**Name:** Alaa Zarban

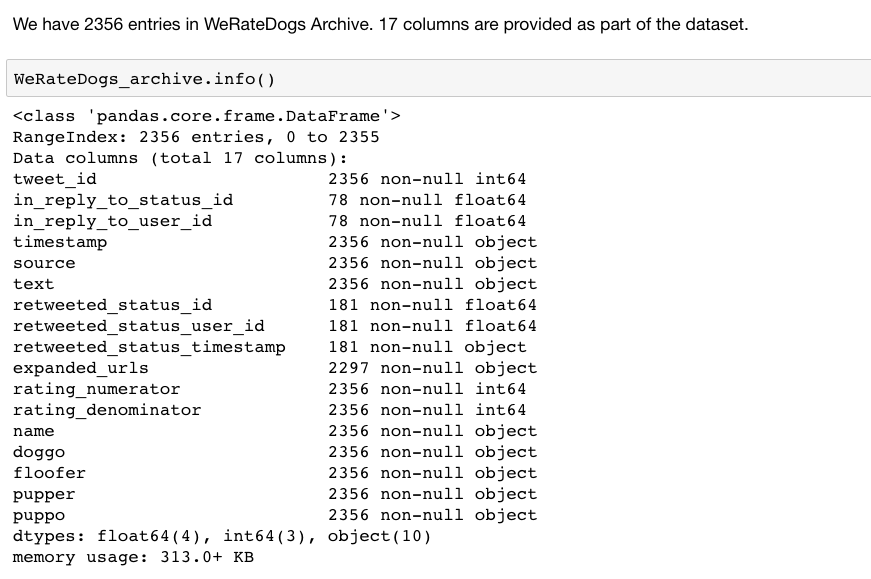
# Introduction

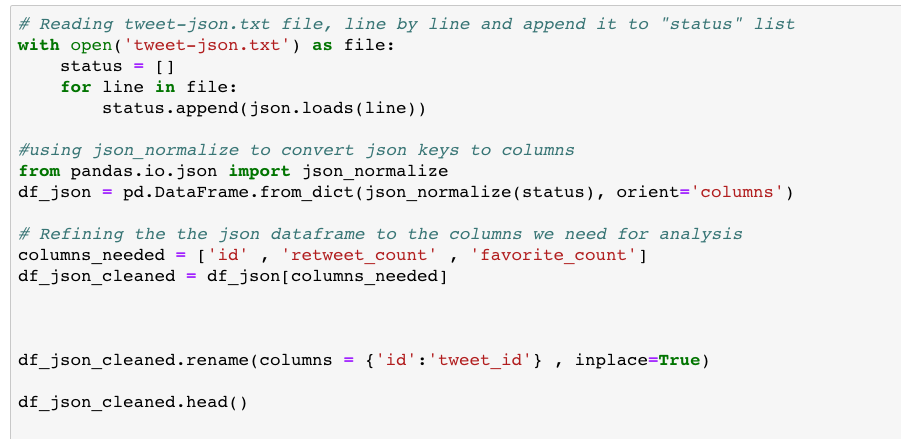
The analysis done in this project is from three sources that we combined together in one dataframe. The dataset is from WeRateDogs twitter account that rates dogs on unique and strange scale of x over 10. Where x the dog rate which usualy is more than 10!.

# Gathering

We started our analysis with gathering the data. The first source was from WeRateDogs archive provided. Second, the data given from image predictions. Third, tweets data that has been taken from text file provided. Unfortunately, I didn’t use Twitter API due to a delay of receiving the required authentication IDs.

1. WeRateDogs archive data



1. Image predictions data using request library through url provided   
   
2. Twitter data from text file provided. Note that JSON python library has been used to ease the conversion from JSON string to dataframe. JSON normalize function was a magical function that made my gathering at its best.  
     
   

# Assessing

I did the following in my assessment phase of data wrangling:

* Check tweet IDs for uniqueness.
* Check sources columns value counts.
* Visually, check the tweet text, rating system, dog stages, dog names.
* Visually and Programmatically, check how people call dog names in their tweets.
* Programmatically, check inapplicable and unreal names ( less than 2 characters, lower-case)
* Show tweets with decimal ratings and cross check that with the actual rating.
* Check denominator rating scale consistency.
* Check tweets with no images.

As a result, the following has been addressed:

## Quality issues:

* Tweets with wrong rating numerator because of decimal number in text.
* Dog names with less than 2 letter.
* Dog names with lower-case ( Not real or wrong ).
* Dog names with inappliacable names ( e.g. "The" , "a" ).
* Missing dog stage values ( All None ).
* inconsistent denominator rating ( 150 , 120 , 130 ).
* inadequate datatypes (timestamp)
* Tweets with no expanded URLs.

## Tidiness issues:

* Dogs stages in different columns, better to be unified in one columns called Dog\_Stage.
* Merge all 3 dataframes into one.

# Cleaning

We started by making a copy of all dataframes to avoid missing with the original data in case we needed them later. First, we started with the first tidiness issue of dog stages and merged all of the four stages under one column called **Dog\_stage.** Second, we merged all the three dataframes into one to ease analysis, insights and visualizations later.

As a result of our assessments, we cleared all quality issues mentioned before and did proper testing to make sure all issues has been addressed.