```
import pandas as pd
import numpy as np
import html
import re
import nltk
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
import os
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
from nltk.tokenize import RegexpTokenizer
import matplotlib.pyplot as plt
import seaborn as sns
from textblob import TextBlob
```

```
dataframe = pd.read_csv('/content/sample_data/AppleTwitterData20_21.csv')
dataframe.head()
```

	Datetime	Text
0	2020-06-29 22:56:32+00:00	#GBPAUD 145 PIPS Profit🔥🔥\n\nFor free signals,...
1	2020-06-29 22:20:05+00:00	APPLE ANALYSIS 15 MIN (BUY S...
2	2020-06-29 21:35:23+00:00	EURNZD TP2 Hit! 180 PIPS PROFIT 🔥👍⚡\n\nFor F...
3	2020-06-29 21:17:03+00:00	Apple Inc price at close, 2020-06-29, is 361.7...
4	2020-06-29 20:40:54+00:00	#NZDCAD Target Hit! 90 PIPS Profit 🔥\n\nChecko...

```
df = dataframe.sort_values(by = 'Datetime', ignore_index = True)
df
```

	Datetime	Text
0		NaN
1	2020-01-01 00:24:13+00:00	4 &amp; 1 hour charts for all 78 instruments ...
2	2020-01-01 00:56:57+00:00	Top 10 trades of the decade: Number 4: Going l...
3	2020-01-01 01:24:41+00:00	4 &amp; 1 hour charts for the Group 3 instrume...
4	2020-01-01 01:26:34+00:00	4 &amp; 1 hour charts for all 78 instruments a...
...	...	...
33363	2021-09-20 22:10:27+00:00	AAPL 20210917 Weekly Price Pattern Coordinates...
33364	2021-09-20 22:15:53+00:00	Apple Inc price at close, 2021-09-20, is 142.9...
33365	2021-09-20 23:15:03+00:00	@MacRumors @julipuli Now, if only any of us ou...
33366	Datetime	Text
33367	Datetime	Text

33368 rows × 2 columns

```
df.drop_duplicates(keep='first') #delete the duplicates by dropping them and store the result value to a new variable
df.head()
```

	Datetime	Text
0		NaN
1	2020-01-01 00:24:13+00:00	4 &amp; 1 hour charts for all 78 instruments ...
2	2020-01-01 00:56:57+00:00	Top 10 trades of the decade: Number 4: Going l...
3	2020-01-01 01:24:41+00:00	4 &amp; 1 hour charts for the Group 3 instrume...
4	2020-01-01 01:26:34+00:00	4 &amp; 1 hour charts for all 78 instruments a...

```
df.shape
```

```
(33368, 2)
```

```
df.isnull().sum()
```

```
Datetime    0
Text        1
dtype: int64
```

```
df.dropna()
```

	Datetime	Text
1	2020-01-01 00:24:13+00:00	4 &amp; 1 hour charts for all 78 instruments ...
2	2020-01-01 00:56:57+00:00	Top 10 trades of the decade: Number 4: Going l...
3	2020-01-01 01:24:41+00:00	4 &amp; 1 hour charts for the Group 3 instrume...
4	2020-01-01 01:26:34+00:00	4 &amp; 1 hour charts for all 78 instruments a...
5	2020-01-01 01:30:11+00:00	Made one big mistake Day Trading Apple (AAPL) ...
...	...	...
33363	2021-09-20 22:10:27+00:00	AAPL 20210917 Weekly Price Pattern Coordinates...
33364	2021-09-20 22:15:53+00:00	Apple Inc price at close, 2021-09-20, is 142.9...
33365	2021-09-20 23:15:03+00:00	@MacRumors @julipuli Now, if only any of us ou...
33366	Datetime	Text
33367	Datetime	Text

33367 rows × 2 columns

```
df.shape
```

```
(33368, 2)
```

```
def preprocessing_text(df):
    #put everythin in lowercase
    #put everythin in lowercase
    df['Text'] = df['Text'].str.lower()
    #Replace rt indicating that was a retweet
    df['Text'] = df['Text'].str.replace('rt', '')
    #Replace occurences of mentioning @UserNames
    df['Text'] = df['Text'].replace(r'@\w+', '', regex=True)
    #Replace links contained in the tweet
    df['Text']= df['Text'].replace(r'http\S+', '', regex=True)
    df['Text'] = df['Text'].replace(r'www.[^ ]+', '', regex=True)
    #remove numbers
    df['Text'] = df['Text'].replace(r'[0-9]+', '', regex=True)
    #replace special characters and puntuation marks
    df['Text'] = df['Text'].replace(r'[!"#$%&()*+,-./:;<=>?@[\]^_`{|}~]\n', '', regex=True)
    return df
```

```
preprocessing_text(df)
```

	Datetime	Text
0		NaN
1	2020-01-01 00:24:13+00:00	&amp; hour chas for all instruments are av...
2	2020-01-01 00:56:57+00:00	top trades of the decade: number : going long...
3	2020-01-01 01:24:41+00:00	&amp; hour chas for the group instruments a...
4	2020-01-01 01:26:34+00:00	&amp; hour chas for all instruments are ava...
...	...	...
33363	2021-09-20 22:10:27+00:00	aapl weekly price pattern coordinates\nannota...
33364	2021-09-20 22:15:53+00:00	apple inc price at close, --, is .. #apple #aapl
33365	2021-09-20 23:15:03+00:00	now, if only any of us outside the us could ...
33366	Datetime	text
33367	Datetime	text

33368 rows × 2 columns

```
df.dropna()
```

	Datetime	Text
1	2020-01-01 00:24:13+00:00	&amp; hour chas for all instruments are av...
2	2020-01-01 00:56:57+00:00	top trades of the decade: number : going long...
3	2020-01-01 01:24:41+00:00	&amp; hour chas for the group instruments a...
4	2020-01-01 01:26:34+00:00	&amp; hour chas for all instruments are ava...
5	2020-01-01 01:30:11+00:00	made one big mistake day trading apple (aapl) ...
...	...	...
33363	2021-09-20 22:10:27+00:00	aapl weekly price pattern coordinates\nannota...
33364	2021-09-20 22:15:53+00:00	apple inc price at close, --, is .. #apple #aapl
33365	2021-09-20 23:15:03+00:00	now, if only any of us outside the us could ...
33366	Datetime	text
33367	Datetime	text

33367 rows × 2 columns

```
df.head(20)
```

	Datetime	Text
0		NaN
1	2020-01-01 00:24:13+00:00	&amp; hour chas for all instruments are av...
2	2020-01-01 00:56:57+00:00	top trades of the decade: number : going long...
3	2020-01-01 01:24:41+00:00	&amp; hour chas for the group instruments a...
4	2020-01-01 01:26:34+00:00	&amp; hour chas for all instruments are ava...
5	2020-01-01 01:30:11+00:00	made one big mistake day trading apple (aapl) ...
6	2020-01-01 07:32:26+00:00	was a great year for me.. thanks to the unive...
7	2020-01-01 09:06:20+00:00	total returns..bitcoin: +nasdaq : +% ...
8	2020-01-01 16:40:06+00:00	made one big mistake day trading apple (aapl) ...
9	2020-01-01 16:45:09+00:00	$spy\n\n staed - what’s next\n- i staed lookin...
10	2020-01-01 16:52:00+00:00	time for a wrap on my strava stats. 🚴+🏃= ,mi...
11	2020-01-01 17:40:00+00:00	#aapl - aapl +% en el - tradingview -
12	2020-01-01 17:46:33+00:00	happy productivity in with #pdfzone #app #mac...
13	2020-01-01 19:38:19+00:00	$es_f thu jobless claims/pmi\ngap up &amp; go ...
14	2020-01-01 19:46:17+00:00	$es_f\n\nif there is a gap up open today (... ...
15	2020-01-01 19:52:39+00:00	$es_f\n\nhourly cha\n\nlook like upside is not...
16	2020-01-01 19:53:50+00:00	$es_f\n\nanother view.\n#es_f $spx #trading #f...
17	2020-01-01 19:59:02+00:00	$es_f\n\nso, for a bear case..\ngap was paiall...
18	2020-01-01 20:36:24+00:00	$es_f\n\nit look like a toss up\n\n\nor\n\n\n-...
19	2020-01-01 20:47:14+00:00	$aapl #aapl new years fireworks coming? buyers...

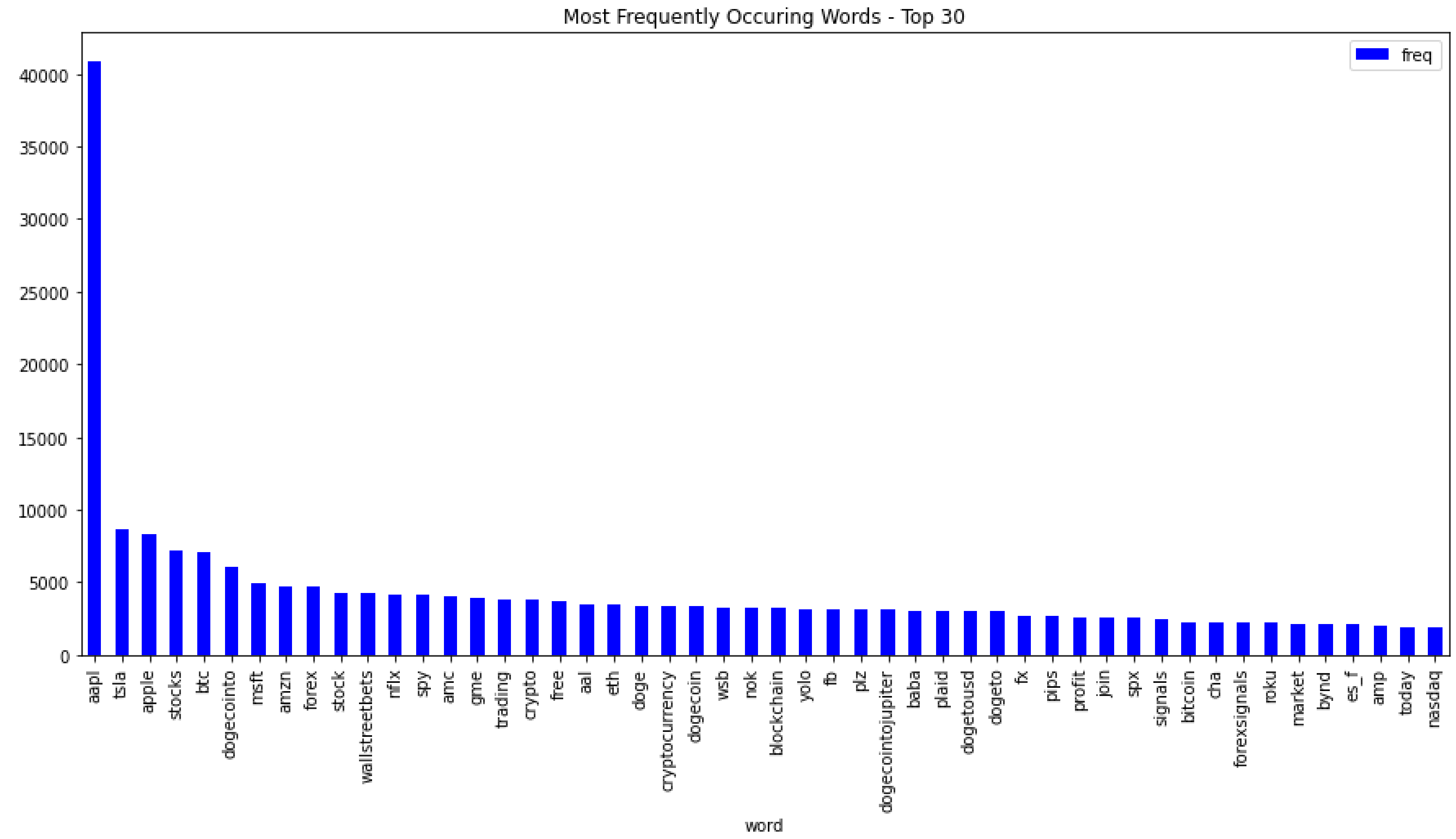
```
df = df.dropna()
```

```
from sklearn.feature_extraction.text import CountVectorizer
```

```
cv = CountVectorizer(stop_words = 'english')
words = cv.fit_transform(df.Text)

sum_words = words.sum(axis=0)
```

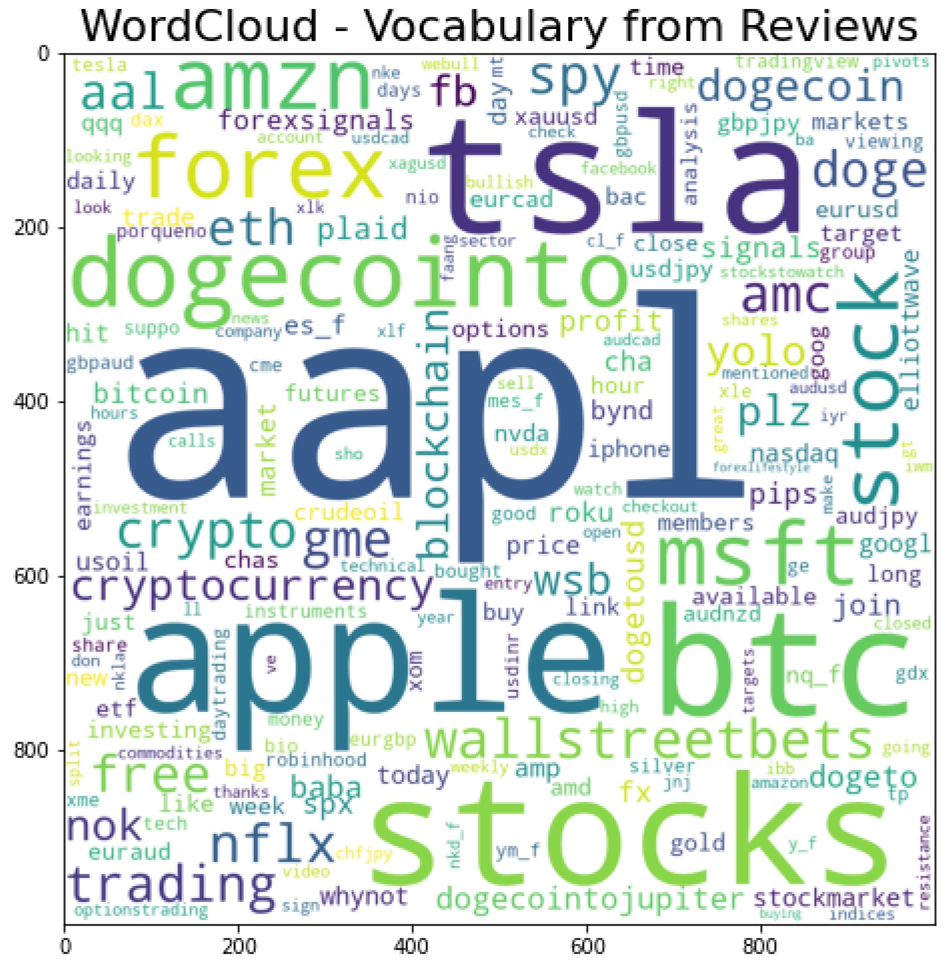
```
words_freq = [(word, sum_words[0, i]) for word, i in cv.vocabulary_.items()]
words_freq = sorted(words_freq, key = lambda x: x[1], reverse = True)

frequency = pd.DataFrame(words_freq, columns=['word', 'freq'])

frequency.head(50).plot(x='word', y='freq', kind='bar', figsize=(15, 7), color = 'blue')
plt.title("Most Frequently Occuring Words - Top 30")
```

```
Text(0.5, 1.0, 'Most Frequently Occuring Words - Top 30')
```



```
from wordcloud import WordCloud

wordcloud = WordCloud(background_color = 'white', width = 1000, height = 1000).generate_from_frequencies(dict(words_freq))

plt.figure(figsize=(10,8))
plt.imshow(wordcloud)
plt.title("WordCloud - Vocabulary from Reviews", fontsize = 22)
```

```
Text(0.5, 1.0, 'WordCloud - Vocabulary from Reviews')
```



```
def getSubjectivity(df):
    return TextBlob(df).sentiment.subjectivity

def getPolarity(df):
    return TextBlob(df).sentiment.polarity

df['Subjectivity'] = df['Text'].apply(getSubjectivity)
df['Polarity'] = df['Text'].apply(getPolarity)

df
```

```
    /usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:7: SettingWithCopyWarning:
    A value is trying to be set on a copy of a slice from a DataFrame.
    Try using .loc[row_indexer,col_indexer] = value instead

    See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
```

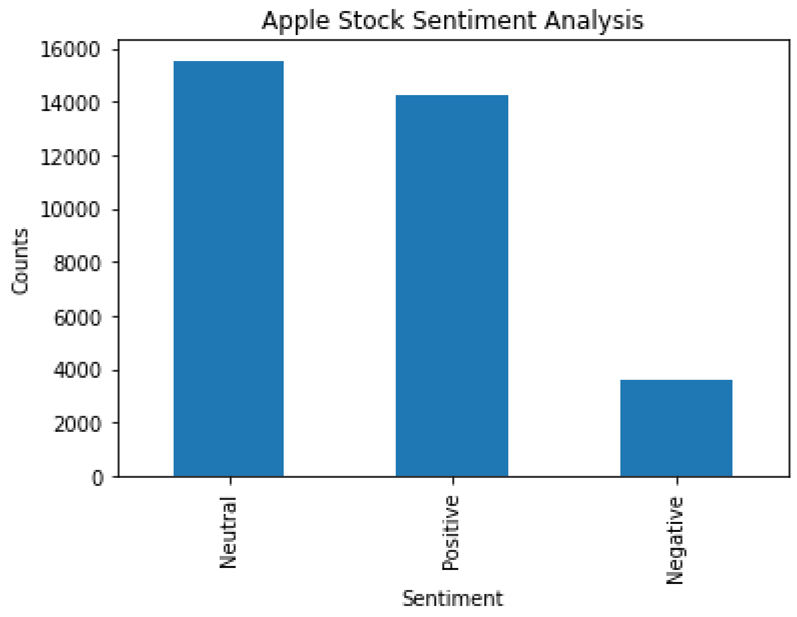
```
def getAnalysis(score):
    if score < 0:
        return 'Negative'
    elif score == 0:
        return 'Neutral'
    else:
        return 'Positive'

df['Analysis'] = df['Polarity'].apply(getAnalysis)
df
```

```
    /usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:9: SettingWithCopyWarning:
    A value is trying to be set on a copy of a slice from a DataFrame.
    Try using .loc[row_indexer,col_indexer] = value instead

    See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
      if __name__ == '__main__':
```

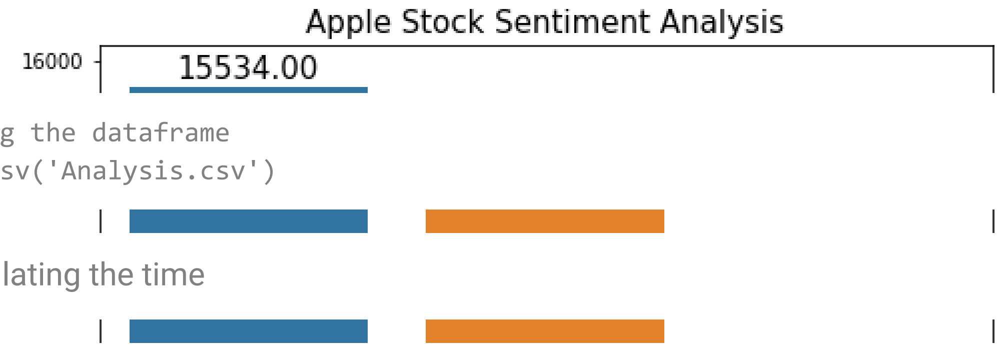
	Datetime	Text	Subjectivity	Polarity	Analysis
1	2020-01-01 00:24:13+00:00	&amp; hour chas for all instruments are av...	0.400000	0.400000	Positive
2	2020-01-01 00:56:57+00:00	top trades of the decade: number : going long...	0.450000	0.218750	Positive
3	2020-01-01 01:24:41+00:00	&amp; hour chas for the group instruments a...	0.400000	0.400000	Positive
4	2020-01-01 01:26:34+00:00	&amp; hour chas for all instruments are ava...	0.400000	0.400000	Positive
5	2020-01-01 01:30:11+00:00	made one big mistake day trading apple (aapl) ...	0.100000	0.000000	Neutral
...	...	...	...	...	...
33363	2021-09-20 22:10:27+00:00	aapl weekly price pattern coordinates\nannota...	0.000000	0.000000	Neutral
33364	2021-09-20 22:15:53+00:00	apple inc price at close, --, is .. #apple #aapl	0.000000	0.000000	Neutral
33365	2021-09-20 23:15:03+00:00	now, if only any of us outside the us could ...	0.395139	-0.010417	Negative
33366	Datetime	text	0.000000	0.000000	Neutral
33367	Datetime	text	0.000000	0.000000	Neutral

33367 rows × 5 columns

```
df['Analysis'].value_counts()

plt.title('Apple Stock Sentiment Analysis')
plt.xlabel('Sentiment')
plt.ylabel('Counts')
df['Analysis'].value_counts().plot(kind = 'bar')
plt.show()
```



```
df_status = df.groupby('Analysis').size().reset_index(name="Counts").sort_values(by="Counts",ascending=False)

plt.figure(figsize=(8, 8))
plots = sns.barplot(x="Analysis", y="Counts", data=df_status)
for bar in plots.patches:

  # passing the coordinates where the annotation shall be done. x-coordinate: bar.get_x() + bar.get_width() / 2, y-coordinate: bar.get_height()
  # free space to be left to make graph pleasing: (0, 8)  # ha and va stand for the horizontal and vertical alignment
    plots.annotate(format(bar.get_height(), '.2f'),
                   (bar.get_x() + bar.get_width() / 2,
                    bar.get_height()), ha='center', va='center',
                   size=15, xytext=(0, 8),
                   textcoords='offset points')
plt.xlabel("Sentiment", size=15)
plt.ylabel("Counts",size=15)
plt.title("Apple Stock Sentiment Analysis", size = 15)
plt.show()
```



```
# saving the dataframe
df.to_csv('Analysis.csv')
```

Manipulating the time

```
df['Datetime'] = df['Datetime'].apply(lambda x:x[:19])
a = list(df["Datetime"].apply(lambda x:len(x)>4))
df["Date"] = pd.to_datetime(df["Datetime"],errors="coerce")
df
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  """Entry point for launching an IPython kernel.
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  This is separate from the ipykernel package so we can avoid doing imports until
```

	Datetime	Text	Subjectivity	Polarity	Analysis	Date
1	2020-01-01 00:24:13	&amp; hour chas for all instruments are av...	0.400000	0.400000	Positive	2020-01-01 00:24:13
2	2020-01-01 00:56:57	top trades of the decade: number : going long...	0.450000	0.218750	Positive	2020-01-01 00:56:57
3	2020-01-01 01:24:41	&amp; hour chas for the group instruments a...	0.400000	0.400000	Positive	2020-01-01 01:24:41
4	2020-01-01 01:26:34	&amp; hour chas for all instruments are ava...	0.400000	0.400000	Positive	2020-01-01 01:26:34
5	2020-01-01 01:30:11	made one big mistake day trading apple (aapl) ...	0.100000	0.000000	Neutral	2020-01-01 01:30:11
...	...	...	...	...	...	...
33363	2021-09-20 22:10:27	aapl weekly price pattern coordinates\nannota...	0.000000	0.000000	Neutral	2021-09-20 22:10:27
33364	2021-09-20 22:15:53	apple inc price at close, --, is .. #apple #aapl	0.000000	0.000000	Neutral	2021-09-20 22:15:53
33365	2021-09-20 23:15:03	now, if only any of us outside the us could ...	0.395139	-0.010417	Negative	2021-09-20 23:15:03
33366	Datetime	text	0.000000	0.000000	Neutral	NaT
33367	Datetime	text	0.000000	0.000000	Neutral	NaT

33367 rows × 6 columns

```
df2 = pd.get_dummies(df['Analysis'])
df2
```

	Negative	Neutral	Positive
1	0	0	1
2	0	0	1
3	0	0	1
4	0	0	1
5	0	1	0
...	...	...	...
33363	0	1	0
33364	0	1	0
33365	1	0	0
33366	0	1	0
33367	0	1	0

33367 rows × 3 columns

```
df
```

	Datetime	Text	Subjectivity	Polarity	Analysis	Date
1	2020-01-01 00:24:13	&amp; hour chas for all instruments are av...	0.400000	0.400000	Positive	2020-01-01 00:24:13
2	2020-01-01 00:56:57	top trades of the decade: number : going long...	0.450000	0.218750	Positive	2020-01-01 00:56:57
3	2020-01-01 01:24:41	&amp; hour chas for the group instruments a...	0.400000	0.400000	Positive	2020-01-01 01:24:41
4	2020-01-01 01:26:34	&amp; hour chas for all instruments are ava...	0.400000	0.400000	Positive	2020-01-01 01:26:34
5	2020-01-01 01:30:11	made one big mistake day trading apple (aapl) ...	0.100000	0.000000	Neutral	2020-01-01 01:30:11
...	...	...	...	...	...	...
33363	2021-09-20 22:10:27	aapl weekly price pattern coordinates\nannota...	0.000000	0.000000	Neutral	2021-09-20 22:10:27
33364	2021-09-20 22:15:53	apple inc price at close, --, is .. #apple #aapl	0.000000	0.000000	Neutral	2021-09-20 22:15:53
33365	2021-09-20 23:15:03	now, if only any of us outside the us could ...	0.395139	-0.010417	Negative	2021-09-20 23:15:03
33366	Datetime	text	0.000000	0.000000	Neutral	NaT
33367	Datetime	text	0.000000	0.000000	Neutral	NaT

33367 rows × 6 columns

```
df3 = pd.concat([df,df2],axis=1)
df3
```

	Datetime	Text	Subjectivity	Polarity	Analysis	Date	Negative	Neutral	Positive
1	2020-01-01 00:24:13	&amp; hour chas for all instruments are av...	0.400000	0.400000	Positive	2020-01-01 00:24:13	0	0	1
2	2020-01-01 00:56:57	top trades of the decade: number : going long...	0.450000	0.218750	Positive	2020-01-01 00:56:57	0	0	1
3	2020-01-01 01:24:41	&amp; hour chas for the group instruments a...	0.400000	0.400000	Positive	2020-01-01 01:24:41	0	0	1
4	2020-01-01 01:26:34	&amp; hour chas for all instruments are ava...	0.400000	0.400000	Positive	2020-01-01 01:26:34	0	0	1
5	2020-01-01 01:30:11	made one big mistake day trading apple (aapl) ...	0.100000	0.000000	Neutral	2020-01-01 01:30:11	0	1	0
...	...	...	...	...	...	...	...	...	...

```
df3 = df3.drop(['Datetime'],axis=1)
df3.head()
```

	Text	Subjectivity	Polarity	Analysis	Date	Negative	Neutral	Positive
1	&amp; hour chas for all instruments are av...	0.40	0.40000	Positive	2020-01-01 00:24:13	0	0	1
2	top trades of the decade: number : going long...	0.45	0.21875	Positive	2020-01-01 00:56:57	0	0	1
3	&amp; hour chas for the group instruments a...	0.40	0.40000	Positive	2020-01-01 01:24:41	0	0	1
4	&amp; hour chas for all instruments are ava...	0.40	0.40000	Positive	2020-01-01 01:26:34	0	0	1
5	made one big mistake day trading apple (aapl) ...	0.10	0.00000	Neutral	2020-01-01 01:30:11	0	1	0

```
df3['Year'] = df3['Date'].dt.year
df3['Month'] = df3['Date'].dt.month
df3['Day'] = df3['Date'].dt.day
df3
```

	Text	Subjectivity	Polarity	Analysis	Date	Negative	Neutral	Positive	Year	Month	Day
1	&amp; hour chas for all instruments are av...	0.400000	0.400000	Positive	2020-01-01 00:24:13	0	0	1	2020.0	1.0	1.0
2	top trades of the decade: number : going long...	0.450000	0.218750	Positive	2020-01-01 00:56:57	0	0	1	2020.0	1.0	1.0
3	&amp; hour chas for the group instruments a...	0.400000	0.400000	Positive	2020-01-01 01:24:41	0	0	1	2020.0	1.0	1.0
4	&amp; hour chas for all instruments are ava...	0.400000	0.400000	Positive	2020-01-01 01:26:34	0	0	1	2020.0	1.0	1.0
5	made one big mistake day trading apple (aapl) ...	0.100000	0.000000	Neutral	2020-01-01 01:30:11	0	1	0	2020.0	1.0	1.0
...	...	...	...	...	...	...	...	...	...	...	...
33363	aapl weekly price pattern coordinates\nannota...	0.000000	0.000000	Neutral	2021-09-20 22:10:27	0	1	0	2021.0	9.0	20.0
33364	apple inc price at close, --, is .. #apple #aapl	0.000000	0.000000	Neutral	2021-09-20 22:15:53	0	1	0	2021.0	9.0	20.0
33365	now, if only any of us outside the us could ...	0.395139	-0.010417	Negative	2021-09-20 23:15:03	1	0	0	2021.0	9.0	20.0
33366	text	0.000000	0.000000	Neutral	NaT	0	1	0	NaN	NaN	NaN
33367	text	0.000000	0.000000	Neutral	NaT	0	1	0	NaN	NaN	NaN

33367 rows × 11 columns

```
data2020 = df3.drop(['Text', 'Subjectivity','Polarity', 'Analysis', 'Date', 'Negative', 'Neutral',  'Positive'],axis=1)
data2020
```

	Year	Month	Day
1	2020.0	1.0	1.0
2	2020.0	1.0	1.0
3	2020.0	1.0	1.0
4	2020.0	1.0	1.0
5	2020.0	1.0	1.0
...	...	...	...
33363	2021.0	9.0	20.0
33364	2021.0	9.0	20.0
33365	2021.0	9.0	20.0
33366	NaN	NaN	NaN
33367	NaN	NaN	NaN

33367 rows × 3 columns

```
df3.insert(3, 'StockName', 'APPLE')
df3
```

	Text	Subjectivity	Polarity	StockName	Analysis	Date	Negative	Neutral	Positive	Year	Month	Day
1	&amp; hour chas for all instruments are av...	0.400000	0.400000	APPLE	Positive	2020-01-01 00:24:13	0	0	1	2020.0	1.0	1.0
2	top trades of the decade: number : going long...	0.450000	0.218750	APPLE	Positive	2020-01-01 00:56:57	0	0	1	2020.0	1.0	1.0
3	&amp; hour chas for the group instruments a...	0.400000	0.400000	APPLE	Positive	2020-01-01 01:24:41	0	0	1	2020.0	1.0	1.0
4	&amp; hour chas for all instruments are ava...	0.400000	0.400000	APPLE	Positive	2020-01-01 01:26:34	0	0	1	2020.0	1.0	1.0
5	made one big mistake day trading apple (aapl) ...	0.100000	0.000000	APPLE	Neutral	2020-01-01 01:30:11	0	1	0	2020.0	1.0	1.0
...	...	...	...	...	...	...	...	...	...	...	...	...
33363	aapl weekly price pattern coordinates\nannota...	0.000000	0.000000	APPLE	Neutral	2021-09-20 22:10:27	0	1	0	2021.0	9.0	20.0
33364	apple inc price at close, --, is .. #apple #aapl	0.000000	0.000000	APPLE	Neutral	2021-09-20 22:15:53	0	1	0	2021.0	9.0	20.0
33365	now, if only any of us outside the us could ...	0.395139	-0.010417	APPLE	Negative	2021-09-20 23:15:03	1	0	0	2021.0	9.0	20.0
33366	text	0.000000	0.000000	APPLE	Neutral	NaT	0	1	0	NaN	NaN	NaN
33367	text	0.000000	0.000000	APPLE	Neutral	NaT	0	1	0	NaN	NaN	NaN

33367 rows × 12 columns

```
df4 = df3.drop(['Text', 'Subjectivity', 'Polarity','Analysis'], axis = 1)
df4
```

	StockName	Date	Negative	Neutral	Positive	Year	Month	Day
1	APPLE	2020-01-01 00:24:13	0	0	1	2020.0	1.0	1.0
2	APPLE	2020-01-01 00:56:57	0	0	1	2020.0	1.0	1.0
3	APPLE	2020-01-01 01:24:41	0	0	1	2020.0	1.0	1.0
4	APPLE	2020-01-01 01:26:34	0	0	1	2020.0	1.0	1.0
5	APPLE	2020-01-01 01:30:11	0	1	0	2020.0	1.0	1.0
...	...	...	...	...	...	...	...	...
33363	APPLE	2021-09-20 22:10:27	0	1	0	2021.0	9.0	20.0
33364	APPLE	2021-09-20 22:15:53	0	1	0	2021.0	9.0	20.0
33365	APPLE	2021-09-20 23:15:03	1	0	0	2021.0	9.0	20.0
33366	APPLE	NaT	0	1	0	NaN	NaN	NaN

```
df5 = df4.drop(['Year','Month','Day'], axis = 1)
df5
```

	StockName	Date	Negative	Neutral	Positive
1	APPLE	2020-01-01 00:24:13	0	0	1
2	APPLE	2020-01-01 00:56:57	0	0	1
3	APPLE	2020-01-01 01:24:41	0	0	1
4	APPLE	2020-01-01 01:26:34	0	0	1
5	APPLE	2020-01-01 01:30:11	0	1	0
...	...	...	...	...	...
33363	APPLE	2021-09-20 22:10:27	0	1	0
33364	APPLE	2021-09-20 22:15:53	0	1	0
33365	APPLE	2021-09-20 23:15:03	1	0	0
33366	APPLE	NaT	0	1	0
33367	APPLE	NaT	0	1	0

33367 rows × 5 columns

```
df6 = df5.groupby([df4['Date'].dt.date]).sum()
df6
```

	Negative	Neutral	Positive
Date			
2020-01-01	2.0	8.0	10.0
2020-01-02	11.0	31.0	42.0
2020-01-03	11.0	12.0	21.0
2020-01-04	1.0	8.0	3.0
2020-01-05	1.0	9.0	4.0
...	...	...	...
2021-09-16	6.0	14.0	19.0
2021-09-17	8.0	14.0	9.0
2021-09-18	3.0	5.0	13.0
2021-09-19	3.0	9.0	12.0
2021-09-20	13.0	22.0	21.0

627 rows × 3 columns

```
df6.head()
```

	Negative	Neutral	Positive
Date			
2020-01-01	2.0	8.0	10.0
2020-01-02	11.0	31.0	42.0
2020-01-03	11.0	12.0	21.0
2020-01-04	1.0	8.0	3.0
2020-01-05	1.0	9.0	4.0

```
df6['Total Tweets'] = df6['Positive']+  df6['Negative']+  df6['Neutral']
df6
```

Negative Neutral Positive Total Tweets

```
df7 = df6.reset_index()
df7
```

	Date	Negative	Neutral	Positive	Total Tweets
0	2020-01-01	2.0	8.0	10.0	20.0
1	2020-01-02	11.0	31.0	42.0	84.0
2	2020-01-03	11.0	12.0	21.0	44.0
3	2020-01-04	1.0	8.0	3.0	12.0
4	2020-01-05	1.0	9.0	4.0	14.0
...	...	...	...	...	...
622	2021-09-16	6.0	14.0	19.0	39.0
623	2021-09-17	8.0	14.0	9.0	31.0
624	2021-09-18	3.0	5.0	13.0	21.0
625	2021-09-19	3.0	9.0	12.0	24.0
626	2021-09-20	13.0	22.0	21.0	56.0

627 rows × 5 columns

```
df8 = df7[['Date','Positive','Neutral' , 'Negative','Total Tweets']]
df8
```

	Date	Positive	Neutral	Negative	Total Tweets
0	2020-01-01	10.0	8.0	2.0	20.0
1	2020-01-02	42.0	31.0	11.0	84.0
2	2020-01-03	21.0	12.0	11.0	44.0
3	2020-01-04	3.0	8.0	1.0	12.0
4	2020-01-05	4.0	9.0	1.0	14.0
...	...	...	...	...	...
622	2021-09-16	19.0	14.0	6.0	39.0
623	2021-09-17	9.0	14.0	8.0	31.0
624	2021-09-18	13.0	5.0	3.0	21.0
625	2021-09-19	12.0	9.0	3.0	24.0
626	2021-09-20	21.0	22.0	13.0	56.0

627 rows × 5 columns

```
df8['Date']=pd.to_datetime(df8.Date, format='%Y/%m/%d')
df8['Year'] = df8['Date'].dt.year
df8['Month'] = df8['Date'].dt.month
df8['Day'] = df8['Date'].dt.day
df8
```

	Date	Positive	Neutral	Negative	Total Tweets	Year	Month	Day
0	2020-01-01	10.0	8.0	2.0	20.0	2020	1	1
1	2020-01-02	42.0	31.0	11.0	84.0	2020	1	2
2	2020-01-03	21.0	12.0	11.0	44.0	2020	1	3
3	2020-01-04	3.0	8.0	1.0	12.0	2020	1	4
4	2020-01-05	4.0	9.0	1.0	14.0	2020	1	5
...	...	...	...	...	...	...	...	...
622	2021-09-16	19.0	14.0	6.0	39.0	2021	9	16
623	2021-09-17	9.0	14.0	8.0	31.0	2021	9	17
624	2021-09-18	13.0	5.0	3.0	21.0	2021	9	18
625	2021-09-19	12.0	9.0	3.0	24.0	2021	9	19
626	2021-09-20	21.0	22.0	13.0	56.0	2021	9	20

627 rows × 8 columns

```
df9 = df8.drop(['Date'],axis=1)
df9
```

	Positive	Neutral	Negative	Total Tweets	Year	Month	Day
0	10.0	8.0	2.0	20.0	2020	1	1
1	42.0	31.0	11.0	84.0	2020	1	2
2	21.0	12.0	11.0	44.0	2020	1	3
3	3.0	8.0	1.0	12.0	2020	1	4
4	4.0	9.0	1.0	14.0	2020	1	5
...	...	...	...	...	...	...	...
622	19.0	14.0	6.0	39.0	2021	9	16
623	9.0	14.0	8.0	31.0	2021	9	17
624	13.0	5.0	3.0	21.0	2021	9	18
625	12.0	9.0	3.0	24.0	2021	9	19
626	21.0	22.0	13.0	56.0	2021	9	20

627 rows × 7 columns

```
df10 = df9[['Year', 'Month', 'Day', 'Positive','Negative','Neutral','Total Tweets']]

df10.insert(3, 'StockName', 'APPLE')
df10
```

	Year	Month	Day	StockName	Positive	Negative	Neutral	Total Tweets
0	2020	1	1	APPLE	10.0	2.0	8.0	20.0
1	2020	1	2	APPLE	42.0	11.0	31.0	84.0
2	2020	1	3	APPLE	21.0	11.0	12.0	44.0
3	2020	1	4	APPLE	3.0	1.0	8.0	12.0
4	2020	1	5	APPLE	4.0	1.0	9.0	14.0
...	...	...	...	...	...	...	...	...
622	2021	9	16	APPLE	19.0	6.0	14.0	39.0
623	2021	9	17	APPLE	9.0	8.0	14.0	31.0
624	2021	9	18	APPLE	13.0	3.0	5.0	21.0
625	2021	9	19	APPLE	12.0	3.0	9.0	24.0
626	2021	9	20	APPLE	21.0	13.0	22.0	56.0

627 rows × 8 columns

```
df10.to_csv('Apple_final.csv')
```

```
df10['Date']=pd.to_datetime(df8.Date, format='%Y/%m/%d')
df10
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  """Entry point for launching an IPython kernel.
```

	Year	Month	Day	StockName	Positive	Negative	Neutral	Total Tweets	Date
0	2020	1	1	APPLE	10.0	2.0	8.0	20.0	2020-01-01
1	2020	1	2	APPLE	42.0	11.0	31.0	84.0	2020-01-02
2	2020	1	3	APPLE	21.0	11.0	12.0	44.0	2020-01-03
3	2020	1	4	APPLE	3.0	1.0	8.0	12.0	2020-01-04
4	2020	1	5	APPLE	4.0	1.0	9.0	14.0	2020-01-05
...	...	...	...	...	...	...	...	...	...
622	2021	9	16	APPLE	19.0	6.0	14.0	39.0	2021-09-16
623	2021	9	17	APPLE	9.0	8.0	14.0	31.0	2021-09-17
624	2021	9	18	APPLE	13.0	3.0	5.0	21.0	2021-09-18
625	2021	9	19	APPLE	12.0	3.0	9.0	24.0	2021-09-19
626	2021	9	20	APPLE	21.0	13.0	22.0	56.0	2021-09-20

627 rows × 9 columns

Merging with the stock data

```
df11 = pd.read_csv('/content/sample_data/AppleStockData20_21.csv')
df11
```

	Date	Close/Last	Volume	Open	High	Low
0	09/21/2021	$143.43	75833960	$143.93	$144.6	$142.78
1	09/20/2021	$142.94	123478900	$143.8	$144.84	$141.27
2	09/17/2021	$146.06	129868800	$148.82	$148.82	$145.76
3	09/16/2021	$148.79	68034150	$148.44	$148.97	$147.221
4	09/15/2021	$149.03	83281320	$148.56	$149.44	$146.37
...	...	...	...	...	...	...
429	01-08-2020	$75.7975	132363800	$74.29	$76.11	$74.289
430	01-07-2020	$74.5975	111510640	$74.96	$75.225	$74.37
431	01-06-2020	$74.95	118578560	$73.4475	$74.99	$73.1875
432	01-03-2020	$74.3575	146535520	$74.2875	$75.145	$74.125
433	01-02-2020	$75.0875	135647440	$74.06	$75.15	$73.7975

434 rows × 6 columns

```
df11['Date'] = pd.to_datetime(df11['Date']).apply(lambda x: x.strftime('%Y-%m-%d')if not pd.isnull(x) else '')
df11
```

	Date	Close/Last	Volume	Open	High	Low
0	2021-09-21	$143.43	75833960	$143.93	$144.6	$142.78
1	2021-09-20	$142.94	123478900	$143.8	$144.84	$141.27
2	2021-09-17	$146.06	129868800	$148.82	$148.82	$145.76
3	2021-09-16	$148.79	68034150	$148.44	$148.97	$147.221
4	2021-09-15	$149.03	83281320	$148.56	$149.44	$146.37
...	...	...	...	...	...	...
429	2020-01-08	$75.7975	132363800	$74.29	$76.11	$74.289
430	2020-01-07	$74.5975	111510640	$74.96	$75.225	$74.37
431	2020-01-06	$74.95	118578560	$73.4475	$74.99	$73.1875
432	2020-01-03	$74.3575	146535520	$74.2875	$75.145	$74.125
433	2020-01-02	$75.0875	135647440	$74.06	$75.15	$73.7975

434 rows × 6 columns

```
df11['Date'] = pd.to_datetime(df11.Date, errors='coerce',format='%Y/%m/%d')
```

```
df11.rename(columns = {'Close/Last':'Close'}, inplace = True)
```

```
merge_left = pd.merge(df10, df11, on = 'Date', how='left')
merge_left.set_index('Date', inplace = True)
```

```
merge_left.reset_index(inplace=True)
merge_left
```

	Date	Year	Month	Day	StockName	Positive	Negative	Neutral	Total Tweets	Close	Volume	Open	High	Low
0	2020-01-01	2020	1	1	APPLE	10.0	2.0	8.0	20.0	NaN	NaN	NaN	NaN	NaN
1	2020-01-02	2020	1	2	APPLE	42.0	11.0	31.0	84.0	$75.0875	135647440.0	$74.06	$75.15	$73.7975
2	2020-01-03	2020	1	3	APPLE	21.0	11.0	12.0	44.0	$74.3575	146535520.0	$74.2875	$75.145	$74.125
3	2020-01-04	2020	1	4	APPLE	3.0	1.0	8.0	12.0	NaN	NaN	NaN	NaN	NaN
4	2020-01-05	2020	1	5	APPLE	4.0	1.0	9.0	14.0	NaN	NaN	NaN	NaN	NaN
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
622	2021-09-16	2021	9	16	APPLE	19.0	6.0	14.0	39.0	$148.79	68034150.0	$148.44	$148.97	$147.221
623	2021-09-17	2021	9	17	APPLE	9.0	8.0	14.0	31.0	$146.06	129868800.0	$148.82	$148.82	$145.76
624	2021-09-18	2021	9	18	APPLE	13.0	3.0	5.0	21.0	NaN	NaN	NaN	NaN	NaN
625	2021-09-19	2021	9	19	APPLE	12.0	3.0	9.0	24.0	NaN	NaN	NaN	NaN	NaN
626	2021-09-20	2021	9	20	APPLE	21.0	13.0	22.0	56.0	$142.94	123478900.0	$143.8	$144.84	$141.27

627 rows × 14 columns

```
merge_left['Close'] = merge_left['Close'].astype(str).map(lambda x: x.lstrip('$'))
merge_left['Open'] = merge_left['Open'].astype(str).map(lambda x: x.lstrip('$'))
merge_left['Low'] = merge_left['Low'].astype(str).map(lambda x: x.lstrip('$'))
merge_left['High'] = merge_left['High'].astype(str).map(lambda x: x.lstrip('$'))
merge_left['dayOfWeek'] = merge_left['Date'].dt.day_name()
merge_2 = merge_left.sort_index(ascending=True)

#Add stock name column = APPL
merge_2['Stock name']= 'APPL'
colum_names = ['Stock name','Date','Year','Month','Day','dayOfWeek','Close','Open','High','Low','Volume','Positive','Negative','Neutral','Total Tweets']
merge_3 = merge_2.reindex(columns=colum_names)
merge_3['Close'] = merge_3['Close'].astype(float)
merge_3['Open'] = merge_3['Open'].astype(float)
merge_3['High'] = merge_3['High'].astype(float)
merge_3['Low'] = merge_3['Low'].astype(float)
merge_4 = merge_3.interpolate(method = 'linear', limit_direction='backward')
merge_4
```

	Stock name	Date	Year	Month	Day	dayOfWeek	Close	Open	High	Low	Volume	Positive	Negative	Neutral	Total Tweets
0	APPL	2020-01-01	2020	1	1	Wednesday	75.0875	74.060000	75.150000	73.797500	1.356474e+08	10.0	2.0	8.0	20.0
1	APPL	2020-01-02	2020	1	2	Thursday	75.0875	74.060000	75.150000	73.797500	1.356474e+08	42.0	11.0	31.0	84.0
2	APPL	2020-01-03	2020	1	3	Friday	74.3575	74.287500	75.145000	74.125000	1.465355e+08	21.0	11.0	12.0	44.0
3	APPL	2020-01-04	2020	1	4	Saturday	74.5550	74.007500	75.093333	73.812500	1.372165e+08	3.0	1.0	8.0	12.0
4	APPL	2020-01-05	2020	1	5	Sunday	74.7525	73.727500	75.041667	73.500000	1.278975e+08	4.0	1.0	9.0	14.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
622	APPL	2021-09-16	2021	9	16	Thursday	148.7900	148.440000	148.970000	147.221000	6.803415e+07	19.0	6.0	14.0	39.0
623	APPL	2021-09-17	2021	9	17	Friday	146.0600	148.820000	148.820000	145.760000	1.298688e+08	9.0	8.0	14.0	31.0
624	APPL	2021-09-18	2021	9	18	Saturday	145.0200	147.146667	147.493333	144.263333	1.277388e+08	13.0	3.0	5.0	21.0
625	APPL	2021-09-19	2021	9	19	Sunday	143.9800	145.473333	146.166667	142.766667	1.256089e+08	12.0	3.0	9.0	24.0
626	APPL	2021-09-20	2021	9	20	Monday	142.9400	143.800000	144.840000	141.270000	1.234789e+08	21.0	13.0	22.0	56.0

627 rows × 15 columns

```
merge_4.to_csv('final_merge.csv')
```

✓ 0s completed at 1:56 AM