# Data Wrangling of WeRateDogs Dataset

## Introduction:

The WeRateDogs dataset, sourced from Twitter,had several data quality and tidiness issues. The data had plenty of inconsistencies with missing data and formatting issues. These issues were compounded as three distinct sources were required, each with their own set of problems. The data required significant wrangling before it could be used for any meaningful analysis and insights.

### Data Gathering:

The three distinct data sources that were gathered were:

1. The WeRateDogs Twitter archive (‘twitter-archive-enhanced.csv’). This archive included the ratings, dog names and dog stages from the account’s tweet history.
2. The image predictions file for the tweets (‘image-predictions.tsv’). This file contained predictions for the breed of dog in the image of the tweet.
3. Data from the Twitter API (‘tweet-json’). To increase the information available for analysis, this data provided additional context on each tweet.

### Data Assessment:

The data was assessed visually and programmatically. Several quality and tidiness issues were identified:

*Quality Issues:*

1. The 'timestamp' in df_archive is not stored in the datetime format.
2. Dog breeds in df_predictions are inconsistent in their format, as some do not start with uppercase letters.
3. Underscores are used instead of spaces for dog breeds in df_predictions.
4. Names of some dogs are incorrect - all lowercase or 'None'.
5. The 'source' column contained raw HTML tags, making extraction of meaningful data cumbersome.
6. The inclusion of replies and retweets diluted the dataset's focus on original ratings.

7. Some tweets were missing associated images, which were crucial for breed predictions.
8. ‘tweet_id' columns were not stored as strings, risking potential data loss or misinterpretation during operations.

*Tidiness Issues:*

1. Dog categories (e.g., doggo, pupper) were spread across multiple columns, making data querying and manipulation an unnecessary burden.
2. Information necessary for analysis was spread across the three datasets and necessitated a consolidated dataset.

## Data Cleaning:

With the issues identified, the data was cleaned using the Define-Code-Test framework. This methodology ensured that the issue was clearly defined, handled with the relevant code, and verified the intended solution. Once the issues were addressed, the three sources were merged into a master dataset for further analysis.