```
import pandas as pd
import numpy as np

import matplotlib.pyplot as plt
import plotly.express as px
import seaborn as sns

import nltk
import string
#from itertools import chain
from collections import Counter
from wordcloud import WordCloud


from tensorflow.keras.preprocessing.text import Tokenizer
from nltk.tokenize import RegexpTokenizer
from nltk.corpus import stopwords



import nltk
nltk.download('stopwords')
```

```
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
True
```

```
data = pd.read_csv('/content/Corona_NLP_test.csv')
```

```
data.head()
```

	UserName	ScreenName	Location	TweetAt	OriginalTweet	Sentiment
0	1	44953	NYC	02-03-2020	TRENDING: New Yorkers encounter empty supermar...	Extremely Negative
1	2	44954	Seattle, WA	02-03-2020	When I couldn't find hand sanitizer at Fred Me...	Positive
2	3	44955	NaN	02-03-2020	Find out how you can protect yourself and love...	Extremely Positive
3	4	44956	Chicagoland	02-03-2020	#Panic buying hits #NewYork City as anxious sh...	Negative
4	5	44957	Melbourne, Victoria	03-03-2020	#toiletpaper #dunnypaper #coronavirus #coronav...	Neutral

```
data.isnull().sum()
```

```
UserName           0
ScreenName         0
Location         834
TweetAt            0
OriginalTweet      0
Sentiment          0
dtype: int64
```

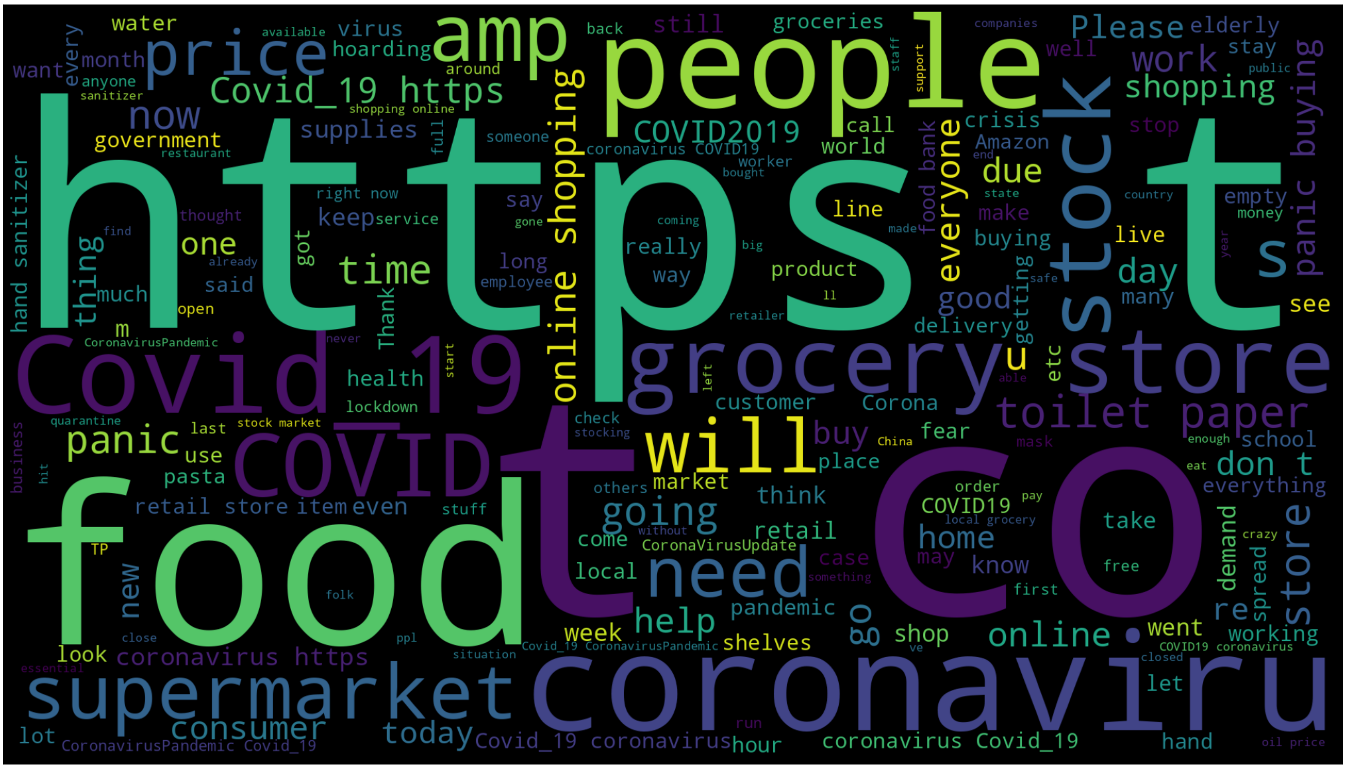
```
data['OriginalTweet'] = data['OriginalTweet']
```

```
stop_words = set(stopwords.words('english'))
to_remove = ['•', '!', '"', '#', '”', '“', '$', '%', '&', "'", '–', '(', ')', '*', '+', ',', '-', '.', '/', ':', ';', '<', '=', '>', '?'
stop_words.update(to_remove)
print('Number of stopwords:', len(stop_words))

def preprocess_text(OriginalTweet):
    OriginalTweet = OriginalTweet.lower()
    OriginalTweet = re.sub(r'http\S+', '', OriginalTweet)
    OriginalTweet = re.sub('\[[^]]*\]', '', OriginalTweet)
    OriginalTweet = (" ").join([word for word in OriginalTweet.split() if not word in stop_words])
    OriginalTweet = "".join([char for char in OriginalTweet if not char in to_remove])
    return OriginalTweet
```

```
 Number of stopwords: 226
```

```
import matplotlib.pyplot as plt
from wordcloud import WordCloud

# define the text data to be analyzed

OriginalTweet = ' '.join(data['OriginalTweet'].tolist())

# generate the word cloud
```

```python
wordcloud = WordCloud(width=1920, height=1080).generate(OriginalTweet)

# display the word cloud
fig = plt.figure(figsize=(20, 20), facecolor='k', edgecolor='k')
plt.imshow(wordcloud)
plt.axis('off')
plt.tight_layout(pad=0)
plt.show()
```




```python
mostrepeated = Counter(" ".join(data["OriginalTweet"]).split()).most_common(100)
mostrepeated
```

```
[('the', 3848),
 ('to', 3655),
 ('and', 2337),
 ('of', 2035),
 ('a', 1706),
 ('in', 1685),
 ('#Covid_19', 1383),
 ('for', 1277),
 ('is', 1245),
 ('I', 1070),
 ('are', 1067),
 ('#coronavirus', 1021),
 ('on', 993),
 ('food', 902),
 ('you', 874),
 ('at', 861),
 ('grocery', 749),
 ('store', 736),
 ('up', 660),
 ('have', 651),
 ('be', 647),
 ('that', 637),
 ('stock', 628),
 ('people', 606),
 ('this', 579),
 ('with', 564),
 ('or', 541),
 ('all', 514),
 ('&amp;', 510),
 ('your', 500),
 ('will', 486),
 ('my', 454),
 ('not', 443),
 ('out', 434),
 ('from', 432),
 ('it', 424),
 ('as', 415),
 ('we', 408),
```

```
 ('shopping', 395),
 ('online', 381),
 ('The', 361),
 ('they', 355),
 ('about', 354),
 ('supermarket', 345),
 ('can', 345),
 ('panic', 333),
 ('toilet', 326),
 ('but', 324),
 ('prices', 315),
 ('need', 314),
 ('our', 311),
 ('?', 305),
 ('-', 302),
 ('if', 302),
 ('like', 300),
 ('get', 298),
 ('their', 296),
 ('has', 292),
```

```
X = [d.split() for d in data['OriginalTweet'].tolist()]
```

```
print(X[0])
```

```
['TRENDING:', 'New', 'Yorkers', 'encounter', 'empty', 'supermarket', 'shelves', '(pictured,', 'Wegmans', 'in', 'Brooklyn),', 'sold-
```

```
# Tokenising the model
tokenizer = Tokenizer()
tokenizer.fit_on_texts(X)
X = tokenizer.texts_to_sequences(X)
```

```
X
```

```
[[6369,
  155,
  3048,
  4097,
  143,
  44,
  96,
  6370,
  4098,
  5,
  6371,
  4099,
  41,
  1219,
  6372,
  6373,
  37,
  6374,
  433,
  20,
  24,
  6375,
  6376],
 [81,
  11,
  1033,
  235,
  115,
  243,
  16,
  4100,
  6377,
  11,
  2469,
  2,
  6378,
  46,
  6379,
  8,
  6,
  146,
  955,
  4,
  6380,
  38,
  69,
  10,
  628,
  12,
  1345,
  24,
```

```
 351,
 6381],
[235, 38, 69, 15, 49, 416, 728, 3, 2081, 770, 40, 366, 58],
[889,
 72,
 2082,
 2083.
```

```
tokenizer.word_index
```

```
{'the': 1,
 'to': 2,
 'and': 3,
 'of': 4,
 'in': 5,
 'a': 6,
 '#covid_19': 7,
 'for': 8,
 'is': 9,
 '#coronavirus': 10,
 'i': 11,
 'are': 12,
 'on': 13,
 'food': 14,
 'you': 15,
 'at': 16,
 'grocery': 17,
 'this': 18,
 'store': 19,
 'stock': 20,
 'be': 21,
 'people': 22,
 'have': 23,
 'up': 24,
 'that': 25,
 'with': 26,
 'we': 27,
 'all': 28,
 'or': 29,
 'my': 30,
 'your': 31,
 'it': 32,
 'if': 33,
 'not': 34,
 '&amp;': 35,
 'will': 36,
 'as': 37,
 'out': 38,
 'they': 39,
 'from': 40,
 'online': 41,
 'shopping': 42,
 'no': 43,
 'supermarket': 44,
 'panic': 45,
 'but': 46,
 'just': 47,
 'about': 48,
 'can': 49,
 'so': 50,
 'covid-19': 51,
 'our': 52,
 'toilet': 53,
 'prices': 54,
 'need': 55,
 'get': 56,
 'like': 57,
 '?': 58,
```