# Data Wrangling Project: WeRateDogs

The following report describes the data wrangling efforts through the project, the purpose of this project is to practice gathering data from a variety of sources and in a variety of formats, assess its quality and tidiness, then clean it.

## 1- Data Gathering

The data was collected from three different sources:

- Importing data from a CSV file.
- Using the Requests library to download files from URL.
- Using Tweepy to query Twitter's API for additional data.

### The data sources:

- **Enhanced Twitter Archive file.**

The WeRateDogs Twitter archive contains basic tweet data for all 5000+ of their tweets, I downloaded the file manually from Udacity sources.

- **Image Prediction file.**

A table full of image predictions (the top three only) alongside each tweet ID, image URL, and the image number that corresponded to the most confident prediction (numbered 1 to 4 since tweets can have up to four images).
I used the requests library to download the file from the provided URL.

- **Twitter API**

I couldn’t set up a Twitter developer account using the steps mentioned in the project overview, so I loaded the provided file directly into the dataframe.

## 2- Assessing the data

After gathering the data from the sources and loading them into dataframes, i started to assess them visually and programmatically for quality and tidiness issues.

### The Output was:

### Quality

- Some tweets have no images.
- Remove the incorrect dog names and convert the none values to nan type.
- Change columns data types.
- Remove retweets columns.
- Change the source content to human readable form.
- Capitalize the first letter of dog names.
- Display full content of the “text” column.
- Fixing the [rating_denominator] that have values != 10

### Tidiness

- doggo, floofer, pupper, and puppo columns in twitter_archive file should be values not headers.
- Merge json file’ and 'image_predictions' to 'twitter_archive'

## 3- Cleaning the data

All issues identified in the assessment phase were successfully cleaned using Python and pandas after making copies of each dataframe.
The define, code, and test steps of the cleaning process were clearly documented.