# Data wrangling and analysis report

## Introduction

**Project overview** – this project is a good one to practice web scraping as we gather data, complete the missing ones by wrangling techniques from twitter api, extracting data from a link, and finally manually from downloaded files later on we look after quality and tidiness issues and the last step in wrangling is cleaning the data .

**Libraries imported** - pandas - NumPy - requests - re - tweepy - json - data - timeit – matplotlib – seaborn - warnings

**Action done** - Data gathering - Data assessing - Data cleaning - Data analysis

**Case Study** WeRateDogs is popular for dog rating according to type of breed, accuracy, size and popularity according to number of tweets and number of favorites counts this ratings is done by a denominator of 10 and numerator always more than 10 to express the love of those people to dogs.
it has a popularity of millions of followers as it gained media coverage in this project we are having two types of data :

1. Udacity data which provide us with twitter-archive-enhanced which is a csv file and image-predictions.tsv wich is downloaded using request library using get url method and link provided by udacity
2. twitter api which is extracted using twitter developer account which provide us with the needed data to access the data from weratedogs including tweet_id and favourite_counts.

So the ipynb file consists of sections :

## Gathering data :

Pd.read_csv('csvfile') : to extract data from csv file into ipynb
Reading the file line by line techniques
Getting data using requests library to extract data from url

## Assessing data : we uses two types of assessing which are programmatically and visually

The used data frames are
pd.dataFrame.info() : to know the detailed information about the data frame their types , dimensions and no. of rows, columns.
pd.dataFrame.describe() : to determine the minimum and maximum values.
pd.value_counts() : to count the values in the data frame
pd.duplicated() : to determine the duplicated rows
pd.Series.query() : to extract certain values from the data frame
pd.loc and pd.iloc : index the contents of the data frame

## Quality issues:

- Dog names with problems should be replaced by nans.
- tweet_id datatype must be changed all to int64 to be merged correctly.
- timestamp dtype should be changed to datetime.
- missing values must be treated.

- numerator problems must be corrected if less than denominator.
- Dropping retweets from the data
- Removing html text which will be missy with data
- Duplicated jpg images as a result of retweets
- Rows which contain images for animals or anything that aren't dogs
- Columns containing string none instead of python nan
- Missy dog named containing word separators as – or _
- Favourite column dtype correction

**Tidiness issues :**

- Removing useless columns
- Dog classes columns can be collapsed to one column
- Merge three tweed_id columns in the three data frames
- Three bread probability can be changed with top most probability
- Three probability confidences can be changed by the top most accurate

## Data Cleaning :

First thing to do is to create copies of data frames so the data will be changed in that copied data frame
lambda functions to manipulate columns
pd.dropna() drop raws having nan values
pd.drop() drop columns
pd.Series.str.extract() to extract specified pattern from data frame
pd.Series.replace() to replace data
pd.merge() to merge columns of different data frame