# Wrangle Report

## Gather

- Twitter-archive table
  Use Pandas read_csv function to read in the csv file
- Image-prediction table
  Use response function to get the content of the table and read it in with Pandas read_csv function
- Tweet-json table
  Use tweepy api to extract information with given twitter id

## Access and Clean

Twitter-Archive table

- Quality Issue #1: drop the re-tweet rows
  Since we only care about the original tweet, I only kept rows where *retweeted_status_id* is *NaN*
- Quality Issue #2 Drop rows without pictures
  Use str.match function to flag rows which contains `https://twitter.com/dog_rates/status/../photo/1` in the `expanded_url` column and keep them. Also keep in mind there might be other strings before or after the photo url. I also replaced NaN with False value.
- Quality Issue #3: check if any tweets are after 1-Aug-2017
  Convert timestamp column to date-time object, then sort values to see if any time stamp is after 1-Aug-2017
- Quality Issue #4: some rating_numerator and rating_denominator are parsed incorrectly
  Part1: The rating_denominator should be 10 or multiples of 10. Drop rows whose rating_denominator are not multiples of 10.
  Part2: Created a new column rating_result as the result of rating_numerator devided by rating_denominator. Looked into rows whose rating_result are greater than 2.6 and found they were not either parsed incorrectly or were given a absurd number. It was safe to drop those five rows.
- Quality issue #5: tweet_id should be changed to data type str
  Tweet_id column need to be changed to data type str to match other tables because later we need to merge it with other table.
- Tidy issue #1: the columns 'doggo', 'floofer', 'pupper', 'puppo' are values of variable stage
  Replace ‘None’ with 0, 'doggo', 'floofer', 'pupper', 'puppo’ with 1.

There were 10 rows with multiple stage values. In this project, I am only going to analysis tweets with single stage so I flagged them as outliers and dropped them.
I created a new column ‘stage’ and assign the corresponding stage values to it.

- Tidy issue #2: create a new column and remove unnecessary columns
  Create a new column multi_dog to flag if there are more than one dog in the picture.
  Keep columns 'tweet_id', 'rating_result', 'stage', 'multi_dog' only for further processing.

## Tweet_json table

- Quality Issue #1: favorite_count , retweet_count should be converted to integer
- Tidy issue #1: unnessery table
  - This table describes the attributes of each tweet_id, which is the same as `archive_clean` table, so it could be merged with the archive_clean table. I merged `favorite_count` and `retweet_count` columns to `archive_clean` table.
  - From the merged table tweet_merge I found 3 with null values so I dropped them.

## img_predit table

- Quality issue #1: wrong data types
  To merge this table with `tweet_merge` table, we need convert `tweet_id` column to data type character
- Quality issue #2: mixed lower and upper case letters for p1 output
  All we needed from this table is the dog breed. I took p1, which was the most confident choice, as the result of breed prediction. Some of them start with upper case letter and some of them don't. I converted them all to lower cases. I also replaced values in p1 with NaN if the result was not a dog.
- Tidy issue #1: unnessery table
  This table described the attributes of each tweet_id, which was the same as tweet_merge table. I merged p1 column to tweet_merge table.