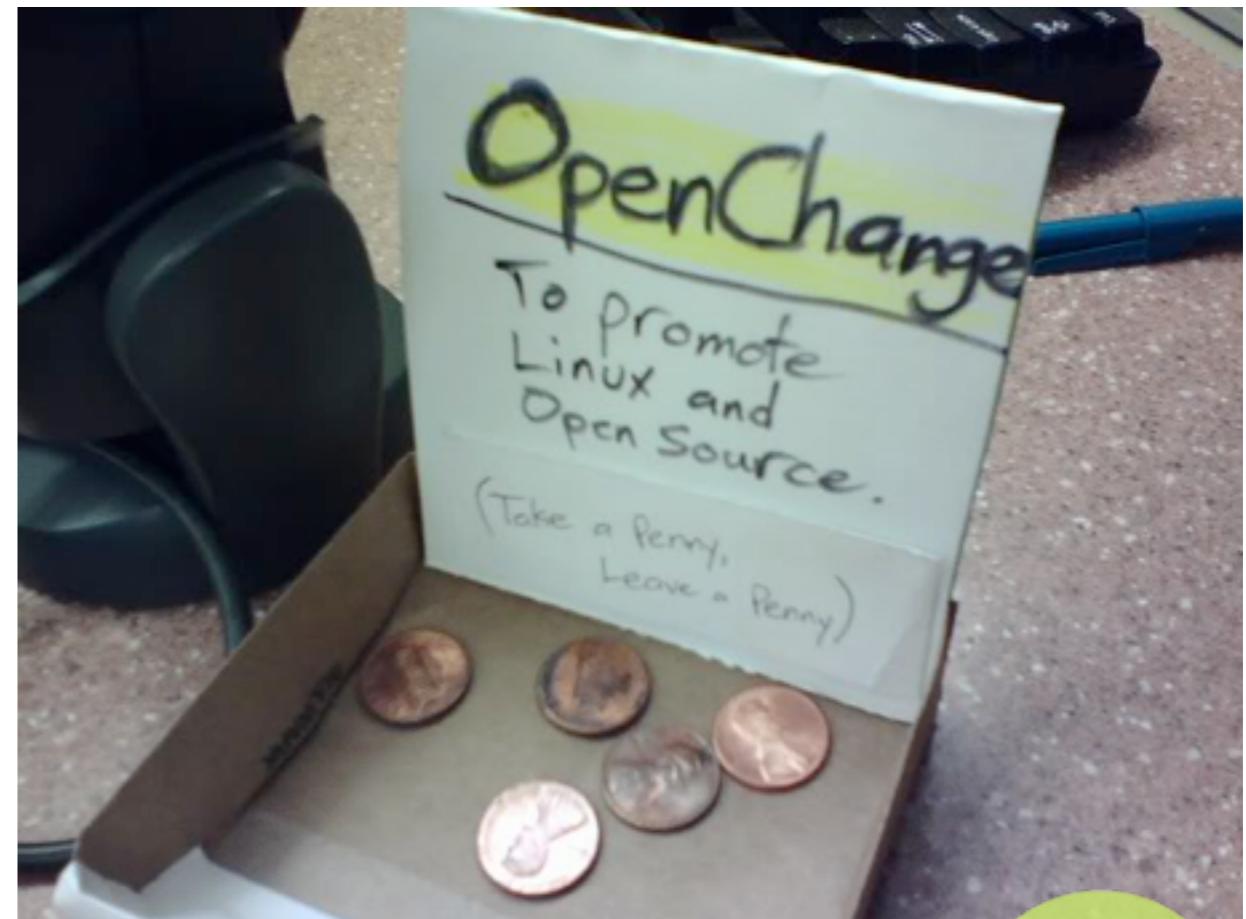
FOSS

There's a lot of FOSS Python code available.

<http://ubuntuforums.org/showthread.php?t=549713>

Free and Open Source Software

- Yay, free stuff!
- Treasure trove for projects
- Use it, give back too



http://en.wikipedia.org/wiki/Free_and_open-source_software

code is fun

PACKAGE EXAMPLE

The screenshot shows a web browser window displaying the Python Package Index (PyPI) page for the `binaryornot` package version 0.3.0. The browser's title bar reads "binaryornot 0.3.0 : Python". The address bar shows the URL <https://pypi.python.org/pypi/binaryornot/0.3.0>. The page itself has a header with the Python logo and navigation links like "Package Index > binaryornot > 0.3.0". On the left, there's a sidebar with links for package browsing, submission, classifiers, and various PyPI documentation and support resources. The main content area features the package name "binaryornot 0.3.0" and a brief description: "Ultra-lightweight pure Python package to check if a file is binary or text." It includes a green "Download" button for the tar.gz file. Below the description, there are badges for PyPI package status (0.3.0), build status (passing), and download count (3.3k/month). A detailed description follows, mentioning it's a heuristic similar to Perl's `pp_is_text`. There are also bullet points about the BSD license and documentation. The right sidebar shows user authentication options ("Not Logged In") and a "Status" section indicating "Nothing to report".

binaryornot 0.3.0

Ultra-lightweight pure Python package to check if a file is binary or text.

[Download
binaryornot-0.3.0.tar.gz](#)

pypi package 0.3.0 build passing downloads 3.3k/month

Ultra-lightweight pure Python package to guess whether a file is binary or text, using a heuristic similar to Perl's `pp_is_text` and its analysis by @eliben.

- Free software: BSD license
- Documentation: <http://binaryornot.readthedocs.org>

Status

It works, and I'm using this package in various places. But it doesn't cover all edge cases yet.

The code could be improved. Pull requests welcome! As of now, it is based on these snippets, but that may change:

- <http://stackoverflow.com/questions/898669/how-can-i-detect-if-a-file-is-binary-non-text-in-python>
- <http://stackoverflow.com/questions/1446549/how-to-identify-binary-and-text-files-using-python>
- <http://code.activestate.com/recipes/173220/>
- <http://eli.thegreenplace.net/2011/10/19/perl-s-guess-if-file-is-text-or-binary-implemented-in-python/>

Features

Has tests for these file types:

- Text: .css, .json, .txt, .svg

EXERCISE: EXPLORE A PACKAGE

Go to <https://pypi.python.org>:

- Search for *binaryornot*
 - or “binary or not”

EXERCISE: EXPLORE A PACKAGE

Study the package page:

The screenshot shows a web browser window with the following details:

- Address Bar:** https://pypi.python.org/pypi/binaryornot/
- Page Title:** binaryornot 0.3.0 : Python
- Header:** Python Software Foundation [US]
- Logo:** Python logo
- Search Bar:** search
- Breadcrumbs:** » Package Index > binaryornot > 0.3.0
- Left Sidebar (PACKAGE INDEX):**
 - Browse packages
 - Package submission
 - List trove classifiers
 - List packages
 - RSS (latest 40 updates)
 - RSS (newest 40 packages)
 - Python 3 Packages
 - PyPI Tutorial
 - PyPI Security
 - PyPI Support
 - PyPI Bug Reports
 - PyPI Discussion
 - PyPI Developer Info
- Right Sidebar (Not Logged In):**
 - Login
 - Register
 - Lost Login?
 - Use OpenID 8 IP

Status: Nothing to report
- Main Content:**

binaryornot 0.3.0

Ultra-lightweight pure Python package to check if a file is binary or text.

Download binaryornot-0.3.0.tar.gz

pypi package 0.3.0 build passing downloads 3.3k/month

Ultra-lightweight pure Python package to guess whether a file is binary or text, using a heuristic similar to Perl's `pp_fttext` and its analysis by @eliben.

 - Free software: BSD license
 - Documentation: <http://binaryornot.readthedocs.org>

Status

It works, and I'm using this package in various places. But it doesn't cover all edge cases yet.

The code could be improved. Pull requests welcome! As of now, it is based on these snippets, but that may change:

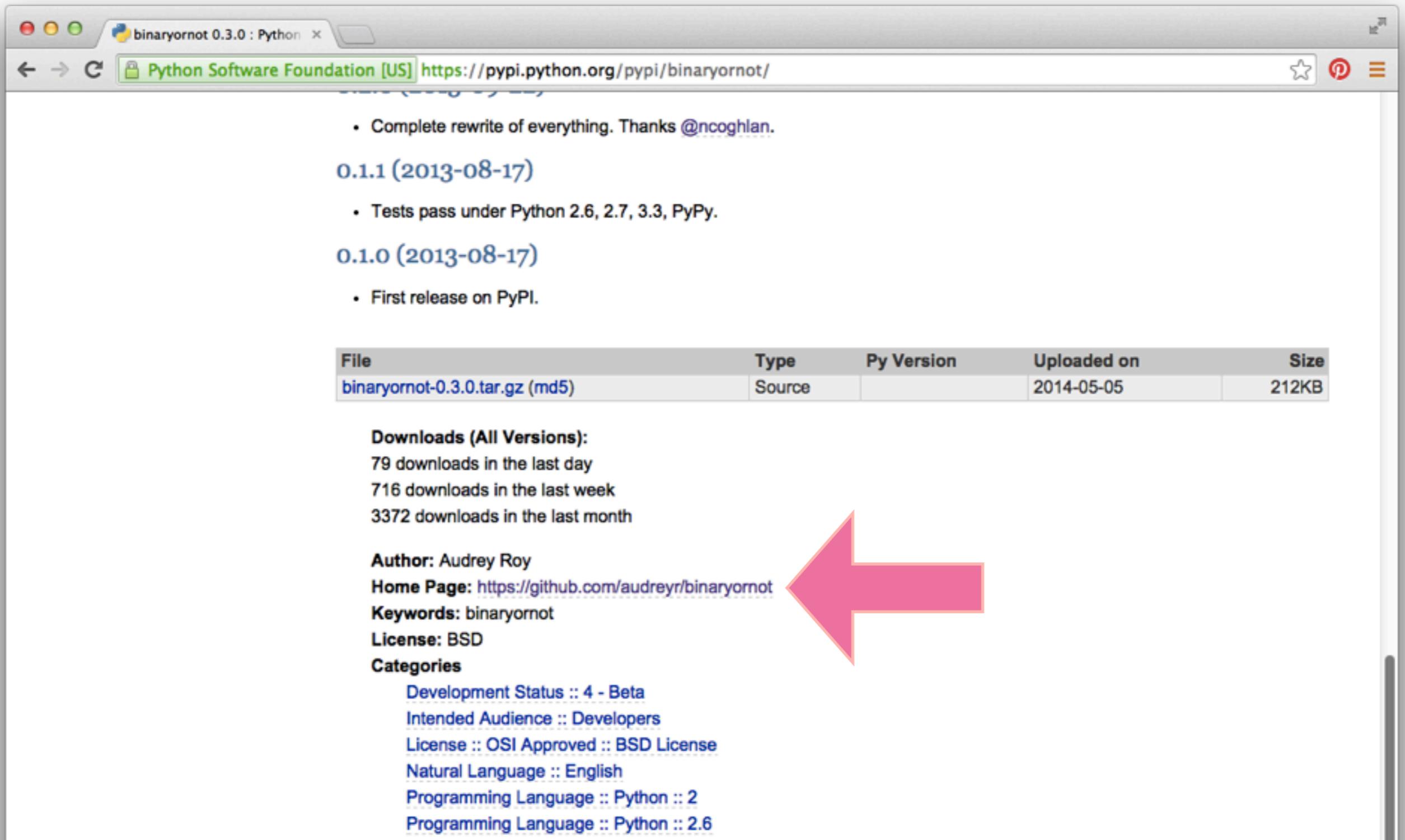
 - <http://stackoverflow.com/questions/898669/how-can-i-detect-if-a-file-is-binary-non-text-in-python>
 - <http://stackoverflow.com/questions/1446549/how-to-identify-binary-and-text-files-using-python>
 - <http://code.activestate.com/recipes/173220/>
 - <http://eli.thegreenplace.net/2011/10/19/perl-s-guess-if-file-is-text-or-binary-implemented-in-python/>

Features

Has tests for these file types:

EXERCISE: EXPLORE A PACKAGE

Scroll down and find the package's Home Page:



A screenshot of a web browser displaying the PyPI page for the `binaryornot` package. The browser window title is "binaryornot 0.3.0 : Python". The address bar shows the URL <https://pypi.python.org/pypi/binaryornot/>. The page content includes release notes for version 0.1.1 (2013-08-17) and 0.1.0 (2013-08-17), a file download table, download statistics, and package metadata.

File

File	Type	Py Version	Uploaded on	Size
binaryornot-0.3.0.tar.gz (md5)	Source		2014-05-05	212KB

Downloads (All Versions):
79 downloads in the last day
716 downloads in the last week
3372 downloads in the last month

Author: Audrey Roy
Home Page: <https://github.com/audreyr/binaryornot> ←
Keywords: binaryornot
License: BSD
Categories
Development Status :: 4 - Beta
Intended Audience :: Developers
License :: OSI Approved :: BSD License
Natural Language :: English
Programming Language :: Python :: 2
Programming Language :: Python :: 2.6

EXERCISE: EXPLORE A PACKAGE

Now browse the code for the package:

The screenshot shows a GitHub repository page for the 'binaryornot' package. The repository is owned by 'audreyr'. The page includes a summary bar with 63 commits, 3 branches, 3 releases, and 2 contributors. Below this, a list of recent commits is shown, including releases 0.3.0 and 0.3.1, and various bug fixes and improvements. On the right side, there are links for Code, Issues, Pull Requests, Wiki, Pulse, Graphs, and Settings. An SSH clone URL is also provided.

Ultra-lightweight pure Python package to check if a file is binary or text. — Edit

63 commits 3 branches 3 releases 2 contributors

branch: master binaryornot / +

Bump status to beta.

audreyr authored on May 5 latest commit e797740aa8

File	Description	Time
binaryornot	Release 0.3.0.	5 months ago
docs	Better PYTHONPATH handling when building Sphinx docs.	a year ago
tests	Read files in binary mode to simplify implementation	a year ago
.gitignore	Ignore coverage output, profiling stats file.	a year ago
.travis.yml	Initial commit.	a year ago
AUTHORS.rst	Add @vincentbernat to AUTHORS.	5 months ago
CONTRIBUTING.rst	Initial commit.	a year ago
HISTORY.rst	Release 0.3.0.	5 months ago

Code Issues Pull Requests Wiki Pulse Graphs Settings

SSH clone URL
git@github.com:audreyr/binaryornot
You can clone with HTTPS, SSH, or Subversion.

EXERCISE: EXPLORE A PACKAGE

The actual Python code is in the ***binaryornot*** directory:

- ***check.py***
- ***helpers.py***

Look at those files. They're just modules containing simple Python functions.

EXERCISE: EXPLORE A PACKAGE

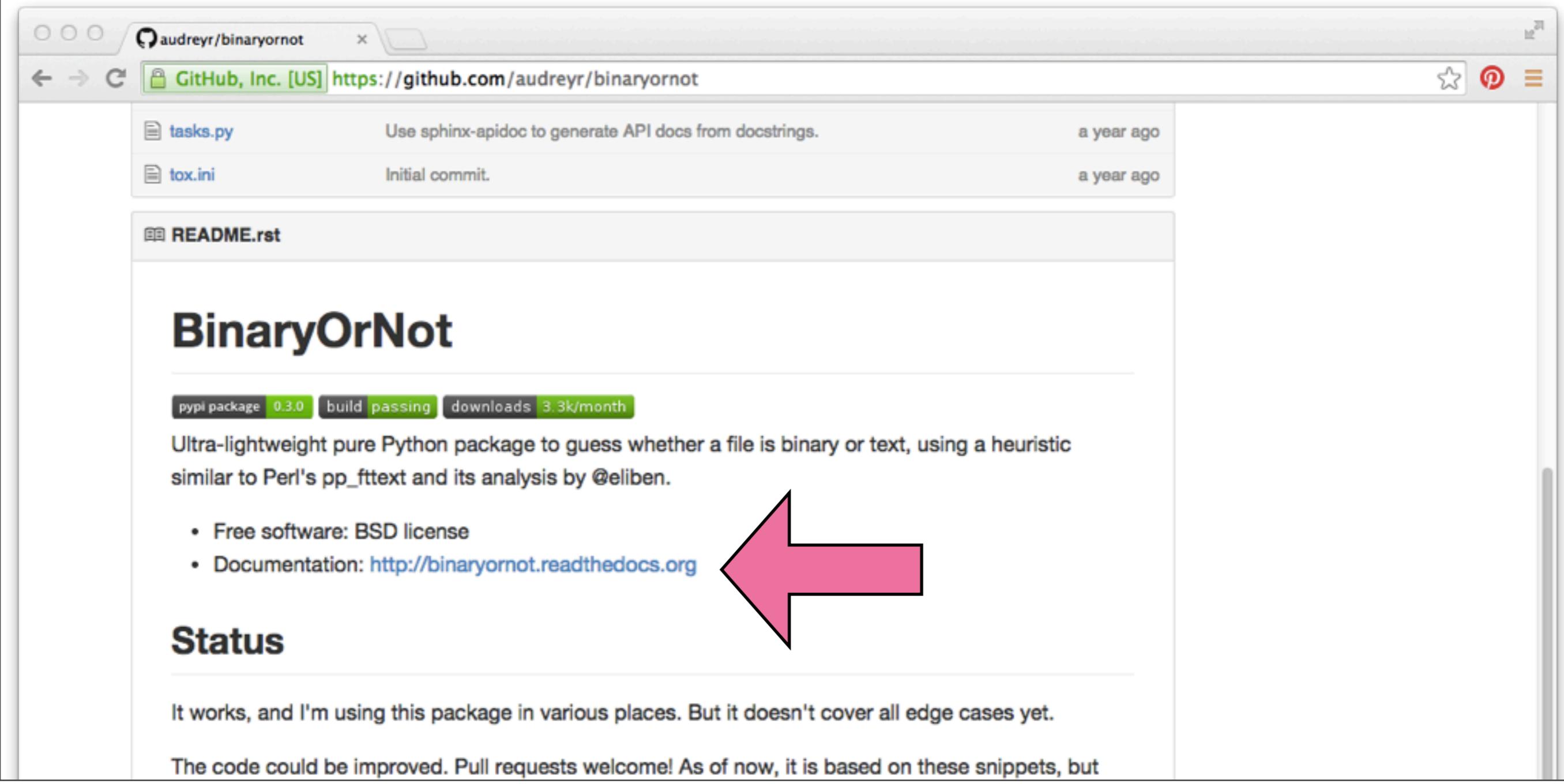
What about all the other files?

- `__init__.py` is needed for Python 2 compatibility
- `.rst` files, `docs/` are for documentation
- `setup.py`, `MANIFEST.in` is packaging cruft
- `tox`, `travis`, `tests/` are for testing

All this boilerplate can be auto-generated, don't worry

EXERCISE: EXPLORE A PACKAGE

Find the link to the package's documentation:



audreyr/binaryornot

GitHub, Inc. [US] <https://github.com/audreyr/binaryornot>

tasks.py Use sphinx-apidoc to generate API docs from docstrings. a year ago

tox.ini Initial commit. a year ago

README.rst

BinaryOrNot

pypi package 0.3.0 build passing downloads 3.3k/month

Ultra-lightweight pure Python package to guess whether a file is binary or text, using a heuristic similar to Perl's pp_fttext and its analysis by @eliben.

- Free software: BSD license
- Documentation: <http://binaryornot.readthedocs.org>

Status

It works, and I'm using this package in various places. But it doesn't cover all edge cases yet.

The code could be improved. Pull requests welcome! As of now, it is based on these snippets, but

EXERCISE: EXPLORE A PACKAGE

The documentation for a package teaches you how to install and use it.

A screenshot of a web browser window showing the documentation for the `BinaryOrNot` package. The URL in the address bar is `binaryornot.readthedocs.org/en/latest/`. The page title is "Welcome to BinaryOrNot's documentation!". On the left sidebar, there are sections for "Project Versions" (latest), "RTD Search" (with a search input field and "Go" button), "Table Of Contents" (listing "Welcome to BinaryOrNot's doc" and "Indices and tables"), "Next topic" (listing "BinaryOrNot"), and "This Page" (listing "Show Source" and "Show on GitHub"). The main content area displays a table of contents with the following items:

- BinaryOrNot
 - Status
 - Features
 - Why?
 - Credits
- Installation
- Quickstart
- Contributing
 - Types of Contributions
 - Get Started!
 - Pull Request Guidelines
 - Tips
- Credits
 - Development Lead

INSTALLING PACKAGES

Common package installation tools:

- pip
- anaconda (“conda”)
- easy_install

Always read a package’s docs for installation advice, though.

CREATING NEW PACKAGES

If you continue with Python coding, you'll want to create and release your own packages

- Look up **cookiecutter**
 - Tool for generating the package boilerplate files

WHERE TO GO FROM HERE

Your next adventures in Python coding

REVIEW ON YOUR OWN! STUDY!

Start with the official Python docs & tutorial:

The screenshot shows a web browser window displaying the Python 3.4.1 documentation. The title bar reads "Overview — Python 3.4.1". The address bar shows "Python Software Foundation [US] https://docs.python.org/3/". The page content includes a sidebar with links for "Download", "Docs for other versions" (listing Python 2.7, 3.3, 3.5, and Old versions), "Other resources" (PEP Index, Beginner's Guide, Book List, Audio/Visual Talks), and "Quick search". The main content area features the title "Python 3.4.1 documentation" and a welcome message: "Welcome! This is the documentation for Python 3.4.1, last updated Sep 18, 2014." It lists several documentation sections: "Parts of the documentation:", "What's new in Python 3.4?", "Tutorial", "Library Reference", "Language Reference", "Python Setup and Usage", "Python HOWTOs", "Indices and tables:", "Installing Python Modules", "Distributing Python Modules", "Extending and Embedding", "Python/C API", and "FAQs".

Download
Download these documents

Docs for other versions
Python 2.7 (stable)
Python 3.3 (stable)
Python 3.5 (in development)
Old versions

Other resources
PEP Index
Beginner's Guide
Book List
Audio/Visual Talks

Quick search
Enter search terms or a module, class or function name.

Go

Python 3.4.1 documentation

Welcome! This is the documentation for Python 3.4.1, last updated Sep 18, 2014.

Parts of the documentation:

What's new in Python 3.4?
or all "What's new" documents since 2.0

Tutorial
start here

Library Reference
keep this under your pillow

Language Reference
describes syntax and language elements

Python Setup and Usage
how to use Python on different platforms

Python HOWTOs
in-depth documents on specific topics

Indices and tables:

modules | index

FREE ONLINE UNIVERSITY COURSES

Normally something like this would cost \$\$\$, but we are lucky to get it for **free**:

coursera | Global Partners

Courses Specializations Institutions About ▾ | Sign In Sign Up



An Introduction to Interactive Programming in Python

Part of the "[Fundamentals of Computing Specialization](#) »

This course is designed to be a fun introduction to the basics of programming in Python. Our main focus will be on building simple interactive games such as Pong, Blackjack and Asteroids.



Look! It just started! Sign up!

About the Course

This course is designed to help students with very little or no computing background learn the basics of building simple interactive applications. Our language of choice,

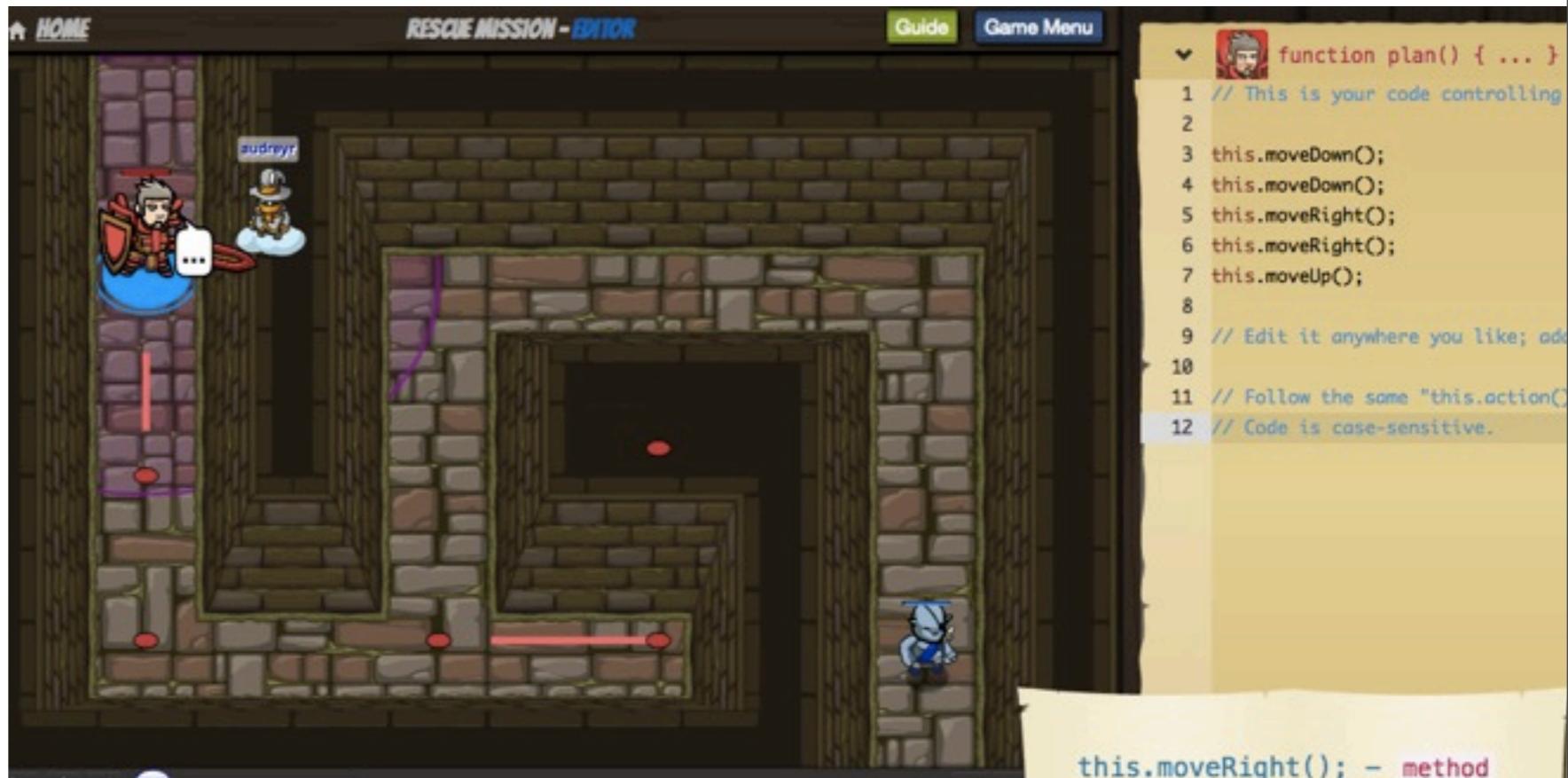
Sessions

Sep 15th 2014 - Nov 16th 2014

INTERACTIVE PYTHON WALKTHROUGHS/GAMES

Websites that turn learning Python into an interactive experience or puzzle game:

- [codewars.com/?language=python](https://www.codewars.com/?language=python)
- [codecademy.com/en/tracks/python](https://www.codecademy.com/en/tracks/python)
- [codecombat.com](https://www.codecombat.com)
- pythontutor.com



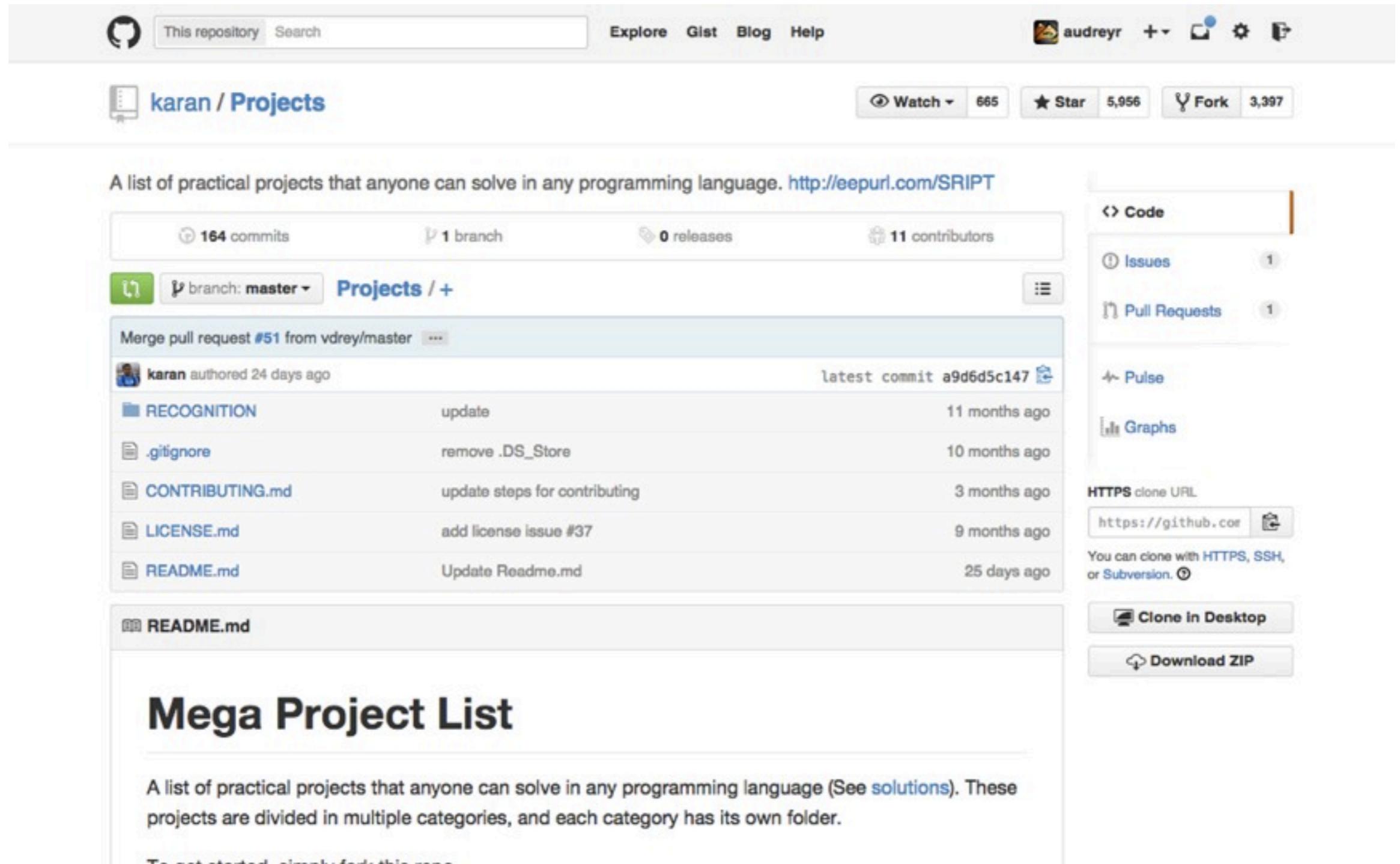
BOOKS (“OLD TECHNOLOGY”)

There's still nothing quite like learning from a book the old-fashioned way.



YOUR FIRST CODING PROJECT

Stuck without ideas? See Karan's Mega Project List:
<https://github.com/karan/Projects>



The screenshot shows the GitHub repository page for 'karan / Projects'. The repository has 164 commits, 1 branch, 0 releases, and 11 contributors. It contains files like RECOGNITION, .gitignore, CONTRIBUTING.md, LICENSE.md, and README.md. A merge pull request #51 from vdrey/master is shown. The repository has 665 stars and 3,397 forks. The 'Code' tab is selected, showing issues (1), pull requests (1), and graphs. A clone URL is provided: <https://github.com/karan/Projects>. Buttons for 'Clone in Desktop' and 'Download ZIP' are also present.

A list of practical projects that anyone can solve in any programming language. <http://eepurl.com/SRIPT>

164 commits 1 branch 0 releases 11 contributors

branch: master Projects / +

Merge pull request #51 from vdrey/master ...

karan authored 24 days ago latest commit a9d6d5c147

File	Action	Time
RECOGNITION	update	11 months ago
.gitignore	remove .DS_Store	10 months ago
CONTRIBUTING.md	update steps for contributing	3 months ago
LICENSE.md	add license issue #37	9 months ago
README.md	Update Readme.md	25 days ago

README.md

Mega Project List

A list of practical projects that anyone can solve in any programming language (See [solutions](#)). These projects are divided in multiple categories, and each category has its own folder.

To get started, simply fork this repo.

ASK QUESTIONS

Shy? Use a pseudonym. Many of us do that.

StackExchange ▾ 🔍 ⚡ 4,201 • 9 • 30 • 49 review help ▾ [python]

 stackoverflow Questions Tags Users Badges Unanswered Ask Question

Tagged Questions info newest 14 featured frequent votes active unanswered

Python is a dynamic and strongly typed programming language that is designed to emphasize usability. Two similar but incompatible versions of Python are in widespread use (2 and 3). Please consider mentioning the version and implementation that you are using when asking a question about Python.

[learn more...](#) | [improve tag wiki](#) | [top users](#) | [synonyms \(3\)](#) | [python jobs](#) 607

2801 What does the yield keyword do in Python?
What is the use of the yield keyword in Python? What does it do? For example, I'm trying to understand this code (**): def node._get_child_candidates(self, distance, min_dist, max_dist): if ...
19 answers 545k views

1877 What is a metaclass in Python?
What are metaclasses? What do you use them for?
9 answers 253k views

1321 How can I make a chain of function decorators in Python?

341,811 questions tagged python about »

Featured on Meta
 New! Smarter profile creation and syncing

Hot Meta Posts

- 11 People turning regular code blocks in others' question into snippets
- 58 Why won't the system allow me to ask questions for several days?
- 11 Automatically select jQuery in snippet editor if tagged with jQuery

Looking for a job?

ON THE INTERNET, YOU CAN BE ANONYMOUS

Take advantage of it:

- Ask for help liberally!
- Put your code online, e.g. GitHub
- Try putting a package on PyPI and see what happens

ON THE INTERNET, NOBODY KNOWS YOU'RE A NEW CODER

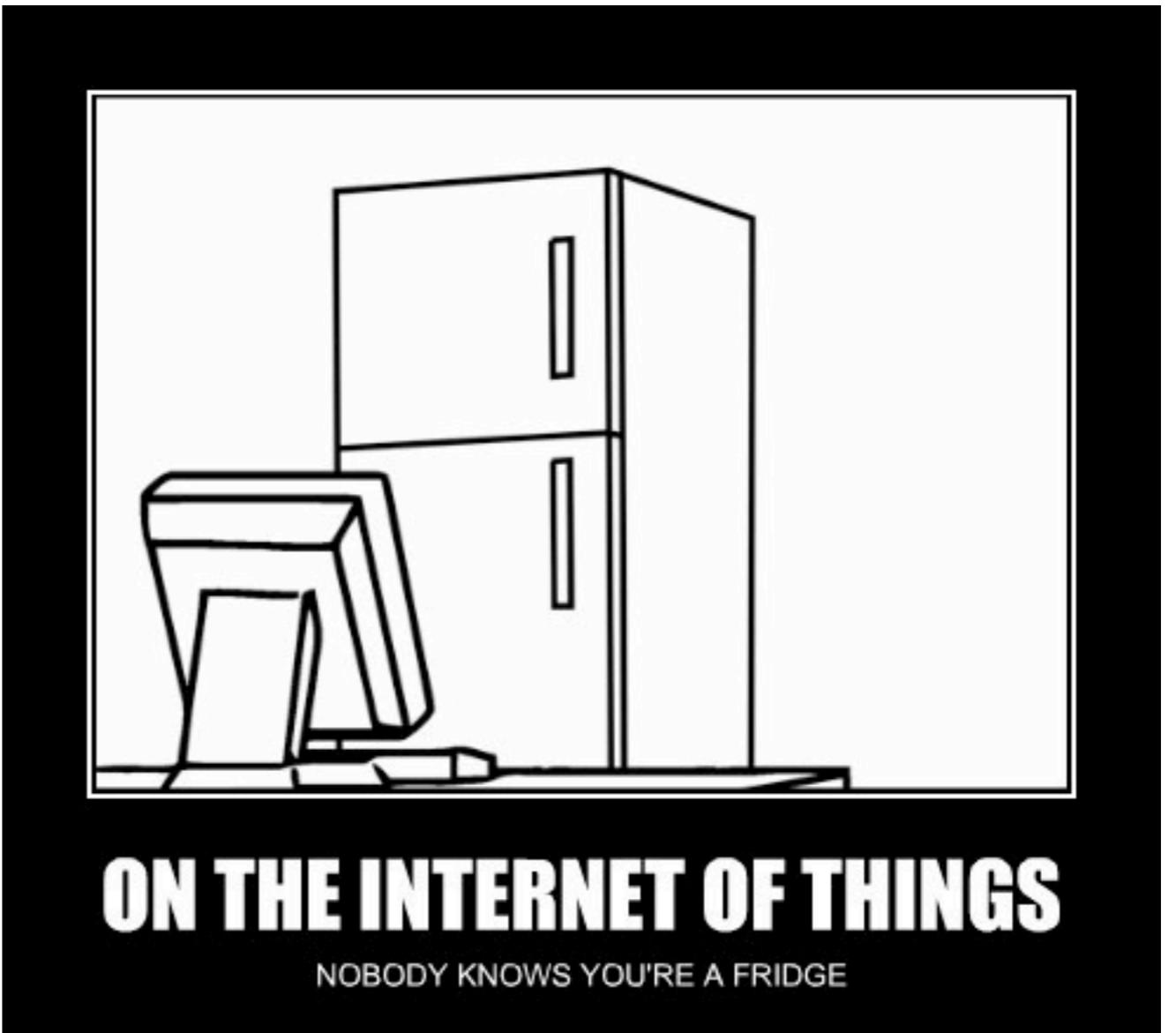


For all they know,
you could be a dog.

"On the Internet, nobody knows you're a dog."

ON THE INTERNET, NOBODY KNOWS YOU'RE A NEW CODER

Or even a fridge.



ON THE INTERNET OF THINGS

NOBODY KNOWS YOU'RE A FRIDGE

ATTEND CODING MEETUPS

Gotta catch ‘em all (free education, meet peers)

The screenshot shows the Meetup.com homepage with a blue header bar. In the top left, there's a red 'Meetup' button, followed by 'Find a Meetup Group' and 'Start a Meetup Group'. On the right side of the header are profile icons for a user (a woman), a message icon, a notification bell, and three dots for more options.

Your Next Meetup

SEP 20 Inland Empire Pyladies
Intro to Python Workshop (joint event)
Saturday, September 20 at 9:30 AM | 7 Comments

81 Meetups in your groups

57 Meetups with friends

Below the header, there's a search bar with 'Programming' and 'within 25 miles of San Diego, CA' filters, along with 'Groups' and 'Calendar' buttons. The main content area shows a list of events for Saturday, Sep 20, including the Python workshop mentioned above.

SATURDAY, SEP 20

9:30 AM Inland Empire Pyladies
Yes **Intro to Python Workshop (joint event)**
4 Pythonistas going

1 friend

All Meetups

My Meetups & suggestions

My Meetups

User stats

JOIN THE NEAREST PUG

PUG means Python User Group



San Diego Python

<http://www.meetup.com/pythonsd/>



San Diego Pyladies

<http://www.meetup.com/pythonsd/>

Too far from home? Search for others like Inland Empire
Pyladies, Inland Empire Python, SoCal Python, LA
Python, LA Pyladies, OC Python, etc.

DEVELOPER CONFERENCES

Like a firehose of coding knowledge.

Great upcoming opportunity:
BarCamp Django SF (\$15 tix!)



Other really good ones:



PyCon 2015
Montréal • April 8-16



KEEP ON CODING!

We are all unpaid volunteers, here because we want to help the community.

- So take the time to keep learning.
- You get as much out of this workshop as you put in.

Thank you!



QUICK SURVEY

Please help us by doing this survey **right now**:

- <https://www.surveymonkey.com/s/8KX9GGQ>

Your feedback helps us plan future community workshops and events.

