

# Twitter data

## Copyright and Licensing

You are free to use or adapt this notebook for any purpose you'd like. However, please respect the [Simplified BSD License](https://github.com/ptwobrussell/Mining-the-Social-Web-2nd-Edition/blob/master/LICENSE.txt) (<https://github.com/ptwobrussell/Mining-the-Social-Web-2nd-Edition/blob/master/LICENSE.txt>) that governs its use.

## Twitter API Access

Twitter implements OAuth 1.0A as its standard authentication mechanism, and in order to use it to make requests to Twitter's API, you'll need to go to <https://dev.twitter.com/apps> (<https://dev.twitter.com/apps>) and create a sample application.

Choose any name for your application, write a description and use `http://google.com` for the website.

Under **Key and Access Tokens**, there are four primary identifiers you'll need to note for an OAuth 1.0A workflow:

- consumer key,
- consumer secret,
- access token, and
- access token secret (Click on Create Access Token to create those).

Note that you will need an ordinary Twitter account in order to login, create an app, and get these credentials.

The first time you execute the notebook, add all credentials so that you can save them in the `pk1` file, then you can remove the secret keys from the notebook because they will just be loaded from the `pk1` file.

The `pk1` file contains sensitive information that can be used to take control of your twitter account, **do not share it**.

In [1]:

```
import pickle
import os
```

In [2]:

```
if not os.path.exists('secret_twitter_credentials.pk1'):
    Twitter={}
    Twitter['Consumer Key'] = 'kqLnYdYhzGhkUYk7FkDOcSNAj'
    Twitter['Consumer Secret'] = 'KI0CzlOzarCKF2RMq5Pd3L3JvLJsLlnFofwVawF2kUroeBrPFI'
    Twitter['Access Token'] = '1067767943944036353-oszJolzqwKniO2haVACrTsoKOOjiZR'
    Twitter['Access Token Secret'] = 'lsF2AwR6Ez3vVFDXSoKXyrm0GgcSIFkqwmkn0Ju5zL2gT'
    with open('secret_twitter_credentials.pk1', 'wb') as f:
        pickle.dump(Twitter, f)
else:
    Twitter=pickle.load(open('secret_twitter_credentials.pk1', 'rb'))
```

Install the `twitter` package to interface with the Twitter API

In [3]:

```
!pip install twitter
```

Requirement already satisfied: twitter in /Users/johnny/anaconda3/lib/python3.6/site-packages (1.18.0)

## Example 1. Authorizing an application to access Twitter account data

In [4]:

```
import twitter

auth = twitter.oauth.OAuth(Twitter['Access Token'],
                             Twitter['Access Token Secret'],
                             Twitter['Consumer Key'],
                             Twitter['Consumer Secret'])

twitter_api = twitter.Twitter(auth=auth)

# Nothing to see by displaying twitter_api except that it's now a
# defined variable

print(twitter_api)
```

<twitter.api.Twitter object at 0x1112177b8>

## Example 2. Retrieving trends

Twitter identifies locations using the Yahoo! Where On Earth ID.

The Yahoo! Where On Earth ID for the entire world is 1. See

<https://dev.twitter.com/docs/api/1.1/get/trends/place> (<https://dev.twitter.com/docs/api/1.1/get/trends/place>)  
and <http://developer.yahoo.com/geo/geoplanet/> (<http://developer.yahoo.com/geo/geoplanet/>)

look at the BOSS placefinder here: <https://developer.yahoo.com/boss/placefinder/>  
(<https://developer.yahoo.com/boss/placefinder/>)

In [5]:

```
WORLD_WOE_ID = 1
US_WOE_ID = 23424977
```

Look for the WOEID for [Los Angeles](http://woeid.rosselliot.co.nz/lookup/los%20angeles) (<http://woeid.rosselliot.co.nz/lookup/los%20angeles>)

You can change it to another location.

In [6]:

```
LOCAL_WOE_ID= 2442047

# Prefix ID with the underscore for query string parameterization.
# Without the underscore, the twitter package appends the ID value
# to the URL itself as a special case keyword argument.

world_trends = twitter_api.trends.place(_id=WORLD_WOE_ID)
us_trends = twitter_api.trends.place(_id=US_WOE_ID)
local_trends = twitter_api.trends.place(_id=LOCAL_WOE_ID)
```

In [7]:

```
world_trends[:2]
```

Out[7]:

```
[{'as_of': '2018-12-01T00:27:54Z',
  'created_at': '2018-12-01T00:21:44Z',
  'locations': [{'name': 'Worldwide', 'woeid': 1}],
  'trends': [{'name': '#ThankUNextVideo',
    'promoted_content': None,
    'query': '%23ThankUNextVideo',
    'tweet_volume': 544478,
    'url': 'http://twitter.com/search?q=%23ThankUNextVideo'},
    {'name': 'Colón',
    'promoted_content': None,
    'query': 'Col%C3%B3n',
    'tweet_volume': 58930,
    'url': 'http://twitter.com/search?q=Col%C3%B3n'},
    {'name': '#LateLateToyShow',
    'promoted_content': None,
    'query': '%23LateLateToyShow',
    'tweet_volume': 51511,
    'url': 'http://twitter.com/search?q=%23LateLateToyShow'}],
  'name': 'Colón',
  'promoted_content': None,
  'query': 'Col%C3%B3n',
  'tweet_volume': 58930,
  'url': 'http://twitter.com/search?q=Col%C3%B3n'},
  {'name': '#LateLateToyShow',
  'promoted_content': None,
  'query': '%23LateLateToyShow',
  'tweet_volume': 51511,
  'url': 'http://twitter.com/search?q=%23LateLateToyShow'}]
```

In [8]:

```
trends=local_trends
print(type(trends))
print(list(trends[0].keys()))
print(trends[0]['trends'])
```

```
<class 'twitter.api.TwitterListResponse'>
['trends', 'as_of', 'created_at', 'locations']
[{'name': '#ThankUNextVideo', 'url': 'http://twitter.com/search?q=%23ThankUNextVideo', 'promoted_content': None, 'query': '%23ThankUNextVideo', 'tweet_volume': 544478}, {'name': 'Ariana Grande', 'url': 'http://twitter.com/search?q=%22Ariana+Grande%22', 'promoted_content': None, 'query': '%22Ariana+Grande%22', 'tweet_volume': 239785}, {'name': 'Kareem Hunt', 'url': 'http://twitter.com/search?q=%22Kareem+Hunt%22', 'promoted_content': None, 'query': '%22Kareem+Hunt%22', 'tweet_volume': 66940}, {'name': 'Nick Nolte', 'url': 'http://twitter.com/search?q=%22Nick+Nolte%22', 'promoted_content': None, 'query': '%22Nick+Nolte%22', 'tweet_volume': None}, {'name': '#AskMeek', 'url': 'http://twitter.com/search?q=%23AskMeek', 'promoted_content': None, 'query': '%23AskMeek', 'tweet_volume': None}, {'name': 'Anchorage', 'url': 'http://twitter.com/search?q=Anchorage', 'promoted_content': None, 'query': 'Anchorage', 'tweet_volume': 155357}, {'name': 'Yan Gomes', 'url': 'http://twitter.com/search?q=%22Yan+Gomes%22', 'promoted_content': None, 'query': '%22Yan+Gomes%22', 'tweet_volume': None}, {'name': '#AlaskaEarthquake', 'url': 'http://twitter.com/search?q=%23AlaskaEarthquake', 'promoted_content': None, 'query': '%23AlaskaEarthquake', 'tweet_volume': None}]
```

### Example 3. Displaying API responses as pretty-printed JSON

In [9]:

```
import json

print((json.dumps(us_trends[:2], indent=1)))
```

```
[
  {
    "trends": [
      {
        "name": "#ThankUNextVideo",
        "url": "http://twitter.com/search?q=%23ThankUNextVideo",
        "promoted_content": null,
        "query": "%23ThankUNextVideo",
        "tweet_volume": 544478
      },
      {
        "name": "Kareem Hunt",
        "url": "http://twitter.com/search?q=%22Kareem+Hunt%22",
        "promoted_content": null,
        "query": "%22Kareem+Hunt%22",
        "tweet_volume": 66940
      },
      {
        "name": "#AskMeek",
        "url": "http://twitter.com/search?q=%23AskMeek",
        "promoted_content": null,
        "query": "%23AskMeek",
        "tweet_volume": None
      }
    ]
  }
]
```

### Example 4. Computing the intersection of two sets of trends

[illegible]

```
for loc in ['world', 'us', 'los angeles']:
    print((-'*10, loc))
    print((', '.join(trends_set[loc])))
```

( '-----' , 'us' )  
#BalanceTonPost,#ElMandamas,#ChadwickRenuncia,#SagerStrong,#ViernesDeEliminacion,#GrahamNorton,Tim Means,#ThankUNextVideo,#LateLateToyShow,#CARWOL,Wilmer Flores,#SchleFaZ,#HarryPotterEnCuatro,#اوصف\_شعورك\_بكلمه ,  
#UnCuentoInfantilConDuque,#ウルトラマンループ,#propagandalive,#bakeoffitalia,#AMLOFest,Anchorage,#AskMeek,#CrownPrinceInG20,#G20Summit2018,#UltimoViernesDeMesY,#قمه\_لقاء\_محمد\_بن\_سلمان,Kareem Hunt,Amber Guyger,#PagaráPremiumSuMadre,Alaska,Julio Bocca,#MilenioLive,#ادعم\_جروس,Colón,Matías Suárez,#ديسمبر,#EnDiciembreMeEspera,#RememberSKAM,#RomComMyLife,Ya  
No Gomes,Sam Gyimah,#TCMS10,#TheVoiceKids,#平ジエネFOREVER,#ل\_نصحوں\_بالز ,  
Pinilla,#HotWheelsSebas,#scherziaparte,Flavia Palmiero,#As AventurasDePoliana143,#ASSEFCN  
( '-----' , 'us' )  
#KIISJingleBall,Jeff Green,Jada Byers,#SagerStrong,Kesha,Daniel Johnson,#Pac12FCG,#Bellator210,Tim Means,#ThankUNextVideo,#WhenISeeTheFirstSnowflake,#LateLateToyShow,The RAC,White Sox,#WCollegeCup,Wilmer Flores,throne 2,Ariana Grande,Wichita Falls,Anchorage,#AskMeek,#Dbacks,#AlaskaEarthquake,Kareem Hunt,ICONIC,#ClimateFriday,Amber Guyger,Brenda Snipes,Victor Oladipo,Addison Russell,Brandon Knight,Hollywood Walk of Fame,#BadeFest Golden Nick Brand #Producers Nicholas Deakins #DemGemsVid

In [12]:

```
print(( '='*10,'intersection of world and us'))
print((trends_set['world'].intersection(trends_set['us'])))

print(( '='*10,'intersection of us and san-diego'))
print((trends_set['los angeles'].intersection(trends_set['us'])))

('=====', 'intersection of world and us')
{'#RomComMyLife', 'Amber Guyger', 'Yan Gomes', '#SagerStrong', 'Tim Means', '#ThankUNextVideo', '#LateLateToyShow', 'Anchorage', '#AskMeek', 'Colón', 'Kareem Hunt', 'Wilmer Flores'}
('=====', 'intersection of us and san-diego')
{'#KIISJingleBall', 'Jeff Green', 'Jada Byers', '#SagerStrong', 'Kesh a', 'Daniel Johnson', '#Pac12FCG', '#Bellator210', 'Tim Means', '#ThankUNextVideo', '#WhenISeeTheFirstSnowFlake', '#LateLateToyShow', 'White Sox', '#WCollegeCup', 'throne 2', 'Wilmer Flores', 'Ariana Grande', 'W ichita Falls', 'Anchorage', '#AskMeek', '#Dbacks', '#AlaskaEarthquake', 'Kareem Hunt', 'ICONIC', '#ClimateFriday', 'Amber Guyger', 'Brenda Snipes', 'Victor Oladipo', 'Addison Russell', 'Brandon Knight', 'Hollywood Walk of Fame', '#Redsfest', 'wayne estopinal', 'Colón', '#MACTION', 'Nicklas Backstrom', '#RomComMyLife', 'Spencer Ware', 'Yan Gomes', 'PJ Dozier', '#CapsDevils', 'Jonathan Schoop', 'Former Dallas', '#StrangeJobPostings', 'Tornado Watch', 'The Indians', 'The RAC', 'Nanah Wai ya', 'Shelby Miller'}
```

## Example 5. Collecting search results

Set the variable `q` to a trending topic, or anything else for that matter. The example query below was a trending topic when this content was being developed and is used throughout the remainder of this chapter

In [13]:

```
q = '#FelizMiércoles'

number = 100

# See https://dev.twitter.com/docs/api/1.1/get/search/tweets

search_results = twitter_api.search.tweets(q=q, count=number)

statuses = search_results['statuses']
```

In [14]:

```
len(statuses)
#print(statuses)
```

Out[14]:

56

Twitter often returns duplicate results, we can filter them out checking for duplicate texts:

In [15]:

```
all_text = []
filtered_statuses = []
for s in statuses:
    if not s["text"] in all_text:
        filtered_statuses.append(s)
        all_text.append(s["text"])
statuses = filtered_statuses
```

In [16]:

```
len(statuses)
```

Out[16]:

51

In [17]:

```
[s['text'] for s in search_results['statuses']]
```

Out[17]:

```
[ 'RT @sumalen59: Buenos días a tod@s! #FelizMiércoles republicano!Prep
arándonos para tuitear con el HT de @MiercolesRepubl Si tienes ganas d
e...',
'RT @j davidcabello: #FelizMiércoles hoy continuamos trabajando por nu
estra Patria, la de Bolívar y Chávez, hombres que lucharon contra opre
s...',
'RT @PeterPeterbox: «El ciclismo es para todos». Bernard Hinault \n\n
#VivaLaBici #bicicleta #labicimola #FelizMiércoles\n¿Vamos a coger
las...',
'RT @zaidacastillo51: 🐘🌺🌺🌺🌺🌺\n¡Buenos días mundo twittero!!
\n\n"Que tu meta de hoy sea... \nganarle la batalla a tu mejor excus
a..." \n\n#Elefant...',
'RT @Mayesocialista: #FelizMiércoles Patria amada, afinando los últim
os detalles para iniciar en unos minutos el ciclo de ponencias del Con
g...',
'RT @nisiu07: Uno pertenece a donde\nSe le quiere,se le desea,se le a
ma\nY en repetidas ocasiones\n" se le extraña "\nPero sobre todo\nA do
nde se...',
'RT @LosChamosMx: #FelizMiércoles ya probaste nuestro pasticho? http
s://t.co/nYlrNhQgYt', (https://t.co/nYlrNhQgYt',)
'RT @LeyreArgenta: (\\__/\n(=\'.\'=)\nE[:]|||[:]3\n(")_(" \n Bue
nos días mundo 🌲 \n\n#FelizDia\n#FelizMiercoles',
'RT @laspancheritas: ¡Hola a tod@s! Estamos felices porque Pancho ya
está en su natal Argentina 🇦🇷 donde está rodeado de mucho amor y cari
ño...',
'RT @MariaIranzoBio: La spirulina puede atacar a las células tumorale
s espontáneamente y sin dañar a las sanas... ¡Una nueva herramienta nano
t...',
'RT @SoyardEcu: Es hora del #RetoSoyard de esta semana! 🤔💡\nRecuerd
a que Soyard es rico en omega 3, 6 y 9 lo cual te ayuda a fortalecer t
u sa...',
'RT @fodenorca: #Ahora 🗣️| Se desarrolla la creación del CT49 Cuero,
calzados y afines para normalizar este rubro y velar por la calidad q
ue...',
'RT @Mippcivzla: #FelizMiércoles Buenos días te presentamos las porta
das de los diarios📖 que circulan la mañana de hoy #28Nov. https://t.c
o/...', (https://t.co/...',)
'RT @Reportes360: #TránsitoCCS ¡FELIZ MIÉRCOLES! Iniciamos desde la V
alle-Coche donde hay libre desplazamiento para conectar con la #AFF. S
e...',
'RT @pazinteriorv: NAMASTÉ🙏🌱\nFeliz Miércoles! 💖\nHappy Wednesday!
\n #love #Paz #FelizMiércoles #meditation #frases #Mindfulness #Wednes
dayWisd...',
'RT @Region3Policia: ¡Nuestra actitud de servicio es reconocida por l
a ciudadanía! \nhttps://t.co/LefZmEicV8 \n@PoliciaColombia @PoliciaPer
eir...',
'Parece mentira que con la app gratuita de #Kindle puedas leer tantas
novelas y relatos eróticos en tu móvil. En... https://t.co/xw6MlACe0D',
(https://t.co/xw6MlACe0D',)
'RT @vmm7773: Dice Casado que en España "no hay sitio para el que vie
ne a delinquir"\n\nEso significa ... ¿Que van a ser expulsados de nues
tro...',
'RT @abelhuitron: Marketing digital, educación, seguridad en internet
y liderazgo femenino, son algunos de los temas que se están abordand
o...',
```



'RT @EmergenciasMad: En una situación de #Emergencia hay que guardar la calma y ayudar a quienes lo tienen más difícil. No cuesta nada apr en...',

'RT @DMZenvios: En @DMZcomputer les deseamos un muy #felizmiercoles h <https://t.co/dRpnpcNTEJ>', (<https://t.co/dRpnpcNTEJ>'),

'RT @DMZenvios: En @DMZcomputer les deseamos un muy #felizmiercoles h <https://t.co/H55pLVC5Zd>', (<https://t.co/H55pLVC5Zd>'),

'RT @PrimosMX: Hay que dejar ir y dejarse llevar a veces para vivir y disfrutar 🥰🥰🥰🥰 !!\n#PurosPrimos #PrimosMx #LosPrimosMx20Aniver sario \n🎵...',

'RT @RPuntoSobreLai: #FelizMiércoles @juanantonio6289 habla sobre el amor y las relaciones en pareja en su entrevista en #Teleformula y en s...',

'RT @HdAnchiano: Artesanía con forma de moneda y precisión de relojer o. #FelizMiércoles <https://t.co/Cvr0vnnSO>r', (<https://t.co/Cvr0vnnSO>r'),

'RT @mymarbellaochoa: La paz empieza en nuestros corazones y el amor a lo niños es reflejo de esta paz #RETOFEPaZ #ViralizaLaPaz #RETOFEPa Z...',

'RT @ABC760AM: ¿Sera tu día de suerte? 🧐 Descúbrelo leyendo tu #Horos copoDeHoy 🎰🎰🎰\n\n➡ <https://t.co/koBODDz68n>\n\n#FelizMiercoles (htt ps://t.co/koBODDz68n\n\n#FelizMiercoles) <https://t.co/f...>', (<https://t.co/f...>'),

'RT @ABC760AM: 📰 #BuenosDías, hoy #EnPortada de @laprensaom \n ¡CHAPO FIFÍ!\n📰 Lee La Prensa en Línea 📰\n<https://t.co/IUHq61Yx...>',

'RT @jdavidcabello: #FelizMiércoles hoy continuamos trabajando por nu estra Patria, la de Bolívar y Chávez, hombres que lucharon contra opre s...',

'RT @FuerzaAereaCol: #FelizMiércoles ➡ Por surcar a diario los cielos colombianos, somos testigos de un amanecer libre y un anocheecer sober a...',

'RT @premiumflorist: ¡Encontraron agua en Marte! 🤖 ¿Y sabes qué? 🤖 ¡Contigo voy a Marte y a cualquier otra parte! 🤖👩🏻👨🏻👧🏻👦🏻\n\n#FelizMierco les',

'RT @LeyreArgenta: (\\\_\_/\n(=\'.\'=)\nE[:]|||[:]3\n(")\_(" \n Bue nos días mundo 🌲\n\n#FelizDia\n#FelizMiercoles',

'RT @premiumflorist: ¡Truco o trato! 🍷 Que esta noche funcionen todo s tus hechizos de amor y felicidad 💖 ¡Feliz Halloween! 🍷\n\n#HappyHa llowee...',

'RT @lverboencarnado: No hay #PretextosParaNoIr a rezar el #SantoRosa rio este #FelizMiércoles 28 noviembre 2018 <https://t.co/wZ8008KIYS>', (<https://t.co/wZ8008KIYS>'),

'RT @SevillaEsconde: En el centro de la Plaza Nueva se levanta desde 1924 el monumento al rey Fernando III, conocido como El Santo. Fue el c...',

'RT @lverboencarnado: No hay #PretextosParaNoIr y participar de la #S antaMisa este #FelizMiércoles 28 noviembre 2018 <https://t.co/VfDimJrG0M>', (<https://t.co/VfDimJrG0M>'),

"RT @FMacias\_: ¡No queda nada para '1016'! Felicidades por este sueño que ya es realidad y del que nos has hecho partícipe desde el inicio. ...",

'RT @lverboencarnado: No hay #PretextosParaNoIr a rezar el #SantoRosa rio este #FelizMiércoles 28 noviembre 2018 <https://t.co/oYXOUjHm65>', (<https://t.co/oYXOUjHm65>'),

'RT @lverboencarnado: No hay #PretextosParaNoIr a rezar el #SantoRosa rio este #FelizMiércoles 28 noviembre 2018 <https://t.co/wZ8008KIYS>', (<https://t.co/wZ8008KIYS>'),

'RT @lverboencarnado: No hay #PretextosParaNoIr a rezar el #SantoRosa rio este #FelizMiércoles 28 noviembre 2018 <https://t.co/3e7G466vTJ>', (<https://t.co/3e7G466vTJ>'),

http://localhost:8888/notebooks/UC%20San%20Diego/Week%208%20-%20Working%20with%20Text%20and%20Databases/Week-8-NLP-Databases/Using... 10/18

```
# Show one sample search result by slicing the list...
print(json.dumps(statuses[0], indent=1))
```

In [19]:

```
# The result of the list comprehension is a list with only one element that
# can be accessed by its index and set to the variable t
t = statuses[0]
#[ status for status in statuses
#     if status['id'] == 316948241264549888 ][0]

# Explore the variable t to get familiarized with the data structure...

print(t['retweet_count'])
print(t['retweeted'])
```

90  
False

### Example 6. Extracting text, screen names, and hashtags from tweets

In [20]:

```
status_texts = [ status['text']
                  for status in statuses ]

screen_names = [ user_mention['screen_name']
                  for status in statuses
                  for user_mention in status['entities']['user_mentions'] ]

hashtags = [ hashtag['text']
              for status in statuses
              for hashtag in status['entities']['hashtags'] ]

# Compute a collection of all words from all tweets
words = [ w
          for t in status_texts
          for w in t.split() ]
```

In [21]:

```
# Explore the first 5 items for each...
```

```
print(json.dumps(status_texts[0:5], indent=1))
print(json.dumps(screen_names[0:5], indent=1))
print(json.dumps(hashtags[0:5], indent=1))
print(json.dumps(words[0:5], indent=1))
```

```
[
  "RT @sumalen59: Buenos d\u00edas a tod@s! #FelizMi\u00e9rcoles republ  

  icano!Prepar\u00e9ndonos para tuitear con el HT de @MiercolesRepubl Si  

  tienes ganas de\u2026",
  "RT @jdavidcabello: #FelizMi\u00e9rcoles hoy continuamos trabajando p  

  or nuestra Patria, la de Bol\u00edvar y Ch\u00e9vez, hombres que lucha  

  ron contra opres\u2026",
  "RT @PeterPeterbox: \u00abEl ciclismo es para todos\u00bb. Bernard Hi  

  nault \n\n#VivaLaBici #bicicleta #labicimola #FelizMi\u00e9rcoles\n  

  \u00bfVamos a coger las\u2026",
  "RT @zaidacastillo51: \ud83d\udc18\ud83c\udf3e\ud83c\udf37\ud83c\udf4  

  0 \ud83c\udf3a\ud83c\udf40\ud83d\udc18\n\u00a1Buenos d\u00edas mundo t  

  wittero!!\n\n\"Que tu meta de hoy sea... \nganarle la batalla a tu mej  

  or excusa...\n\n#Elefant\u2026",
  "RT @Mayesocialista: #FelizMi\u00e9rcoles Patria amada, afinando los  

  \u00faltimos detalles para iniciar en unos minutos el ciclo de ponenci  

  as del Cong\u2026"
]
[
  "sumalen59",
  "MiercolesRepubl",
  "jdavidcabello",
  "PeterPeterbox",
  "zaidacastillo51"
]
[
  "FelizMi\u00e9rcoles",
  "FelizMi\u00e9rcoles",
  "VivaLaBici",
  "bicicleta",
  "labicimola"
]
[
  "RT",
  "@sumalen59:",
  "Buenos",
  "d\u00edas",
  "a"
]
```

## Example 7. Creating a basic frequency distribution from the words in tweets

In [22]:

```

from collections import Counter

for item in [words, screen_names, hashtags]:
    c = Counter(item)
    print(c.most_common()[:10]) # top 10
    print()

[('RT', 49), ('de', 26), ('a', 24), ('y', 21), ('#FelizMiércoles', 20), ('el', 17), ('en', 15), ('la', 14), ('que', 13), ('hay', 11)]

[('lverboencarnado', 7), ('DMZenvios', 2), ('DMZcomputer', 2), ('ABC760AM', 2), ('premiumflorist', 2), ('s_um_mer0', 2), ('1bloggerist', 2), ('jucilnacional', 2), ('sumalen59', 1), ('MiercolesRepubl', 1)]

[('FelizMiércoles', 20), ('PretextosParaNoIr', 9), ('SantoRosario', 7), ('FelizMiercoles', 4), ('felizmiercoles', 2), ('RETOFEPAZ', 2), ('HappyWednesday', 2), ('WorldsMostBeautifulBride', 2), ('ShahOnAajTa', 2), ('VivaLaBici', 1)]

```

## Example 8. Create a prettyprint function to display tuples in a nice tabular format

In [23]:

```

def prettyprint_counts(label, list_of_tuples):
    print("\n{: ^20} | {: ^6}".format(label, "Count"))
    print("*"*40)
    for k,v in list_of_tuples:
        print("{:20} | {:>6}".format(k,v))

```

In [24]:

```

for label, data in (('Word', words),
                    ('Screen Name', screen_names),
                    ('Hashtag', hashtags)):

    c = Counter(data)
    prettyprint_counts(label, c.most_common()[:10])

```

Word	Count
RT	49
de	26
a	24
y	21
#FelizMiércoles	20
el	17
en	15
la	14
que	13
hay	11

Screen Name	Count
lverboencarnado	7
DMZenvios	2
DMZcomputer	2
ABC760AM	2
premiumflorist	2
s_um_mer0	2
lbloggerist	2
jucilnacional	2
sumalen59	1
MiercolesRepubl	1

Hashtag	Count
FelizMiércoles	20
PretextosParaNoIr	9
SantoRosario	7
FelizMiercoles	4
felizmiercoles	2
RETOFEPAZ	2
HappyWednesday	2
WorldsMostBeautifulBride	2
ShahOnAajTak	2
VivaLaBici	1

## Example 9. Finding the most popular retweets

In [25]:

```
retweets = [
    # Store out a tuple of these three values ...
    (status['retweet_count'],
     status['retweeted_status']['user']['screen_name'],
     status['text'].replace("\n", "\\n"))

    # ... for each status ...
    for status in statuses

    # ... so long as the status meets this condition.
    if 'retweeted_status' in status
]
```

We can build another `prettyprint` function to print entire tweets with their retweet count.

We also want to split the text of the tweet in up to 3 lines, if needed.

In [26]:

```
row_template = "{:^7} | {:^15} | {:50}"
def prettyprint_tweets(list_of_tuples):
    print()
    print(row_template.format("Count", "Screen Name", "Text"))
    print("*"*80)
    for count, screen_name, text in list_of_tuples:
        print(row_template.format(count, screen_name, text[:50]))
        if len(text) > 50:
            print(row_template.format("", "", text[50:100]))
            if len(text) > 100:
                print(row_template.format("", "", text[100:]))
```



In [27]:

```
# Slice off the first 5 from the sorted results and display each item in the tuple
prettyprint_tweets(sorted(retweets, reverse=True)[:10])
```

Count	Screen Name	Text
2147	vmm7773	RT @vmm7773: Dice Casado que en España "no hay sitio para el que viene a delinquir"\Eso significa . .. ¿Que van a ser expulsados de nuestro...
1838	HdAnchiano	RT @HdAnchiano: Artesanía con forma de moneda y precisión de relojero. #FelizMiércoles <a href="https://t.co/Cvr0vnnSOr">https://t.co/</a> (https://t.co/)
699	s_um_mer0	RT @s_um_mer0:  \#Happy Wednesday #FelizMiércoles  ***** <a href="https://t.co/VyXlIHeNkr">https://t.co/VyXlIHeNkr</a> (https://t.co/VyXlIHeNkr)
379	jdavidcabello	RT @jdavidcabello: #FelizMiércoles hoy con tinuamos trabajando por nuestra Patria, la de Bolívar y Chávez, hombres que lucharon contra opres...
320	HectorSulaimanS	RT @HectorSulaimanS: "No es héroe quien carece de frenta y miedos. Un héroe es quien lo siente, lo enfrenta y lo supera."...
259	VOX_Jaen	RT @VOX_Jaen: Así ha aparecido la puerta de nuestra sede en #JAEN esta mañana.\No se enteran. Cosas como éstas nos dan más fuerzas para lle...
253	FMacias_	RT @FMacias_: ¡No queda nada para '1016'! des por este sueño que ya es realidad y de s has hecho partícipe desde el inicio....
127	EmergenciasMad	RT @EmergenciasMad: En una situación de #Emergencia hay que guardar la calma y ayudar a quienes lo tienen más difícil. No cuesta nada aprende...
120	s_um_mer0	RT @s_um_mer0:  \#quot

```
| e Hay palabras que de tanto repetirlas pie  
rden el |  
| sentido. #HappyWednesday #FelizMiercole...  
104 | Mippcivzla | RT @Mippcivzla: #FelizMiércoles Buenos día  
s te pre |  
| sentamos las portadas de los diarios 📰 que  
circulan |  
| la mañana de hoy #28Nov. https://t.co/...  
(https://t.co/...)
```

In [ ]: