```
#upload dataset
from google.colab import files
uploaded = files.upload()
```

Choose Files tweets.csv

- **tweets.csv**(text/csv) - 1615005 bytes, last modified: 11/12/2020 - 100% done

```
Saving tweets.csv to tweets.csv
```

```
#reading dataset
import pandas as pd
df=pd.read_csv("tweets.csv")
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 11370 entries, 0 to 11369
Data columns (total 5 columns):
 #   Column    Non-Null Count  Dtype
---  ------    --------------  -----
 0   id        11370 non-null  int64
 1   keyword   11370 non-null  object
 2   location  7952 non-null   object
 3   text      11370 non-null  object
 4   target    11370 non-null  int64
dtypes: int64(2), object(3)
memory usage: 444.3+ KB
```

```
#checking for null values
df.isnull()
```

	id	keyword	location	text	target
0	False	False	True	False	False
1	False	False	True	False	False
2	False	False	False	False	False
3	False	False	False	False	False
4	False	False	True	False	False
...	...	...	...	...	...
11365	False	False	False	False	False
11366	False	False	False	False	False
11367	False	False	False	False	False
11368	False	False	False	False	False
11369	False	False	True	False	False

11370 rows × 5 columns

```
#count of null values
df.isnull().sum()
```

	0
id	0
keyword	0
location	3418
text	0
target	0

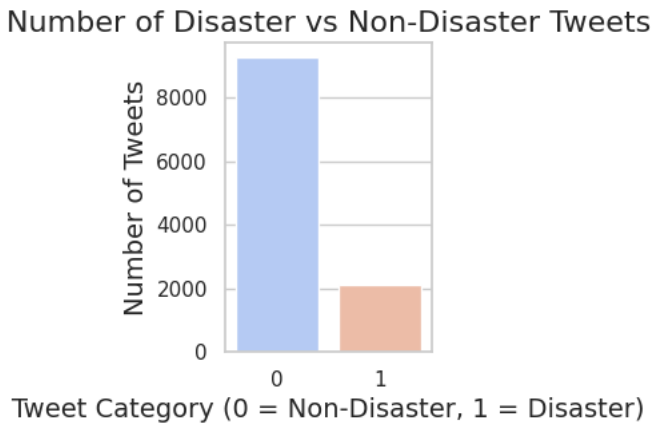
dtype: int64

```
#To analyze numerical data
df.describe()
```

	id	target
count	11370.000000	11370.000000
mean	5684.500000	0.185928
std	3282.380615	0.389066
min	0.000000	0.000000
25%	2842.250000	0.000000
50%	5684.500000	0.000000
75%	8526.750000	0.000000
max	11369.000000	1.000000

```
#data exploration and visualization
import seaborn as sns
import matplotlib.pyplot as plt

sns.set(style="whitegrid")
plt.figure(figsize=(2,3))
sns.countplot(x='target', data=df, palette='coolwarm')
plt.title('Number of Disaster vs Non-Disaster Tweets', fontsize=16)
plt.xlabel('Tweet Category (0 = Non-Disaster, 1 = Disaster)', fontsize=14)
plt.ylabel('Number of Tweets', fontsize=14)

plt.show()
```

```
<ipython-input-51-409f21b90e71>:7: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable to `hue`

  sns.countplot(x='target', data=df, palette='coolwarm')
```




```
# Required Libraries
from wordcloud import WordCloud
import matplotlib.pyplot as plt

disaster_tweets = df[df['target'] == 1]['text']  # Filter disaster-related tweets

# Combine all tweets into one string
all_disaster_tweets = ' '.join(disaster_tweets)

# Generate Word Cloud
wordcloud = WordCloud(width=800, height=400, background_color='white').generate(all_disaster_tweets)

# Display the word cloud
plt.figure(figsize=(5, 3))
plt.imshow(wordcloud, interpolation='bilinear')
plt.axis('off')
plt.title('Word Cloud of Disaster-Related Tweets')
plt.show()
```

Start coding or generate with AI.

```
#importing necessary libraries
import re
import nltk
from sklearn.model_selection import train_test_split

#Differentiating features and target
X = df[["id", "keyword", "location", "text"]]  # Features
y = df["target"]  # Labels

# Split into training and test sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

```
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Package punkt is already up-to-date!
             id                                               text
3912    3912  [why, the, hell, would, want, to, join, the, K...
5902    5902  [Citizens, United, wreaked, havoc, on, our, de...
11305  11305  [Through, all, the, happiness, and, sorrow, ,,...
3691    3691  [Remember, when, this, cheer, derailed, the, c...
11340  11340  [My, first, listen, was, also, in, the, whip, ...
```

```
#Function to remove URLs
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
def Remove_Url(string):
    return re.sub(r'(https|http)?:\/\/(\w|\.|\/|\?|\=|\&|\%|\-)*\b', '', string)
```

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

```
#remvoing URLs and displaying
print("Before removing URL: \n", X_train['text'][22], end = "\n\n")
X_train['text'] = X_train['text'].apply(Remove_Url)
print("After removing URL: \n", X_train['text'][22])
```

```
Before removing URL:
 #ThankfulTuesday Isaiah 43:2 When you pass through the waters, I will be with you; and when you pass through the r… htt

After removing URL:
 #ThankfulTuesday Isaiah 43:2 When you pass through the waters, I will be with you; and when you pass through the r…
```

```
#importing necesarry package to remove emojis
!pip install demoji
import demoji

#function to remove emojies
demoji.download_codes()
def Handle_emoji(string):
    return demoji.replace_with_desc(string)
```

```
Requirement already satisfied: demoji in /usr/local/lib/python3.10/dist-packages (1.1.0)
<ipython-input-25-ef59f837539a>:4: FutureWarning: The demoji.download_codes attribute is deprecated and will be removed
  demoji.download_codes()
```

```
#Removing emojis and displaying
print("Before Handling Emojis: \n", X_train['text'][429],end = "\n\n")
X_train['text'] = X_train['text'].apply(Handle_emoji)
print("After Handling Emojis: \n", X_train['text'][429])
```

```
Before Handling Emojis:
 pre-order untuk Map of the Soul: 7 oleh ARMY China 🇨🇳 telah mencapai 230.192 copy 😮gileee gileee #BestFanArmy #BTSARMY

After Handling Emojis:
 pre-order untuk Map of the Soul: 7 oleh ARMY China :flag: China: telah mencapai 230.192 copy :face with open mouth:gile
```

```
#function to remove useless characters
def Remove_UC(string):
    thestring = re.sub(r'[^a-zA-Z\s]','', string)
    # remove word of length less than 2
    thestring = re.sub(r'\b\w{1,2}\b', '', thestring)
    #https://www.geeksforgeeks.org/python-remove-unwanted-spaces-from-string/
    return re.sub(' +', ' ', thestring)
```

```
#removing useless characters and displaying
print("Example of text before Removing Useless Character: \n", X_train['text'][17],end = "\n\n")
X_train['text'] = X_train['text'].apply(Remove_UC)
print("Example of text after Removing Useless Character: \n", X_train['text'][17])
```

```
Example of text before Removing Useless Character:
 Rengoku sets my heart ablaze:pensive face::red heart::fire: P.s. I missed this style of coloring I do so here it is c:

Example of text after Removing Useless Character:
 Rengoku sets heart ablazepensive facered heartfire missed this style coloring here
```

```
#importing necessary libraries to remove stopwords and stemming
from nltk.stem.snowball import SnowballStemmer
from nltk.corpus import stopwords
from nltk.stem.porter import PorterStemmer
ps = PorterStemmer()
nltk.download('punkt')
nltk.download('stopwords')
stemmer  = SnowballStemmer('english')
stopword = stopwords.words('english')

#Function to remove stop words and stemming
def Remove_StopAndStem(string):
    string_list = string.split()
    return ' '.join([stemmer.stem(i) for i in string_list if i not in stopword])
```

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

```
#removing stopwords and stemming
print("Example of text before Removing Stopwords: \n", X_train['text'][17],end = "\n\n")
X_train['text'] = X_train['text'].apply(Remove_StopAndStem)
print("Example of text after Removing Stopwords and Stemming: \n", X_train['text'][17])
```

```
Example of text before Removing Stopwords:
 Rengoku sets heart ablazepensive facered heartfire missed this style coloring here

Example of text after Removing Stopwords and Stemming:
 rengoku set heart ablazepens facer heartfir miss style color
```

```
#Tokenizing the data
from nltk.tokenize import word_tokenize

# Tokenize the 'text' column in place
X_train['text'] = X_train['text'].apply(lambda x: word_tokenize(x))

# Display the tokenized output
print(X_train[['id', 'text']].head())  # Show the first few tokenized rows
```

```
           id                                               text
3912    3912  [why, the, hell, would, want, to, join, the, K...
5902    5902  [Citizens, United, wreaked, havoc, on, our, de...
11305  11305  [Through, all, the, happiness, and, sorrow, ,,...
3691    3691  [Remember, when, this, cheer, derailed, the, c...
11340  11340  [My, first, listen, was, also, in, the, whip, ...
```

```
from sklearn.feature_extraction.text import CountVectorizer

count_vectorizer = CountVectorizer(max_features=5000, stop_words='english')
X_count = count_vectorizer.fit_transform(df['text'])

print(X_count.shape)
```

```
(11370, 5000)
```

Start coding or generate with AI.