

APIs + Python

A HearMeCode API workshop!

Ready to get started? Make sure you have pip installed. Instructions here:

https://github.com/shannonturner/python-lessons/blob/master/installing_pip.md

- What are APIs?
- How do we access them with Python?
 - Sample code on GitHub!
- Awesome API examples!
- API workshop time

What is an API?

- “Application Programmer Interface”
- A way to interact programmatically with a website or web service
- Get data or make a website, service or device do something

APIs? Websites?

- The easiest way to interact with (many) APIs is through the browser
- You can visit an *endpoint* to get or post data, and use *parameters* to refine what you need

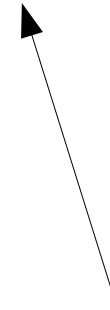
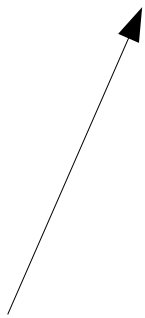
APIs in the browser

Example:

<http://openstates.org/api/v1/legislators>

Main API url


Endpoint




APIs in the browser

Example:

[http://openstates.org/api/v1/legislators?
api_key=blah](http://openstates.org/api/v1/legislators?api_key=blah)



Special parameter: API Key
gives you access



?: tells you that parameters
are coming next

API Keys

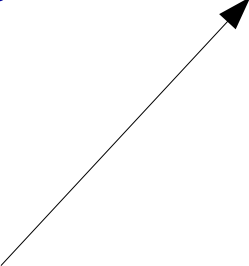
- Many APIs require you to get a key in order to use them
- Sometimes you have to pay, but often you just have to agree to terms of service
- Lets the API's owner keep stats about different kinds of users...
- ...or shut you down easily if you start abusing the service

APIs in the browser

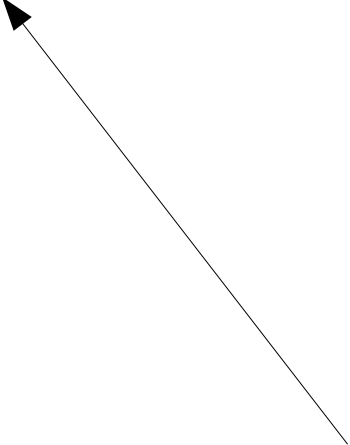
Example:

[http://openstates.org/api/v1/legislators?
api_key=blah&state=WI](http://openstates.org/api/v1/legislators?api_key=blah&state=WI)

&: separates parameters
from each other



Search parameters narrow the
information you get back



APIs in Python

- Use requests to act like the browser
- Store the information you get back
- Process it/combine it/display it
- Profit!

Detour: Python Modules

- Python comes with lots of stuff built in
 - `print "abc"`
 - `"abc def".split()`
 - `"WeiRd CaSES".lower()`
 - ...

Sometimes you have to ask...

- Some libraries that come with Python aren't loaded right away
- Keeps things running faster and cuts down on clutter
- Examples
 - csv deals with CSV documents
 - datetime is smart about dealing with date formats
 - os lets you play with your operating system to do things like move files around
- To use these, you have to import them, but they're already installed

Importing modules

- There are several ways to import modules, but the easiest is:

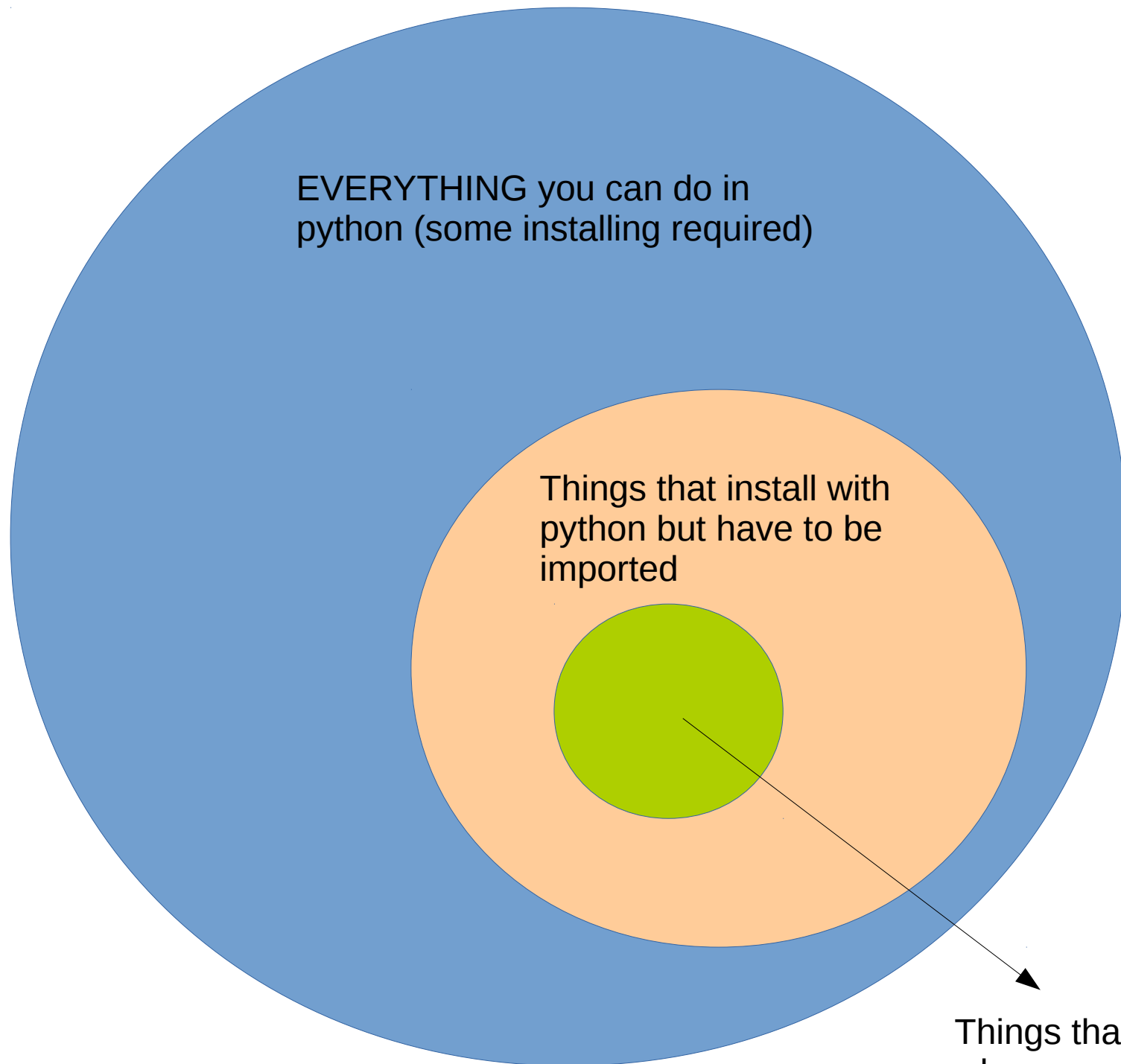
```
import [modulename]
```

Ex:

```
import csv  
  
with open("myfile.csv", "r") as f:  
    r = csv.reader(f)  
    ...
```

Some modules have to be installed

- Keeps python download from being HUGE
- Tends to be for modules that are less common, newer, or written by random folks
 - Yes, anyone can write python modules!
- Ex:
 - django (for websites)
 - numpy (for complicated math)
 - requests (for hitting APIs and websites – we'll use this today!)



EVERYTHING you can do in
python (some installing required)

Things that install with
python but have to be
imported

Things that get loaded
when you start python

Installing modules with Pip

- It can be hard to keep track of what you've installed and get the right versions
- Pip takes care of it all automagically!
- Installing with pip:

```
pip install packagename
```

Installing requests

- Open a terminal/shell window
- Type:
`sudo pip install requests`
- Enter your system password when prompted
- To confirm:
 - Open python (either through IDLE or by typing `python` on the command line)
 - Type `import requests` and run!
 - No error? Good to go!

Example API: Github!

- You can access public information on github via their API!
- For example, I want to know what issues have been closed recently on a project I've been working on.
- I can see that information in my browser by going to this link:

<https://api.github.com/repos/sunlightlabs/openstates/issues?state=closed>

Or...via python!

```
import requests
```

```
r=requests.get("https://api.github.  
com/repos/sunlightlabs/openstates/i  
ssues?state=closed")
```

```
print r.json()
```

Why python?

- Now I can play with the results!
- API responses come in standard formats (in this case, JSON, which is very much like a python dictionary)
- Examples of what you can do with API data in python?

Sample API code

- On Github:
 - What are the first names of the people in congress? How many congresspeople have the same first name? (opencongress)
 - List the movie titles that contain your first name, and the years those movies came out. (Movies, NOT TV episodes!) (imdb)
 - Find all contributions from a github repo and find out the most common day of the week for commits to be submitted (github)

<https://github.com/rshorey/api-sample-code>

Fun APIs to play with

- Wikipedia:
http://www.mediawiki.org/wiki/API:Main_page
- Twitter: <https://dev.twitter.com/>
- Census:
<http://www.census.gov/data/developers/data-sets.html>
- Yelp:
<http://www.yelp.com/developers/documentation>

APIs to play with

- Talk like a pirate API:
<http://isithackday.com/arrpi.php>
- My little pony images:
<http://www.programmableweb.com/api/ponyfaces>
- Lorem Ipsum generator:
<https://www.mashape.com/montanaflynn/lorem-text-generator>