

Notebook 1 - Twitter API

By Michael Carlin and Brock Ogle

```
In [5]: from requests_oauthlib import OAuth1Session
from requests_oauthlib import OAuth1
import requests
import json
import webbrowser
#import the currently required modules
```

To create the Twitter API we visited '<https://apps.twitter.com> (<https://apps.twitter.com>)' and after login in, navigated from 'Create New App' through the process of creating a new RESTful API. After Creating the API the consumer key and consumer secret can be found by navigating back to '<https://apps.twitter.com> (<https://apps.twitter.com>)', selecting the App you are using, and navigating to the 'Keys and Access Tokens' tab. It is here the Consumer Key and Consumer Secret are located. There is also a section to generate an access code, although this method is used only for the API owner to use the API. The steps to the OAuth1 dance were learned and adjusted using the resources provided by '<https://developers.twitter.com> (<https://developers.twitter.com>)' as well as trial and error.

```
In [6]: #Common URLs for use in OAuth1 Dance
REQUEST_TOKEN_URL = 'https://api.twitter.com/oauth/request_token'
ACCESS_TOKEN_URL = 'https://api.twitter.com/oauth/access_token'
AUTHORIZATION_URL = 'https://api.twitter.com/oauth/authorize'
```

```
In [7]: #Found in 'Keys and Access Tokens' for API on https://apps.twitter.com
consumer_key = 'mKaokNDq5tfujhkCHSxvn9Z3j'
consumer_secret = 'yETdUxoAGR0I3stzBUd57wTBTW8b1PMa8sIJHrRS5oIZH88Y8D'
```

```
In [8]: #Initial step of OAuth1 dance, establishes client connection
oauth_client = OAuth1Session(consumer_key,consumer_secret)
```

```
In [9]: #fetches request authorization token to be used in Authorization URL
resp = oauth_client.fetch_request_token(REQUEST_TOKEN_URL)
resp
```

```
Out[9]: {u'oauth_callback_confirmed': u'true',
u'oauth_token': u'NPrQwAAAAAAA3fpXAAABYEoEAgy',
u'oauth_token_secret': u'p6J5Vwua0XfWMJZAQdbpQb7IG2FgVJwq'}
```

```
In [10]: #establishes an authorization endpoint
url = oauth_client.authorization_url(AUTHORIZATION_URL)
```

```
In [11]: #Takes user to authorization endpoint and requires login before moving on
webbrowser.open(url)
```

```
Out[11]: True
```

```
In [12]: #User must type in code provided by authorization endpoint to verify they
#at the correct authorization endpoint
pincode = raw_input('\nCopy and paste code: ')
```

Copy and paste code: 1919289

```
In [13]: #Establishes new client connection
oauth_client = OAuth1Session(consumer_key,client_secret=consumer_secret,
                             resource_owner_key=resp.get('oauth_token')
                             resource_owner_secret=resp.get('oauth_token_secret')
                             verifier=pincode)
```

```
In [14]: #Requests and returns access token and access token secret
resp1 = oauth_client.fetch_access_token(ACCESS_TOKEN_URL)
```

```
In [15]: #Sets access token and secret to variables
access_token = resp1['oauth_token']
access_token_secret = resp1['oauth_token_secret']
```

```
In [16]: #If response is 200, authorization tokens are valid
url = 'https://api.twitter.com/1.1/account/verify_credentials.json'
auth = OAuth1(consumer_key, consumer_secret, access_token, access_token_secret)
requests.get(url, auth=auth)
```

Out[16]: <Response [200]>

```
In [17]: #Creates a dictionary of valid authorization tokens
tokens = {'consumer':{'consumer_key':consumer_key,
                     'consumer_secret':consumer_secret},
          'access_token':{'access_token':access_token,
                          'access_secret':access_token_secret}}
```

```
In [18]: #Uses json module to dump tokens dictionary to file in order
#to be used in Notebook 2
with open('tokens1.json','w') as f:
    json.dump(tokens,f)
```

In []: