# WeRateDogs Twitter Archive Wrangle Report

**For Udacity Nanodegree project**

**By Tenifayo Fagbemi**

This report outlines the steps carried out while wrangling the data required to analyze the WeRateDogs twitter archive.

These steps are:

- **Gathering Data**
  I gathered data from 3 different sources using different methods which are:
  1) twitter_enhanced_archive.csv which was manually downloaded from Udacity's server.
  2) image_predictions.tsv which was downloaded programmatically from Udacity's server using the Requests library.
  3) tweet_json.txt which was obtained by querying the Twitter API for each tweet's JSON data (tweet id, favourite count and retweet count) using python's tweepy library and storing it.

  After, I loaded them into dataframes df1, df2 and df3 respectively.

- **Assessing Data**
  I assessed all the three pieces of data visually and programmatically and detected the following quality and tidiness issues.

  **Quality issues**

  `df1` table,
  1. There are rows where retweeted_status_id and in_reply_to_status are not null. These tweets are either retweets or replies(which most likely do not have images).
  2. Some tweets have invalid rating_denominators (not equal to 10).
  3. Some rating_numerators are invalid.
  4. The source column contains html tags
  5. Some tweets have more than 1 dog stage.
  6. Erroneous datatypes of tweet_id and timestamp columns

  `df2` table
  7. Erroneous datatypes of tweet_id and img_num columns.
  8. Some tweets have images that do not contain dogs.

  `df3` table

9. Erroneous datatype of id column.

**Tidiness issues**

1. Dog stages are in four columns instead of one.

2. df1, df2 and df3 should form a single table.

- **Cleaning Data**
  Here I defined the issues listed above, wrote to clean these issues, and tested to make sure the issues were solved.

  1) As specified in the instructions, we only need original ratings and not retweets or replies. To remove these, I filtered out the rows where either `retweeted_status_id` column or `in_reply_to_status_id` was not null. After, I dropped the `retweeted_status_id`, `retweeted_status_user_id`, `retweeted_status_timestamp`,`in_reply_to_status_id` and `in_reply_to_user_id` columns since they were all null and were no longer useful
  2) The `rating_denominator` was supposed to be 10 but some were either greater or lesser than 10. I cleaned the ones that were combined rating for a group of dogs by dividing so that I could have 10 as the denominator and get the rating for just 1 dog. For the rating that was not correctly extracted, I checked the text and replaced it with the correct one. The rating denominators lesser than 10 were not dog ratings so I dropped the corresponding rows.
  3) For the `rating_numerator` greater than 17 which is supposed to be the highest, on checking the text, I realized that some were decimals which were cut off during extraction (like 11.25 was extracted as 25), so I replaced them with the correct rating and the others were not dog ratings, so I filtered them out. There was a rating numerator of 0 which was not a dog rating, so I filtered it out too.
  4) I extracted the source from the html tags in the `source` column using a regex pattern.
  5) For the rows with more than one dog stage, I picked one and set the other to "None".
  6) In the 3 tables, I changed the datatypes of `tweet_id` and id to string, `img_num` to category and `timestamp` to datetime.
  7) For rows where `p1_dog`, `p2_dog` and `p3_dog` was FALSE, it means the model didn't see any dogs in the images soI filtered them out. After I created two new columns: breed and conf for the breed with the higher confidence level. Then, I dropped the `p1`,`p1_conf`,`p1_dog`,`p2`,`p2_conf`,`p2_dog`,`p3`,`p3_conf` and `p3_dog` columns.

8) `doggo`, `floofer`, `puppo` and `pupper` should be in one dog_stage column. I created a dog_stage column and assign values to it based on the values in those columns and dropped them after.
9) Lastly, I merged all three tables together and dropped the `id and expanded_urls` columns

- **Storing Data**
  Here I saved the final dataframe to a CSV file named “twitter_archive_master.csv”.