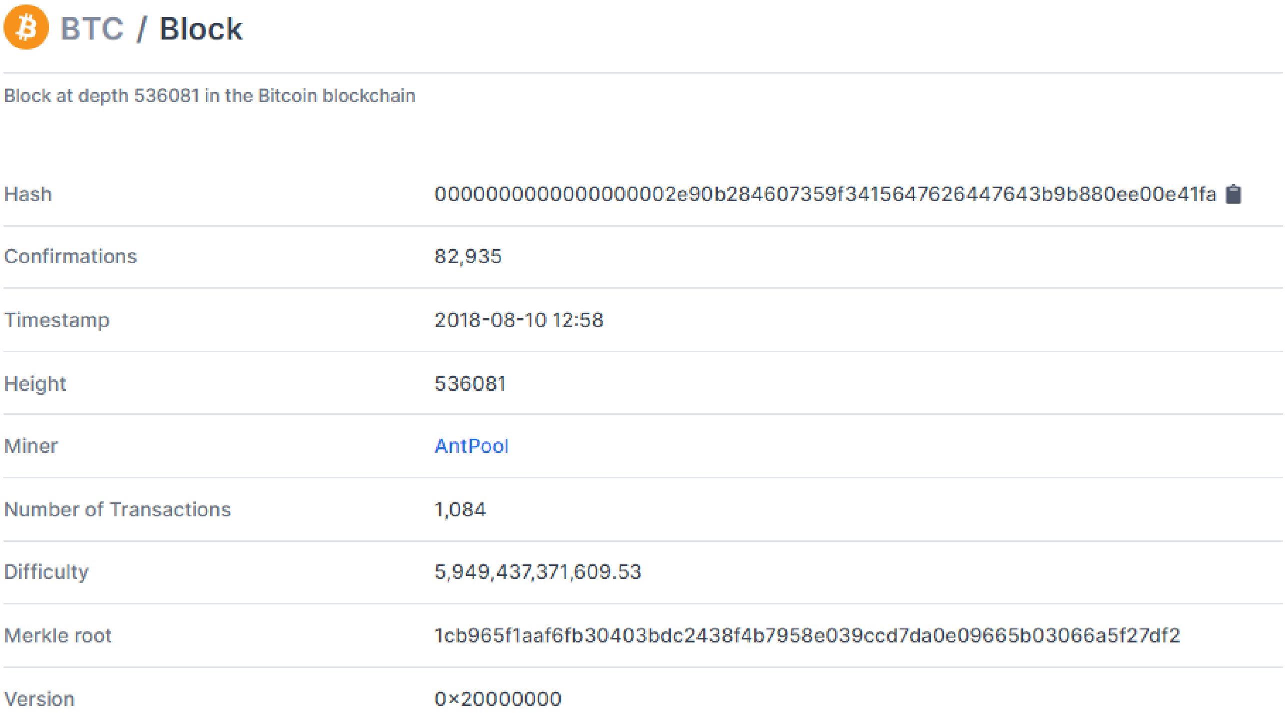
Blockchain Technology and Data Economics

Practise 2 - Blockchain programming with Python

Version: 2020 Spring

Introduction

Blockchain.info is one of the most popular blockchain and bitcoin network explorers and wallet providers. From the web, you can view the block level and see all the transactions that have happened. For example, by going to this particular block—that is, Block #536081—you can see all the transactions, as well as some other information, as shown in the following screenshot:

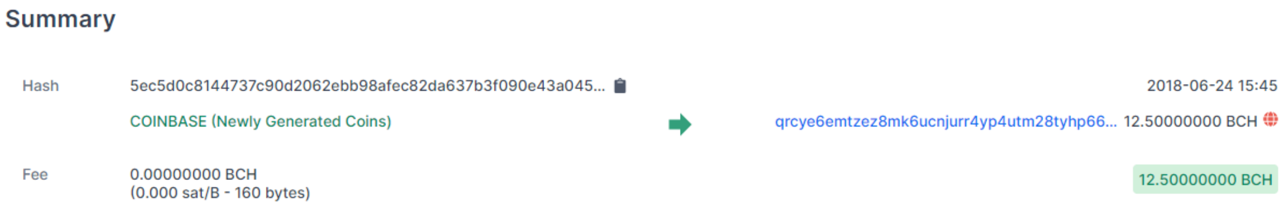


BTC / Block

Block at depth 536081 in the Bitcoin blockchain

Hash	00000000000000000002e90b284607359f3415647626447643b9b880ee00e41fa
Confirmations	82,935
Timestamp	2018-08-10 12:58
Height	536081
Miner	AntPool
Number of Transactions	1,084
Difficulty	5,949,437,371,609.53
Merkle root	1cb965f1aaf6fb30403bdc2438f4b7958e039ccd7da0e09665b03066a5f27df2
Version	0×20000000

The following screenshot shows the statistical data (DATA | Stats). This is great, and useful as well; however, for a developer building applications or performing analysis based on this data, it is important to get this data programmatically:



Summary

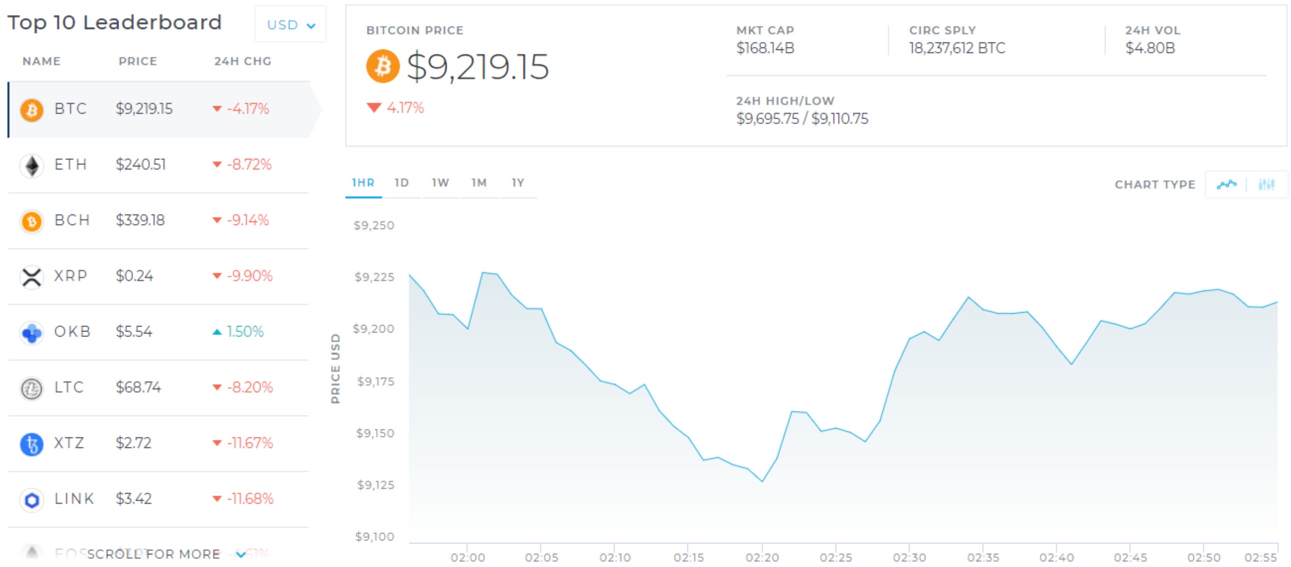
Hash 5ec5d0c8144737c90d2062ebb98afec82da637b3f090e43a045... 2018-06-24 15:45

COINBASE (Newly Generated Coins) ➡ qrcye6emtzez8mk6ucnjurr4yp4utm28tyhp66... 12.50000000 BCH

Fee 0.00000000 BCH
(0.000 sat/B - 160 bytes)

12.50000000 BCH

The following screenshot shows the market data (DATA | Markets):



Top 10 Leaderboard USD

NAME	PRICE	24H CHG
BTC	$9,219.15	▼ -4.17%
ETH	$240.51	▼ -8.72%
BCH	$339.18	▼ -9.14%
XRP	$0.24	▼ -9.90%
OKB	$5.54	▲ 1.50%
LTC	$68.74	▼ -8.20%
XTZ	$2.72	▼ -11.67%
LINK	$3.42	▼ -11.68%

SCROLL FOR MORE

BITCOIN PRICE
$9,219.15
▼ 4.17%

MKT CAP $168.14B
CIRC SPLY 18,237,612 BTC
24H VOL $4.80B
24H HIGH/LOW $9,695.75 / $9,110.75

I. Installing the Blockchain.info Python library

The following are the steps to install the *blockchain* Python library:

1. Open the command-line program on your computer

2. Run the *pip install blockchain* command to install the *blockchain* library.

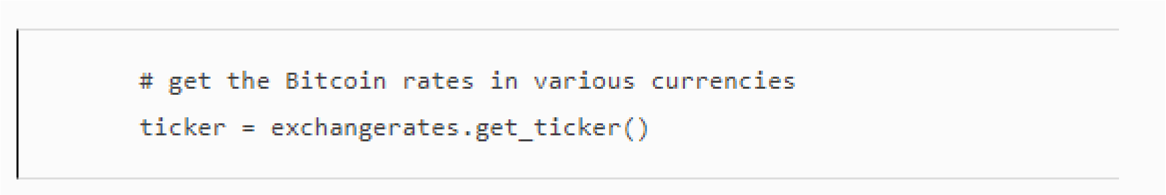
II. Getting bitcoin exchange rates from Blockchain.info

The following steps shows the method for bitcoin exchange rates:

1. First, import the *exchangerates* classes from the *blockchain* library:

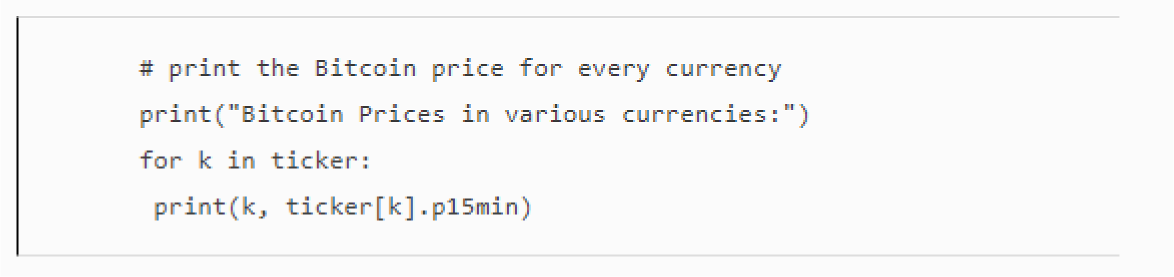
```
#!/usr/bin/env python

# import blockchain library
from blockchain import exchangerates
```

2. Exchange rates define a *get_ticker* method, which returns the exchange rates data in a dictionary object. Call this method and save the resulting object. The *ticker* dictionary object that we have has currency symbols as keys:



```
# get the Bitcoin rates in various currencies
ticker = exchangerates.get_ticker()
```

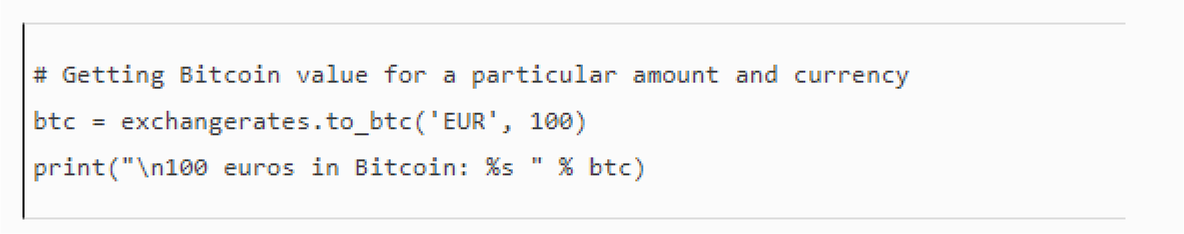
3. By running over these keys, data about the various rates can be pulled. For example, the latest bitcoin rates can be obtained in each currency by getting the *p15min* minimum value:



```
# print the Bitcoin price for every currency
print("Bitcoin Prices in various currencies:")
for k in ticker:
 print(k, ticker[k].p15min)
```

Shows the list of currencies and the equivalent bitcoin rate for those currencies at that moment or from the last 15 minutes to TA.

4. A particular currency can also be converted to bitcoin. For example, you can pass the *to_btc* method and pass in the currency and the amount that we want to convert to *btc*, and get the result as bitcoin. The following code shows how to do this for a value of 100 euros:



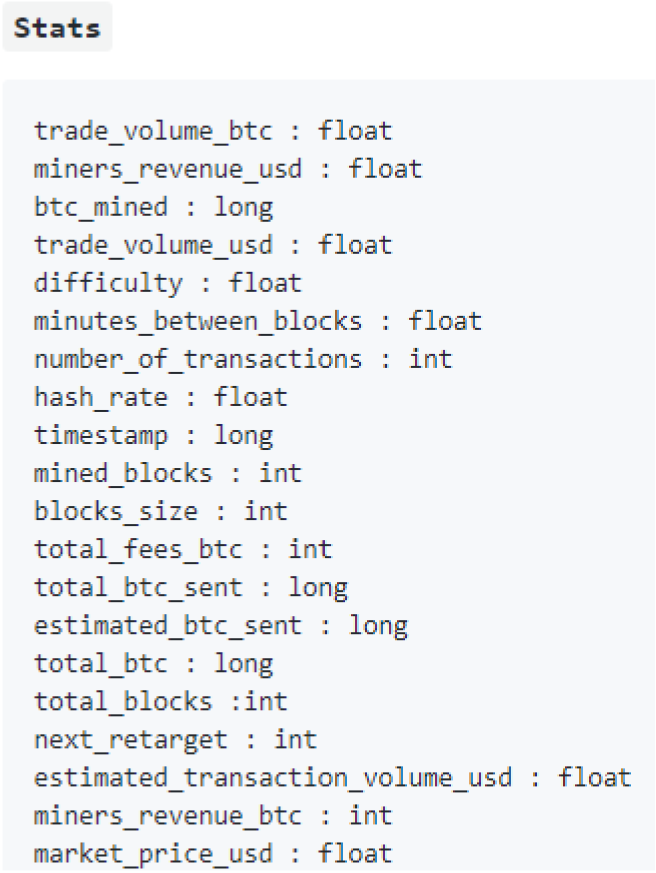
```
# Getting Bitcoin value for a particular amount and currency
btc = exchangerates.to_btc('EUR', 100)
print("\n100 euros in Bitcoin: %s " % btc)
```

Show the output for 100 CNY in bitcoin to TA.

III. Statistics

The next class from the bitcoin blockchain library is called *statistics*.

There are a number of methods that can be called to get a wide variety of blockchain stats data, for example, as shown in the following screenshot:

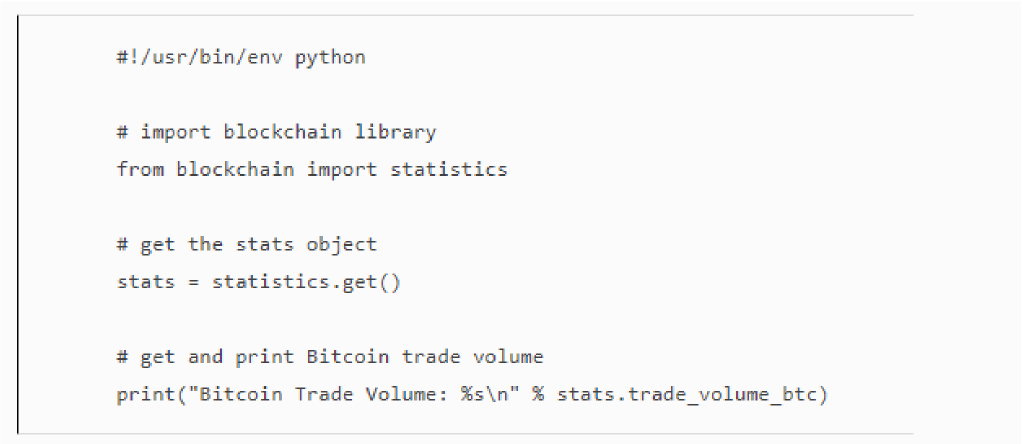


Stats

```
trade_volume_btc : float
miners_revenue_usd : float
btc_mined : long
trade_volume_usd : float
difficulty : float
minutes_between_blocks : float
number_of_transactions : int
hash_rate : float
timestamp : long
mined_blocks : int
blocks_size : int
total_fees_btc : int
total_btc_sent : long
estimated_btc_sent : long
total_btc : long
total_blocks :int
next_retarget : int
estimated_transaction_volume_usd : float
miners_revenue_btc : int
market_price_usd : float
```

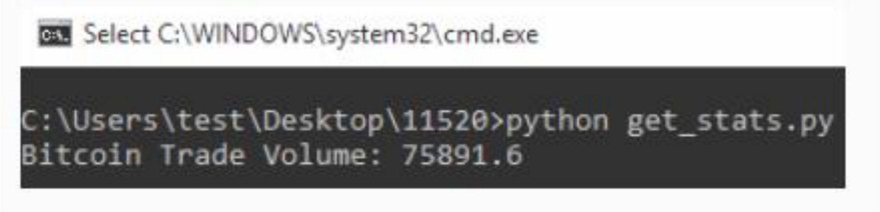
You can call the different methods as follows:

1. Import the relevant class, call the *get* method on *statistics*, and save that object. For example, to get the bitcoin trade volume, we should get the *trade_volume_btc* property from the *stats* object that was created, as shown in the following code:



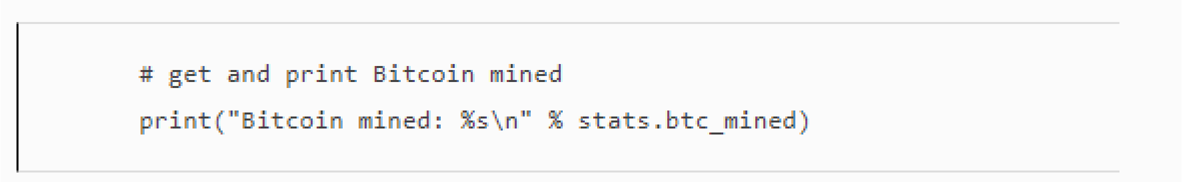
```
#!/usr/bin/env python

# import blockchain library
from blockchain import statistics

# get the stats object
stats = statistics.get()

# get and print Bitcoin trade volume
print("Bitcoin Trade Volume: %s\n" % stats.trade_volume_btc)
```

The following screenshot shows the bitcoin trade volume:



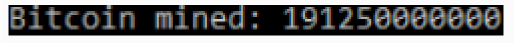
```
C:\Users\test\Desktop\11520>python get_stats.py
Bitcoin Trade Volume: 75891.6
```

2. To get the total bitcoins mined, call the *btc_mined* property on *stats* object, as shown here:



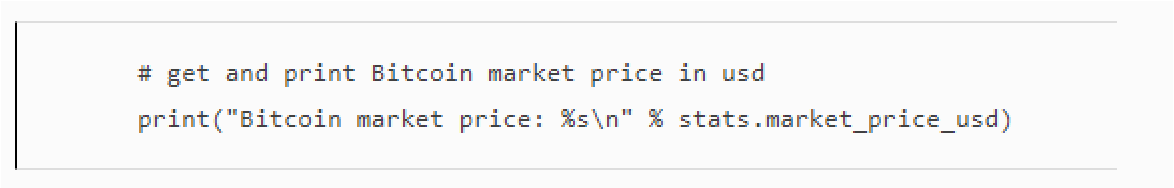
```
# get and print Bitcoin mined
print("Bitcoin mined: %s\n" % stats.btc_mined)
```

The following screenshot shows the output of the number of bitcoins mined:



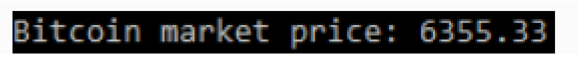
```
Bitcoin mined: 1912500000000
```

3. To get the bitcoin market price, use the *stats* class, call the market price and append that with the particular currency:



```
# get and print Bitcoin market price in usd
print("Bitcoin market price: %s\n" % stats.market_price_usd)
```

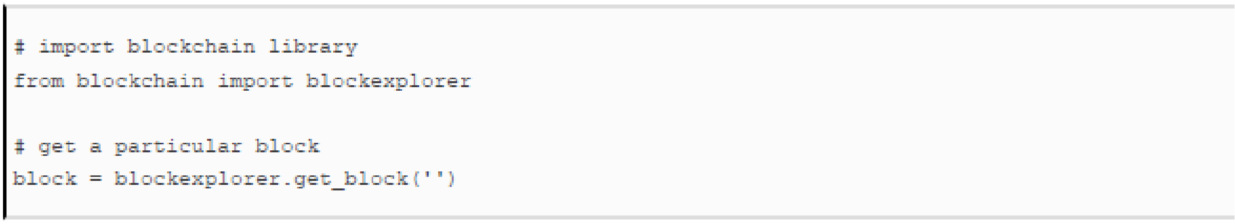
The current bitcoin price is shown in US dollars as follows:



```
Bitcoin market price: 6355.33
```

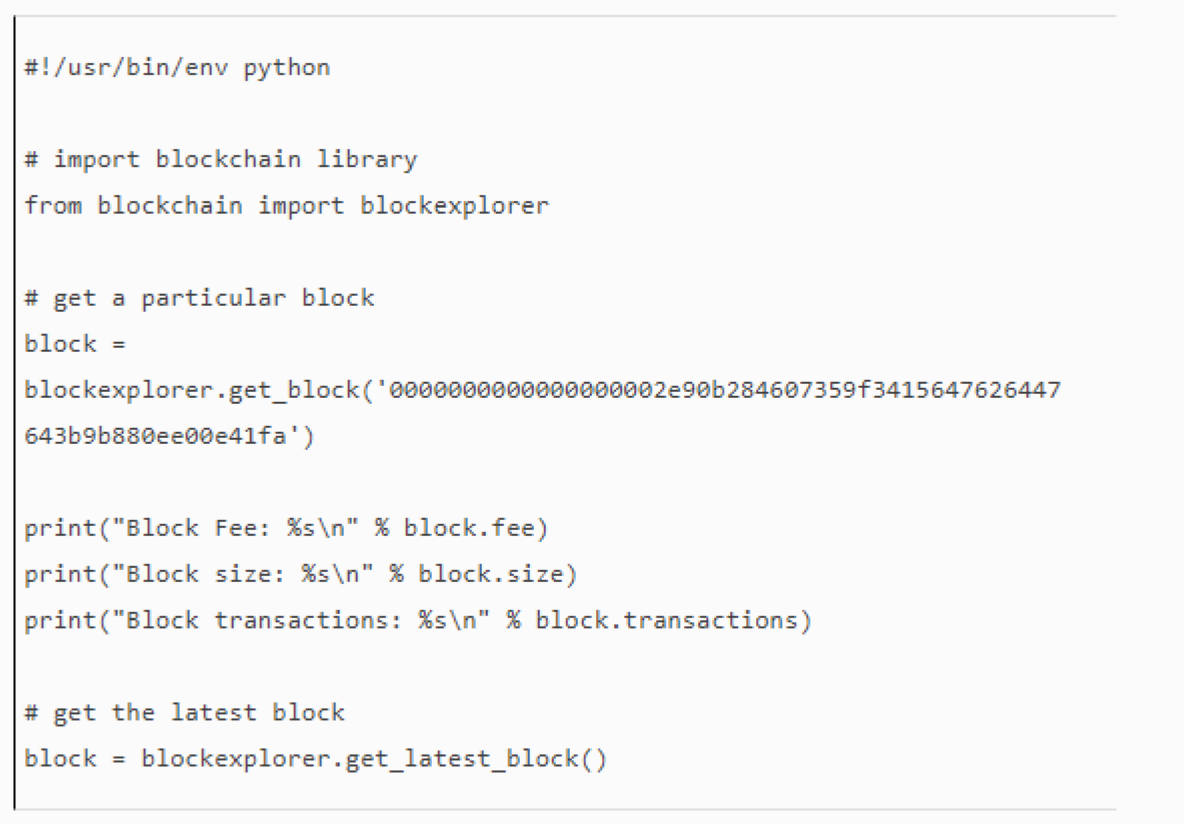
IV. Block explorer methods

For block explorer methods, start by importing the relevant classes from the *blockchain* library. To get a particular block, call the *get_block* method as shown in the following code. It expects a block to be passed in as the parameter.



```
# import blockchain library
from blockchain import blockexplorer

# get a particular block
block = blockexplorer.get_block('')
```

Now let’s get some information about the block. For example, the block fee, block size, and block transactions can be obtained by using *fee*, *size*, and *transactions* properties respectively on the block object created, as shown in the following code:



```
#!/usr/bin/env python

# import blockchain library
from blockchain import blockexplorer

# get a particular block
block =
blockexplorer.get_block('0000000000000000002e90b284607359f3415647626447
643b9b880ee00e41fa')

print("Block Fee: %s\n" % block.fee)
print("Block size: %s\n" % block.size)
print("Block transactions: %s\n" % block.transactions)

# get the latest block
block = blockexplorer.get_latest_block()
```

Shows the block fee, block size, and block transactions output to TA.

There are also many available features in the Blockchain.info library; there are a few that are more related to, for example, wallets, creating wallets, and so on.

Well done!!

The End of Practice 2