Reporting: Wrangle Report

Methods used in data wrangling for this analysis are gathering, assessing, and cleaning. This wrangling process was based on data gathered from the Twitter user **WeRateDogs**. Three different datasets were required to be obtained during the gathering stage. These datasets had to be imported to the workspace (Jupyter Notebook) using different methods.

Gathering:

The first dataset was provided on the Udacity project page to be downloaded manually to my local machine and read it to my notebook using the pandas function .read_csv(). The second dataset had to be downloaded programmatically using python's requests library with the URL that was provided. After downloading it programmatically, I read it to dataframe specifying the delimiter as a tab because it is a tsv file.

The third dataset had to be obtained using the tweepy library. I created a Twitter developer account, which gave me some access codes to help me query WeRateDogs page for a json file collecting the tweet id, favorite count, and retweet count but unfortunately it didn't work out. So I proceeded to use the json file provided by Udacity, downloading it manually. I created an empty list to hold the three attributes (columns). Using a loop, I run through the json file line by line to add the tweet id, favorite count, and retweet count to the empty list created earlier. After that, I converted the list to a dataframe, using the pandas function and specifying the column names.

Assessing and Cleaning:

After gathering all the datasets, I assessed all three individually manually, and programmatically. In the process of assessing, I looked out for quality and tidiness issues. I discovered that they all had quality issues by only two had tidiness issues.

Quality

Assessing	**Cleaning**
Tweet ids in all data frames were integers instead of it being a string.	Convert the column tweet_id to strings
There are columns in both twitter_archive and image_prediction dataframes that have a lot of missing data and are not needed.	Use pandas' .drop() method to drop columns
The name column in twitter_archive df does not have a consistent letter casing.	Use the .capitalize() function in pandas to iterate through the whole column to make changes
Rename the name column in the twitter_archive to a more specific name to help with column identification	Change column name to dog_name using the rename function.

Values in p1, p2, and p3 are not consistent in terms of letter casing	Change values in columns to lowercase using str.lower()
Timestamp column is an object datatype instead of a datetime datatype	Convert the timestamp column to datetime
The project description describes the ratings in the data to have a denominator of 10, but the dataset has 18 different denominators.	Change the existing values in rating_denominator column by assigning the column to 10
Twiiter_archive df should contain only original tweets.	Remove rows where retweets and replies are not null. This leaves only original tweets in our df. Drop retweets and replies columns.

Tidiness

Assessing	Cleaning
Doggo, floofer, pupper and puppo should be in one column.	Convert none values in all four columns to NaN. Create a new column dog_type to combine all columns by filling in nan values.
The dataframes are related to tweets and should be merged. Favorite count and retweet count should be part of the twitter_archive df which follows the rule each type of observational unit forms a table.	Join the two tables using the merge function with their common attribute 'tweet_id'.

After finishing my assessing and cleaning phase, I saved my final dataframes as a csv file.