Accessing APIs: Easier Than Web Scraping

Metis Fall 2018

Brenner Heintz

What are APIs?

- Application Programming Interfaces
- Think of them like user interfaces, but for a different kind of user – developers
- Allow you to ask a server for specific information
    - Yelp: A restaurant review
    - Rotten Tomatoes: A movie rating
    - Google Maps: Your house's GPS coordinates

We could all use more RESTful APIs

- REpresentational State Transfer
- Restful APIs will allow you to fetch information, and modify information, in standardized, predictable way
- Most APIs use the RESTful format
- Some also allow SQL queries
- ‘GET’, ‘PUT’, ‘POST’, and ‘DELETE’

APIs don’t have to be tough

- Many libraries in Python for specific APIs
- Google Maps, Reddit, etc. have “batteries included” APIs that other developers have created to make it really easy
- What if you are querying a smaller API?
- What did anyone do *before* batteries were included?
- Requests module

Let's See Who's On Fire Right Now

An Example

[22]:
```
import requests
import pprint
```

Let's make an API call!

[41]:
```
# Find the endpoint for our data
url = 'https://data.seattle.gov/resource/fire-911.json?'
```

[3]:
```
# Use Requests module to query the API
r = requests.get(url)
```

[4]:
```
# Make sure our request went through as planned
# 200 = OK, 400 = Client Error, 404 = Not Found
r.status_code
```

[4]:
```
200
```

[5]:
```
# Process the data, and make it pretty
data = r.json()
pprint.pprint(data[:3])
```

```
[{'type': ' --T::00'},
 {'address': '10049 College Way N',
  'incident_number': 'F110104009',
  'latitude': '47.701756',
  'longitude': '-122.335022',
  'report_location': {'latitude': '47.701756',
                      'longitude': '-122.335022',
                      'needs_recoding': False},
  'type': 'Aid Response'},
 {'address': '5929 Beach Dr Sw',
  'incident_number': 'F110104008',
  'latitude': '47.550431',
  'longitude': '-122.397816',
  'report_location': {'latitude': '47.550431',
                      'longitude': '-122.397816',
                      'needs_recoding': False},
  'type': 'Aid Response'}]
```

[ ]:
```

```

Now let's take it a step further - filtering our data set to an address we're looking for

[23]:
```
# We can tell the API what data we want by using a filter. Let's filter for a specific address
filters = 'address=5929 Beach Dr Sw'
```

[24]:
```
new_url = url + filters
```

[25]:
```
# Add HTML 'Headers', including our API key - this allows us to make more calls of the API
# NOTE!!! Use %load_dotenv if your API key is sensitive - never post your key on Github!
headers = {'APP_TOKEN': 'IEEzdsdfEalN53LtblAij63MIIym'}
```

[26]:
```
# Calling the API - with our new filter, and the API key
r = requests.get(new_url, headers=headers)
```

[27]:
```
r.status_code
```

[27]:
```
200
```

[28]:
```
data = r.json()
pprint.pprint(data[:6])
```

```
[{'address': '5929 Beach Dr Sw',
  'incident_number': 'F110104008',
  'latitude': '47.550431',
  'longitude': '-122.397816',
  'report_location': {'latitude': '47.550431',
                      'longitude': '-122.397816',
                      'needs_recoding': False},
  'type': 'Aid Response'}]
```

Thank You

Brenner Heintz

Github: @athena15

brenner.heintz@gmail.com