2_Dataset

September 21, 2023

0.0.1 Observe the data fetched from http://www.nasdaqtrader.com/dynamic/SymDir/nasdaqtrade

[1]:
```python
# import libraries
import pandas as pd
```

[2]:
```python
# get the CSV data with pandas
df = pd.read_csv("http://www.nasdaqtrader.com/dynamic/SymDir/nasdaqtraded.txt",␣
 ↪sep='|')
```

[3]:
```python
df.head()
```

[3]:
```
  Nasdaq Traded Symbol                                      Security Name  \
0             Y      A            Agilent Technologies, Inc. Common Stock
1             Y     AA                    Alcoa Corporation Common Stock
2             Y    AAA  Investment Managers Series Trust II AXS First …
3             Y   AAAU             Goldman Sachs Physical Gold ETF Shares
4             Y    AAC  Ares Acquisition Corporation Class A Ordinary …

  Listing Exchange Market Category ETF  Round Lot Size Test Issue  \
0                N                   N           100.0          N
1                N                   N           100.0          N
2                P                   Y           100.0          N
3                Z                   Y           100.0          N
4                N                   N           100.0          N

  Financial Status CQS Symbol NASDAQ Symbol NextShares
0              NaN          A             A          N
1              NaN         AA            AA          N
2              NaN        AAA           AAA          N
3              NaN       AAAU          AAAU          N
4              NaN        AAC           AAC          N
```

[4]:
```python
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11312 entries, 0 to 11311
Data columns (total 12 columns):
 #   Column            Non-Null Count  Dtype
```

```
---  ------             --------------  -----
 0   Nasdaq Traded     11312 non-null  object
 1   Symbol            11310 non-null  object
 2   Security Name     11311 non-null  object
 3   Listing Exchange  11311 non-null  object
 4   Market Category   11311 non-null  object
 5   ETF               11311 non-null  object
 6   Round Lot Size    11311 non-null  float64
 7   Test Issue        11311 non-null  object
 8   Financial Status  5130 non-null   object
 9   CQS Symbol        6181 non-null   object
 10  NASDAQ Symbol     11310 non-null  object
 11  NextShares        11311 non-null  object
dtypes: float64(1), object(11)
memory usage: 1.0+ MB
```

[5]:
```
# remove NaNs from interested columns and test data records
df = df[df['Test Issue'] == 'N']
df.dropna(subset = ["NASDAQ Symbol", "ETF"], inplace=True)
```

[6]:
```
# Check the numbers of ETFs and Stocks in the symbols source
etfs = df[df['ETF'] == 'Y']['NASDAQ Symbol'].tolist()
stocks = df[df['ETF'] == 'N']['NASDAQ Symbol'].tolist()

print("The Nasdaqtraded data contains {} ETF and {} Stock symbols".
  ↪format(len(etfs), len(stocks)))
```

```
The Nasdaqtraded data contains 3206 ETF and 8070 Stock symbols
```

0.0.2 Observe a sample symbol from the nasdaqtraded dataset

[7]:
```
# import libraries
import yfinance as yf
```

[8]:
```
# download Apple Inc. (AAPL) historical data
AAPL = yf.download("AAPL", period="max")
```

```
[*********************100%%***********************]  1 of 1 completed
```

[9]:
```
AAPL.info()
```

```
<class 'pandas.core.frame.DataFrame'>
DatetimeIndex: 10784 entries, 1980-12-12 to 2023-09-21
Data columns (total 6 columns):
 #   Column     Non-Null Count  Dtype
---  ------     --------------  -----
 0   Open       10784 non-null  float64
 1   High       10784 non-null  float64
 2   Low        10784 non-null  float64
 3   Close      10784 non-null  float64
 4   Adj Close  10784 non-null  float64
 5   Volume     10784 non-null  int64
dtypes: float64(5), int64(1)
memory usage: 589.8 KB
```

[10]:

```
AAPL.describe()
```

[10]:

```
               Open          High           Low         Close     Adj Close  \
count  10784.000000  10784.000000  10784.000000  10784.000000  10784.000000
mean      19.014140     19.225556     18.809344     19.025461     18.294860
std       39.978130     40.434202     39.548238     40.009281     39.627338
min        0.049665      0.049665      0.049107      0.049107      0.038050
25%        0.292411      0.300223      0.284821      0.292411      0.240734
50%        0.502232      0.511161      0.494978      0.504464      0.413983
75%       18.535713     18.673125     18.363839     18.549465     16.070243
max      196.240005    198.229996    195.279999    196.449997    196.185074

             Volume
count  1.078400e+04
mean   3.231960e+08
std    3.367551e+08
min    0.000000e+00
25%    1.176843e+08
50%    2.108792e+08
75%    4.032098e+08
max    7.421641e+09
```

[11]:

```
AAPL.head()
```

[11]:

```
                Open      High       Low     Close  Adj Close     Volume
Date
1980-12-12  0.128348  0.128906  0.128348  0.128348   0.099449  469033600
1980-12-15  0.122210  0.122210  0.121652  0.121652   0.094261  175884800
1980-12-16  0.113281  0.113281  0.112723  0.112723   0.087343  105728000
1980-12-17  0.115513  0.116071  0.115513  0.115513   0.089504   86441600
1980-12-18  0.118862  0.119420  0.118862  0.118862   0.092099   73449600
```