# Data Wrangling Project

## Introduction

Real-world data rarely comes clean. So Using Python and its libraries helps very much in cleaning messy, untidy data. In this project data will be gathered from a variety of sources and in a variety of formats. Then it will be assessed to check its quality and tidiness, after that comes the cleaning process. This is what is called data wrangling.

All this wrangling process will be documented in a Jupyter Notebook, plus showcase them through analyses and visualizations using Python (and its libraries) and/or SQL.

Wrangling, analyzing and visualizing will be done using tweet archive of Twitter user at WeRateDogs.

WeRateDogs is a Twitter account that rates people's dogs with a humorous comment about the dog. These ratings almost always have a denominator of 10. The numerators, though? Almost always greater than 10. Why? Because "they're good dogs Brent." WeRateDogs has over 4 million followers and has received international media coverage.

*The goal of this project is to wrangle data from WeRateDogs Twitter's account using Python and document the whole process in a Jupyter Notebook named wrangle_act.ipynb. My aim is to wrangle this data for interesting and trustworthy analyses using visualizations.*

### Project Details

Assessing and cleaning the entire dataset and get only a subset of its issues (eight quality issues and two tidiness issues at minimum) needed to be assessed and cleaned.
The tasks for this project are:

- Data wrangling, which consists of:
- Gathering data
- Assessing data
- Cleaning data
- Storing, analyzing, and visualizing our wrangled data
- Reporting on 1) our data wrangling efforts and 2) our data analyses and visualizations

Gathering Data for this Project

1. 'Enhanced Twitter Archive' the WeRateDogs Twitter archive provided by Udacity. I manually downloaded this file.
2. 'Image Predictions File' the tweet image predictions, is present in each tweet according to a neural network. This file (image_predictions.tsv) is hosted on Udacity's servers and should be downloaded programmatically using the Requests.
3. 'Additional Data via the Twitter API' each tweet's, retweet count and favorite ("like") count at minimum.

Action made to fix data issue:

**1. Quality issues to be fixed:**

I. remove duplicated retweets rows
II. empty columns from 1st file "twitter archive":
    a. retweeted_status_id column is almost empty
    b. retweeted_status_user_id column is almost empty
    c. retweeted_status_timestamp column is almost empty
    d. in_reply_to_status_id column is almost empty
    e. in_reply_to_user_id column is almost empty
III. In 1st file timestamp column is string and it should in datetime formate
IV. Some Dog names are meaningless in 1st file "twitter archive"

V. Assign 10 to rating_denominator column value in 1st file "twitter archive"
VI. Rename the rating_numerator column to be rating in 1st file "twitter archive"
VII. In 2nd file image prediction the column img_num is unnecessary (similar to 2nd issue but in different files)
VIII. Remove 66 repeated url rows in 2nd file
IX. Ratings with decimal values incorrectly extracted - Quality issue in 1st file

**2. Tidiness issues:**

i. 1st file "twitter archive":
   a. Merge 4 dog columns into 1 column
   b. drop 4 unnecessary columns in 1st file [ doggo, floofer, pupper, puppo] and be one column called [stage]
ii. 2nd file 'image_Prediction'
   a. Merge 3 columns [p1_dog, p2_dog, p3_dog] into [p_dog]
   b. drop 3 unnecessary columns [p1_dog, p2_dog, p3_dog]

** This could be similar to the pervious issue but here I'm adding columns in different way since they are Booleans and the resulting column is Boolean as well.

<u>Storing, Analyzing, and Visualizing Data for this Project</u>

- Store the clean DataFrame(s) in a CSV file with the main one named twitter_archive_master.csv.
- Analyze and visualize your wrangled data.