Analyzing and Visualizing Data

In this section, I analyzed and visualized the cleaned dataset. Here are the insights from the data:

Insights:

1. What are the common dog names?
2. What is the most common dog stage?
3. What is the most liked tweet?
4. What is the average likes for dog stages?
5. Dog(s) with the highest rating.
6. What is the most used tweet source?

In [1]:
```
# Import neccesary libraries and modules that will be used in the notebook

import pandas as pd
import requests
import os
import numpy as np
import tweepy
import json
from timeit import default_timer as timer
from datetime import timedelta
%matplotlib inline
from matplotlib import pyplot as plt
import seaborn as sns
from PIL import Image
from io import BytesIO
from wordcloud import WordCloud
import random
```

In [74]:
```
# Load dataframe for analysis
df = pd.read_csv('twitter_archive_master.csv')
```

Checking the data info

In [75]:
```
df.head(10)
```

Out[75]:

	tweet_id	timestamp	tweet_source	text	expanded_urls
0	892177421306343426	2017-08-01 00:17:27+00:00	iphone	This is Tilly. She's just checking pup on you....	https://twitter.com/dog_rates/status/892177421...
1	891815181378084864	2017-07-31 00:18:03+00:00	iphone	This is Archie. He is a rare Norwegian Pouncin...	https://twitter.com/dog_rates/status/891815181...
2	891689557279858688	2017-07-30 15:58:51+00:00	iphone	This is Darla. She commenced a snooze mid meal...	https://twitter.com/dog_rates/status/891689557...

3	891327558926688256	2017-07-29 16:00:24+00:00	iphone	This is Franklin. He would like you to stop ca...	https://twitter.com/dog_rates/status/891327558...	Fr
4	891087950875897856	2017-07-29 00:08:17+00:00	iphone	Here we have a majestic great white breaching ...	https://twitter.com/dog_rates/status/891087950...	
5	890971913173991426	2017-07-28 16:27:12+00:00	iphone	Meet Jax. He enjoys ice cream so much he gets ...	https://gofundme.com/ydvmve-surgery-for-jax,ht...	
6	890729181411237888	2017-07-28 00:22:40+00:00	iphone	When you watch your owner call another dog a g...	https://twitter.com/dog_rates/status/890729181...	
7	890609185150312448	2017-07-27 16:25:51+00:00	iphone	This is Zoey. She doesn't want to be one of th...	https://twitter.com/dog_rates/status/890609185...	
8	890240255349198849	2017-07-26 15:59:51+00:00	iphone	This is Cassie. She is a college pup. Studying...	https://twitter.com/dog_rates/status/890240255...	
9	890006608113172480	2017-07-26 00:31:25+00:00	iphone	This is Koda. He is a South Australian decksha...	https://twitter.com/dog_rates/status/890006608...	

In [76]:
```
df.tail(10)
```

Out[76]:

	tweet_id	timestamp	tweet_source	text	expanded_urls
1668	802239329049477120	2016-11-25 19:55:35+00:00	iphone	This is Loki. He'll do your taxes for you. Can...	https://twitter.com/dog_rates/status/802239329...
1669	793195938047070209	2016-10-31 21:00:23+00:00	iphone	Say hello to Lily. She's pupset that her costu...	https://twitter.com/dog_rates/status/793195938...
1670	790946055508652032	2016-10-25 16:00:09+00:00	iphone	This is Betty. She's assisting with the dishes...	https://twitter.com/dog_rates/status/790946055...

1671	787717603741622272	2016-10-16 18:11:26+00:00	iphone	This is Tonks. She is a service puppo. Can hea...	https://twitter.com/dog_rates/status/787717603...
1672	756275833623502848	2016-07-21 23:53:04+00:00	iphone	When ur older siblings get to play in the deep...	https://twitter.com/dog_rates/status/756275833...
1673	752519690950500352	2016-07-11 15:07:30+00:00	iphone	Hopefully this puppo on a swing will help get ...	https://twitter.com/dog_rates/status/752519690...
1674	751132876104687617	2016-07-07 19:16:47+00:00	iphone	This is Cooper. He's just so damn happy. 10/10...	https://twitter.com/dog_rates/status/751132876...
1675	744995568523612160	2016-06-20 20:49:19+00:00	iphone	This is Abby. She got her face stuck in a glas...	https://twitter.com/dog_rates/status/744995568...
1676	743253157753532416	2016-06-16 01:25:36+00:00	iphone	This is Kilo. He cannot reach the snackum. Nif...	https://twitter.com/dog_rates/status/743253157...
1677	738537504001953792	2016-06-03 01:07:16+00:00	iphone	This is Bayley. She fell asleep trying to esca...	https://twitter.com/dog_rates/status/738537504...

In [77]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1678 entries, 0 to 1677
Data columns (total 15 columns):
 #   Column          Non-Null Count  Dtype 
---  ------          --------------  ----- 
 0   tweet_id        1678 non-null   int64 
 1   timestamp       1678 non-null   object
 2   tweet_source    1678 non-null   object
 3   text            1678 non-null   object
 4   expanded_urls   1678 non-null   object
 5   name            1197 non-null   object
 6   dog_stage       259 non-null    object
 7   rating          1678 non-null   int64 
```

```
 8   raw_rating      1678 non-null   object
 9   retweet_count   1678 non-null   int64
 10  favorite_count  1678 non-null   int64
 11  jpg_url         1678 non-null   object
 12  img_num         1678 non-null   int64
 13  breed           1678 non-null   object
 14  confidence      1678 non-null   float64
dtypes: float64(1), int64(5), object(9)
memory usage: 196.8+ KB
```

In [78]:
```
# Convert datatype to string
df.tweet_id = df.tweet_id.astype('str')
```

In [79]:
```
df.describe()
```

Out[79]:

	rating	retweet_count	favorite_count	img_num	confidence
count	1678.000000	1678.000000	1678.000000	1678.000000	1678.000000
mean	0.985697	2274.331943	7976.435042	1.216329	0.135266
std	0.157607	4140.556771	11755.949188	0.577078	0.101238
min	0.000000	11.000000	66.000000	1.000000	0.000010
25%	1.000000	512.500000	1797.000000	1.000000	0.052987
50%	1.000000	1126.500000	3660.000000	1.000000	0.118710
75%	1.000000	2580.500000	9852.500000	1.000000	0.197506
max	3.000000	70427.000000	144401.000000	4.000000	0.467678

Insights

In [80]:
```
# to duplicte the dataframe before working
df_clean = df.copy()
```

In [81]:
```
# Convert datatypes appropriately
df_clean.tweet_id = df_clean.tweet_id.astype('str')
df_clean.dog_stage = df_clean.dog_stage.astype('category')
```

In [82]:
```
# Sets the style for the visuals
sns.set_theme(style='darkgrid')
```

---

1. What are the common dog names?

In [83]:
```
df_clean.name.value_counts().head(13)
```

Out[83]:
```
Cooper     10
Oliver      9
Charlie     9
Lucy        9
Tucker      9
Penny       8
Daisy       7
Sadie       7
Winston     7
Koda        6
Toby        6
```

```
Jax          6
Lola         6
Name: name, dtype: int64
```

These names all appear more than 5 times as names of dogs.

2. What is the most common dog stage??

In [84]:
```
df_clean.dog_stage.value_counts()
```

Out[84]:
```
pupper     168
doggo       63
puppo       21
floofer      7
Name: dog_stage, dtype: int64
```

The most common dog stage is **pupper**

3. What is the most liked tweet?

In [85]:
```
# assign tweet(s) with highest likes to max_likes
max_likes = df_clean.favorite_count.max()

df_clean.query('favorite_count == {}'.format(max_likes))
```

Out[85]:

	tweet_id	timestamp	tweet_source	text	expanded_urls	na
621	744234799360020481	2016-06-18 18:26:18+00:00	iphone	Here's a doggo realizing you can stand in a po...	https://twitter.com/dog_rates/status/744234799...	N

In [86]:
```
# Return text of tweet with highest likes
df_clean.query('favorite_count == {}'.format(max_likes)).text.item()
```

Out[86]:
```
"Here's a doggo realizing you can stand in a pool. 13/10 enlightened af (vid by Tina Con
rad) https://t.co/7wE9LTEXC4"
```

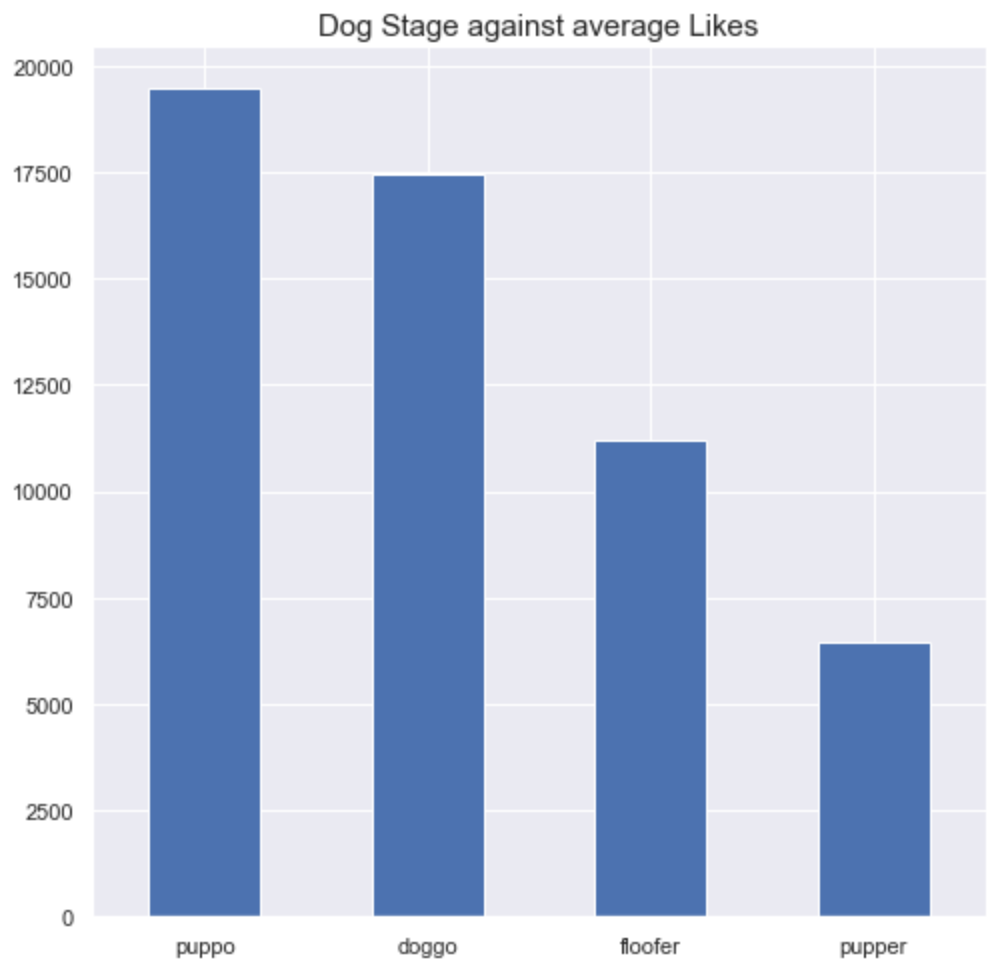
In [87]:
```
# Get image of dog(s) in tweet
url = df_clean.query('favorite_count == {}'.format(max_likes)).jpg_url.item()
r = requests.get(url)

Image.open(BytesIO(r.content))
```

Out[87]:

4. What is the average likes for dog stages?

In [88]:
```
df_clean.groupby('dog_stage')['favorite_count'].mean().sort_values(ascending=False).plot
plt.title('Dog Stage against average Likes', fontsize=15)
plt.xlabel('');
```



On average, Puppos get more tweet likes.

5. Dog(s) with the highest rating?

In [89]:
```
# Assign tweets with highest calculated ratings to max_rating
max_rating = df_clean.rating.max()

df_clean.query('rating == {}'.format(max_rating))
```

Out[89]:

	tweet_id	timestamp	tweet_source	text	expanded_urls
312	810984652412424192	2016-12-19 23:06:23+00:00	iphone	Meet Sam. She smiles 24/7 & secretly aspir...	https://www.gofundme.com/sams-smile,https://tw...
1045	680494726643068929	2015-12-25 21:06:00+00:00	iphone	Here we have uncovered an entire battalion of ...	https://twitter.com/dog_rates/status/680494726...
1515	778027034220126208	2016-09-20 00:24:34+00:00	iphone	This is Sophie. She's a Jubilant Bush Pupper. ...	https://twitter.com/dog_rates/status/778027034...

In [90]:
```
# URL of such tweets
url = df_clean.query('rating == {}'.format(max_rating)).jpg_url.to_list()

r1 = requests.get(url[0])
r2 = requests.get(url[1])
r3 = requests.get(url[2])

# Open their pictures
Image.open(BytesIO(r1.content))
```

Out[90]:

In [91]:
```
Image.open(BytesIO(r2.content))
```

Out[91]:

In [92]:
```
Image.open(BytesIO(r3.content))
```

Out[92]:

6. What is the most used tweet source?

In [93]:
```
df_clean.tweet_source.value_counts()
```

Out[93]:
```
iphone                1648
twitter web client      22
tweetdeck                8
Name: tweet_source, dtype: int64
```

7. Most liked dog breeds

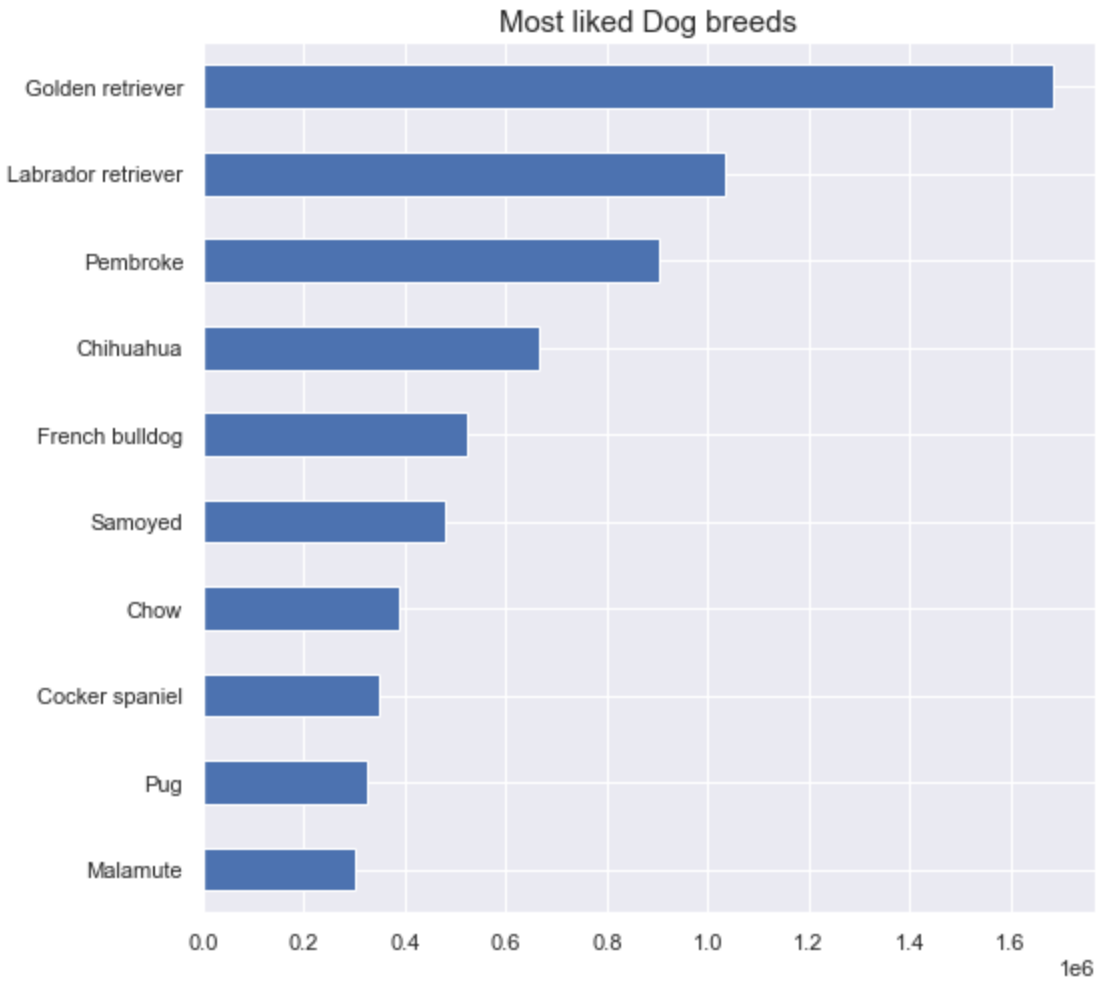
In [94]:
```
df_clean.groupby('breed').sum().sort_values(by = 'favorite_count', ascending = False).he
```

Out[94]:

breed	rating	retweet_count	favorite_count	img_num	confidence
Golden retriever	159	475396	1683384	201	17.470215
Labrador retriever	105	312301	1036514	124	14.235984
Pembroke	93	235959	903248	119	13.365032
Chihuahua	88	210051	667314	112	10.091322
French bulldog	30	131691	524721	35	2.900188
Samoyed	41	155418	480653	48	4.386557
Chow	48	106755	388434	62	5.521922
Cocker spaniel	30	118294	351164	37	4.286072
Pug	61	94035	324427	77	5.288428
Malamute	33	88161	303844	40	6.056324

In [95]:
```
df_clean.groupby('breed').sum().favorite_count.sort_values(ascending = False).head(10).s
plt.title('Most liked Dog breeds', fontsize=15)
```

```
plt.xlabel('');
plt.ylabel('');
```



Visualization

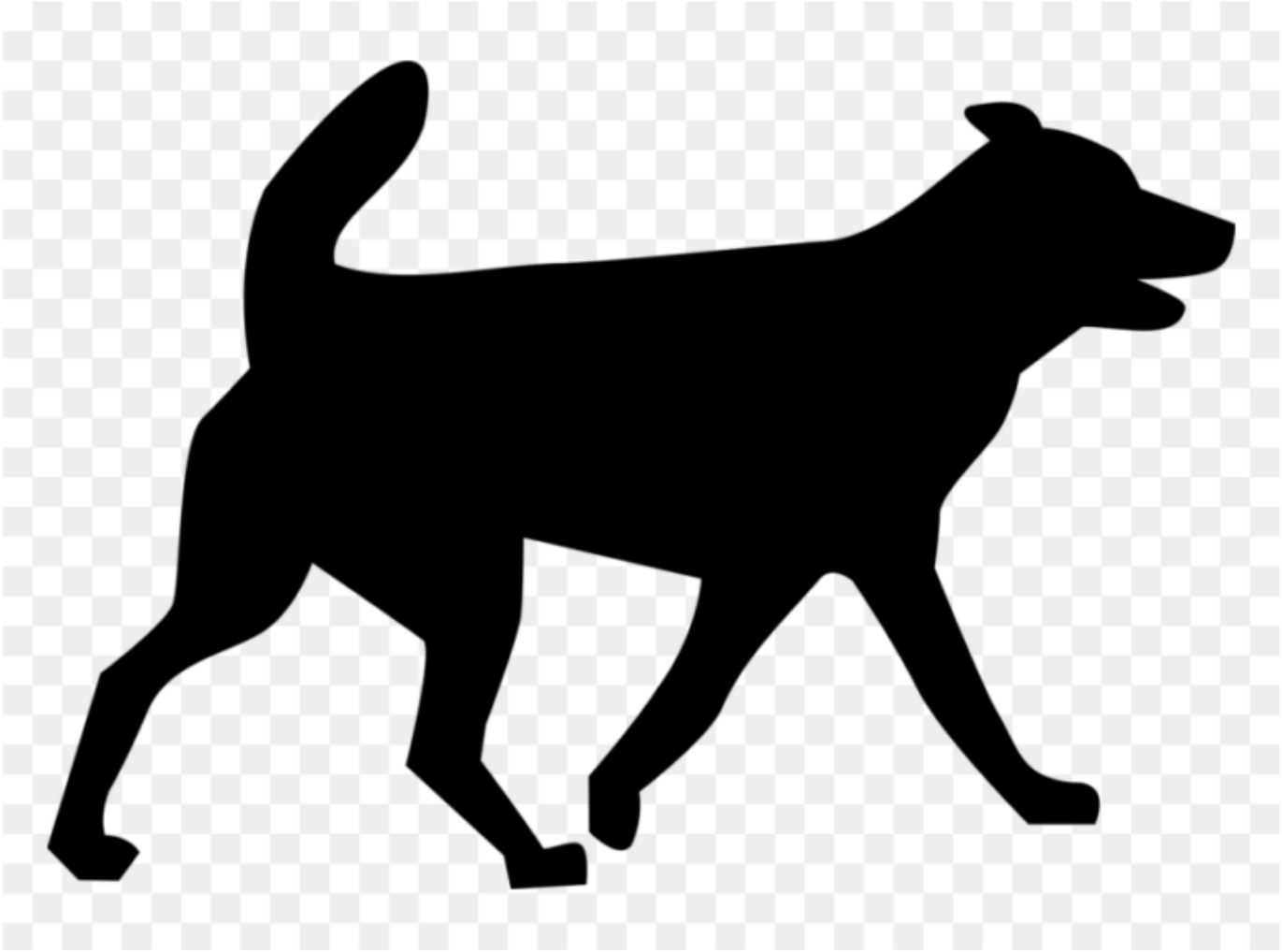
In [96]:
```
# Import dog silhoutte
url = 'https://www.dlf.pt/dfpng/middlepng/57-578964_silhouette-transparent-background-do
r = requests.get(url)
folder_name = 'C:/Users/Zion/Documents/Udacity_Wrangling'

# Download image for wordcloud
i = Image.open(BytesIO(r.content))
i.save(folder_name + "/" + 'dog_clipart' + '.' + 'png')

# Load image for wordcloud
image = np.array(Image.open('dog_clipart.png'))

fig = plt.figure() # Instantiate the figure object
fig.set_figwidth(14) # set width
fig.set_figheight(18) # set height

plt.imshow(image, cmap=plt.cm.gray, interpolation='bilinear') # Display data as an image
plt.axis('off') # Remove axis
plt.show() # Display image
```



In [97]:
```
# Create function to generate the blue colour for the Word CLoud

def blue_color_func(word, font_size, position, orientation, random_state=None,**kwargs):
    return "hsl(210, 100%%, %d%%)" % random.randint(50, 70)
```

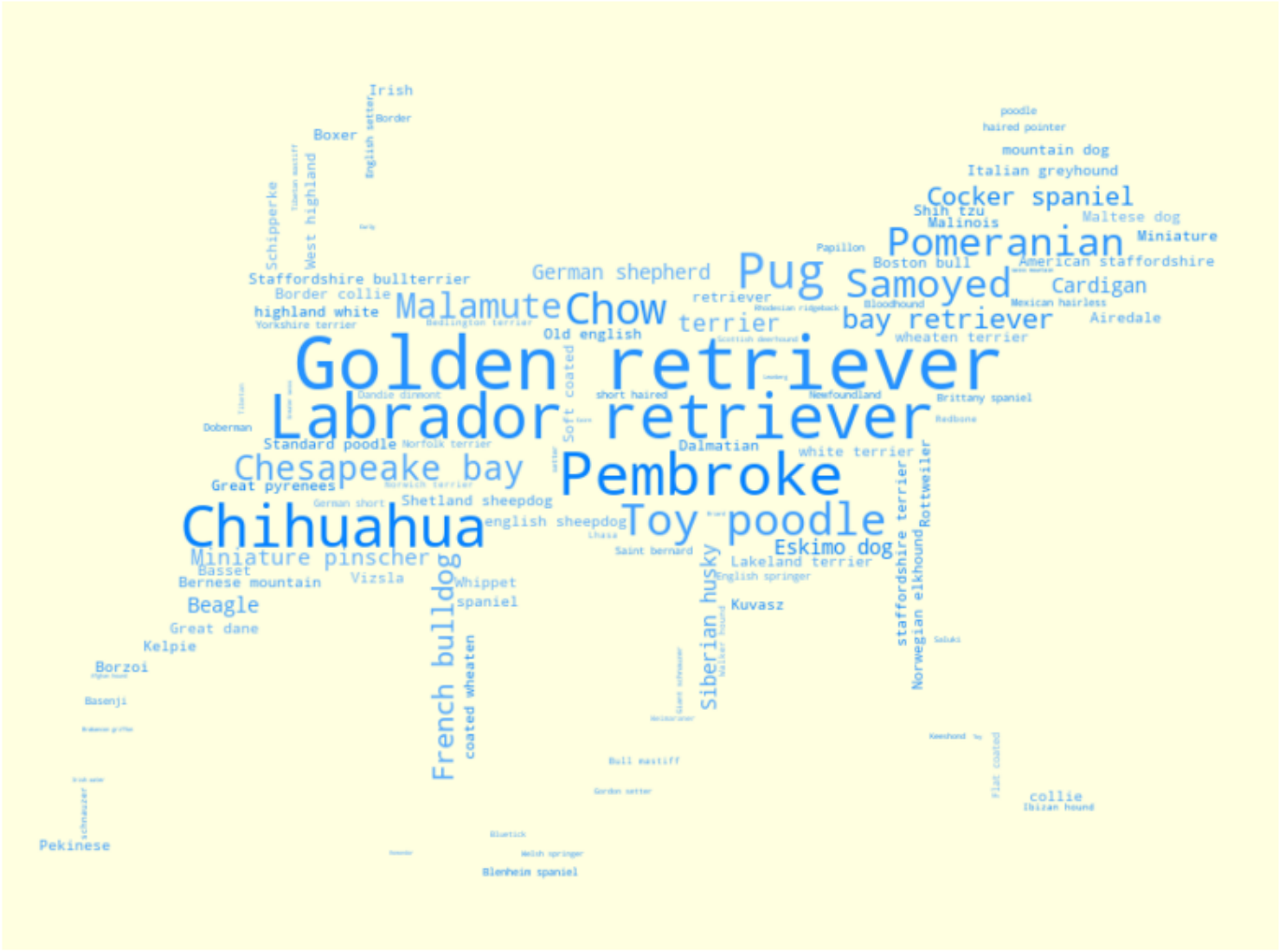
In [98]:
```
# Extract all breed into one long string separated by space
breeds_long_string = df_clean['breed'].replace(" ", "_").tolist()
breeds_long_string = " ".join(breeds_long_string)
```

In [99]:
```
# Instantiate the Twitter word cloud object
wc = WordCloud(mode='RGBA',background_color='lightyellow', max_words=1500, mask=image)

# generate the word cloud
wc.generate(breeds_long_string)

# display the word cloud
fig = plt.figure()
fig.set_figwidth(14)  # set width
fig.set_figheight(18)  # set height

plt.imshow(wc.recolor(color_func=blue_color_func, random_state=3),
           interpolation="bilinear")
plt.axis('off')
plt.show()
```



In [100…

```
# Save to a png file
wc.to_file("breed_wordcloud.png")
```

Out[100]:

```
<wordcloud.wordcloud.WordCloud at 0x18c3a33eca0>
```