My wrangling efforts began with gathering tweet information and image prediction data from various sources. 'Twitter-Archive-Enhanced.csv' file which contained the tweet ids, dog names, and dog ratings was provided. After an application, and subsequent acceptance, for a twitter developer account, I wrote a script with Tweepy which created json files of all the tweets in 'Twitter-Archive-Enhanced.csv' and wrote them onto individual lines in 'tweet_json.txt'. This file could then be read into a Pandas DataFrame like a CSV file. After investigating the dataset, I decided that the only two columns I needed from it were retweet and favorite counts, which I added to the original dataset. Using Requests I downloaded the 'Image-Prediction.tsv' file and read it into a dataframe with Pandas and joined it to the main dataframe. Finally, I dropped several columns from the dataset that were unnecessary.

Some problems with the dataset were obvious from the start of the assessment, but many problems only arose during cleaning or into the analysis phase: when I had a better idea about what I wanted from the dataset. Starting with the quality assessment and cleaning. Visual assessment of the dog names and prediction columns showed many issues that needed fixing. The prediction software was not limited to only dogs so there was no standard for breed names. In order to make it more readable. I changed the underscores to spaces and capitalized each word. The only dog breed that needed specific correction was the Entle Bucher. Many words in the name column were not proper nouns, so they were replaced with None to match the other cells with no name.The prediction columns, that indicate whether or not the prediction is a dog, were strings that were easy to convert to booleans. Similarly, I converted the timestamp column from string to datetime, and the retweet and favorite count columns to integers from floats. With all of the data type problems fixed I moved to consolidating variables into single columns. Since I was only interested in what breed was in each image, I created a column that had the name of the breed most likely in the picture. If the first guesses were not dogs it would skip over them, so if none of the predictions were dog breeds the value in this column would be Null. This allowed me to drop the predictions columns.

The only two tidiness issues were data for tweets being separated into three separate datasets, and the dog stages were spread out over four columns. One I had retrieved the data from their various sources, I joined them all into a single dataset using their unique tweet ids. Lastly, I created a 'Dog Stage' column to consolidate the descriptions (Doggo, Floofer, Pupper, Puppo) in one column.