

1. Importing Required Dependencies

```
In [0]: #importing tweepy for twitter api
import tweepy as tw
#importing pandas for data manipulation
import pandas as pd
#importing files from google collaboratory to download csv
from google.colab import files
```

2. Twitter User Secret Credentials

```
In [0]: consumer_key = 'LViaFgMpZcbVXnYAYNH1GsBgZ'
consumer_secret = '6vdzHqULYgD7j2tqrftBSE8PswZdEqTeS3jhzB42Zkzw1g3A1L'
access_token = '1142044773240455168-RwNN1B4KVI97m9gJTF1A6c1mTmOGmI'
access_token_secret = 'zS1gvJWXLBFJEKJsKx6p6yOY9abEugiNJ5NZEEpyGjmV'
```

3. Twiteer Credentials Authentication

```
In [0]: auth = tw.OAuthHandler(consumer_key, consumer_secret)
auth.set_access_token(access_token, access_token_secret)
api = tw.API(auth, wait_on_rate_limit=True)
```

4. Bitcoin Tweets Mining

```
In [0]: # tweets by hashtags to be mined
search_words = '#bitcoin' and '#btc'

# tweets from date
date_since = "2019-07-01"

# filtering to exclude retweets
new_search = search_words + " -filter:retweets"

#cursor parameter with each of our requests
btc = tw.Cursor(api.search, q=new_search, lang="en", since=date_since).items(3000)

# extracting only user name, tweet, location and date
btc = [[tweet.user.screen_name, tweet.text, tweet.user.location, tweet.created_at] for tweet in btc]

#converting the data extracted into pandas dataframe
btc = pd.DataFrame(data=btc, columns=['user', 'text', 'location', 'date'])

#shape of dataframe
print(df.shape)

#first 5 rows of the dataframe
print(btc.head())

#dataframe to csv
btc.to_csv('bitcoin_tweets.csv', sep='\t', encoding='utf-8')

#download csv file
files.download('bitcoin_tweets.csv')
```

5. Ethereum Tweets Mining

```
In [0]: # tweets by hashtags to be mined
search_words = '#ethereum' and '#eth'

# tweets from date
date_since = "2019-07-01"

# filtering to exclude retweets
new_search = search_words + " -filter:retweets"

#cursor parameter with each of our requests
eth = tw.Cursor(api.search, q=new_search, lang="en", since=date_since).items(3000)

# extracting only user name, tweet, location and date
eth = [[tweet.user.screen_name, tweet.text, tweet.user.location, tweet.created_at] for tweet in eth]
```

```
In [106]: #converting the data extracted into pandas dataframe
eth = pd.DataFrame(data=eth, columns=['user', 'text', 'location', 'date'])

#shape of dataframe
print(eth.shape)

#first 5 rows of the dataframe
print(eth.head())

#dataframe to csv
eth.to_csv('ethereum_tweets.csv', sep='\t', encoding='utf-8')

#download csv file
files.download('ethereum_tweets.csv')
```

```
(1602, 4)
      user  ...      date
0  HeartlanderPD  ...  2019-07-18 23:44:14
1  KrakenPrices  ...  2019-07-18 23:44:06
2  ChaEYou53800684  ...  2019-07-18 23:43:47
3  ChaEYou53800684  ...  2019-07-18 23:43:21
4      ABreddy18  ...  2019-07-18 23:43:18
```

```
[5 rows x 4 columns]
```

6. Ripple Tweets Mining

```
In [0]: # tweets by hashtags to be mined
search_words = '#ripple' and '#xrp'

# tweets from date
date_since = "2019-07-01"

# filtering to exclude retweets
new_search = search_words + " -filter:retweets"

#cursor parameter with each of our requests
xrp = tw.Cursor(api.search, q=new_search, lang="en", since=date_since).items(3000)

# extracting only user name, tweet, location and date
xrp = [[tweet.user.screen_name, tweet.text, tweet.user.location, tweet.created_at] for tweet in xrp]
```

```
In [117]: #converting the data extracted into pandas dataframe
xrp = pd.DataFrame(data=xrp, columns=['user', 'text', 'location', 'date'])

#shape of dataframe
print(xrp.shape)

#first 5 rows of the dataframe
print(xrp.head())

#dataframe to csv
xrp.to_csv('ripple_tweets.csv', sep='\t', encoding='utf-8')

#download csv file
files.download('ripple_tweets.csv')
```

```
(2000, 4)
      user  ...      date
0  CashlessSocys  ...  2019-07-19 00:59:49
1    Rob92608209  ...  2019-07-19 00:59:42
2    TickerTop  ...  2019-07-19 00:59:25
3  jasonclarktwit  ...  2019-07-19 00:59:21
4  jasonclarktwit  ...  2019-07-19 00:59:17

[5 rows x 4 columns]
```

7. Litecoin Tweets Mining

```
In [0]: # tweets by hashtags to be mined
search_words = '#litecoin' and '#ltc'

# tweets from date
date_since = "2019-07-01"

# filtering to exclude retweets
new_search = search_words + " -filter:retweets"

#cursor parameter with each of our requests
ltc = tw.Cursor(api.search, q=new_search, lang="en", since=date_since).items(3000)

# extracting only user name, tweet, Location and date
ltc = [[tweet.user.screen_name, tweet.text, tweet.user.location, tweet.created_at] for tweet i
n ltc]
```

```
In [120]: #converting the data extracted into pandas dataframe
ltc = pd.DataFrame(data=ltc, columns=['user', 'text', 'location', 'date'])

#shape of dataframe
print(ltc.shape)

#first 5 rows of the dataframe
print(ltc.head())

#dataframe to csv
ltc.to_csv('litecoin_tweets.csv', sep='\t', encoding='utf-8')

#download csv file
files.download('litecoin_tweets.csv')
```

```
(2245, 4)
      user  ...      date
0  cryptonewslab  ...  2019-07-19 01:03:16
1    CryptomaQ  ...  2019-07-19 01:02:15
2    CoinCapsAi  ...  2019-07-19 01:00:59
3  litecoin_price_  ...  2019-07-19 01:00:52
4    CoinTigoDev  ...  2019-07-19 01:00:30

[5 rows x 4 columns]
```

8. Bitcoin Cash Data Mining

```
In [0]: # tweets by hashtags to be mined
search_words = '#bitcoincash' and '#bch'

# tweets from date
date_since = "2019-07-01"

# filtering to exclude retweets
new_search = search_words + " -filter:retweets"

#cursor parameter with each of our requests
bch = tw.Cursor(api.search, q=new_search, lang="en", since=date_since).items(3000)

# extracting only user name, tweet, location and date
bch = [[tweet.user.screen_name, tweet.text, tweet.user.location, tweet.created_at] for tweet i
n bch]
```

```
In [7]: #converting the data extracted into pandas dataframe
bch = pd.DataFrame(data=bch, columns=['user', 'text', 'location', 'date'])

#shape of dataframe
print(bch.shape)

#first 5 rows of the dataframe
print(bch.head())

#dataframe to csv
bch.to_csv('bitcoincash_tweets.csv', sep='\t', encoding='utf-8')

#download csv file
files.download('bitcoincash_tweets.csv')
```

```
(3000, 4)
      user  ...      date
0  CashTechNews  ...  2019-07-19 10:12:04
1      bch_price  ...  2019-07-19 10:02:03
2   dailycoininfo  ...  2019-07-19 10:00:43
3      coinstats  ...  2019-07-19 10:00:22
4  webnowcompany  ...  2019-07-19 09:58:13
```

```
[5 rows x 4 columns]
```

9. EOS Tweets Mining

```
In [0]: # tweets by hashtags to be mined
search_words = '#EOS'

# tweets from date
date_since = "2019-07-01"

# filtering to exclude retweets
new_search = search_words + " -filter:retweets"

#cursor parameter with each of our requests
eos = tw.Cursor(api.search, q=new_search, lang="en", since=date_since).items(3000)

# extracting only user name, tweet, location and date
eos = [[tweet.user.screen_name, tweet.text, tweet.user.location, tweet.created_at] for tweet i
n eos]
```

```
In [9]: #converting the data extracted into pandas dataframe
eos = pd.DataFrame(data=eos, columns=['user', 'text', 'location', 'date'])

#shape of dataframe
print(eos.shape)

#first 5 rows of the dataframe
print(eos.head())

#dataframe to csv
eos.to_csv('eos_tweets.csv', sep='\t', encoding='utf-8')

#download csv file
files.download('eos_tweets.csv')
```

```
(3000, 4)
      user  ...      date
0  hodlcl...  ...  2019-07-19 10:31:56
1      IostPlay  ...  2019-07-19 10:31:55
2  hodlcl...  ...  2019-07-19 10:31:39
3  hodlcl...  ...  2019-07-19 10:28:37
4  hodlcl...  ...  2019-07-19 10:28:22

[5 rows x 4 columns]
```

10. Binance Coin Tweets Mining

```
In [0]: # tweets by hashtags to be mined
search_words1 = 'binancecoin' and 'bnb'
search_words2 = '#binancecoin' and '#bnb'
search_words = search_words1 + search_words2

# tweets from date
date_since = "2019-07-01"

# filtering to exclude retweets
new_search = search_words + " -filter:retweets"

#cursor parameter with each of our requests
bnb = tw.Cursor(api.search, q=new_search, lang="en", since=date_since).items(3000)

# extracting only user name, tweet, location and date
bnb = [[tweet.user.screen_name, tweet.text, tweet.user.location, tweet.created_at] for tweet i
n bnb]
```

```
In [87]: #converting the data extracted into pandas dataframe
bnb = pd.DataFrame(data=bnb, columns=['user', 'text', 'location', 'date'])

#shape of dataframe
print(bnb.shape)

#first 5 rows of the dataframe
print(bnb.head())

#dataframe to csv
bnb.to_csv('binance_tweets.csv', sep='\t', encoding='utf-8')

#download csv file
files.download('binance_tweets.csv')
```

```
(2799, 4)

   user  ...      date
0  CryptoCompare  ...  2019-07-19 13:23:49
1  northnorfolknow  ...  2019-07-19 13:15:21
2      Alt_Cash  ...  2019-07-19 13:06:03
3      quadency  ...  2019-07-19 13:00:54
4      coinstats  ...  2019-07-19 13:00:27

[5 rows x 4 columns]
```

11. Tether Tweets Mining

```
In [0]: # tweets by hashtags to be mined
search_words = '#tether' and '#usdt'
#search_words2 = 'tether' and 'usdt'
#search_words = search_words1 + search_words2

# tweets from date
date_since = "2019-07-01"

# filtering to exclude retweets
new_search = search_words + " -filter:retweets"

#cursor parameter with each of our requests
usdt = tw.Cursor(api.search, q=new_search, lang="en", since=date_since).items(3000)

# extracting only user name, tweet, location and date
usdt = [[tweet.user.screen_name, tweet.text, tweet.user.location, tweet.created_at] for tweet
in usdt]
```

```
In [19]: #converting the data extracted into pandas dataframe
usdt = pd.DataFrame(data=usdt, columns=['user', 'text', 'location', 'date'])

#shape of dataframe
print(usdt.shape)

#first 5 rows of the dataframe
print(usdt.head())

#dataframe to csv
usdt.to_csv('tether_tweets.csv', sep='\t', encoding='utf-8')

#download csv file
files.download('tether_tweets.csv')
```

```
(1240, 4)
      user  ...      date
0  xbtusdBITCOIN  ...  2019-07-19 21:17:49
1    GannMarkets  ...  2019-07-19 21:05:07
2    coinstats  ...  2019-07-19 21:00:33
3  CoinoMonitor  ...  2019-07-19 20:33:01
4    KadaneZod  ...  2019-07-19 20:12:14

[5 rows x 4 columns]
```

12. Bitcoin SV Tweets Mining

```
In [0]: # tweets by hashtags to be mined
search_words = '#bitcoinsv' and '#bsv'

# tweets from date
date_since = "2019-07-01"

# filtering to exclude retweets
new_search = search_words + " -filter:retweets"

#cursor parameter with each of our requests
bsv = tw.Cursor(api.search, q=new_search, lang="en", since=date_since).items(3000)

# extracting only user name, tweet, location and date
bsv = [[tweet.user.screen_name, tweet.text, tweet.user.location, tweet.created_at] for tweet i
n bsv]
```

```
In [22]: #converting the data extracted into pandas dataframe
bsv = pd.DataFrame(data=bsv, columns=['user', 'text', 'location', 'date'])

#shape of dataframe
print(bsv.shape)

#first 5 rows of the dataframe
print(bsv.head())

#dataframe to csv
bsv.to_csv('bitcoinsv_tweets.csv', sep='\t', encoding='utf-8')

#download csv file
files.download('bitcoinsv_tweets.csv')
```

```
(1706, 4)
      user  ...      date
0  CoinoMonitor  ...  2019-07-19 22:05:05
1    AutoRass  ...  2019-07-19 21:54:11
2    FnqComp  ...  2019-07-19 21:51:40
3  XCashFoundation  ...  2019-07-19 21:47:18
4    ejfhp  ...  2019-07-19 21:45:13

[5 rows x 4 columns]
```

13. Stellar Tweets Mining

```
In [0]: # tweets by hashtags to be mined
search_words = '#stellar' and '#xlm'

# tweets from date
date_since = "2019-07-01"

# filtering to exclude retweets
new_search = search_words + " -filter:retweets"

#cursor parameter with each of our requests
xlm = tw.Cursor(api.search, q=new_search, lang="en", since=date_since).items(3000)

# extracting only user name, tweet, location and date
xlm = [[tweet.user.screen_name, tweet.text, tweet.user.location, tweet.created_at] for tweet i
n xlm]
```

```
In [25]: #converting the data extracted into pandas dataframe
xlm = pd.DataFrame(data=xlm, columns=['user', 'text', 'location', 'date'])

#shape of dataframe
print(xlm.shape)

#first 5 rows of the dataframe
print(xlm.head())

#dataframe to csv
xlm.to_csv('stellar_tweets.csv', sep='\t', encoding='utf-8')

#download csv file
files.download('stellar_tweets.csv')
```

```
(2328, 4)
      user  ...      date
0  RedditStellar  ...  2019-07-19 22:23:15
1    govip_site  ...  2019-07-19 22:20:52
2  webnowcompany  ...  2019-07-19 22:12:50
3    KoinKnight  ...  2019-07-19 22:01:20
4      xlmprice  ...  2019-07-19 22:00:04
```

```
[5 rows x 4 columns]
```