wrangle_act

January 5, 2021

In [1]:
```
import numpy as np
import os
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import requests
import tweepy
import json
from timeit import default_timer as timer
from tweepy import OAuthHandler
```

1 Gathering Data

In [2]:
```
#Loading the twitter-archive-enhanced.csv into a DataFrame [WeRateDogs Twitter archive]
df = pd.read_csv('twitter-archive-enhanced.csv')
```

In [3]:
```
#Loading the tweet image predictions from Udacity's servers
r = requests.get('https://d17h27t6h515a5.cloudfront.net/topher/2017/August/599fd2ad_imag

with open('image-predictions.tsv', mode='wb') as file:
    file.write(r.content)
```

In [4]:
```
image_df = pd.read_csv('image-predictions.tsv', sep='\t')
```

In [5]:
```
# Query Twitter API for each tweet in the Twitter archive and save JSON in a text file
# These are hidden to comply with Twitter's API terms and conditions
consumer_key ='XXXX'
consumer_secret ='XXXX'
access_token ='XXXX'
access_secret='XXXX'

auth = OAuthHandler(consumer_key, consumer_secret)
auth.set_access_token(access_token, access_secret)

api = tweepy.API(auth, wait_on_rate_limit=True)
```

In [ ]:
```
# NOTE TO STUDENT WITH MOBILE VERIFICATION ISSUES:
# df_1 is a DataFrame with the twitter_archive_enhanced.csv file. You may have to
```

```
        # change line 17 to match the name of your DataFrame with twitter_archive_enhanced.csv
        # NOTE TO REVIEWER: this student had mobile verification issues so the following
        # Twitter API code was sent to this student from a Udacity instructor
        # Tweet IDs for which to gather additional data via Twitter's API
        tweet_ids = df.tweet_id.values
        len(tweet_ids)

        # Query Twitter's API for JSON data for each tweet ID in the Twitter archive
        count = 0
        fails_dict = {}
        start = timer()
        # Save each tweet's returned JSON as a new line in a .txt file
        with open('tweet_json.txt', 'w') as outfile:
            # This loop will likely take 20-30 minutes to run because of Twitter's rate limit
            for tweet_id in tweet_ids:
                count = 1
                print(str(count) + ": " + str(tweet_id))
                try:
                    tweet = api.get_status(tweet_id, tweet_mode='extended')
                    print("Success")
                    json.dump(tweet._json, outfile)
                    outfile.write('\n')
                except tweepy.TweepError as e:
                    print("Fail")
                    fails_dict[tweet_id] = e
                    pass
        end = timer()
        print(end - start)
        print(fails_dict)
```

```
In [5]: df_tweet_json = pd.DataFrame(columns=['tweet_id', 'retweet_count', 'favorite_count'])
        with open('tweet_json.txt') as data_file:
            for line in data_file:
                tweet = json.loads(line)
                tweet_id = tweet['id_str']
                retweet_count = tweet['retweet_count']
                favorite_count = tweet['favorite_count']
                df_tweet_json = df_tweet_json.append(pd.DataFrame([[tweet_id, retweet_count, fav
                columns=['tweet_id', 'retweet_count', 'favorite_count']))
                df_tweet_json = df_tweet_json.reset_index(drop=True)
```

2 Assessing Data

2.1 Assess df

```
In [7]: # Display the df table
        df
```

Out[7]:

	tweet_id	in_reply_to_status_id	in_reply_to_user_id	\
0	892420643555336193	NaN	NaN	
1	892177421306343426	NaN	NaN	
2	891815181378084864	NaN	NaN	
3	891689557279858688	NaN	NaN	
4	891327558926688256	NaN	NaN	
5	891087950875897856	NaN	NaN	
6	890971913173991426	NaN	NaN	
7	890729181411237888	NaN	NaN	
8	890609185150312448	NaN	NaN	
9	890240255349198849	NaN	NaN	
10	890006608113172480	NaN	NaN	
11	889880896479866881	NaN	NaN	
12	889665388333682689	NaN	NaN	
13	889638837579907072	NaN	NaN	
14	889531135344209921	NaN	NaN	
15	889278841981685760	NaN	NaN	
16	888917238123831296	NaN	NaN	
17	888804989199671297	NaN	NaN	
18	888554962724278272	NaN	NaN	
19	888202515573088257	NaN	NaN	
20	888078434458587136	NaN	NaN	
21	887705289381826560	NaN	NaN	
22	887517139158093824	NaN	NaN	
23	887473957103951883	NaN	NaN	
24	887343217045368832	NaN	NaN	
25	887101392804085760	NaN	NaN	
26	886983233522544640	NaN	NaN	
27	886736880519319552	NaN	NaN	
28	886680336477933568	NaN	NaN	
29	886366144734445568	NaN	NaN	
...	...	...	...	
2326	666411507551481857	NaN	NaN	
2327	666407126856765440	NaN	NaN	
2328	666396247373291520	NaN	NaN	
2329	666373753744588802	NaN	NaN	
2330	666362758909284353	NaN	NaN	
2331	666353288456101888	NaN	NaN	
2332	666345417576210432	NaN	NaN	
2333	666337882303524864	NaN	NaN	
2334	666293911632134144	NaN	NaN	
2335	666287406224695296	NaN	NaN	
2336	666273097616637952	NaN	NaN	
2337	666268910803644416	NaN	NaN	
2338	666104133288665088	NaN	NaN	
2339	666102155909144576	NaN	NaN	
2340	666099513787052032	NaN	NaN	
2341	666094000022159362	NaN	NaN	

```
2342  666082916733198337                    NaN                    NaN
2343  666073100786774016                    NaN                    NaN
2344  666071193221509120                    NaN                    NaN
2345  666063827256086533                    NaN                    NaN
2346  666058600524156928                    NaN                    NaN
2347  666057090499244032                    NaN                    NaN
2348  666055525042405380                    NaN                    NaN
2349  666051853826850816                    NaN                    NaN
2350  666050758794694657                    NaN                    NaN
2351  666049248165822465                    NaN                    NaN
2352  666044226329800704                    NaN                    NaN
2353  666033412701032449                    NaN                    NaN
2354  666029285002620928                    NaN                    NaN
2355  666020888022790149                    NaN                    NaN

                      timestamp  \
0     2017-08-01 16:23:56 +0000
1     2017-08-01 00:17:27 +0000
2     2017-07-31 00:18:03 +0000
3     2017-07-30 15:58:51 +0000
4     2017-07-29 16:00:24 +0000
5     2017-07-29 00:08:17 +0000
6     2017-07-28 16:27:12 +0000
7     2017-07-28 00:22:40 +0000
8     2017-07-27 16:25:51 +0000
9     2017-07-26 15:59:51 +0000
10    2017-07-26 00:31:25 +0000
11    2017-07-25 16:11:53 +0000
12    2017-07-25 01:55:32 +0000
13    2017-07-25 00:10:02 +0000
14    2017-07-24 17:02:04 +0000
15    2017-07-24 00:19:32 +0000
16    2017-07-23 00:22:39 +0000
17    2017-07-22 16:56:37 +0000
18    2017-07-22 00:23:06 +0000
19    2017-07-21 01:02:36 +0000
20    2017-07-20 16:49:33 +0000
21    2017-07-19 16:06:48 +0000
22    2017-07-19 03:39:09 +0000
23    2017-07-19 00:47:34 +0000
24    2017-07-18 16:08:03 +0000
25    2017-07-18 00:07:08 +0000
26    2017-07-17 16:17:36 +0000
27    2017-07-16 23:58:41 +0000
28    2017-07-16 20:14:00 +0000
29    2017-07-15 23:25:31 +0000
...                        ...
2326  2015-11-17 00:24:19 +0000
```

```
2327  2015-11-17 00:06:54 +0000
2328  2015-11-16 23:23:41 +0000
2329  2015-11-16 21:54:18 +0000
2330  2015-11-16 21:10:36 +0000
2331  2015-11-16 20:32:58 +0000
2332  2015-11-16 20:01:42 +0000
2333  2015-11-16 19:31:45 +0000
2334  2015-11-16 16:37:02 +0000
2335  2015-11-16 16:11:11 +0000
2336  2015-11-16 15:14:19 +0000
2337  2015-11-16 14:57:41 +0000
2338  2015-11-16 04:02:55 +0000
2339  2015-11-16 03:55:04 +0000
2340  2015-11-16 03:44:34 +0000
2341  2015-11-16 03:22:39 +0000
2342  2015-11-16 02:38:37 +0000
2343  2015-11-16 01:59:36 +0000
2344  2015-11-16 01:52:02 +0000
2345  2015-11-16 01:22:45 +0000
2346  2015-11-16 01:01:59 +0000
2347  2015-11-16 00:55:59 +0000
2348  2015-11-16 00:49:46 +0000
2349  2015-11-16 00:35:11 +0000
2350  2015-11-16 00:30:50 +0000
2351  2015-11-16 00:24:50 +0000
2352  2015-11-16 00:04:52 +0000
2353  2015-11-15 23:21:54 +0000
2354  2015-11-15 23:05:30 +0000
2355  2015-11-15 22:32:08 +0000

                                                  source  \
0     <a href="http://twitter.com/download/iphone" r...
1     <a href="http://twitter.com/download/iphone" r...
2     <a href="http://twitter.com/download/iphone" r...
3     <a href="http://twitter.com/download/iphone" r...
4     <a href="http://twitter.com/download/iphone" r...
5     <a href="http://twitter.com/download/iphone" r...
6     <a href="http://twitter.com/download/iphone" r...
7     <a href="http://twitter.com/download/iphone" r...
8     <a href="http://twitter.com/download/iphone" r...
9     <a href="http://twitter.com/download/iphone" r...
10    <a href="http://twitter.com/download/iphone" r...
11    <a href="http://twitter.com/download/iphone" r...
12    <a href="http://twitter.com/download/iphone" r...
13    <a href="http://twitter.com/download/iphone" r...
14    <a href="http://twitter.com/download/iphone" r...
15    <a href="http://twitter.com/download/iphone" r...
16    <a href="http://twitter.com/download/iphone" r...
```

```
17    <a href="http://twitter.com/download/iphone" r...
18    <a href="http://twitter.com/download/iphone" r...
19    <a href="http://twitter.com/download/iphone" r...
20    <a href="http://twitter.com/download/iphone" r...
21    <a href="http://twitter.com/download/iphone" r...
22    <a href="http://twitter.com/download/iphone" r...
23    <a href="http://twitter.com/download/iphone" r...
24    <a href="http://twitter.com/download/iphone" r...
25    <a href="http://twitter.com/download/iphone" r...
26    <a href="http://twitter.com/download/iphone" r...
27    <a href="http://twitter.com/download/iphone" r...
28    <a href="http://twitter.com/download/iphone" r...
29    <a href="http://twitter.com/download/iphone" r...
...                                                 ...
2326  <a href="http://twitter.com/download/iphone" r...
2327  <a href="http://twitter.com/download/iphone" r...
2328  <a href="http://twitter.com/download/iphone" r...
2329  <a href="http://twitter.com/download/iphone" r...
2330  <a href="http://twitter.com/download/iphone" r...
2331  <a href="http://twitter.com/download/iphone" r...
2332  <a href="http://twitter.com/download/iphone" r...
2333  <a href="http://twitter.com/download/iphone" r...
2334  <a href="http://twitter.com/download/iphone" r...
2335  <a href="http://twitter.com/download/iphone" r...
2336  <a href="http://twitter.com/download/iphone" r...
2337  <a href="http://twitter.com/download/iphone" r...
2338  <a href="http://twitter.com/download/iphone" r...
2339  <a href="http://twitter.com/download/iphone" r...
2340  <a href="http://twitter.com/download/iphone" r...
2341  <a href="http://twitter.com/download/iphone" r...
2342  <a href="http://twitter.com/download/iphone" r...
2343  <a href="http://twitter.com/download/iphone" r...
2344  <a href="http://twitter.com/download/iphone" r...
2345  <a href="http://twitter.com/download/iphone" r...
2346  <a href="http://twitter.com/download/iphone" r...
2347  <a href="http://twitter.com/download/iphone" r...
2348  <a href="http://twitter.com/download/iphone" r...
2349  <a href="http://twitter.com/download/iphone" r...
2350  <a href="http://twitter.com/download/iphone" r...
2351  <a href="http://twitter.com/download/iphone" r...
2352  <a href="http://twitter.com/download/iphone" r...
2353  <a href="http://twitter.com/download/iphone" r...
2354  <a href="http://twitter.com/download/iphone" r...
2355  <a href="http://twitter.com/download/iphone" r...

                                                 text  retweeted_status_id  \
0     This is Phineas. He's a mystical boy. Only eve...                  NaN
1     This is Tilly. She's just checking pup on you...                  NaN
```

2	This is Archie. He is a rare Norwegian Pouncin...	NaN
3	This is Darla. She commenced a snooze mid meal...	NaN
4	This is Franklin. He would like you to stop ca...	NaN
5	Here we have a majestic great white breaching ...	NaN
6	Meet Jax. He enjoys ice cream so much he gets ...	NaN
7	When you watch your owner call another dog a g...	NaN
8	This is Zoey. She doesn't want to be one of th...	NaN
9	This is Cassie. She is a college pup. Studying...	NaN
10	This is Koda. He is a South Australian decksha...	NaN
11	This is Bruno. He is a service shark. Only get...	NaN
12	Here's a puppo that seems to be on the fence a...	NaN
13	This is Ted. He does his best. Sometimes that'...	NaN
14	This is Stuart. He's sporting his favorite fan...	NaN
15	This is Oliver. You're witnessing one of his m...	NaN
16	This is Jim. He found a fren. Taught him how t...	NaN
17	This is Zeke. He has a new stick. Very proud o...	NaN
18	This is Ralphus. He's powering up. Attempting ...	NaN
19	RT @dog_rates: This is Canela. She attempted s...	8.874740e+17
20	This is Gerald. He was just told he didn't get...	NaN
21	This is Jeffrey. He has a monopoly on the pool...	NaN
22	I've yet to rate a Venezuelan Hover Wiener. Th...	NaN
23	This is Canela. She attempted some fancy porch...	NaN
24	You may not have known you needed to see this ...	NaN
25	This... is a Jubilant Antarctic House Bear. We...	NaN
26	This is Maya. She's very shy. Rarely leaves he...	NaN
27	This is Mingus. He's a wonderful father to his...	NaN
28	This is Derek. He's late for a dog meeting. 13...	NaN
29	This is Roscoe. Another pupper fallen victim t...	NaN
...	...	...
2326	This is quite the dog. Gets really excited whe...	NaN
2327	This is a southern Vesuvius bumblegruff. Can d...	NaN
2328	Oh goodness. A super rare northeast Qdoba kang...	NaN
2329	Those are sunglasses and a jean jacket. 11/10 ...	NaN
2330	Unique dog here. Very small. Lives in containe...	NaN
2331	Here we have a mixed Asiago from the Galápagos...	NaN
2332	Look at this jokester thinking seat belt laws ...	NaN
2333	This is an extremely rare horned Parthenon. No...	NaN
2334	This is a funny dog. Weird toes. Won't come do...	NaN
2335	This is an Albanian 3 1/2 legged Episcopalian...	NaN
2336	Can take selfies 11/10 https://t.co/ws2AMaNwPW	NaN
2337	Very concerned about fellow dog trapped in com...	NaN
2338	Not familiar with this breed. No tail (weird)...	NaN
2339	Oh my. Here you are seeing an Adobe Setter giv...	NaN
2340	Can stand on stump for what seems like a while...	NaN
2341	This appears to be a Mongolian Presbyterian mi...	NaN
2342	Here we have a well-established sunblockerspan...	NaN
2343	Let's hope this flight isn't Malaysian (lol). ...	NaN
2344	Here we have a northern speckled Rhododendron...	NaN

```
2345  This is the happiest dog you will ever see. Ve...                NaN
2346  Here is the Rand Paul of retrievers folks! He'...                NaN
2347  My oh my. This is a rare blond Canadian terrie...                NaN
2348  Here is a Siberian heavily armored polar bear ...                NaN
2349  This is an odd dog. Hard on the outside but lo...                NaN
2350  This is a truly beautiful English Wilson Staff...                NaN
2351  Here we have a 1949 1st generation vulpix. Enj...                NaN
2352  This is a purebred Piers Morgan. Loves to Netf...                NaN
2353  Here is a very happy pup. Big fan of well-main...                NaN
2354  This is a western brown Mitsubishi terrier. Up...                NaN
2355  Here we have a Japanese Irish Setter. Lost eye...                NaN

      retweeted_status_user_id retweeted_status_timestamp  \
0                          NaN                         NaN
1                          NaN                         NaN
2                          NaN                         NaN
3                          NaN                         NaN
4                          NaN                         NaN
5                          NaN                         NaN
6                          NaN                         NaN
7                          NaN                         NaN
8                          NaN                         NaN
9                          NaN                         NaN
10                         NaN                         NaN
11                         NaN                         NaN
12                         NaN                         NaN
13                         NaN                         NaN
14                         NaN                         NaN
15                         NaN                         NaN
16                         NaN                         NaN
17                         NaN                         NaN
18                         NaN                         NaN
19                4.196984e+09  2017-07-19 00:47:34 +0000
20                         NaN                         NaN
21                         NaN                         NaN
22                         NaN                         NaN
23                         NaN                         NaN
24                         NaN                         NaN
25                         NaN                         NaN
26                         NaN                         NaN
27                         NaN                         NaN
28                         NaN                         NaN
29                         NaN                         NaN
...                        ...                         ...
2326                       NaN                         NaN
2327                       NaN                         NaN
2328                       NaN                         NaN
2329                       NaN                         NaN
```

```
2330                        NaN                         NaN
2331                        NaN                         NaN
2332                        NaN                         NaN
2333                        NaN                         NaN
2334                        NaN                         NaN
2335                        NaN                         NaN
2336                        NaN                         NaN
2337                        NaN                         NaN
2338                        NaN                         NaN
2339                        NaN                         NaN
2340                        NaN                         NaN
2341                        NaN                         NaN
2342                        NaN                         NaN
2343                        NaN                         NaN
2344                        NaN                         NaN
2345                        NaN                         NaN
2346                        NaN                         NaN
2347                        NaN                         NaN
2348                        NaN                         NaN
2349                        NaN                         NaN
2350                        NaN                         NaN
2351                        NaN                         NaN
2352                        NaN                         NaN
2353                        NaN                         NaN
2354                        NaN                         NaN
2355                        NaN                         NaN

                                              expanded_urls  rating_numerator  \
0     https://twitter.com/dog_rates/status/892420643...                  13
1     https://twitter.com/dog_rates/status/892177421...                  13
2     https://twitter.com/dog_rates/status/891815181...                  12
3     https://twitter.com/dog_rates/status/891689557...                  13
4     https://twitter.com/dog_rates/status/891327558...                  12
5     https://twitter.com/dog_rates/status/891087950...                  13
6     https://gofundme.com/ydvmve-surgery-for-jax,ht...                  13
7     https://twitter.com/dog_rates/status/890729181...                  13
8     https://twitter.com/dog_rates/status/890609185...                  13
9     https://twitter.com/dog_rates/status/890240255...                  14
10    https://twitter.com/dog_rates/status/890006608...                  13
11    https://twitter.com/dog_rates/status/889880896...                  13
12    https://twitter.com/dog_rates/status/889665388...                  13
13    https://twitter.com/dog_rates/status/889638837...                  12
14    https://twitter.com/dog_rates/status/889531135...                  13
15    https://twitter.com/dog_rates/status/889278841...                  13
16    https://twitter.com/dog_rates/status/888917238...                  12
17    https://twitter.com/dog_rates/status/888804989...                  13
18    https://twitter.com/dog_rates/status/888554962...                  13
19    https://twitter.com/dog_rates/status/887473957...                  13
```

```
20    https://twitter.com/dog_rates/status/888078434...                12
21    https://twitter.com/dog_rates/status/887705289...                13
22    https://twitter.com/dog_rates/status/887517139...                14
23    https://twitter.com/dog_rates/status/887473957...                13
24    https://twitter.com/dog_rates/status/887343217...                13
25    https://twitter.com/dog_rates/status/887101392...                12
26    https://twitter.com/dog_rates/status/886983233...                13
27    https://www.gofundme.com/mingusneedsus,https:/...                13
28    https://twitter.com/dog_rates/status/886680336...                13
29    https://twitter.com/dog_rates/status/886366144...                12
...                                                  ...               ...
2326  https://twitter.com/dog_rates/status/666411507...                 2
2327  https://twitter.com/dog_rates/status/666407126...                 7
2328  https://twitter.com/dog_rates/status/666396247...                 9
2329  https://twitter.com/dog_rates/status/666373753...                11
2330  https://twitter.com/dog_rates/status/666362758...                 6
2331  https://twitter.com/dog_rates/status/666353288...                 8
2332  https://twitter.com/dog_rates/status/666345417...                10
2333  https://twitter.com/dog_rates/status/666337882...                 9
2334  https://twitter.com/dog_rates/status/666293911...                 3
2335  https://twitter.com/dog_rates/status/666287406...                 1
2336  https://twitter.com/dog_rates/status/666273097...                11
2337  https://twitter.com/dog_rates/status/666268910...                10
2338  https://twitter.com/dog_rates/status/666104133...                 1
2339  https://twitter.com/dog_rates/status/666102155...                11
2340  https://twitter.com/dog_rates/status/666099513...                 8
2341  https://twitter.com/dog_rates/status/666094000...                 9
2342  https://twitter.com/dog_rates/status/666082916...                 6
2343  https://twitter.com/dog_rates/status/666073100...                10
2344  https://twitter.com/dog_rates/status/666071193...                 9
2345  https://twitter.com/dog_rates/status/666063827...                10
2346  https://twitter.com/dog_rates/status/666058600...                 8
2347  https://twitter.com/dog_rates/status/666057090...                 9
2348  https://twitter.com/dog_rates/status/666055525...                10
2349  https://twitter.com/dog_rates/status/666051853...                 2
2350  https://twitter.com/dog_rates/status/666050758...                10
2351  https://twitter.com/dog_rates/status/666049248...                 5
2352  https://twitter.com/dog_rates/status/666044226...                 6
2353  https://twitter.com/dog_rates/status/666033412...                 9
2354  https://twitter.com/dog_rates/status/666029285...                 7
2355  https://twitter.com/dog_rates/status/666020888...                 8

      rating_denominator      name  doggo floofer  pupper  puppo
0                     10   Phineas   None    None    None   None
1                     10     Tilly   None    None    None   None
2                     10    Archie   None    None    None   None
3                     10     Darla   None    None    None   None
4                     10  Franklin   None    None    None   None
```

5	10	None	None	None	None	None
6	10	Jax	None	None	None	None
7	10	None	None	None	None	None
8	10	Zoey	None	None	None	None
9	10	Cassie	doggo	None	None	None
10	10	Koda	None	None	None	None
11	10	Bruno	None	None	None	None
12	10	None	None	None	None	puppo
13	10	Ted	None	None	None	None
14	10	Stuart	None	None	None	puppo
15	10	Oliver	None	None	None	None
16	10	Jim	None	None	None	None
17	10	Zeke	None	None	None	None
18	10	Ralphus	None	None	None	None
19	10	Canela	None	None	None	None
20	10	Gerald	None	None	None	None
21	10	Jeffrey	None	None	None	None
22	10	such	None	None	None	None
23	10	Canela	None	None	None	None
24	10	None	None	None	None	None
25	10	None	None	None	None	None
26	10	Maya	None	None	None	None
27	10	Mingus	None	None	None	None
28	10	Derek	None	None	None	None
29	10	Roscoe	None	None	pupper	None
...	...	...	...	...	...	...
2326	10	quite	None	None	None	None
2327	10	a	None	None	None	None
2328	10	None	None	None	None	None
2329	10	None	None	None	None	None
2330	10	None	None	None	None	None
2331	10	None	None	None	None	None
2332	10	None	None	None	None	None
2333	10	an	None	None	None	None
2334	10	a	None	None	None	None
2335	2	an	None	None	None	None
2336	10	None	None	None	None	None
2337	10	None	None	None	None	None
2338	10	None	None	None	None	None
2339	10	None	None	None	None	None
2340	10	None	None	None	None	None
2341	10	None	None	None	None	None
2342	10	None	None	None	None	None
2343	10	None	None	None	None	None
2344	10	None	None	None	None	None
2345	10	the	None	None	None	None
2346	10	the	None	None	None	None
2347	10	a	None	None	None	None

```
2348                  10         a   None    None    None   None
2349                  10        an   None    None    None   None
2350                  10         a   None    None    None   None
2351                  10      None   None    None    None   None
2352                  10         a   None    None    None   None
2353                  10         a   None    None    None   None
2354                  10         a   None    None    None   None
2355                  10      None   None    None    None   None

[2356 rows x 17 columns]
```

```
In [8]: df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2356 entries, 0 to 2355
Data columns (total 17 columns):
tweet_id                      2356 non-null int64
in_reply_to_status_id         78 non-null float64
in_reply_to_user_id           78 non-null float64
timestamp                     2356 non-null object
source                        2356 non-null object
text                          2356 non-null object
retweeted_status_id           181 non-null float64
retweeted_status_user_id      181 non-null float64
retweeted_status_timestamp    181 non-null object
expanded_urls                 2297 non-null object
rating_numerator              2356 non-null int64
rating_denominator            2356 non-null int64
name                          2356 non-null object
doggo                         2356 non-null object
floofer                       2356 non-null object
pupper                        2356 non-null object
puppo                         2356 non-null object
dtypes: float64(4), int64(3), object(10)
memory usage: 313.0+ KB
```

```
In [9]: df.head()
```

```
Out[9]:              tweet_id  in_reply_to_status_id  in_reply_to_user_id  \
        0  892420643555336193                    NaN                  NaN
        1  892177421306343426                    NaN                  NaN
        2  891815181378084864                    NaN                  NaN
        3  891689557279858688                    NaN                  NaN
        4  891327558926688256                    NaN                  NaN

                         timestamp  \
        0  2017-08-01 16:23:56 +0000
        1  2017-08-01 00:17:27 +0000
```

```
2  2017-07-31 00:18:03 +0000
3  2017-07-30 15:58:51 +0000
4  2017-07-29 16:00:24 +0000

                                              source  \
0  <a href="http://twitter.com/download/iphone" r...
1  <a href="http://twitter.com/download/iphone" r...
2  <a href="http://twitter.com/download/iphone" r...
3  <a href="http://twitter.com/download/iphone" r...
4  <a href="http://twitter.com/download/iphone" r...

                                                text  retweeted_status_id  \
0  This is Phineas. He's a mystical boy. Only eve...                  NaN
1  This is Tilly. She's just checking pup on you...                  NaN
2  This is Archie. He is a rare Norwegian Pouncin...                  NaN
3  This is Darla. She commenced a snooze mid meal...                  NaN
4  This is Franklin. He would like you to stop ca...                  NaN

   retweeted_status_user_id retweeted_status_timestamp  \
0                       NaN                        NaN
1                       NaN                        NaN
2                       NaN                        NaN
3                       NaN                        NaN
4                       NaN                        NaN

                                      expanded_urls  rating_numerator  \
0  https://twitter.com/dog_rates/status/892420643...                13
1  https://twitter.com/dog_rates/status/892177421...                13
2  https://twitter.com/dog_rates/status/891815181...                12
3  https://twitter.com/dog_rates/status/891689557...                13
4  https://twitter.com/dog_rates/status/891327558...                12

   rating_denominator      name doggo floofer pupper puppo
0                  10   Phineas  None    None   None  None
1                  10     Tilly  None    None   None  None
2                  10    Archie  None    None   None  None
3                  10     Darla  None    None   None  None
4                  10  Franklin  None    None   None  None
```

1. Missing data in the following columns: in_reply_to_status_id, in_reply_to_user_id, retweeted_status_id, etweeted_status_user_id, retweeted_status_timestamp, expanded_urls
2. Timestamp and retweeted_status_timestamp is an object
3. Source columns have HTML tags
4. This dataset includes retweets, which means there is duplicated data

```
In [12]: # checks for duplicated entries in df
         df[df.duplicated()].shape[0]

Out[12]: 0

In [13]: df["rating_numerator"].max()

Out[13]: 1776

In [14]: df["rating_denominator"].max()

Out[14]: 170

In [15]: df[df.name.str.islower()].name.value_counts()

Out[15]: a               55
         the              8
         an               7
         very             5
         just             4
         one              4
         quite            4
         getting          2
         actually         2
         mad              2
         not              2
         this             1
         unacceptable     1
         light            1
         life             1
         all              1
         officially       1
         by               1
         my               1
         his              1
         old              1
         infuriating      1
         such             1
         incredibly       1
         space            1
         Name: name, dtype: int64

In [16]: df[df.name.str.isupper()].name.value_counts()

Out[16]: JD    1
         O     1
         Name: name, dtype: int64
```

Dogs name have 'None', or 'a', or 'an.' or 'O' or 'by' and some more lower case words as names

2.2 Access image_df

In [17]:
```
# Display the image_df table
image_df
```

Out[17]:

	tweet_id	jpg_url
0	666020888022790149	https://pbs.twimg.com/media/CT4udn0WwAA0aMy.jpg
1	666029285002620928	https://pbs.twimg.com/media/CT42GRgUYAA5iDo.jpg
2	666033412701032449	https://pbs.twimg.com/media/CT4521TWwAEvMyu.jpg
3	666044226329800704	https://pbs.twimg.com/media/CT5Dr8HUEAA-lEu.jpg
4	666049248165822465	https://pbs.twimg.com/media/CT5IQmsXIAAKY4A.jpg
5	666050758794694657	https://pbs.twimg.com/media/CT5Jof1WUAEuVxN.jpg
6	666051853826850816	https://pbs.twimg.com/media/CT5KoJ1WoAAJash.jpg
7	666055525042405380	https://pbs.twimg.com/media/CT5N9tpXIAAifs1.jpg
8	666057090499244032	https://pbs.twimg.com/media/CT5PY90WoAAQGLo.jpg
9	666058600524156928	https://pbs.twimg.com/media/CT5Qw94XAAA_2dP.jpg
10	666063827256086533	https://pbs.twimg.com/media/CT5Vg_wXIAAXfnj.jpg
11	666071193221509120	https://pbs.twimg.com/media/CT5cN_3WEAAl0oZ.jpg
12	666073100786774016	https://pbs.twimg.com/media/CT5d9DZXAAALcwe.jpg
13	666082916733198337	https://pbs.twimg.com/media/CT5m4VGWEAAtKc8.jpg
14	666094000022159362	https://pbs.twimg.com/media/CT5w9gUW4AAsBNN.jpg
15	666099513787052032	https://pbs.twimg.com/media/CT51-JJUEAA6hV8.jpg
16	666102155909144576	https://pbs.twimg.com/media/CT54YGiWUAEZnoK.jpg
17	666104133288665088	https://pbs.twimg.com/media/CT56LSZWoAAlJj2.jpg
18	666268910803644416	https://pbs.twimg.com/media/CT8QCd1WEAADXws.jpg
19	666273097616637952	https://pbs.twimg.com/media/CT8T1mtUwAA3aqm.jpg
20	666287406224695296	https://pbs.twimg.com/media/CT8g3BpUEAAuFjg.jpg
21	666293911632134144	https://pbs.twimg.com/media/CT8mx7KW4AEQu8N.jpg
22	666337882303524864	https://pbs.twimg.com/media/CT9OwFIWEAMuRje.jpg
23	666345417576210432	https://pbs.twimg.com/media/CT9Vn7PWoAA_ZCM.jpg
24	666353288456101888	https://pbs.twimg.com/media/CT9cxOtUEAAhNN_.jpg
25	666362758909284353	https://pbs.twimg.com/media/CT9lXGsUcAAyUFt.jpg
26	666373753744588802	https://pbs.twimg.com/media/CT9vZEYWUAAlZO5.jpg
27	666396247373291520	https://pbs.twimg.com/media/CT-D2ZHWIAA3gK1.jpg
28	666407126856765440	https://pbs.twimg.com/media/CT-NvwmW4AAugGZ.jpg
29	666411507551481857	https://pbs.twimg.com/media/CT-RugiWIAELEaq.jpg
...	...	...
2045	886366144734445568	https://pbs.twimg.com/media/DE0BTnQUwAApKEH.jpg
2046	886680336477933568	https://pbs.twimg.com/media/DE4fEDzWAAAyHMM.jpg
2047	886736880519319552	https://pbs.twimg.com/media/DE5Se8FXcAAJFx4.jpg
2048	886983233522544640	https://pbs.twimg.com/media/DE8yicJW0AAAvBJ.jpg
2049	887101392804085760	https://pbs.twimg.com/media/DE-eAq6UwAA-jaE.jpg
2050	887343217045368832	https://pbs.twimg.com/ext_tw_video_thumb/88734...
2051	887473957103951883	https://pbs.twimg.com/media/DFDw2tyUQAAAFke.jpg
2052	887517139158093824	https://pbs.twimg.com/ext_tw_video_thumb/88751...
2053	887705289381826560	https://pbs.twimg.com/media/DFHDQBbXgAEqY7t.jpg
2054	888078434458587136	https://pbs.twimg.com/media/DFMWn56WsAAkA7B.jpg
2055	888202515573088257	https://pbs.twimg.com/media/DFDw2tyUQAAAFke.jpg
2056	888554962724278272	https://pbs.twimg.com/media/DFTH_0-UQAACu20.jpg

```
2057  888804989199671297    https://pbs.twimg.com/media/DFWra-3VYAA2piG.jpg
2058  888917238123831296    https://pbs.twimg.com/media/DFYRgsOUQAARGhO.jpg
2059  889278841981685760  https://pbs.twimg.com/ext_tw_video_thumb/88927...
2060  889531135344209921    https://pbs.twimg.com/media/DFg_2PVWOAEHN3p.jpg
2061  889638837579907072    https://pbs.twimg.com/media/DFihzFfXsAYGDPR.jpg
2062  889665388333682689    https://pbs.twimg.com/media/DFi579UWsAAatzw.jpg
2063  889880896479866881    https://pbs.twimg.com/media/DFl99B1WsAITKsg.jpg
2064  890006608113172480    https://pbs.twimg.com/media/DFnwSY4WAAAMliS.jpg
2065  890240255349198849    https://pbs.twimg.com/media/DFrEyVuWOAAO3t9.jpg
2066  890609185150312448    https://pbs.twimg.com/media/DFwUU__XcAEpyXI.jpg
2067  890729181411237888    https://pbs.twimg.com/media/DFyBahAVwAAhUTd.jpg
2068  890971913173991426    https://pbs.twimg.com/media/DF1eOmZXUAALUcq.jpg
2069  891087950875897856    https://pbs.twimg.com/media/DF3HwyEWsAABqE6.jpg
2070  891327558926688256    https://pbs.twimg.com/media/DF6hr6BUMAAzZgT.jpg
2071  891689557279858688    https://pbs.twimg.com/media/DF_q7IAWsAEuuN8.jpg
2072  891815181378084864    https://pbs.twimg.com/media/DGBdLU1WsAANxJ9.jpg
2073  892177421306343426    https://pbs.twimg.com/media/DGGmoV4XsAAUL6n.jpg
2074  892420643555336193    https://pbs.twimg.com/media/DGKD1-bXoAAIAUK.jpg

      img_num                          p1   p1_conf  p1_dog  \
0           1       Welsh_springer_spaniel  0.465074    True
1           1                     redbone  0.506826    True
2           1              German_shepherd  0.596461    True
3           1          Rhodesian_ridgeback  0.408143    True
4           1           miniature_pinscher  0.560311    True
5           1         Bernese_mountain_dog  0.651137    True
6           1                   box_turtle  0.933012   False
7           1                         chow  0.692517    True
8           1                shopping_cart  0.962465   False
9           1             miniature_poodle  0.201493    True
10          1             golden_retriever  0.775930    True
11          1                Gordon_setter  0.503672    True
12          1                 Walker_hound  0.260857    True
13          1                          pug  0.489814    True
14          1                    bloodhound  0.195217    True
15          1                        Lhasa  0.582330    True
16          1              English_setter  0.298617    True
17          1                          hen  0.965932   False
18          1            desktop_computer  0.086502   False
19          1           Italian_greyhound  0.176053    True
20          1                Maltese_dog  0.857531    True
21          1            three-toed_sloth  0.914671   False
22          1                          ox  0.416669   False
23          1            golden_retriever  0.858744    True
24          1                    malamute  0.336874    True
25          1                   guinea_pig  0.996496   False
26          1  soft-coated_wheaten_terrier  0.326467    True
27          1                   Chihuahua  0.978108    True
```

```
28          1      black-and-tan_coonhound  0.529139    True
29          1                         coho  0.404640   False
...       ...                          ...       ...     ...
2045        1               French_bulldog  0.999201    True
2046        1                  convertible  0.738995   False
2047        1                       kuvasz  0.309706    True
2048        2                    Chihuahua  0.793469    True
2049        1                      Samoyed  0.733942    True
2050        1             Mexican_hairless  0.330741    True
2051        2                     Pembroke  0.809197    True
2052        1                    limousine  0.130432   False
2053        1                       basset  0.821664    True
2054        1               French_bulldog  0.995026    True
2055        2                     Pembroke  0.809197    True
2056        3               Siberian_husky  0.700377    True
2057        1             golden_retriever  0.469760    True
2058        1             golden_retriever  0.714719    True
2059        1                      whippet  0.626152    True
2060        1             golden_retriever  0.953442    True
2061        1               French_bulldog  0.991650    True
2062        1                     Pembroke  0.966327    True
2063        1               French_bulldog  0.377417    True
2064        1                      Samoyed  0.957979    True
2065        1                     Pembroke  0.511319    True
2066        1                Irish_terrier  0.487574    True
2067        2                   Pomeranian  0.566142    True
2068        1                  Appenzeller  0.341703    True
2069        1    Chesapeake_Bay_retriever  0.425595    True
2070        2                       basset  0.555712    True
2071        1                  paper_towel  0.170278   False
2072        1                    Chihuahua  0.716012    True
2073        1                    Chihuahua  0.323581    True
2074        1                       orange  0.097049   False

                         p2   p2_conf  p2_dog                         p3  \
0                   collie  0.156665    True            Shetland_sheepdog
1          miniature_pinscher  0.074192    True          Rhodesian_ridgeback
2                  malinois  0.138584    True                  bloodhound
3                   redbone  0.360687    True           miniature_pinscher
4                Rottweiler  0.243682    True                     Doberman
5            English_springer  0.263788    True   Greater_Swiss_Mountain_dog
6                 mud_turtle  0.045885   False                      terrapin
7              Tibetan_mastiff  0.058279    True                     fur_coat
8              shopping_basket  0.014594   False             golden_retriever
9                  komondor  0.192305    True  soft-coated_wheaten_terrier
10             Tibetan_mastiff  0.093718    True           Labrador_retriever
11           Yorkshire_terrier  0.174201    True                     Pekinese
12            English_foxhound  0.175382    True                 Ibizan_hound
```

13	bull_mastiff	0.404722	True	French_bulldog
14	German_shepherd	0.078260	True	malinois
15	Shih-Tzu	0.166192	True	Dandie_Dinmont
16	Newfoundland	0.149842	True	borzoi
17	cock	0.033919	False	partridge
18	desk	0.085547	False	bookcase
19	toy_terrier	0.111884	True	basenji
20	toy_poodle	0.063064	True	miniature_poodle
21	otter	0.015250	False	great_grey_owl
22	Newfoundland	0.278407	True	groenendael
23	Chesapeake_Bay_retriever	0.054787	True	Labrador_retriever
24	Siberian_husky	0.147655	True	Eskimo_dog
25	skunk	0.002402	False	hamster
26	Afghan_hound	0.259551	True	briard
27	toy_terrier	0.009397	True	papillon
28	bloodhound	0.244220	True	flat-coated_retriever
29	barracouta	0.271485	False	gar
...	...	...	...	...
2045	Chihuahua	0.000361	True	Boston_bull
2046	sports_car	0.139952	False	car_wheel
2047	Great_Pyrenees	0.186136	True	Dandie_Dinmont
2048	toy_terrier	0.143528	True	can_opener
2049	Eskimo_dog	0.035029	True	Staffordshire_bullterrier
2050	sea_lion	0.275645	False	Weimaraner
2051	Rhodesian_ridgeback	0.054950	True	beagle
2052	tow_truck	0.029175	False	shopping_cart
2053	redbone	0.087582	True	Weimaraner
2054	pug	0.000932	True	bull_mastiff
2055	Rhodesian_ridgeback	0.054950	True	beagle
2056	Eskimo_dog	0.166511	True	malamute
2057	Labrador_retriever	0.184172	True	English_setter
2058	Tibetan_mastiff	0.120184	True	Labrador_retriever
2059	borzoi	0.194742	True	Saluki
2060	Labrador_retriever	0.013834	True	redbone
2061	boxer	0.002129	True	Staffordshire_bullterrier
2062	Cardigan	0.027356	True	basenji
2063	Labrador_retriever	0.151317	True	muzzle
2064	Pomeranian	0.013884	True	chow
2065	Cardigan	0.451038	True	Chihuahua
2066	Irish_setter	0.193054	True	Chesapeake_Bay_retriever
2067	Eskimo_dog	0.178406	True	Pembroke
2068	Border_collie	0.199287	True	ice_lolly
2069	Irish_terrier	0.116317	True	Indian_elephant
2070	English_springer	0.225770	True	German_short-haired_pointer
2071	Labrador_retriever	0.168086	True	spatula
2072	malamute	0.078253	True	kelpie
2073	Pekinese	0.090647	True	papillon
2074	bagel	0.085851	False	banana

	p3_conf	p3_dog
0	0.061428	True
1	0.072010	True
2	0.116197	True
3	0.222752	True
4	0.154629	True
5	0.016199	True
6	0.017885	False
7	0.054449	False
8	0.007959	True
9	0.082086	True
10	0.072427	True
11	0.109454	True
12	0.097471	True
13	0.048960	True
14	0.075628	True
15	0.089688	True
16	0.133649	True
17	0.000052	False
18	0.079480	False
19	0.111152	True
20	0.025581	True
21	0.013207	False
22	0.102643	True
23	0.014241	True
24	0.093412	True
25	0.000461	False
26	0.206803	True
27	0.004577	True
28	0.173810	True
29	0.189945	False
...	...	...
2045	0.000076	True
2046	0.044173	False
2047	0.086346	True
2048	0.032253	False
2049	0.029705	True
2050	0.134203	True
2051	0.038915	True
2052	0.026321	False
2053	0.026236	True
2054	0.000903	True
2055	0.038915	True
2056	0.111411	True
2057	0.073482	True
2058	0.105506	True
2059	0.027351	True

```
2060  0.007958    True
2061  0.001498    True
2062  0.004633    True
2063  0.082981   False
2064  0.008167    True
2065  0.029248    True
2066  0.118184    True
2067  0.076507    True
2068  0.193548   False
2069  0.076902   False
2070  0.175219    True
2071  0.040836   False
2072  0.031379    True
2073  0.068957    True
2074  0.076110   False

[2075 rows x 12 columns]
```

In [18]:
```
image_df.head()
```

Out[18]:

	tweet_id	jpg_url	img_num	p1	p1_conf	p1_dog	p2	p2_conf	p2_dog	p3	p3_conf	p3_dog
0	666020888022790149	https://pbs.twimg.com/media/CT4udn0WwAA0aMy.jpg	1	Welsh_springer_spaniel	0.465074	True	collie	0.156665	True	Shetland_sheepdog	0.061428	True
1	666029285002620928	https://pbs.twimg.com/media/CT42GRgUYAA5iDo.jpg	1	redbone	0.506826	True	miniature_pinscher	0.074192	True	Rhodesian_ridgeback	0.072010	True
2	666033412701032449	https://pbs.twimg.com/media/CT4521TWwAEvMyu.jpg	1	German_shepherd	0.596461	True	malinois	0.138584	True	bloodhound	0.116197	True
3	666044226329800704	https://pbs.twimg.com/media/CT5Dr8HUEAA-lEu.jpg	1	Rhodesian_ridgeback	0.408143	True	redbone	0.360687	True	miniature_pinscher	0.222752	True
4	666049248165822465	https://pbs.twimg.com/media/CT5IQmsXIAAKY4A.jpg	1	miniature_pinscher	0.560311	True	Rottweiler	0.243682	True	Doberman	0.154629	True

In [19]:
```
image_df.describe()
```

Out[19]:

	tweet_id	img_num	p1_conf	p2_conf	p3_conf
count	2.075000e+03	2075.000000	2075.000000	2.075000e+03	2.075000e+03
mean	7.384514e+17	1.203855	0.594548	1.345886e-01	6.032417e-02
std	6.785203e+16	0.561875	0.271174	1.006657e-01	5.090593e-02
min	6.660209e+17	1.000000	0.044333	1.011300e-08	1.740170e-10
25%	6.764835e+17	1.000000	0.364412	5.388625e-02	1.622240e-02

```
50%    7.119988e+17     1.000000     0.588230  1.181810e-01  4.944380e-02
75%    7.932034e+17     1.000000     0.843855  1.955655e-01  9.180755e-02
max    8.924206e+17     4.000000     1.000000  4.880140e-01  2.734190e-01
```

dog breeds are not consistently in p1,p2,p3 columns i.e lower or uppercase

```
In [20]: image_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2075 entries, 0 to 2074
Data columns (total 12 columns):
tweet_id    2075 non-null int64
jpg_url     2075 non-null object
img_num     2075 non-null int64
p1          2075 non-null object
p1_conf     2075 non-null float64
p1_dog      2075 non-null bool
p2          2075 non-null object
p2_conf     2075 non-null float64
p2_dog      2075 non-null bool
p3          2075 non-null object
p3_conf     2075 non-null float64
p3_dog      2075 non-null bool
dtypes: bool(3), float64(3), int64(2), object(4)
memory usage: 152.1+ KB
```

```
In [21]: # checks for duplicated entries in image_pred_df
         image_df[image_df.duplicated()].shape[0]
```

```
Out[21]: 0
```

```
In [22]: # Count of duplicate jpg_url
         image_df[image_df.jpg_url.duplicated()].shape[0]
```

```
Out[22]: 66
```

jpg_url contains duplicate items means duplicate image links

```
In [23]: image_df.img_num.value_counts()
```

```
Out[23]: 1    1780
         2     198
         3      66
         4      31
         Name: img_num, dtype: int64
```

2.3 Access df_tweet_json

In [25]:
```
# Display the df_tweet_json table
df_tweet_json
```

Out[25]:

	tweet_id	retweet_count	favorite_count
0	892420643555336193	7437	35276
1	892177421306343426	5527	30527
2	891815181378084864	3652	22954
3	891689557279858688	7616	38561
4	891327558926688256	8197	36842
5	891087950875897856	2751	18574
6	890971913173991426	1780	10797
7	890729181411237888	16634	59410
8	890609185150312448	3799	25556
9	890240255349198849	6446	29152
10	890006608113172480	6467	28116
11	889880896479866881	4390	25586
12	889665388333682689	8817	43904
13	889638837579907072	3943	24701
14	889531135344209921	1991	13916
15	889278841981685760	4691	23057
16	888917238123831296	3950	26659
17	888804989199671297	3731	23406
18	888554962724278272	3048	18050
19	888078434458587136	3059	19945
20	887705289381826560	4762	27730
21	887517139158093824	10372	42462
22	887473957103951883	15844	62823
23	887343217045368832	9282	30829
24	887101392804085760	5260	28077
25	886983233522544640	6732	31833
26	886736880519319552	2801	10930
27	886680336477933568	3946	20609
28	886366144734445568	2790	19337
29	886267009285017600	4	110
...	...	...	...
2301	666411507551481857	285	395
2302	666407126856765440	31	98
2303	666396247373291520	73	155
2304	666373753744588802	78	169
2305	666362758909284353	501	698
2306	666353288456101888	64	194
2307	666345417576210432	128	267
2308	666337882303524864	81	178
2309	666293911632134144	311	450
2310	666287406224695296	57	131
2311	666273097616637952	70	156
2312	666268910803644416	32	94

```
2313  666104133288665088          5783          13256
2314  666102155909144576            11             69
2315  666099513787052032            57            140
2316  666094000022159362            66            153
2317  666082916733198337            41            100
2318  666073100786774016           141            283
2319  666071193221509120            52            133
2320  666063827256086533           190            438
2321  666058600524156928            51            103
2322  666057090499244032           119            263
2323  666055525042405380           213            402
2324  666051853826850816           746           1092
2325  666050758794694657            51            119
2326  666049248165822465            39             95
2327  666044226329800704           123            263
2328  666033412701032449            39            108
2329  666029285002620928            41            118
2330  666020888022790149           445           2350

[2331 rows x 3 columns]
```

```
In [26]: df_tweet_json.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2331 entries, 0 to 2330
Data columns (total 3 columns):
tweet_id          2331 non-null object
retweet_count     2331 non-null object
favorite_count    2331 non-null object
dtypes: object(3)
memory usage: 54.7+ KB
```

```
In [27]: df_tweet_json.head()
```

```
Out[27]:              tweet_id retweet_count favorite_count
         0  892420643555336193          7437          35276
         1  892177421306343426          5527          30527
         2  891815181378084864          3652          22954
         3  891689557279858688          7616          38561
         4  891327558926688256          8197          36842
```

```
In [28]: df_tweet_json.describe()
```

```
Out[28]:                   tweet_id  retweet_count  favorite_count
         count                 2331           2331            2331
         unique                2331           1673            1984
         top     870656317836468226            710               0
         freq                     1              6             163
```

```
In [29]: df_tweet_json.sample(20)
```

```
Out[29]:
```

	tweet_id	retweet_count	favorite_count
2323	666055525042405380	213	402
1322	704480331685040129	1032	3293
307	834209720923721728	4607	20112
1353	701601587219795968	436	2033
582	798673117451325440	5507	0
1531	688894073864884227	659	2177
1250	709198395643068416	614	2375
150	861288531465048066	3790	16052
1179	716285507865542656	1008	2682
1395	698342080612007937	930	2196
2195	668484198282485761	216	393
1447	693993230313091072	389	1821
493	812372279581671427	3623	13559
1084	734559631394082816	382	1448
558	800859414831898624	97	684
63	879674319642796034	10	291
874	759047813560868866	1953	6358
1682	680805554198020098	640	2078
875	758854675097526272	879	3481
2310	666287406224695296	57	131

```
In [30]: df_tweet_json[df_tweet_json.duplicated()]
```

```
Out[30]: Empty DataFrame
         Columns: [tweet_id, retweet_count, favorite_count]
         Index: []
```

No Duplicate entries present

2.4 Quality Issues

2.4.1 df:

1) Missing data in the following columns: in_reply_to_status_id, in_reply_to_user_id, Retweeted_status_id, retweeted_status_user_id, retweeted_status_timestamp, expanded_urls
2) This dataset includes retweets, which means there is duplicated data
3) Timestamp and retweeted_status_timestamp is an object and not correct datetime frame.
4) The source column still has the HTML tags
5) Dogs name have 'None', or 'a', or 'an.' and some more lower case words as names
6) Multiple dog stages occurs such as 'doggo puppo', 'doggo pupper', 'doggo floofer'

2.4.2 image_df:

1) Dog breeds are not consistently in p1,p2,p3 columns: this column contains dog breeds name starts with upper case or lower case.

2.4.3 df_tweet_json:

1) Date type of column tweet_id is an object. It should be an int in order to merge it to the master df.

2.5 Tidiness Issues

2.5.1 df:

1) The variable for the dog's stage (dogoo, floofer, pupper, puppo) is spread in different columns which should be in one column.

2.5.2 image_df:

2) This data set is part of the same observational unit as the data in the archive_df

2.5.3 df_tweet_json:

3) This data set is also part of the same observational unit as the data in the archive_df

3 Cleaning Data

```
In [31]: #Making a copy of the dataframes before cleaning
         df_clean = df.copy()
         image_df_clean = image_df.copy()
         tweet_json_clean = df_tweet_json.copy()
```

3.1 DEFINE-CODE-TEST

1) Convert the tweet_id in tweet_json_clean dataframe into int type for merging into master dataframe

2) Creates a predicted dog breed column, based on the the confidence level of minimum 20% and 'p1_dog', 'p2_dog' and 'p3_dog' statements

3) Create one column for the various dog types: doggo, floofer, pupper, puppo, 'doggo, puppo', 'doggo, pupper', 'doggo, floofer' ascolumn name ' type ' with the categorical dtype

4) Merge the copied df_clean, image_df_clean, and tweet_json_clean dataframes

5) Convert the tweet_id in master_df into object type as there is no use for maths operation in tweet_id

6) Replace 'a', 'an', 'the', 'None' and other lower case words with NaN in name column

7) Remove Inconsistency in pred_breed

8) Delete retweets

9) Remove columns no longer needed: in_reply_to_status_id, in_reply_to_user_id, retweeted_status_id, retweeted_status_user_id, and retweeted_status_timestamp

10) Change the timestamp to correct datetime format

11) Removing HTML tags from source column

12) Dog ratings get standardized for denom of 10.

3.1.1 1. Convert the tweet_id in tweet_json_clean dataframe into int type for merging into master dataframe

```
In [32]: tweet_json_clean['tweet_id'] = tweet_json_clean['tweet_id'].astype('int64')

In [33]: tweet_json_clean.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2331 entries, 0 to 2330
Data columns (total 3 columns):
tweet_id          2331 non-null int64
retweet_count     2331 non-null object
favorite_count    2331 non-null object
dtypes: int64(1), object(2)
memory usage: 54.7+ KB
```

3.1.2 2. Creates a predicted dog breed column, based on the the confidence level of minimum 20% and 'p1_dog', 'p2_dog' and 'p3_dog' statements

```
In [34]: image_df_clean.sample()

Out[34]:                 tweet_id                                        jpg_url  \
         1170  736225175608430592  https://pbs.twimg.com/media/CjeY5DKXEAA3WkD.jpg

               img_num                  p1   p1_conf  p1_dog  \
         1170        1  Labrador_retriever  0.399217    True

                                    p2  p2_conf  p2_dog             p3   p3_conf  \
         1170  West_Highland_white_terrier  0.13771    True  cocker_spaniel  0.062033

               p3_dog
         1170    True

In [35]: image_df_clean['pred_breed'] = [df['p1'] if df['p1_dog'] == True and df['p1_conf'] > 0.
                                 else df['p2'] if df['p2_dog'] == True and df['p2_conf'] > 0.2
                                 else df['p3'] if df['p3_dog'] == True and df['p3_conf'] > 0.2
                                 else np.nan for index, df in image_df_clean.iterrows()]
```

```
In [36]: # Drop 'p1', 'p1_dog', 'p1_conf','p2', 'p2_dog', 'p2_conf','p3', 'p3_dog', 'p3_conf' co
         image_df_clean.drop(['p1', 'p1_dog', 'p1_conf','p2', 'p2_dog', 'p2_conf','p3', 'p3_dog'

In [37]: image_df_clean.head()
```

Out[37]:

	tweet_id	jpg_url	\
0	666020888022790149	https://pbs.twimg.com/media/CT4udn0WwAA0aMy.jpg	
1	666029285002620928	https://pbs.twimg.com/media/CT42GRgUYAA5iDo.jpg	
2	666033412701032449	https://pbs.twimg.com/media/CT4521TWwAEvMyu.jpg	
3	666044226329800704	https://pbs.twimg.com/media/CT5Dr8HUEAA-lEu.jpg	
4	666049248165822465	https://pbs.twimg.com/media/CT5IQmsXIAAKY4A.jpg	

	img_num	pred_breed
0	1	Welsh_springer_spaniel
1	1	redbone
2	1	German_shepherd
3	1	Rhodesian_ridgeback
4	1	miniature_pinscher

3.1.3 3) Create one column for the various dog types: doggo, floofer, pupper, puppo, 'doggo, puppo', 'doggo, pupper', 'doggo, floofer' ascolumn name ' type ' with the categorical dtype

```
In [38]: #Number of columns in df_clean
         df_clean.columns
```

```
Out[38]: Index(['tweet_id', 'in_reply_to_status_id', 'in_reply_to_user_id', 'timestamp',
                'source', 'text', 'retweeted_status_id', 'retweeted_status_user_id',
                'retweeted_status_timestamp', 'expanded_urls', 'rating_numerator',
                'rating_denominator', 'name', 'doggo', 'floofer', 'pupper', 'puppo'],
               dtype='object')
```

```
In [39]: # as there are separate columns for dogs type 'doggo','floofer','pupper'and so on...
         #i will convert them into one column
         df_clean.doggo.replace(np.NaN, '', inplace=True)
         df_clean.floofer.replace(np.NaN, '', inplace=True)
         df_clean.pupper.replace(np.NaN, '', inplace=True)
         df_clean.puppo.replace(np.NaN, '', inplace=True)
         df_clean.doggo.replace('None', '', inplace=True)
         df_clean.floofer.replace('None', '', inplace=True)
         df_clean.pupper.replace('None', '', inplace=True)
         df_clean.puppo.replace('None', '', inplace=True)

In [40]: df_clean['stage'] = df_clean.doggo + df_clean.floofer + df_clean.pupper + df_clean.pupp
         df_clean.loc[df_clean.stage == 'doggopupper', 'stage'] = 'doggo, pupper'
         df_clean.loc[df_clean.stage == 'doggopuppo', 'stage'] = 'doggo, puppo'
         df_clean.loc[df_clean.stage == 'doggofloofer', 'stage'] = 'doggo, floofer'

In [41]: # Convert the stage in df_clean into categorical dtype
         df_clean['stage'] = df_clean['stage'].astype('category')
```

```
In [42]: # drop 'doggo', 'floofer', 'pupper', 'puppo' columns
         df_clean.drop(['doggo', 'floofer', 'pupper', 'puppo'], axis=1, inplace=True)
         df_clean.stage.replace('', np.nan, inplace=True)

In [43]: df_clean.info()
         df_clean.stage.value_counts()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2356 entries, 0 to 2355
Data columns (total 14 columns):
tweet_id                      2356 non-null int64
in_reply_to_status_id         78 non-null float64
in_reply_to_user_id           78 non-null float64
timestamp                     2356 non-null object
source                        2356 non-null object
text                          2356 non-null object
retweeted_status_id           181 non-null float64
retweeted_status_user_id      181 non-null float64
retweeted_status_timestamp    181 non-null object
expanded_urls                 2297 non-null object
rating_numerator              2356 non-null int64
rating_denominator            2356 non-null int64
name                          2356 non-null object
stage                         380 non-null category
dtypes: category(1), float64(4), int64(3), object(6)
memory usage: 242.0+ KB


Out[43]: pupper            245
         doggo              83
         puppo              29
         doggo, pupper      12
         floofer             9
         doggo, puppo        1
         doggo, floofer      1
                             0
         Name: stage, dtype: int64

In [44]: df_clean.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2356 entries, 0 to 2355
Data columns (total 14 columns):
tweet_id                      2356 non-null int64
in_reply_to_status_id         78 non-null float64
in_reply_to_user_id           78 non-null float64
timestamp                     2356 non-null object
source                        2356 non-null object
text                          2356 non-null object
```

```
retweeted_status_id           181 non-null float64
retweeted_status_user_id      181 non-null float64
retweeted_status_timestamp    181 non-null object
expanded_urls                 2297 non-null object
rating_numerator              2356 non-null int64
rating_denominator            2356 non-null int64
name                          2356 non-null object
stage                         380 non-null category
dtypes: category(1), float64(4), int64(3), object(6)
memory usage: 242.0+ KB
```

3.1.4 4) Merge the copied df_clean, image_df_clean, and tweet_json_clean dataframes

```
In [45]: from functools import reduce
         data = [df_clean, image_df_clean, tweet_json_clean]
         main_df = reduce(lambda left, right:  pd.merge(left, right,on = 'tweet_id'), data)

In [46]: main_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 2059 entries, 0 to 2058
Data columns (total 19 columns):
tweet_id                      2059 non-null int64
in_reply_to_status_id         23 non-null float64
in_reply_to_user_id           23 non-null float64
timestamp                     2059 non-null object
source                        2059 non-null object
text                          2059 non-null object
retweeted_status_id           72 non-null float64
retweeted_status_user_id      72 non-null float64
retweeted_status_timestamp    72 non-null object
expanded_urls                 2059 non-null object
rating_numerator              2059 non-null int64
rating_denominator            2059 non-null int64
name                          2059 non-null object
stage                         318 non-null category
jpg_url                       2059 non-null object
img_num                       2059 non-null int64
pred_breed                    1460 non-null object
retweet_count                 2059 non-null object
favorite_count                2059 non-null object
dtypes: category(1), float64(4), int64(4), object(10)
memory usage: 308.0+ KB
```

3.1.5 5) Convert the tweet_id in master_df into object type as there is no use for maths operation in tweet_id

```
In [47]: main_df['tweet_id'] = main_df['tweet_id'].astype('object')

In [48]: main_df.info()

<class 'pandas.core.frame.DataFrame'>
Int64Index: 2059 entries, 0 to 2058
Data columns (total 19 columns):
tweet_id                      2059 non-null object
in_reply_to_status_id         23 non-null float64
in_reply_to_user_id           23 non-null float64
timestamp                     2059 non-null object
source                        2059 non-null object
text                          2059 non-null object
retweeted_status_id           72 non-null float64
retweeted_status_user_id      72 non-null float64
retweeted_status_timestamp    72 non-null object
expanded_urls                 2059 non-null object
rating_numerator              2059 non-null int64
rating_denominator            2059 non-null int64
name                          2059 non-null object
stage                         318 non-null category
jpg_url                       2059 non-null object
img_num                       2059 non-null int64
pred_breed                    1460 non-null object
retweet_count                 2059 non-null object
favorite_count                2059 non-null object
dtypes: category(1), float64(4), int64(3), object(11)
memory usage: 308.0+ KB
```

3.1.6 6. Replace 'a', 'an', 'the', 'None' and other lower case words with NaN in name column

```
In [49]: words = main_df[main_df.name.str.islower()].name.unique()

In [50]: main_df['name'] = main_df['name'].replace(words, np.nan)
         main_df['name'] = main_df['name'].replace('None', np.nan)

In [51]: main_df['name'].dropna()

Out[51]: 0          Phineas
         1            Tilly
         2           Archie
         3            Darla
         4         Franklin
         6              Jax
         8             Zoey
```

9	Cassie
10	Koda
11	Bruno
13	Ted
14	Stuart
15	Oliver
16	Jim
17	Zeke
18	Ralphus
19	Gerald
20	Jeffrey
22	Canela
25	Maya
26	Mingus
27	Derek
28	Roscoe
29	Waffles
30	Jimbo
31	Maisey
32	Lilly
34	Earl
35	Lola
36	Kevin
...	
1972	Dook
1974	Hall
1975	Philippe
1978	Reese
1979	Cupcake
1983	Biden
1984	Fwed
1986	Genevieve
1987	Joshwa
1990	Timison
1993	Clarence
1994	Kenneth
1995	Churlie
1996	Bradlay
1997	Pipsy
1999	Gabe
2000	Clybe
2001	Dave
2003	Keet
2005	Klevin
2006	Carll
2011	Jeph
2012	Jockson
2015	Josep

```
2016         Lugan
2018    Christoper
2020       Jimothy
2021      Kreggory
2022         Scout
2028        Walter
Name: name, Length: 1386, dtype: object
```

```
In [52]: main_df.name.value_counts()
```

```
Out[52]: Cooper       10
         Penny        10
         Oliver       10
         Tucker       10
         Charlie      10
         Lucy          9
         Sadie         8
         Bo            8
         Lola          8
         Winston       8
         Daisy         7
         Toby          7
         Scout         6
         Stanley       6
         Koda          6
         Milo          6
         Bella         6
         Jax           6
         Dave          6
         Rusty         6
         Bailey        6
         Louis         5
         Alfie         5
         Oscar         5
         Larry         5
         Leo           5
         Buddy         5
         Chester       5
         Sophie        4
         Bear          4
                      ..
         Barney        1
         Strider       1
         Miguel        1
         Ridley        1
         Chaz          1
         Kota          1
         Mairi         1
```

```
Enchilada     1
Mattie        1
Brudge        1
Noosh         1
Bodie         1
Huck          1
Alexander     1
Mojo          1
Venti         1
Schnozz       1
Howie         1
Bode          1
Keurig        1
Rodney        1
Lilah         1
Nida          1
Pawnd         1
Shawwn        1
Samsom        1
Harrison      1
Reagan        1
Burt          1
Lucky         1
Name: name, Length: 911, dtype: int64
```

3.1.7 7) Delete Retweets

```
In [53]: # Delete the rows which contains retweets
         main_df = main_df.drop(main_df[(main_df['in_reply_to_status_id'].isnull() == False) | (
```

```
In [54]: main_df.shape[0]
```

```
Out[54]: 1964
```

```
In [55]: main_df.shape
```

```
Out[55]: (1964, 19)
```

3.1.8 8) Remove columns no longer needed: in_reply_to_status_id, in_reply_to_user_id, retweeted_status_id, retweeted_status_user_id, and retweeted_status_timestamp

```
In [56]: # drop the reply status and retweet status columns
         main_df.drop(['in_reply_to_status_id', 'in_reply_to_user_id','retweeted_status_id', 're
                'retweeted_status_timestamp'], axis=1, inplace=True)
```

```
In [57]: main_df.columns
```

```
Out[57]: Index(['tweet_id', 'timestamp', 'source', 'text', 'expanded_urls',
                'rating_numerator', 'rating_denominator', 'name', 'stage', 'jpg_url',
                'img_num', 'pred_breed', 'retweet_count', 'favorite_count'],
               dtype='object')
```

```
In [58]: main_df.info()

<class 'pandas.core.frame.DataFrame'>
Int64Index: 1964 entries, 0 to 2058
Data columns (total 14 columns):
tweet_id              1964 non-null object
timestamp             1964 non-null object
source                1964 non-null object
text                  1964 non-null object
expanded_urls         1964 non-null object
rating_numerator      1964 non-null int64
rating_denominator    1964 non-null int64
name                  1342 non-null object
stage                 302 non-null category
jpg_url               1964 non-null object
img_num               1964 non-null int64
pred_breed            1395 non-null object
retweet_count         1964 non-null object
favorite_count        1964 non-null object
dtypes: category(1), int64(3), object(10)
memory usage: 217.1+ KB
```

3.1.9 9) Change the timestamp to correct datetime format

```
In [59]: main_df['timestamp'].sample(5)

Out[59]: 842     2016-06-17 00:05:25 +0000
         1991    2015-11-19 03:10:02 +0000
         198     2017-03-23 18:07:10 +0000
         596     2016-09-21 17:42:10 +0000
         1038    2016-03-17 00:58:46 +0000
         Name: timestamp, dtype: object

In [60]: main_df['timestamp'] = pd.to_datetime(main_df['timestamp'], format='%Y-%m-%d %H:%M:%S')

In [61]: main_df['timestamp'].sample(5)

Out[61]: 260    2017-02-16 17:00:25
         1887   2015-11-24 03:29:51
         64     2017-06-23 16:00:04
         344    2017-01-12 00:55:47
         363    2017-01-05 21:29:55
         Name: timestamp, dtype: datetime64[ns]

In [62]: main_df['timestamp'].describe()

Out[62]: count                    1964
         unique                   1964
```

```
top       2016-08-04 22:52:29
freq                        1
first     2015-11-15 22:32:08
last      2017-08-01 16:23:56
Name: timestamp, dtype: object
```

3.1.10 10) Removing HTML tags from source column

```
In [63]: href = main_df["source"].str.split('"', expand = True)
         main_df["source"] = href[1]

In [64]: main_df.head()
```

Out[64]:

	tweet_id	timestamp	source	text	expanded_urls	rating_numerator	rating_denominator	name	stage	jpg_url	img_num	pred_breed
0	892420643555336193	2017-08-01 16:23:56	http://twitter.com/download/iphone	This is Phineas. He's a mystical boy. Only eve...	https://twitter.com/dog_rates/status/892420643...	13	10	Phineas	NaN	https://pbs.twimg.com/media/DGKD1-bXoAAIAUK.jpg	1	NaN
1	892177421306343426	2017-08-01 00:17:27	http://twitter.com/download/iphone	This is Tilly. She's just checking pup on you...	https://twitter.com/dog_rates/status/892177421...	13	10	Tilly	NaN	https://pbs.twimg.com/media/DGGmoV4XsAAUL6n.jpg	1	Chihuahua
2	891815181378084864	2017-07-31 00:18:03	http://twitter.com/download/iphone	This is Archie. He is a rare Norwegian Pouncin...	https://twitter.com/dog_rates/status/891815181...	12	10	Archie	NaN	https://pbs.twimg.com/media/DGBdLU1WsAANxJ9.jpg	1	Chihuahua
3	891689557279858688	2017-07-30 15:58:51	http://twitter.com/download/iphone	This is Darla. She commenced a snooze mid meal...	https://twitter.com/dog_rates/status/891689557...	13	10	Darla	NaN	https://pbs.twimg.com/media/DF_q7IAWsAEuuN8.jpg	1	NaN
4	891327558926688256	2017-07-29 16:00:24	http://twitter.com/download/iphone	This is Franklin. He would like you to stop ca...	https://twitter.com/dog_rates/status/891327558...	12	10	Franklin	NaN	https://pbs.twimg.com/media/DF6hr6BUMAAzZgT.jpg	2	basset

```
  retweet_count favorite_count
0          7437          35276
1          5527          30527
2          3652          22954
3          7616          38561
4          8197          36842
```

In [65]: href

Out[65]:

```
              0                                      1      2         3  \
0     <a href=  http://twitter.com/download/iphone   rel=  nofollow
1     <a href=  http://twitter.com/download/iphone   rel=  nofollow
2     <a href=  http://twitter.com/download/iphone   rel=  nofollow
3     <a href=  http://twitter.com/download/iphone   rel=  nofollow
4     <a href=  http://twitter.com/download/iphone   rel=  nofollow
5     <a href=  http://twitter.com/download/iphone   rel=  nofollow
6     <a href=  http://twitter.com/download/iphone   rel=  nofollow
7     <a href=  http://twitter.com/download/iphone   rel=  nofollow
8     <a href=  http://twitter.com/download/iphone   rel=  nofollow
9     <a href=  http://twitter.com/download/iphone   rel=  nofollow
10    <a href=  http://twitter.com/download/iphone   rel=  nofollow
11    <a href=  http://twitter.com/download/iphone   rel=  nofollow
12    <a href=  http://twitter.com/download/iphone   rel=  nofollow
13    <a href=  http://twitter.com/download/iphone   rel=  nofollow
14    <a href=  http://twitter.com/download/iphone   rel=  nofollow
15    <a href=  http://twitter.com/download/iphone   rel=  nofollow
16    <a href=  http://twitter.com/download/iphone   rel=  nofollow
17    <a href=  http://twitter.com/download/iphone   rel=  nofollow
18    <a href=  http://twitter.com/download/iphone   rel=  nofollow
19    <a href=  http://twitter.com/download/iphone   rel=  nofollow
20    <a href=  http://twitter.com/download/iphone   rel=  nofollow
21    <a href=  http://twitter.com/download/iphone   rel=  nofollow
22    <a href=  http://twitter.com/download/iphone   rel=  nofollow
23    <a href=  http://twitter.com/download/iphone   rel=  nofollow
24    <a href=  http://twitter.com/download/iphone   rel=  nofollow
25    <a href=  http://twitter.com/download/iphone   rel=  nofollow
26    <a href=  http://twitter.com/download/iphone   rel=  nofollow
27    <a href=  http://twitter.com/download/iphone   rel=  nofollow
28    <a href=  http://twitter.com/download/iphone   rel=  nofollow
29    <a href=  http://twitter.com/download/iphone   rel=  nofollow
...       ...                                    ...    ...       ...
2029  <a href=  http://twitter.com/download/iphone   rel=  nofollow
2030  <a href=  http://twitter.com/download/iphone   rel=  nofollow
2031  <a href=  http://twitter.com/download/iphone   rel=  nofollow
2032  <a href=  http://twitter.com/download/iphone   rel=  nofollow
2033  <a href=  http://twitter.com/download/iphone   rel=  nofollow
2034  <a href=  http://twitter.com/download/iphone   rel=  nofollow
2035  <a href=  http://twitter.com/download/iphone   rel=  nofollow
```

```
2036  <a href=  http://twitter.com/download/iphone   rel=  nofollow
2037  <a href=  http://twitter.com/download/iphone   rel=  nofollow
2038  <a href=  http://twitter.com/download/iphone   rel=  nofollow
2039  <a href=  http://twitter.com/download/iphone   rel=  nofollow
2040  <a href=  http://twitter.com/download/iphone   rel=  nofollow
2041  <a href=  http://twitter.com/download/iphone   rel=  nofollow
2042  <a href=  http://twitter.com/download/iphone   rel=  nofollow
2043  <a href=  http://twitter.com/download/iphone   rel=  nofollow
2044  <a href=  http://twitter.com/download/iphone   rel=  nofollow
2045  <a href=  http://twitter.com/download/iphone   rel=  nofollow
2046  <a href=  http://twitter.com/download/iphone   rel=  nofollow
2047  <a href=  http://twitter.com/download/iphone   rel=  nofollow
2048  <a href=  http://twitter.com/download/iphone   rel=  nofollow
2049  <a href=  http://twitter.com/download/iphone   rel=  nofollow
2050  <a href=  http://twitter.com/download/iphone   rel=  nofollow
2051  <a href=  http://twitter.com/download/iphone   rel=  nofollow
2052  <a href=  http://twitter.com/download/iphone   rel=  nofollow
2053  <a href=  http://twitter.com/download/iphone   rel=  nofollow
2054  <a href=  http://twitter.com/download/iphone   rel=  nofollow
2055  <a href=  http://twitter.com/download/iphone   rel=  nofollow
2056  <a href=  http://twitter.com/download/iphone   rel=  nofollow
2057  <a href=  http://twitter.com/download/iphone   rel=  nofollow
2058  <a href=  http://twitter.com/download/iphone   rel=  nofollow

                          4
0     >Twitter for iPhone</a>
1     >Twitter for iPhone</a>
2     >Twitter for iPhone</a>
3     >Twitter for iPhone</a>
4     >Twitter for iPhone</a>
5     >Twitter for iPhone</a>
6     >Twitter for iPhone</a>
7     >Twitter for iPhone</a>
8     >Twitter for iPhone</a>
9     >Twitter for iPhone</a>
10    >Twitter for iPhone</a>
11    >Twitter for iPhone</a>
12    >Twitter for iPhone</a>
13    >Twitter for iPhone</a>
14    >Twitter for iPhone</a>
15    >Twitter for iPhone</a>
16    >Twitter for iPhone</a>
17    >Twitter for iPhone</a>
18    >Twitter for iPhone</a>
19    >Twitter for iPhone</a>
20    >Twitter for iPhone</a>
21    >Twitter for iPhone</a>
22    >Twitter for iPhone</a>
```

```
23    >Twitter for iPhone</a>
24    >Twitter for iPhone</a>
25    >Twitter for iPhone</a>
26    >Twitter for iPhone</a>
27    >Twitter for iPhone</a>
28    >Twitter for iPhone</a>
29    >Twitter for iPhone</a>
...                        ...
2029  >Twitter for iPhone</a>
2030  >Twitter for iPhone</a>
2031  >Twitter for iPhone</a>
2032  >Twitter for iPhone</a>
2033  >Twitter for iPhone</a>
2034  >Twitter for iPhone</a>
2035  >Twitter for iPhone</a>
2036  >Twitter for iPhone</a>
2037  >Twitter for iPhone</a>
2038  >Twitter for iPhone</a>
2039  >Twitter for iPhone</a>
2040  >Twitter for iPhone</a>
2041  >Twitter for iPhone</a>
2042  >Twitter for iPhone</a>
2043  >Twitter for iPhone</a>
2044  >Twitter for iPhone</a>
2045  >Twitter for iPhone</a>
2046  >Twitter for iPhone</a>
2047  >Twitter for iPhone</a>
2048  >Twitter for iPhone</a>
2049  >Twitter for iPhone</a>
2050  >Twitter for iPhone</a>
2051  >Twitter for iPhone</a>
2052  >Twitter for iPhone</a>
2053  >Twitter for iPhone</a>
2054  >Twitter for iPhone</a>
2055  >Twitter for iPhone</a>
2056  >Twitter for iPhone</a>
2057  >Twitter for iPhone</a>
2058  >Twitter for iPhone</a>

[1964 rows x 5 columns]
```

```
In [66]: main_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1964 entries, 0 to 2058
Data columns (total 14 columns):
tweet_id              1964 non-null object
timestamp             1964 non-null datetime64[ns]
```

```
source                1964 non-null object
text                  1964 non-null object
expanded_urls         1964 non-null object
rating_numerator      1964 non-null int64
rating_denominator    1964 non-null int64
name                  1342 non-null object
stage                 302 non-null category
jpg_url               1964 non-null object
img_num               1964 non-null int64
pred_breed            1395 non-null object
retweet_count         1964 non-null object
favorite_count        1964 non-null object
dtypes: category(1), datetime64[ns](1), int64(3), object(9)
memory usage: 217.1+ KB
```

```
In [67]: main_df.describe()
```

Out[67]:

	rating_numerator	rating_denominator	img_num
count	1964.000000	1964.000000	1964.000000
mean	12.223014	10.479124	1.202138
std	41.708155	6.865424	0.559615
min	0.000000	2.000000	1.000000
25%	10.000000	10.000000	1.000000
50%	11.000000	10.000000	1.000000
75%	12.000000	10.000000	1.000000
max	1776.000000	170.000000	4.000000

3.1.11 12. Standardize dog ratings

```
In [68]: ratings = main_df.text.str.extract('((?:\d+\.)?\d+)\/(\d+)', expand=True)
```

```
In [69]: main_df.rating_numerator = ratings
         main_df['rating_numerator'] = main_df['rating_numerator'].astype('float64')
```

```
In [70]: # standardizing to a denominator of 10 for groups of dogs:

         rating_num = [int(round(num/(denom/10)))  if denom != 10 and num/denom <= 2
                                    else num for num, denom in zip(main_df['rating_numerator'],
         rating_denom = [10 if denom != 10 and num/denom <= 2
                                     else denom for num, denom in zip(main_df['rating_numerator
         main_df['rating_numerator'] = rating_num
         main_df['rating_denominator'] = rating_denom

         main_df = main_df.drop(main_df[((main_df['rating_denominator'] != 10) | (main_df['ratin
```

```
In [71]: main_df['rating_numerator'].unique()
```

```
Out[71]: array([ 13.  ,  12.  ,  14.  ,  13.5 ,  11.  ,   6.  ,  10.  ,   0.  ,
                  9.75,   5.  ,  11.27,   3.  ,   7.  ,   8.  ,   9.  ,   4.  ,
                  2.  ,  11.26,   1.  ])
```

```
In [72]: main_df['rating_denominator'].unique()

Out[72]: array([10])
```

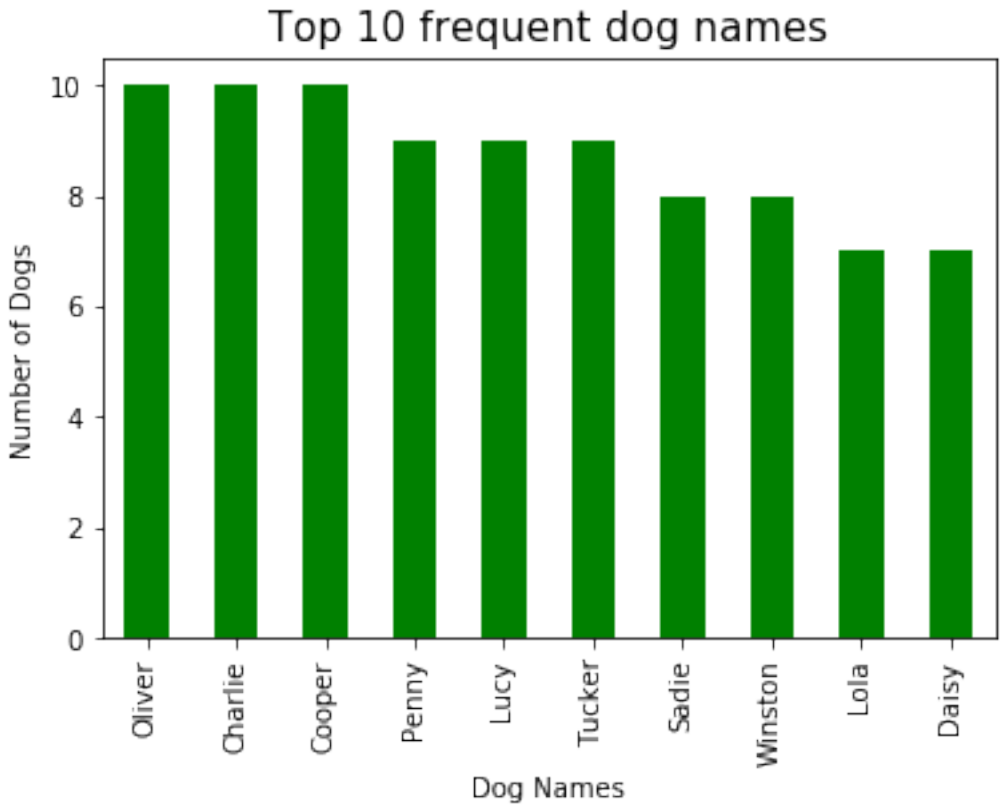
4 Storing, Analyzing, and Visualizing Data

```
In [73]: # storing main dataframe as csv
         main_df.to_csv('twitter_archive_master.csv', encoding='utf-8', index=False)

In [74]: # read twitter_archive_master.csv
         df1 = pd.read_csv('twitter_archive_master.csv')
         df1.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1961 entries, 0 to 1960
Data columns (total 14 columns):
tweet_id              1961 non-null int64
timestamp             1961 non-null object
source                1961 non-null object
text                  1961 non-null object
expanded_urls         1961 non-null object
rating_numerator      1961 non-null float64
rating_denominator    1961 non-null int64
name                  1340 non-null object
stage                 302 non-null object
jpg_url               1961 non-null object
img_num               1961 non-null int64
pred_breed            1394 non-null object
retweet_count         1961 non-null int64
favorite_count        1961 non-null int64
dtypes: float64(1), int64(5), object(8)
memory usage: 214.6+ KB


In [75]: df1.describe()

Out[75]:            tweet_id  rating_numerator  rating_denominator      img_num  \
         count  1.961000e+03       1961.000000              1961.0  1961.000000
         mean   7.358030e+17         10.528700                10.0     1.202448
         std    6.745542e+16          2.179024                 0.0     0.559987
         min    6.660209e+17          0.000000                10.0     1.000000
         25%    6.758457e+17         10.000000                10.0     1.000000
         50%    7.087111e+17         11.000000                10.0     1.000000
         75%    7.877176e+17         12.000000                10.0     1.000000
         max    8.924206e+17         14.000000                10.0     4.000000

                retweet_count  favorite_count
         count    1961.000000     1961.000000
```

```
mean      2385.981642     8108.614482
std       4269.800622    11936.349859
min         11.000000       69.000000
25%        530.000000     1738.000000
50%       1154.000000     3648.000000
75%       2722.000000    10122.000000
max      75085.000000   151947.000000
```

4.1 What are the 10 most frequent dog names?

```
In [84]: df1['name'].value_counts()[0:10].sort_values(ascending=False).plot(kind = 'bar', color=
         plt.ylabel('Number of Dogs')
         plt.title('Top 10 frequent dog names', size=15)
         plt.xlabel('Dog Names')
         plt.plot();
```



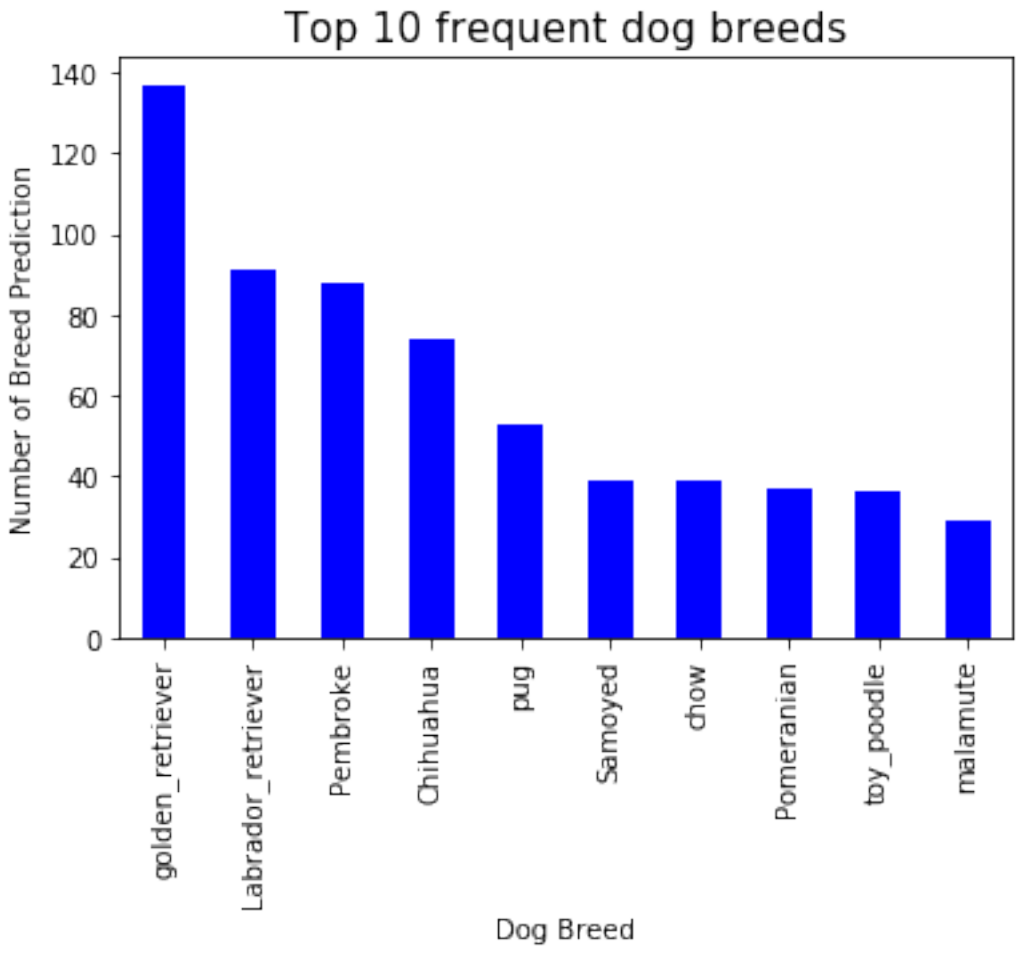
Most of the dogs are of names: OLiver,Charlie, Cooper, Penny, Lucy, Tucker, Sadie, Winston, Lola, Daisy Also, check the count below:

```
In [77]: #Top 10 frequent dog names
         df1['name'].value_counts()[0:10].sort_values(ascending=False)
```

```
Out[77]: Oliver     10
         Charlie    10
         Cooper     10
         Penny       9
         Lucy        9
         Tucker      9
         Sadie       8
         Winston     8
         Lola        7
         Daisy       7
         Name: name, dtype: int64
```

4.2 What are the 10 most frequent predicted dog breeds?

```
In [89]: df1['pred_breed'].value_counts()[0:10].sort_values(ascending=False).plot(kind = 'bar',
         plt.ylabel('Number of Breed Prediction')
         plt.title('Top 10 frequent dog breeds', size=15)
         plt.xlabel('Dog Breed')
         plt.plot();
```



Most of the dogs have golden retriever, labrador retriever as a breed which all are rated

```
In [79]: #Top 10 frequent dog breeds
         df1['pred_breed'].value_counts()[0:10].sort_values(ascending=False)
```

```
Out[79]: golden_retriever      137
         Labrador_retriever     91
         Pembroke               88
         Chihuahua              74
         pug                    53
         Samoyed                39
         chow                   39
         Pomeranian             37
         toy_poodle             36
         malamute               29
         Name: pred_breed, dtype: int64
```

5 Findings of the analysis

1) The pred_breed column is created based on the the confidence level of minimum 20% and 'p1_dog', 'p2_dog' and 'p3_dog' statements
2) Based on dog types: doggo, floofer, pupper, puppo, 'doggo, puppo', 'doggo, pupper', 'doggo, floofer', only one categorical column is created named as 'stage'
3) tweet_id is set as object type as it is not going to use for calculation.
4) A main dataframe is created using df_clean, image_df_clean, and tweet_json_clean dataframes
5) Dog Names Issue got rectified
6) Inconsistency in pred_breed got removed
7) All retweets get deleted to get unique tweets
8) The columns such as in_reply_to_status_id, in_reply_to_user_id, retweeted_status_id, retweeted_status_user_id, and retweeted_status_timestamp is removed which is not needed
9) Timestamp format got corrected to datetime format
10) Extra HTML tags from source column get refracted
11) Dog ratings get standardized for denom of 10.

```
In [80]: from subprocess import call
         call(['python', '-m', 'nbconvert', 'wrangle_act.ipynb'])
```

```
Out[80]: 0
```