Wrangle Project Summary

Wrangling steps:

1- Gathering Data:
   a. Gather data from 'twitter-archive-enhanced.csv': file uploaded directly as csv file (using the read_csv function).
   b. Gather data from "image_predictions.tsv": the file is collected from the provided url (using the requests library), then the data, folder created from data (using the OS library), Read the file, using the tab separation (using read_csv function).
   c. Gather data from "Twitter API": Twitter API json file imported line by line (using json liberary), convert data to dataframe,
   d. All the previous dataframes are saved as csv files (using pandas function to_csv).

2- Assessing data: All the dataframes are inspected for quality and tidiness issues (using functions as info, head, tail, columns value_counts, column sort_values, and isnull().sum()), and the summary of the findings as follow:
   a. image_predictions:
      i. Quality issues:
         Q1- Column Header is undescribtive: img_num p1, p1_conf, p1_dog, p2, p2_conf, p2_dog, p3, p3_conf, p3_dog.
         Q2- some of the dogs types are incorrect, such in p1 column: car_mirror in raw 371, snorkel in row 655, killer_whale in column 337, mousetrap in row 889
      ii. Tiddiness issues:
         T1- img_num column contain the data as the same for the jpg_url column.
         T2- the first column to be removed as it the same as index.
   b. twetter_archive:
      i. Quality issues:
         Q4- For the data in column in_reply_to_status_id: wrong data type (to be object insteade of float).
         Q5- For the data in column in_reply_to_user_id: wrong data type (to be object insteade of float).
         Q6- For the data in column retweeted_status_timestamp: wrong data type (to be date insteade of object).
         Q7- For the data in column in_timestamp: wrong data type (to be date insteade of object).
         Q8- For the data in column retweeted_status_id: wrong data type (to be object insteade of float).
         Q9- For the data in column retweeted_status_user_id: wrong data type (to be object insteade of float).
         Q10- For the data in column retweeted_status_timestamp: wrong data type (to be date insteade of object).

Q11- There are many empty arrays in the data frame.

ii. Tiddiness issues:

T3- the first column to be removed as it the same as index.

T4- In the columns doggo, floofer, pupper, puppo: need to be melted in one column.

c. twettr_api:

i. Quality issues:

Q12- Column Header is undescribtive: extended_entities, sourc, lang

Q13- There are many missing data.

Q14- For the data in column created_at: wrong data type (to be date insteade of object).

Q15- For the data in column in_reply_to_status_id: wrong data type (to be object insteade of float64).

Q16- For the data in column in_reply_to_status_id_str: wrong data type (to be object insteade of float64).

Q17- For the data in column in_reply_to_status_id_str: wrong data type (to be object insteade of float64).

ii. Tiddiness issues:

T5- Columns id, id_str contain the same data.

T6- the first column to be removed as it the same as index.

d. All the 3 data frame:

T7- All the 3 data frame can be merged in one dataframe, using the tweet id as index, and the follwing columns are replicated between them: in_reply_to_status_id, in_reply_to_user_id, source (using duplicated function for all the columns).

3- Cleaning Data:

a. Clean clean_twettr_api:

i. Remove the un-nesseray columns (using drop function), rename the undescriptive columns (using the rename function).

b. Clean clean_twetter_archive:

i. Remove the un-nesseray columns (using drop function), rename the undescriptive columns (using the rename function).

ii. Melt the columns doggo, floofer, pupper, puppo to dog_type new column (using the melt function).

iii. Remove the duplicates of the data frame after melting (using drop_duplicates function).

c. Clean clean_image_predictions:

i. Remove the un-nesseray columns (using drop function), rename the undescriptive columns (using the rename function).

d. Merge the new 3 dataframes into one:

i. Merge the dataframes (using merge function).

ii. Change the None values into NaN values (using replace function).
iii. Save the new dataframe into csv file (using to_csv function).

References:

1- Drop function: https://cmdlinetips.com/2018/04/how-to-drop-one-or-more-columns-in-pandas-dataframe/
2- Merge function: https://www.datacamp.com/community/tutorials/python-rename-column?utm_source=adwords_ppc&utm_campaignid=1455363063&utm_adgroupid=65083631748&utm_device=c&utm_keyword=&utm_matchtype=b&utm_network=g&utm_adpostion=&utm_creative=332602034361&utm_targetid=dsa-429603003980&utm_loc_interest_ms=&utm_loc_physical_ms=1005386&gclid=CjwKCAiA17P9BRB2EiwAMvwNyH3bB3luhVIX_22iCIfPZwCoJjjtqPUT2w-3lNkwFKdfitIMfwL1ABoCCRgQAvD_BwE
3- Rename function: https://www.datacamp.com/community/tutorials/python-rename-column?utm_source=adwords_ppc&utm_campaignid=1455363063&utm_adgroupid=65083631748&utm_device=c&utm_keyword=&utm_matchtype=b&utm_network=g&utm_adpostion=&utm_creative=332602034361&utm_targetid=dsa-429603003980&utm_loc_interest_ms=&utm_loc_physical_ms=1005386&gclid=CjwKCAiA17P9BRB2EiwAMvwNyH3bB3luhVIX_22iCIfPZwCoJjjtqPUT2w-3lNkwFKdfitIMfwL1ABoCCRgQAvD_BwE
4- Replace value in columns: https://www.codegrepper.com/code-examples/delphi/how+to+replace+values+with+nan+in+pandas