Introduction to Using APIs with Python

NaLette Brodnax
Indiana University Bloomington
School of Public & Environmental Affairs
Department of Political Science
Center of Excellence for Women in Technology
CEWiT
CENTER OF EXCELLENCE
FOR WOMEN IN TECHNOLOGY

We started a conversation about women in technology, ensuring all women have a seat at the table in every technology venture.

CEWiT addresses the global need to increase participation of women at all stages of their involvement in technology related fields.

Faculty, staff, alumnae and student alliances hold events, host professional seminars, and give IU women opportunities to build a community.

CONNECT WITH US

iucewit@IU_CEWiT
cewit.indiana.edu

We started a conversation about women in technology, ensuring all women have a seat at the table in every technology venture.

CEWiT addresses the global need to increase participation of women at all stages of their involvement in technology related fields.

Faculty, staff, alumnae and student alliances hold events, host professional seminars, and give IU women opportunities to build a community.

CONNECT WITH US

iucewit@IU_CEWiT
cewit.indiana.edu
what is an API?

An application programming interface (api) is a tool that allows computers to exchange data.
1. static webpage
2. dynamic webpage
3. application programming interface (api)
uses of APIs

• Social – Twitter, Facebook, etc.
• Internet – bit.ly, domain registration
• Mapping – Google Maps, Bing Maps, etc.
• Search – Google, Yahoo, etc.

APIs make information transferred across the web digestible for a computer.
key protocols

• HTTP – communicating with web server
• OAuth – accessing secure information
part 2: python overview
getting the tools

• Use Python on SSRC computer
  • Search for IDLE

• Use Python on your laptop
  • Install requests package using the command line

Sample installation code for Mac OS X using Terminal:

$ pip3 install requests_oauthlib
integrated development environment

1. Python interpreter (required)
2. Text editor (optional)
3. Command line (optional)
python development environment

IDE ➔ IDLE

Interact with Python

Write programs in a separate screen: **File ➔ New File**
Run the program in the interpreter: **F5** or **Run ➔ Run Module**
data types: sequences

**String**—ordered sequence of characters

```javascript
'happy'
```

**List**—ordered sequence of items

```javascript
['Leia', 'Rey', 'Maz']
```

**Dictionary**—unordered sequence of key-value pairs

```javascript
{'name': 'Kylo', 'side': 'dark'}
```
part 3: data collection!
the API we will access

Twitter REST API
http://dev.twitter.com

Workshop Code
https://github.com/nmbrodnax/wim-workshop
• twitter_api.py
• twitter_auth_example.txt
<table>
<thead>
<tr>
<th></th>
<th>Web Server</th>
<th>Web Server + API</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Review</strong></td>
<td>HTML structure (tags, attributes, etc.)</td>
<td>Parameters and structure from documentation</td>
</tr>
<tr>
<td><strong>Access</strong></td>
<td>No registration, no authentication</td>
<td>Registration and sometimes authentication</td>
</tr>
<tr>
<td><strong>Parse</strong></td>
<td>HTML</td>
<td>JSON or XML</td>
</tr>
<tr>
<td><strong>Transform</strong></td>
<td>Nested tables, lists</td>
<td>Nested dictionary</td>
</tr>
<tr>
<td><strong>StORe</strong></td>
<td>Text, CSV</td>
<td>Text, CSV</td>
</tr>
</tbody>
</table>
next steps

- Register as a developer
- Create an application
- Create an authentication document
- Use the API
registration

https://dev.twitter.com/#

Welcome to the Twitter Platform

Dive into documentation

Learn about our products
Create an application

Application Details

Name *

practice_application

Your application name. This is used to attribute the source of a tweet and in user-facing authorization screens. 32 characters max.

Description *

Practice connecting to the API

Your application description, which will be shown in user-facing authorization screens. Between 10 and 200 characters max.
Review • Access • Parse • Transform • stORe
review

https://dev.twitter.com/rest/public/search
Your application has been created. Please take a moment to review and adjust your application's settings.

**wim_practice_application**

**Settings**

Practice connecting to the API

http://oewit.indiana.edu

**Organization**

*Information about the organization or company associated with your application. This information is optional.*

<table>
<thead>
<tr>
<th>Organization</th>
<th>None</th>
</tr>
</thead>
<tbody>
<tr>
<td>Organization website</td>
<td>None</td>
</tr>
</tbody>
</table>
access

Your Twitter developer credentials

```javascript
{
  'consumer_key': 'your_consumer_key',
  'consumer_secret': 'your_consumer_secret',
  'access_token': 'your_access_token',
  'access_secret': 'your_access_secret'
}
```
let’s take a 10-minute break!
Import statements allow you to add functions

```python
import sys
import csv
import from requests_oauthlib import OAuth1Session
```
access

# get authentication parameters from local file
local_file = 'your_file_path'
with open(local_file) as txtfile:
    contents = txtfile.readline()
    credentials = eval(contents.strip('
'))
# api OAuth 1.0 authentication

twitter = OAuth1Session(
    credentials.get('consumer_key'),
    client_secret=credentials.get('consumer_secret'),
    resource_owner_key=credentials.get('access_token'),
    resource_owner_secret=credentials.get('access_secret'))
access

# host location of api
host = 'https://api.twitter.com'

# api GET request for user ids of followers
get_path = '/1.1/search/tweets.json?q=%40IUBloomington'
url = host + get_path
response = twitter.get(url)
# check the HTTP response code
print(response)
# parse the JSON data into a python object
tweets = response.json()
# check the structure of the data
print(len(tweets))
print(type(tweets))
print(tweets.keys())
print(len(tweets['statuses']))

# encode uncommon characters
non_bmp_map = dict.fromkeys(range(0x10000, sys.maxunicode + 1), 0xfffd)
print(str(tweets['statuses']).translate(non_bmp_map))
store

- text file
- CSV file
- other formats
run your script!

From IDLE:
Run the program in the interpreter: F5 or Run → Run Module

OR

From the Command Line:
$ python3 twitter_api.py
Review • Access • Parse • Transform • stORe
Thank you!

email: nbrodanax@indiana.edu

linkedin: nalettebrodnax
github: nmbrodnax
twitter: @nbrodanax