# ETL Project: Comparison of closing price between cryptocurrencies and ETFs

By Morgan Ivey, Adan Bonilla, and Orlando Lepe

Proposal:

Using complete historical Cryptocurrency Financial Data and Stock Market Data obtained from kaggle.com we were interested in developing a database  by combining cryptocurrency and stock market closing price in order to provide a production database that can then be utilized for further analysis.

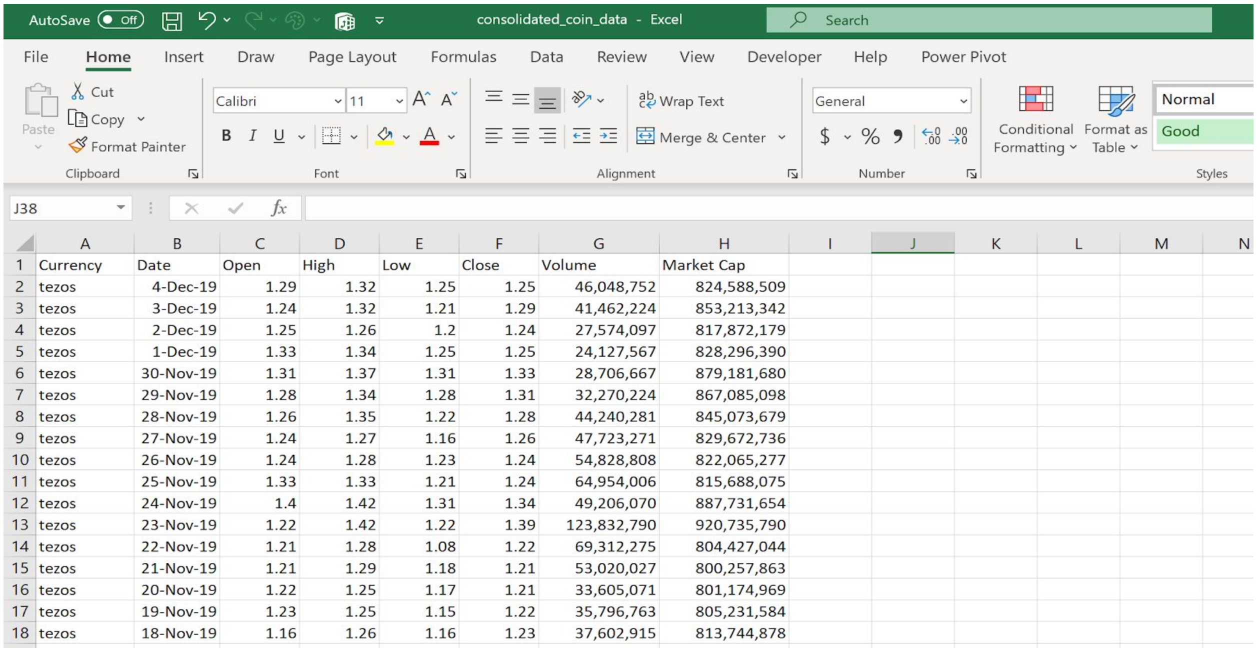
Extraction:

Data was loaded into a Jupyter Notebook for analysis and transformation.

**Data Source:** Complete Historical Cryptocurrency Financial Data | 4/28/13 - 12/4/19 (https://www.kaggle.com/philmohun/cryptocurrency-financial-data)

**Format:** CSV

The cryptocurrency data set is a single .csv file containing 28,944 records of cryptocurrency data spanning from 4/28/13 to 12/4/19.

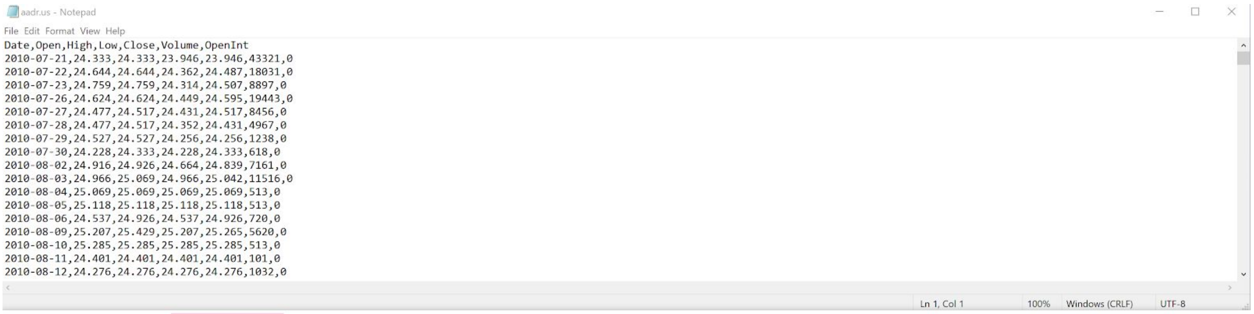


| | A | B | C | D | E | F | G | H |
|---|---|---|---|---|---|---|---|---|
| 1 | Currency | Date | Open | High | Low | Close | Volume | Market Cap |
| 2 | tezos | 4-Dec-19 | 1.29 | 1.32 | 1.25 | 1.25 | 46,048,752 | 824,588,509 |
| 3 | tezos | 3-Dec-19 | 1.24 | 1.32 | 1.21 | 1.29 | 41,462,224 | 853,213,342 |
| 4 | tezos | 2-Dec-19 | 1.25 | 1.26 | 1.2 | 1.24 | 27,574,097 | 817,872,179 |
| 5 | tezos | 1-Dec-19 | 1.33 | 1.34 | 1.25 | 1.25 | 24,127,567 | 828,296,390 |
| 6 | tezos | 30-Nov-19 | 1.31 | 1.37 | 1.31 | 1.33 | 28,706,667 | 879,181,680 |
| 7 | tezos | 29-Nov-19 | 1.28 | 1.34 | 1.28 | 1.31 | 32,270,224 | 867,085,098 |
| 8 | tezos | 28-Nov-19 | 1.26 | 1.35 | 1.22 | 1.28 | 44,240,281 | 845,073,679 |
| 9 | tezos | 27-Nov-19 | 1.24 | 1.27 | 1.16 | 1.26 | 47,723,271 | 829,672,736 |
| 10 | tezos | 26-Nov-19 | 1.24 | 1.28 | 1.23 | 1.24 | 54,828,808 | 822,065,277 |
| 11 | tezos | 25-Nov-19 | 1.33 | 1.33 | 1.21 | 1.24 | 64,954,006 | 815,688,075 |
| 12 | tezos | 24-Nov-19 | 1.4 | 1.42 | 1.31 | 1.34 | 49,206,070 | 887,731,654 |
| 13 | tezos | 23-Nov-19 | 1.22 | 1.42 | 1.22 | 1.39 | 123,832,790 | 920,735,790 |
| 14 | tezos | 22-Nov-19 | 1.21 | 1.28 | 1.08 | 1.22 | 69,312,275 | 804,427,044 |
| 15 | tezos | 21-Nov-19 | 1.21 | 1.29 | 1.18 | 1.21 | 53,020,027 | 800,257,863 |
| 16 | tezos | 20-Nov-19 | 1.22 | 1.25 | 1.17 | 1.21 | 33,605,071 | 801,174,969 |
| 17 | tezos | 19-Nov-19 | 1.23 | 1.25 | 1.15 | 1.22 | 35,796,763 | 805,231,584 |
| 18 | tezos | 18-Nov-19 | 1.16 | 1.26 | 1.16 | 1.23 | 37,602,915 | 813,744,878 |

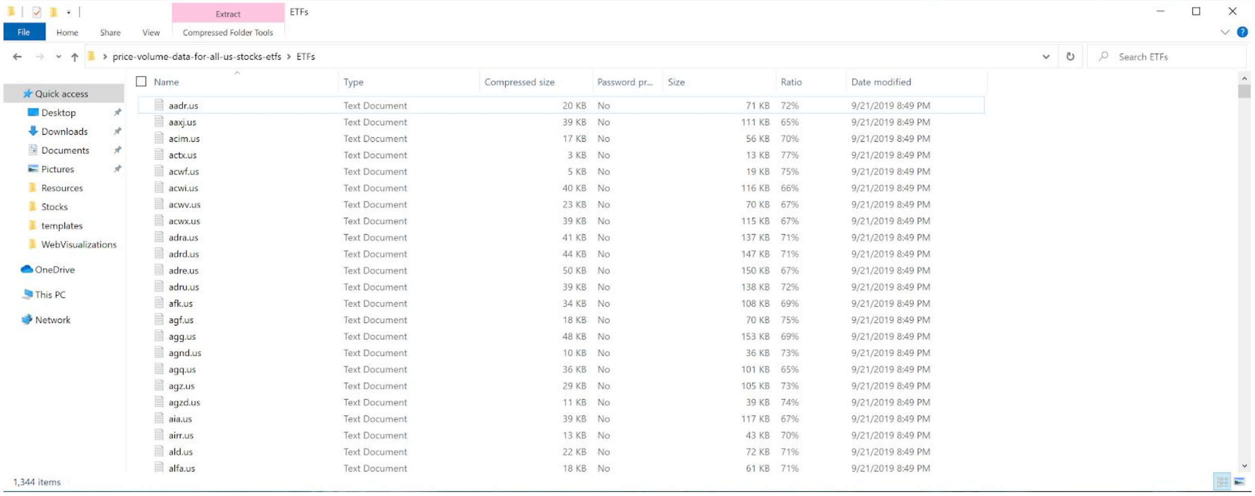
**Data Source:** Huge Market Dataset | 4/1/13 -1/7/16
(https://www.kaggle.com/borismarjanovic/price-volume-data-for-all-us-stocks-etfs#aadr.us.txt)

**Format:** TXT

The Stock Market data set was also obtained from kaggle.com, however, it was comprised of multiple .txt files. Each .txt file contained information for each individual stock. The screenshot below shows what the raw data looks like.

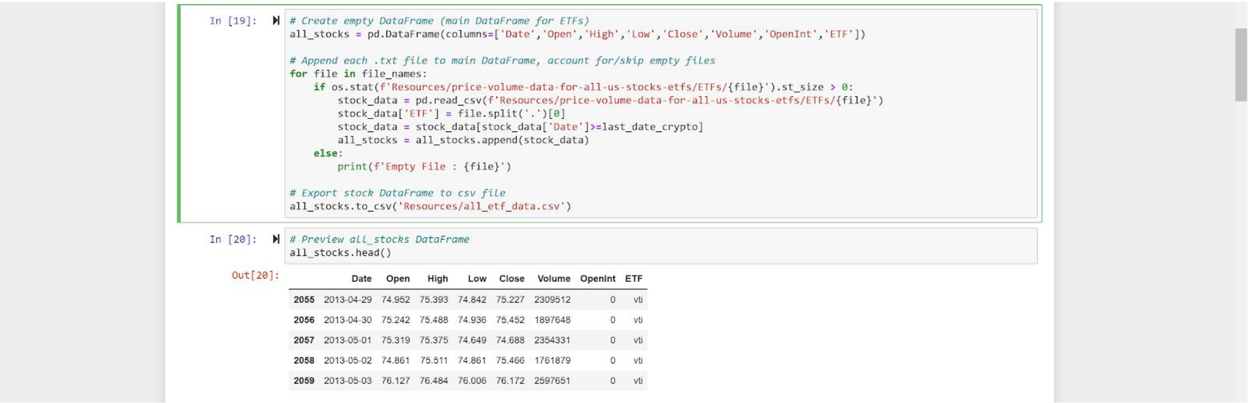


```
Date,Open,High,Low,Close,Volume,OpenInt
2010-07-21,24.333,24.333,23.946,23.946,43321,0
2010-07-22,24.644,24.644,24.362,24.487,18031,0
2010-07-23,24.759,24.759,24.314,24.507,8897,0
2010-07-26,24.624,24.624,24.449,24.595,19443,0
2010-07-27,24.477,24.517,24.431,24.517,8456,0
2010-07-28,24.477,24.517,24.352,24.431,4967,0
2010-07-29,24.527,24.527,24.256,24.256,1238,0
2010-07-30,24.228,24.333,24.228,24.333,618,0
2010-08-02,24.916,24.926,24.664,24.839,7161,0
2010-08-03,24.966,25.069,24.966,25.042,11516,0
2010-08-04,25.069,25.069,25.069,25.069,513,0
2010-08-05,25.118,25.118,25.118,25.118,513,0
2010-08-06,24.537,24.926,24.537,24.926,720,0
2010-08-09,25.207,25.429,25.207,25.265,5620,0
2010-08-10,25.285,25.285,25.285,25.285,513,0
2010-08-11,24.401,24.401,24.401,24.401,101,0
2010-08-12,24.276,24.276,24.276,24.276,1032,0
```



| Name | Type | Compressed size | Password pr... | Size | Ratio | Date modified |
|---|---|---|---|---|---|---|
| aadr.us | Text Document | 20 KB | No | 71 KB | 72% | 9/21/2019 8:49 PM |
| aaxj.us | Text Document | 39 KB | No | 111 KB | 65% | 9/21/2019 8:49 PM |
| acim.us | Text Document | 17 KB | No | 56 KB | 70% | 9/21/2019 8:49 PM |
| actx.us | Text Document | 3 KB | No | 13 KB | 77% | 9/21/2019 8:49 PM |
| acwf.us | Text Document | 5 KB | No | 19 KB | 75% | 9/21/2019 8:49 PM |
| acwi.us | Text Document | 40 KB | No | 116 KB | 66% | 9/21/2019 8:49 PM |
| acwv.us | Text Document | 23 KB | No | 70 KB | 67% | 9/21/2019 8:49 PM |
| acwx.us | Text Document | 39 KB | No | 115 KB | 67% | 9/21/2019 8:49 PM |
| adra.us | Text Document | 41 KB | No | 137 KB | 71% | 9/21/2019 8:49 PM |
| adrd.us | Text Document | 44 KB | No | 147 KB | 71% | 9/21/2019 8:49 PM |
| adre.us | Text Document | 50 KB | No | 150 KB | 67% | 9/21/2019 8:49 PM |
| adru.us | Text Document | 39 KB | No | 138 KB | 72% | 9/21/2019 8:49 PM |
| afk.us | Text Document | 34 KB | No | 108 KB | 69% | 9/21/2019 8:49 PM |
| agf.us | Text Document | 18 KB | No | 70 KB | 75% | 9/21/2019 8:49 PM |
| agg.us | Text Document | 48 KB | No | 153 KB | 69% | 9/21/2019 8:49 PM |
| agnd.us | Text Document | 10 KB | No | 36 KB | 73% | 9/21/2019 8:49 PM |
| agq.us | Text Document | 36 KB | No | 101 KB | 65% | 9/21/2019 8:49 PM |
| agz.us | Text Document | 29 KB | No | 105 KB | 73% | 9/21/2019 8:49 PM |
| agzd.us | Text Document | 11 KB | No | 39 KB | 74% | 9/21/2019 8:49 PM |
| aia.us | Text Document | 39 KB | No | 117 KB | 67% | 9/21/2019 8:49 PM |
| airr.us | Text Document | 13 KB | No | 43 KB | 70% | 9/21/2019 8:49 PM |
| ald.us | Text Document | 22 KB | No | 72 KB | 71% | 9/21/2019 8:49 PM |
| alfa.us | Text Document | 18 KB | No | 61 KB | 71% | 9/21/2019 8:49 PM |

Once we loaded the data in jupyter notebook, converting the cryptocurrency into a ‘Date’ dataframe was very straightforward process, however, in order for the stock data to be usable, we needed to extract the data for each individual .txt file and place the data into a dataframe. In essence what we needed to do is run a for loop to read each .txt file and append the data into an empty dataframe. Once we were able to successfully extract the data from the .txt files we then converted the dataframe to a csv file. See screenshot of jupyter notebook code of the for loop.



In [19]:
```python
# Create empty DataFrame (main DataFrame for ETFs)
all_stocks = pd.DataFrame(columns=['Date','Open','High','Low','Close','Volume','OpenInt','ETF'])

# Append each .txt file to main DataFrame, account for/skip empty files
for file in file_names:
    if os.stat(f'Resources/price-volume-data-for-all-us-stocks-etfs/ETFs/{file}').st_size > 0:
        stock_data = pd.read_csv(f'Resources/price-volume-data-for-all-us-stocks-etfs/ETFs/{file}')
        stock_data['ETF'] = file.split('.')[0]
        stock_data = stock_data[stock_data['Date']>=last_date_crypto]
        all_stocks = all_stocks.append(stock_data)
    else:
        print(f'Empty File : {file}')

# Export stock DataFrame to csv file
all_stocks.to_csv('Resources/all_etf_data.csv')
```

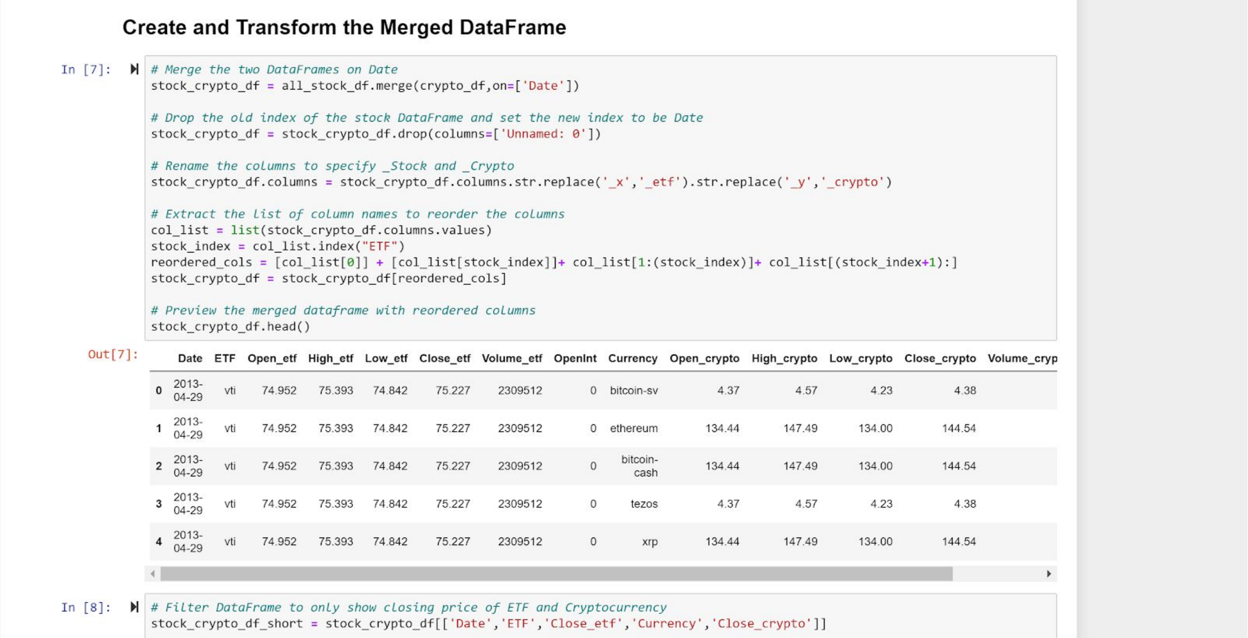
In [20]:
```python
# Preview all_stocks DataFrame
all_stocks.head()
```

Out[20]:

| | Date | Open | High | Low | Close | Volume | OpenInt | ETF |
|---|---|---|---|---|---|---|---|---|
| **2055** | 2013-04-29 | 74.952 | 75.393 | 74.842 | 75.227 | 2309512 | 0 | vti |
| **2056** | 2013-04-30 | 75.242 | 75.488 | 74.936 | 75.452 | 1897648 | 0 | vti |
| **2057** | 2013-05-01 | 75.319 | 75.375 | 74.649 | 74.688 | 2354331 | 0 | vti |
| **2058** | 2013-05-02 | 74.861 | 75.511 | 74.861 | 75.466 | 1761879 | 0 | vti |
| **2059** | 2013-05-03 | 76.127 | 76.484 | 76.006 | 76.172 | 2597651 | 0 | vti |

Transformation:

Once we had both the cryptocurrency data and stock data in dataframes, we were then able to transform the two dataframes so that we could merge the two dataframes by date. This entailed converting the date to the same format and sorting dataframes by date. Additionally, we also had to reorder the columns and relabel them.



### Create and Transform the Merged DataFrame

In [7]:
```python
# Merge the two DataFrames on Date
stock_crypto_df = all_stock_df.merge(crypto_df,on=['Date'])

# Drop the old index of the stock DataFrame and set the new index to be Date
stock_crypto_df = stock_crypto_df.drop(columns=['Unnamed: 0'])

# Rename the columns to specify _Stock and _Crypto
stock_crypto_df.columns = stock_crypto_df.columns.str.replace('_x','_etf').str.replace('_y','_crypto')

# Extract the list of column names to reorder the columns
col_list = list(stock_crypto_df.columns.values)
stock_index = col_list.index("ETF")
reordered_cols = [col_list[0]] + [col_list[stock_index]]+ col_list[1:(stock_index)]+ col_list[(stock_index+1):]
stock_crypto_df = stock_crypto_df[reordered_cols]

# Preview the merged dataframe with reordered columns
stock_crypto_df.head()
```

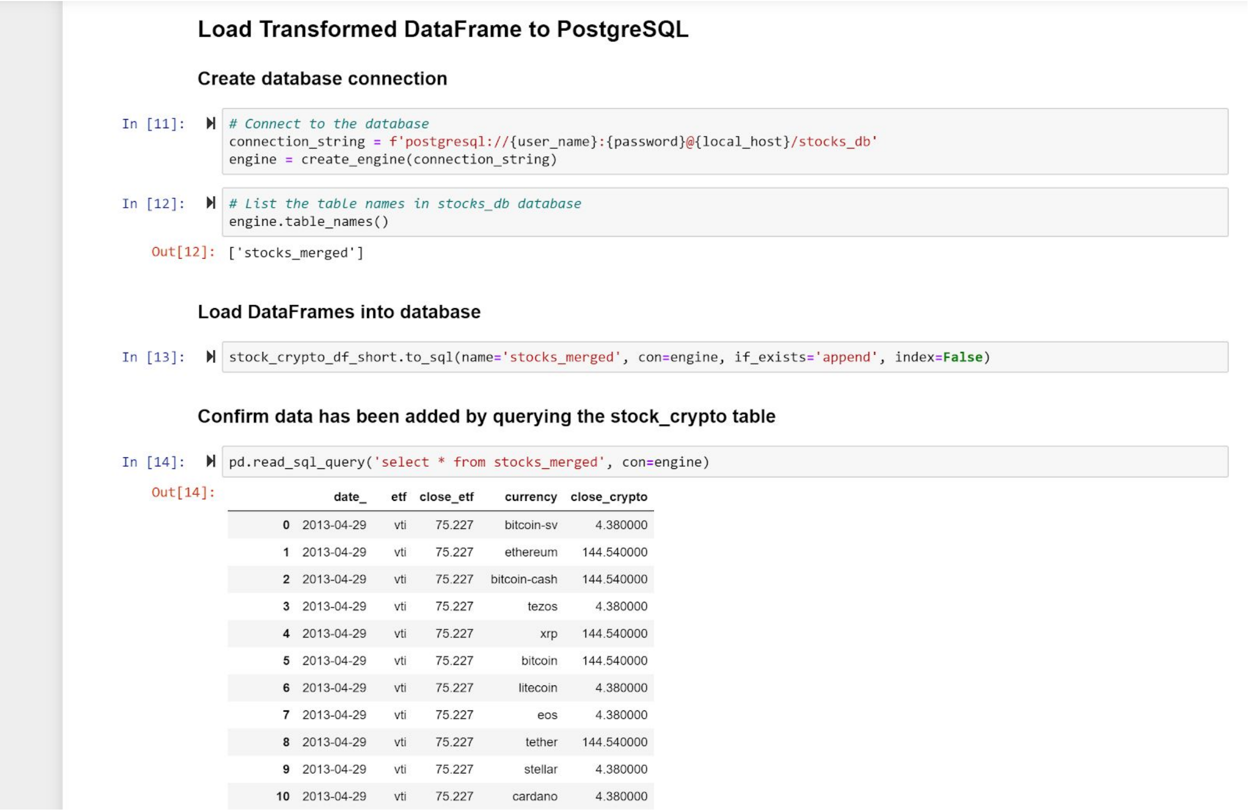
Out[7]:

| | Date | ETF | Open_etf | High_etf | Low_etf | Close_etf | Volume_etf | OpenInt | Currency | Open_crypto | High_crypto | Low_crypto | Close_crypto | Volume_cryp |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 2013-04-29 | vti | 74.952 | 75.393 | 74.842 | 75.227 | 2309512 | 0 | bitcoin-sv | 4.37 | 4.57 | 4.23 | 4.38 | |
| **1** | 2013-04-29 | vti | 74.952 | 75.393 | 74.842 | 75.227 | 2309512 | 0 | ethereum | 134.44 | 147.49 | 134.00 | 144.54 | |
| **2** | 2013-04-29 | vti | 74.952 | 75.393 | 74.842 | 75.227 | 2309512 | 0 | bitcoin-cash | 134.44 | 147.49 | 134.00 | 144.54 | |
| **3** | 2013-04-29 | vti | 74.952 | 75.393 | 74.842 | 75.227 | 2309512 | 0 | tezos | 4.37 | 4.57 | 4.23 | 4.38 | |
| **4** | 2013-04-29 | vti | 74.952 | 75.393 | 74.842 | 75.227 | 2309512 | 0 | xrp | 134.44 | 147.49 | 134.00 | 144.54 | |

In [8]:
```python
# Filter DataFrame to only show closing price of ETF and Cryptocurrency
stock_crypto_df_short = stock_crypto_df[['Date','ETF','Close_etf','Currency','Close_crypto']]
```

Loading:

Once we completed the transformation process and cleaned the dataframe we proceeded to load the new dataframe to PostgreSQL. In this phase of the project we ran into some issues primarily because the merged dataframe with the cryptocurrency and stock data was very large, 74 million rows. As a result, when attempting to load it to PostgreSQL our computer would crash. Consequently, we decided to change course and revisit the stock data and dataframe and load the ETF data instead of the stock data. Once we changed course, we were successful in loading the data to PostgreSQL, however, it took 3 hours to complete. Finally, we confirmed that the data had loaded successfully.



## Load Transformed DataFrame to PostgreSQL

### Create database connection

In [11]:
```
# Connect to the database
connection_string = f'postgresql://{user_name}:{password}@{local_host}/stocks_db'
engine = create_engine(connection_string)
```

In [12]:
```
# List the table names in stocks_db database
engine.table_names()
```

Out[12]:
```
['stocks_merged']
```

### Load DataFrames into database

In [13]:
```
stock_crypto_df_short.to_sql(name='stocks_merged', con=engine, if_exists='append', index=False)
```

### Confirm data has been added by querying the stock_crypto table

In [14]:
```
pd.read_sql_query('select * from stocks_merged', con=engine)
```

Out[14]:

| | date_ | etf | close_etf | currency | close_crypto |
|---|---|---|---|---|---|
| **0** | 2013-04-29 | vti | 75.227 | bitcoin-sv | 4.380000 |
| **1** | 2013-04-29 | vti | 75.227 | ethereum | 144.540000 |
| **2** | 2013-04-29 | vti | 75.227 | bitcoin-cash | 144.540000 |
| **3** | 2013-04-29 | vti | 75.227 | tezos | 4.380000 |
| **4** | 2013-04-29 | vti | 75.227 | xrp | 144.540000 |
| **5** | 2013-04-29 | vti | 75.227 | bitcoin | 144.540000 |
| **6** | 2013-04-29 | vti | 75.227 | litecoin | 4.380000 |
| **7** | 2013-04-29 | vti | 75.227 | eos | 4.380000 |
| **8** | 2013-04-29 | vti | 75.227 | tether | 144.540000 |
| **9** | 2013-04-29 | vti | 75.227 | stellar | 4.380000 |
| **10** | 2013-04-29 | vti | 75.227 | cardano | 4.380000 |