

Staff Lecture: Python + Twitter



Announcements

Fill out course evaluations!

- we care a lot about what you think!
- extra credit for posting screenshot of confirmation page to Canvas

Today

- staff lectures

Plan for Today

Plan for Today

Use Python to build a tree of Tweet replies

- think "tree" in Linux
- Python is cool
- there's a library for (almost) everything
- don't write code that you don't have to write!

Requirements

Requirements

twarc

```
$ pip3 install twarc
```

- created specifically to pull data from Twitter
- handles authentication in one line
- not very well-documented, but it has the reply-searching functionality we're looking for

Requirements

twarc

```
$ pip3 install twarc
```

- created specifically to pull data from Twitter
- handles authentication in one line
- not very well-documented, but it has the reply-searching functionality we're looking for

anytree

```
pip3 install anytree
```

- simple and easy to use
- gives node functionality to any user-defined class
- builds AND renders trees

Let's get started!

Step 1: Starter Code

Go to <https://tinyurl.com/398-tweet>

Step 1: Starter Code

Go to <https://tinyurl.com/398-tweet>

`bot.py` contains starter code for the Python program

`secret.py` will hold your API keys and tokens

- you want to keep your API keys secret, which is why they are placed in a separate file from your program.
- if you choose to put your project on GitHub, do NOT commit `secret.py` !

Step 2: Generate Twitter API Keys and Tokens

Step 2: Generate Twitter API Keys and Tokens

Go to developer.twitter.com

NOTE: You will need a Twitter developer account from this point on.

- sign in then click "Create an app" in the upper right
- fill out the name, description, website, and intent of your application
- check the "Developer Agreement" box and finish creating your app

Step 2: Generate Twitter API Keys and Tokens

Step 2: Generate Twitter API Keys and Tokens

Consumer Keys

- located under App Details, Keys and Tokens
- copy and paste these keys into `secret.py`

Access Tokens

- scroll down and click "Create"
- copy and paste these keys into `secret.py`

Let's code!

Step 3: Import Libraries

Step 3: Import Libraries

TODO: import libraries

- `from twarc import Twarc`
- `from anytree import NodeMixin, RenderTree`
- `from secret import *`

Step 4: Twitter

Step 4: Twitter

TODO: create a new Twitter session

- check out the twarc [documentation](#)
- use keys/tokens from `secret.py` to create the session

Step 4: Twitter

TODO: create a new Twitter session

- check out the twarc [documentation](#)
- use keys/tokens from `secret.py` to create the session

TODO: user input

- ask the user (you) what Tweet your app should build a tree for
- user input is a Tweet ID, which can be found in a Tweet's URL

Step 4: Twitter

Step 4: Twitter

TODO: get the user's requested Tweet

- make twarc do the work for us
- returns a Python dictionary containing all the Tweet information

Step 5: "Take the Recursive Leap of Faith!"

Step 5: "Take the Recursive Leap of Faith!"

TODO: implement `get_replies()`

Step 5: "Take the Recursive Leap of Faith!"

TODO: implement `get_replies()`

Getting Twitter replies is difficult...

Step 5: "Take the Recursive Leap of Faith!"

TODO: implement `get_replies()`

Getting Twitter replies is difficult...

...but twarc will do it for us!

Step 5: "Take the Recursive Leap of Faith!"

TODO: implement `get_replies()`

Getting Twitter replies is difficult...

...but twarc will do it for us!

...but twarc's documentation isn't great.

Step 5: "Take the Recursive Leap of Faith!"

Step 5: "Take the Recursive Leap of Faith!"

TODO: implement `get_replies()`

- check out `test_twarc.py` in the twarc GitHub repo for examples of how to use `replies()`
- `get_replies()` should get the replies of replies of replies of replies, add them as nodes in our tree, and return the root

Step 6: Display the Tree

Step 6: Display the Tree

TODO: call `RenderTree()`

- check out anytree's [documentation](#) for examples of `RenderTree()`
- call string function to print Tweet objects

Congratulations!

You now know how to connect to the Twitter API with Python and display Tweets in a cool way.