Data Wrangling Process:

Data wrangling, which consists of:

- Gathering data
- Assessing data
- Cleaning data
- Storing, analyzing, and visualizing your wrangled data.
- Reporting on:
  - wrangling efforts
  - Data analyses and visualizations

Gathering Data for this Project:

1. Gather each of the three pieces of data:
   1. The Waterdogs Twitter archive: I downloaded the file manually from the website, the file name is twitter_archive_enhanced.csv

   2. The tweet image predictions: This file (image_predictions.tsv) is hosted on Udacity's servers and should be downloaded programmatically using the Requests library and the following URL: https://d17h27t6h515a5.cloudfront.net/topher/2017/August/599fd2ad_image-predictions/image-predictions.tsv

   3. Using the tweet IDs in the WeRateDogs Twitter archive, I query the Twitter API for each tweet's JSON data using Python's Tweepy library and store each tweet's entire set of JSON data in a file called tweet_json.txt file. Each tweet's JSON data is written to its own line. Then I read this .txt file line by line into a pandas DataFram

Assessing Data:

After gathering Data, I assessed the data both visually and programmatically for quality and tidiness. By using the pandas functions and methods we get:

.info(), .head(), .tail(), .duplicated(), .isnull(), and .value_counts()

Quality issues:

1. in archive: some rows has NaN expanded_urls
2. in archive: source column has extra string before and after
3. in archive: tweet_id have wrong type
4. in archive: empty values in dog_type
5. in archive: timestamp is string and it should be datetime
6. in image_prediction: some dogs names are upper case and others are lower case
7. in image_prediction: tweet_id have wrong type
8. in api: id instead of tweet_id
9. in api: tweet_id have wrong type

Tidiness issues:

1. in archive: columns do not need
2. in archive: too many columns for the dog type
3. in api: columns do not need
4. in image_predictions: columns do not need
5. in image_predictions: dog types has 9 columns and can be only one

Cleaning:

1. make a copy from each dataframe

   ```
   archive_clean = archive.copy()
   image_predictions_clean = image_predictions.copy()
   api_clean = api.copy()
   ```

2. I went through each issue to fix it starting with tidiness issues then quality issues
3. After cleaning some issues, I found other ones that need to be fixed so I wrote them in issues and fixed them