**Introduction**

Let’s use Python and grab publicly available Twitter post data (from any user you specify) and convert JSON into a tabular so that we can analyse the data through Excel or insert them into a relational database. Python has a package that is a wrapper around the Twitter API (<https://pypi.python.org/pypi/python-twitter/>). The package is easy to use and works fine. In this post, I used the generic requests package to make API calls to the endpoint for timeline.

**Twitter API** (https://developer.twitter.com)

Twitter API is a REST-based API and use both OAuth1.0 and OAuth2.0 for authentication depending on the application you are making. We use OAuth1.0 here.

To obtain the API credentials, you need your Twitter account and go to the Twitter Apps page (<https://apps.twitter.com/>). Then, click ‘Create New App’ and enter Name, Description and Website. Callback URL can be left empty. You can see Consumer Key (API Key) and Consumer Secret (API Secret) once you create an app. You will need to generate access token and secret through the UI by clicking the button.

**How the code works**

It takes 9 arguments, including the API credentials, Twitter user (whom you want to get the data), number of records to ingest, path to create json response files, path + filename for csv files. The program creates one timeline table for tweet and another table with the latest user information.

The code runs in both Python 2.7 and 3. The major challenge for backward compatibility is the way Python 2.7 handles Unicode (it is quite different from Python 3). Therefore, you need to uncomment or replace a few lines as indicated in the code.

**Example call**

python twitter\_timeline\_scraper.py <api\_key> <api\_secret> <access\_token> <access\_token\_secret> <screen\_name (e.g. CocaCola)> <record\_no (e.g. 3000)> <json\_path (e.g. /tmp/twitter/raw/)> <timeline\_file\_path (e.g. /tmp/twitter/timeline.csv)> <user\_file\_path (e.g. /tmp/twitter/user.csv)> >

**Key Points**

For the user\_timeline data, 200 is the maximum number of the record you can retrieve at a time. To obtain more than 200 records, we need to use the max\_id parameter to specify which record is the first for each iteration.

The possibly\_sensitive field may be missing. Therefore, use the key error exception to assign no value in case it is missing.

Enjoy!