

Criando Processo de coleta do Twitter

Passo 1: Criando conta no Twitter:

O primeiro passo é criarmos uma conta de desenvolvedor no Twitter. Portanto, registre uma nova conta ou faça login em uma conta existente. Inscreva-se para uma conta de desenvolvedor: <https://developer.twitter.com/en.html>. Se solicitado, escolha uma conta pessoal ou individual em vez de uma conta comercial. Ao enviar um uso pretendido, **certifique-se que sua resposta é suficientemente descritiva**, e não apenas alguma declaração de lixo ou cópia qualquer.

Uma vez a conta de desenvolvedor aprovada, crie um App: Registre um aplicativo em <https://developer.twitter.com/en/apps>. Para o campo obrigatório do site, você pode colocar um espaço reservado. Após a configuração, vá para 'Token e Chaves' para gerar um token e um segredo de acesso. Juntamente com as chaves da API do consumidor (já geradas), essas são as credenciais que usamos para nos comunicar com a API do Twitter.

Passo 2: Instale o Tweepy

Existem algumas bibliotecas disponíveis para interagir com a API do Twitter. No nosso laboratório usaremos o Tweepy, ele faz chamadas confiáveis e é adequado para lidar com tweets com mais de 140 caracteres. Para instalá-lo execute o comando como administrador:

```
C:\WINDOWS\system32>pip install tweepy
Collecting tweepy
  Downloading https://files.pythonhosted.org/packages/36/1b/2bd38043d22ade352fc3d3902cf30ce0e2f4bf285be3b304a2782a767aec/tweepy-3.8.0-py3-none-any.whl
Requirement already satisfied: six>=1.10.0 in c:\users\mateus.balen\appdata\roaming\python\python37\site-packages (from tweepy) (1.11.0)
Collecting requests-oauthlib>=0.7.0 (from tweepy)
  Downloading https://files.pythonhosted.org/packages/a3/12/b92740d845ab62ea4edf04d2f416d82532b5a0b083836d4d4e71c6f3d379/requests_oauthlib-1.3.0-py2.py3-none-any.whl
Collecting PySocks>=1.5.7 (from tweepy)
  Downloading https://files.pythonhosted.org/packages/8d/59/b4572118e098ac8e46e399a1dd0f2d85403ce8bbaad9ec79373ed6bada9/PySocks-1.7.1-py3-none-any.whl
Requirement already satisfied: requests>=2.11.1 in c:\users\mateus.balen\appdata\roaming\python\python37\site-packages (from requests-oauthlib>=0.7.0->tweepy) (2.22.0)
Collecting oauthlib>=3.0.0 (from requests-oauthlib>=0.7.0->tweepy)
  Downloading https://files.pythonhosted.org/packages/05/57/ca2e7a8fa7c0afb54a0581b14a65b56e62b5759dbc98e0627142b8a3704/oauthlib-3.1.0-py2.py3-none-any.whl (147kB)
    | 153kB 595kB/s
Requirement already satisfied: idna<2.9,>=2.5 in c:\users\mateus.balen\appdata\roaming\python\python37\site-packages (from requests>=2.11.1->tweepy) (2.8)
Requirement already satisfied: urllib3!=1.25.0,!=1.25.1,<1.26,>=1.21.1 in c:\users\mateus.balen\appdata\roaming\python\python37\site-packages (from requests>=2.11.1->tweepy) (1.25.3)
Requirement already satisfied: chardet<3.1.0,>=3.0.2 in c:\users\mateus.balen\appdata\roaming\python\python37\site-packages (from requests>=2.11.1->tweepy) (3.0.4)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\mateus.balen\appdata\roaming\python\python37\site-packages (from requests>=2.11.1->tweepy) (2019.3.9)
Installing collected packages: oauthlib, requests-oauthlib, PySocks, tweepy
Successfully installed PySocks-1.7.1 oauthlib-3.1.0 requests-oauthlib-1.3.0 tweepy-3.8.0
WARNING: You are using pip version 19.2.1, however version 19.3.1 is available.
You should consider upgrading via the 'python -m pip install --upgrade pip' command.

C:\WINDOWS\system32>
```

Passo 3: Testando rapidamente as credenciais e a biblioteca

Credenciais da API definidas, Tweepy instalado. Agora é hora de ver se funciona, cole o código abaixo em um bloco de notas, insira suas próprias chaves (observe os erros de digitação!) E execute.

```
import tweepy

#Adicione suas credenciais
twitter_keys = {
    'consumer_key': '---YOUR-KEY---',
    'consumer_secret': '---YOUR-KEY---',
    'access_token_key': '---YOUR-KEY---',
    'access_token_secret': '---YOUR-KEY---'
}

#Configura acesso a API
auth = tweepy.OAuthHandler(twitter_keys['consumer_key'], twitter_keys['consumer_secret'])
auth.set_access_token(twitter_keys['access_token_key'], twitter_keys['access_token_secret'])

api = tweepy.API(auth)

#Faz a chamada da sua timeline e imprime cada texto dos tweets
public_tweets = api.home_timeline()
for tweet in public_tweets:
    print(tweet.text)
```

```
import tweepy

#Adicione suas credenciais

twitter_keys = {

    'consumer_key': '---YOUR-KEY---',

    'consumer_secret': '---YOUR-KEY---',

    'access_token_key': '---YOUR-KEY---',

    'access_token_secret': '---YOUR-KEY---'

}

#Configura acesso a API

auth = tweepy.OAuthHandler(twitter_keys['consumer_key'],
twitter_keys['consumer_secret'])

auth.set_access_token(twitter_keys['access_token_key'],
twitter_keys['access_token_secret'])

api = tweepy.API(auth)

#Faz a chamada da sua timeline e imprime cada texto dos tweets

public_tweets = api.home_timeline()

for tweet in public_tweets:

    print(tweet.text)
```

Você acabou de fazer sua primeira chamada à API do Twitter. Agora você deve observar um fluxo de tweets na linha do tempo da sua conta (os tweets de todos usuários que sua conta segue). Vai parecer algo assim:

```
PS C:\Users\mateus.balen\OneDrive - COMPASSO TECNOLOGIA LTDA\Documentos\Capacitações\Bolsas\Exercícios\10_AmazonKinesis\Testes> c
d 'c:\Users\mateus.balen\OneDrive - COMPASSO TECNOLOGIA LTDA\Documentos\Capacitações\Bolsas\Exercícios\10_AmazonKinesis\Testes';
$(env:PYTHONIOENCODING='UTF-8'; ${env:PYTHONUNBUFFERED}=1; & 'C:\Program Files\Python37\python.exe' 'c:\Users\mateus.balen\vs
code\extensions\ms-python.python-2019.3.6558\pythonFiles\ptvsd_launcher.py' '--default' '--client' '--host' 'localhost' '--port'
'60004' 'c:\Users\mateus.balen\OneDrive - COMPASSO TECNOLOGIA LTDA\Documentos\Capacitações\Bolsas\Exercícios\10_AmazonKinesis\Tes
tes\teste_credentials.py')
Luxemburgo já foi alvo de estudo de Jorge Jesus https://t.co/OfbJuwRBry via @UOLEsporte @UOL
RT @RealAlora: Remote vibrator on a date is a must💎
Buy now💎
https://t.co/5UYjf659Uz https://t.co/et24tgilLz
Justiça quebra sigilo do escritório de Thomaz Bastos, eminência do governo do carniça https://t.co/d5l3B6SuaX
Para Marco Aurélio Mello, Congresso afrontaria STF se mudar regra de prisão em segunda instância // Noffa. Não pode... https://t.co
/P8GC73RfoY
```

Passo 4: Inspeccionando um tweets json

Em um novo arquivo e execute `public_tweets[0]` para visualizar o json associado ao primeiro tweet, aproveite a saída.

Para visualizar melhor os dados, não adianta apenas formatar “bonitinho” usando `pprint` ou invés de `print`. Nós Precisamos tornar a saída `tweepy` mais legível para que possamos descobrir qual código precisamos escrever para extrair os dados que queremos. Podemos fazer o seguinte:

```
import json

status = public_tweets[0]

#convert to string
json_str = json.dumps(status._json)

#deserialize string into python object
parsed = json.loads(json_str)

print(json.dumps(parsed, indent=4, sort_keys=True))
```

Passo 5: Analisando a saída dos dados

Se você deseja apenas dados primários em tweets (texto, id, user id), o acesso às informações é relativamente fácil e confiável, como já vimos, podemos acessar o texto de um tweet com `status.text` ao usar a API da linha do tempo da página inicial ligar.

Dependendo da natureza do seu projeto e dos dados necessários, você precisará escolher a chamada de API apropriada para coletá-lo. Para trabalhar com dados da chamada `user_timeline`, que (para uma conta padrão) pode recuperar até 3200 tweets mais recentes de um usuário (desde que o perfil não esteja bloqueado). Pode-se fazer da seguinte maneira:

```
#Configura acesso a API
auth = tweepy.OAuthHandler(twitter_keys['consumer_key'], twitter_keys['consumer_secret'])
auth.set_access_token(twitter_keys['access_token_key'], twitter_keys['access_token_secret'])

api = tweepy.API(auth)

#Faz a chamada da sua timeline e imprime cada texto dos tweets
public_tweets = api.home_timeline()
for tweet in public_tweets:
    print(tweet.text)

status = public_tweets[0]
#convert to string
json_str = json.dumps(status._json)

#deserialise string into python object
parsed = json.loads(json_str)

print(json.dumps(parsed, indent=4, sort_keys=True))
```

Para fazer uso da biblioteca Tweepy para os dados públicos, pode-se fazer da seguinte maneira:

```
#Configura acesso a API
auth = tweepy.OAuthHandler(twitter_keys['consumer_key'], twitter_keys['consumer_secret'])
auth.set_access_token(twitter_keys['access_token_key'], twitter_keys['access_token_secret'])

api = tweepy.API(auth)

#Faz a chamada de tweets a partir de uma data e buscando por uma palavra chave
search_words = "#BigData"
date_since = "2019-11-10"
public_tweets = tweepy.Cursor(api.search,
                               q=search_words,
                               lang="pt-br",
                               since=date_since).items(15)

for tweet in public_tweets:
    print(tweet.text)
```