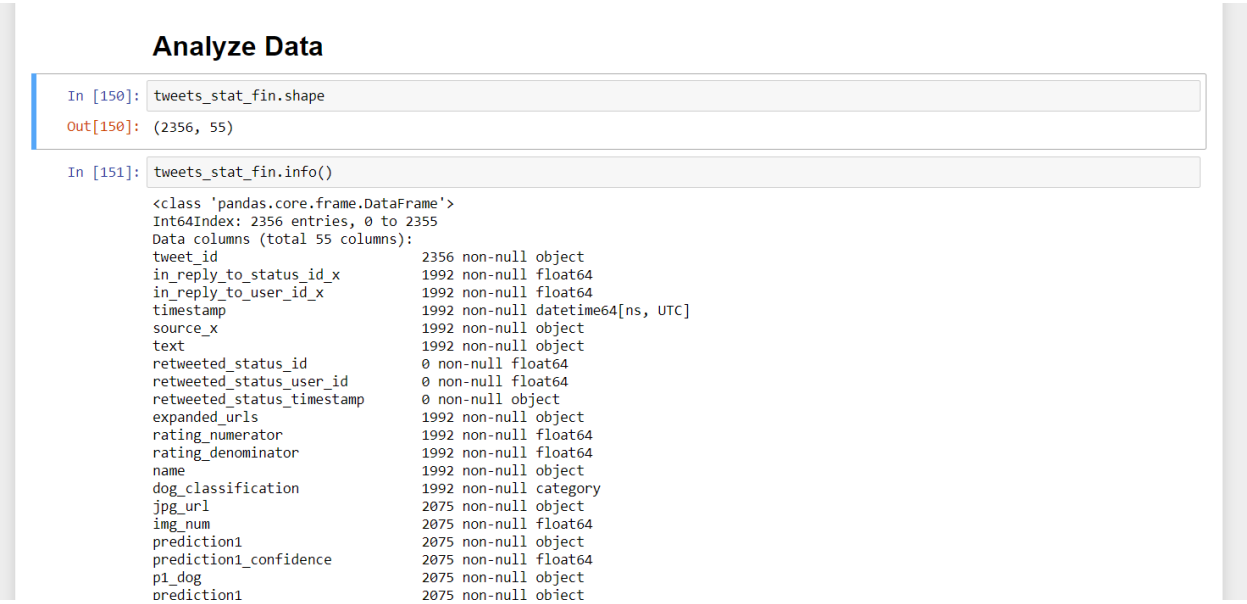
Documentation of analysis and insights

Analyze

- Analyze numerator ratings
- Analyze denominator ratings
- Total of 1231 dogs rated greater than 10, which is almost 60%
- Time trend Analysis
- Quantify multiple sources influence
- Analysis of dog classification
- General stats drill down retweet for count average (tweets_clean)
- General stats drill down for favorite count average (tweets_clean)
- General stats drill down retweet for count average (api)
- General stats drill down for favorite count average (api)
- Check most frequent dogs names



Analyze Data

```
In [150]: tweets_stat_fin.shape
Out[150]: (2356, 55)

In [151]: tweets_stat_fin.info()
<class 'pandas.core.frame.DataFrame'>
Int64Index: 2356 entries, 0 to 2355
Data columns (total 55 columns):
tweet_id                      2356 non-null object
in_reply_to_status_id_x       1992 non-null float64
in_reply_to_user_id_x         1992 non-null float64
timestamp                     1992 non-null datetime64[ns, UTC]
source_x                      1992 non-null object
text                          1992 non-null object
retweeted_status_id           0 non-null float64
retweeted_status_user_id      0 non-null float64
retweeted_status_timestamp    0 non-null object
expanded_urls                 1992 non-null object
rating_numerator              1992 non-null float64
rating_denominator            1992 non-null float64
name                          1992 non-null object
dog_classification            1992 non-null category
jpg_url                       2075 non-null object
img_num                       2075 non-null float64
prediction1                   2075 non-null object
prediction1_confidence        2075 non-null float64
p1_dog                        2075 non-null object
prediction1                   2075 non-null object
```

In [152]:
```
#Analye numerator ratings
tweets_stat_fin['rating_numerator'].value_counts().sort_index()
```

Out[152]:
```
1.0         5
2.0         9
3.0        19
4.0        16
5.0        33
6.0        32
7.0        52
8.0        95
9.0       151
10.0      419
11.0      396
12.0      450
13.0      261
14.0       35
24.0        1
26.0        1
27.0        1
44.0        1
45.0        1
50.0        1
```

In [153]:
```
#Analye denominator ratings
tweets_stat_fin['rating_denominator'].value_counts().sort_index()
```

Out[153]:
```
2.0         1
7.0         1
10.0     1974
11.0        2
20.0        1
40.0        1
50.0        3
70.0        1
80.0        2
```

In [154]:
```
tweets_stat_fin['rating_denominator'].value_counts().sort_index().sum()
```

Out[154]: 1992

In [155]:
```
tweets_stat_fin['rating_numerator'].value_counts().sort_index().sum()
```

Out[155]: 1992

In [156]:
```
#A total of 1161 dogs rated greater than 10, which is almost 60%
tweets_stat_fin['rating_numerator'][tweets_stat_fin['rating_numerator'] > 10].value_counts().sum()
```

Out[156]: 1161

In [157]:
```
#Time trend Analysis
tweets_stat_fin['timestamp'].apply(lambda x: x.strftime('%Y-%m')).value_counts().sort_index()
```

Out[157]:
```
2015-11    296
2015-12    367
2016-01    169
2016-02    111
2016-03    120
2016-04     54
2016-05     57
2016-06     80
2016-07     88
2016-08     59
2016-09     63
2016-10     65
2016-11     53
2016-12     54
2017-01     66
2017-02     63
2017-03     48
2017-04     41
2017-05     43
```

In [158]:
```
#Check tweets duplicaions
tweets_stat_fin.loc[:,'tweet_id'].duplicated().sum()
```

Out[158]:
```
0
```

In [159]:
```
#Detect aggregated missing values.
tweets_stat_fin.isna().sum()
```

Out[159]:
```
tweet_id                        0
in_reply_to_status_id_x       364
in_reply_to_user_id_x         364
timestamp                     364
source_x                      364
text                          364
retweeted_status_id          2356
retweeted_status_user_id     2356
retweeted_status_timestamp   2356
expanded_urls                 364
rating_numerator              364
rating_denominator            364
name                          364
dog_classification            364
jpg_url                       281
img_num                       281
prediction1                   281
prediction1_confidence        281
p1_dog                        281
prediction1                   281
prediction2_confidence        281
p2_dog                        281
prediction3                   281
prediction3_confidence        281
p3_dog                        281
created_at                      2
id_str                          2
full_text                       2
```

Out[160]:
```
<a href="http://twitter.com/download/iphone" rel="nofollow">Twitter for iPhone</a>     1953
<a href="http://twitter.com" rel="nofollow">Twitter Web Client</a>                       28
<a href="https://about.twitter.com/products/tweetdeck" rel="nofollow">TweetDeck</a>      11
Name: source_x, dtype: int64
```

In [161]:
```
#Analysis of dog classification
tweets_stat_fin['dog_classification'][tweets_stat_fin['dog_classification'] == 'None'].value_counts()
```

Out[161]:
```
puppo             0
pupper            0
floofer           0
doggo, puppo      0
doggo, pupper     0
doggo, floofer    0
doggo             0
                  0
Name: dog_classification, dtype: int64
```

In [162]:
```
tweets_stat_fin['name'].value_counts().sort_index()
```

Out[162]:
```
Abby            2
Ace             1
Acro            1
Adele           1
Aiden           1
                ..
such            1
the             7
this            1
unacceptable    1
very            4
Name: name, Length: 936, dtype: int64
```

In [163]:
```
tweets_stat_fin.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 2356 entries, 0 to 2355
Data columns (total 55 columns):
tweet_id                        2356 non-null object
in_reply_to_status_id_x         1992 non-null float64
in_reply_to_user_id_x           1992 non-null float64
timestamp                       1992 non-null datetime64[ns, UTC]
source_x                        1992 non-null object
text                            1992 non-null object
retweeted_status_id             0 non-null float64
retweeted_status_user_id        0 non-null float64
retweeted_status_timestamp      0 non-null object
expanded_urls                   1992 non-null object
rating_numerator                1992 non-null float64
rating_denominator              1992 non-null float64
name                            1992 non-null object
dog_classification              1992 non-null category
jpg_url                         2075 non-null object
img_num                         2075 non-null float64
prediction1                     2075 non-null object
prediction1_confidence          2075 non-null float64
p1_dog                          2075 non-null object
prediction1                     2075 non-null object
prediction2_confidence          2075 non-null float64
p2_dog                          2075 non-null object
prediction3                     2075 non-null object
prediction3_confidence          2075 non-null float64
p3_dog                          2075 non-null object
created_at                      2354 non-null datetime64[ns, UTC]
id str                          2354 non-null float64
```

memory usage: 1015.6+ KB

In [165]:
```
image_clean.shape
```

Out[165]:
```
(2075, 12)
```

In [166]:
```
api_df.head()
```

Out[166]:

	id	retweet_count	favorite_count	followers_count
0	892420643555336193	8853	39467	3200889
1	892177421306343426	6514	33819	3200889
2	891815181378084864	4328	25461	3200889
3	891689557279858688	8964	42908	3200889
4	891327558926688256	9774	41048	3200889

In [167]:
```
tweets_stat_fin.describe()
```

Out[167]:

	in_reply_to_status_id_x	in_reply_to_user_id_x	retweeted_status_id	retweeted_status_user_id	rating_numerator	rating_denominator	img_num	predicti
count	1.992000e+03	1.992000e+03	0.0	0.0	1992.000000	1992.000000	2075.000000	
mean	7.682114e+15	4.635223e+07	NaN	NaN	12.293173	10.532631	1.203855	
std	7.284849e+16	4.387342e+08	NaN	NaN	41.516729	7.324367	0.561875	
min	0.000000e+00	0.000000e+00	NaN	NaN	1.000000	2.000000	1.000000	
25%	0.000000e+00	0.000000e+00	NaN	NaN	10.000000	10.000000	1.000000	
50%	0.000000e+00	0.000000e+00	NaN	NaN	11.000000	10.000000	1.000000	
75%	0.000000e+00	0.000000e+00	NaN	NaN	12.000000	10.000000	1.000000	

In [170]:
```
api_df.describe()
```

Out[170]:

	retweet_count	favorite_count	followers_count
count	2354.000000	2354.000000	2.354000e+03
mean	3164.797366	8080.968564	3.200942e+06
std	5284.770364	11814.771334	4.457302e+01
min	0.000000	0.000000	3.200799e+06
25%	624.500000	1415.000000	3.200898e+06
50%	1473.500000	3603.500000	3.200945e+06
75%	3652.000000	10122.250000	3.200953e+06
max	79515.000000	132810.000000	3.201018e+06

In [171]:
```
# General stats drill down retweet for count average (tweets_clean)
print('The mean retweet count is : {}.'.format(round(tweets_stat_fin.retweet_count.mean())))
```

```
The mean retweet count is : 3165.
```

In [172]:
```
# General stats  drill down for favorite count average  (tweets_clean)
print('The mean favorite count is : {}.'.format(round(tweets_stat_fin.favorite_count.mean())))
```

```
The mean favorite count is : 8081.
```

In [173]:
```
# General stats drill down retweet for count average  (api)
print('The mean retweet count is : {}.'.format(round(tweets_stat_fin.retweet_count.mean())))
```

```
The mean retweet count is : 3165.
```

In [174]:
```
# General stats  drill down for favorite count average  (api)
print('The mean favorite count is : {}.'.format(round(tweets_stat_fin.favorite_count.mean())))
```

In [174]:
```
# General stats  drill down for favorite count average  (api)
print('The mean favorite count is : {}.'.format(round(tweets_stat_fin.favorite_count.mean())))
```

```
The mean favorite count is : 8081.
```

In [175]:
```
#check unique names
tweets_stat_fin['name'].unique()
```

Out[175]:
```
array(['Phineas', 'Tilly', 'Archie', 'Darla', 'Franklin', 'None', 'Jax',
       'Zoey', 'Cassie', 'Koda', 'Bruno', 'Ted', 'Stuart', 'Oliver',
       'Jim', 'Zeke', 'Ralphus', 'Gerald', 'Jeffrey', 'such', 'Canela',
       'Maya', 'Mingus', 'Derek', 'Roscoe', 'Waffles', 'Jimbo', 'Maisey',
       'Earl', 'Lola', 'Kevin', 'Yogi', 'Noah', 'Bella', 'Grizzwald',
       'Rusty', 'Gus', 'Stanley', 'Alfy', 'Koko', 'Rey', 'Gary', 'a',
       'Elliot', 'Louis', 'Jesse', 'Romeo', 'Bailey', 'Duddles', 'Jack',
       'Steven', 'Beau', 'Snoopy', 'Shadow', 'Emmy', 'Aja', 'Penny',
       'Dante', 'Nelly', 'Ginger', 'Benedict', 'Venti', 'Goose', 'Nugget',
       'Cash', 'Jed', 'Sebastian', 'Sierra', 'Monkey', 'Harry', 'Kody',
       'Lassie', 'Rover', 'Napolean', 'Boomer', 'Cody', 'Rumble',
       'Clifford', 'Dewey', 'Scout', 'Gizmo', 'Walter', 'Cooper',
       'Harold', 'Shikha', 'Lili', 'Jamesy', 'Coco', 'Sammy', 'Meatball',
       'Paisley', 'Albus', 'Neptune', 'Belle', 'Quinn', 'Zooey', 'Dave',
       'Jersey', 'Hobbes', 'Burt', 'Lorenzo', 'Carl', 'Jordy', 'Milky',
       'Trooper', 'quite', 'Sophie', 'Wyatt', 'Rosie', 'Thor', 'Oscar',
       'Callie', 'Cermet', 'Marlee', 'Arya', 'Einstein', 'Alice',
       'Rumpole', 'Benny', 'Aspen', 'Jarod', 'Wiggles', 'General',
       'Sailor', 'Iggy', 'Snoop', 'Kyle', 'Leo', 'Riley', 'Noosh', 'Odin',
       'Jerry', 'Georgie', 'Bonta', 'Cannon', 'Furgey', 'Daisy', 'Tuck',
```

In [176]:
```
tweets_stat_fin['name'].value_counts()
#archive_clean['name'].value_counts()[0:30].sort_values(ascending=False)
```

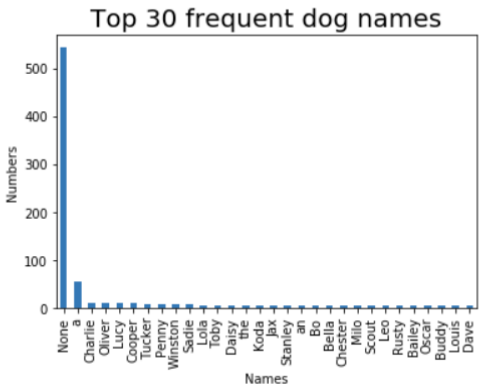
Out[176]:
```
None              544
a                  55
Charlie            11
Oliver             10
Lucy               10
                 ...
Geoff               1
Michelangelope      1
Mo                  1
Mookie              1
Chadrick            1
Name: name, Length: 936, dtype: int64
```

Visualize:

- Most frequent dogs names
- Tweet count trend
- Retweet & favorite count over time
- Source trend
- Prediction confidence

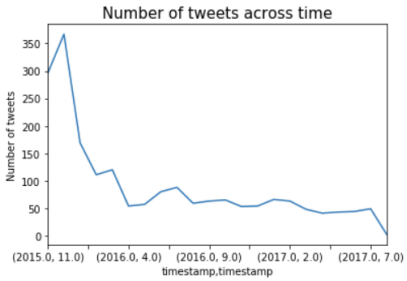
In [177]:
```
#Check most frequent dogs names
plt.title('Top 30 frequent dog names', size=20)
plt.xlabel('Names')
plt.ylabel('Numbers')
plt.savefig('frequent_dogs_names');

archive_clean['name'].value_counts()[0:30].sort_values(ascending=False).plot(kind = 'bar')
```

Out[177]: <matplotlib.axes._subplots.AxesSubplot at 0x1111deffcc8>



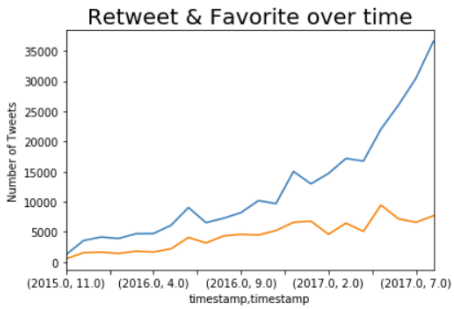
In [178]:
```
#visualize tweet count trend
plt.title('Number of tweets across time', size=15)
plt.xlabel('Time (Year, Month)')
plt.ylabel('Number of tweets')
plt.savefig('tweets_over_time');

tweets_stat_fin['tweet_id'].groupby([archive_clean['timestamp'].dt.year, archive_clean['timestamp'].dt.month]).count().plot(kind
```

Out[178]: <matplotlib.axes._subplots.AxesSubplot at 0x1111ecf76c8>



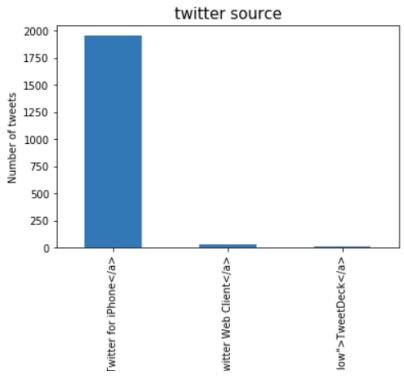
In [179]:
```
#visualize retweet & favorite count over time
plt.title('Retweet & Favorite over time', size =20)
plt.ylabel('Number of Tweets')
plt.xlabel('Time (Year,Month)')
#plt.legend('Retweet Count', 'Favorite Count')
plt.savefig('Retweet_Favorite')

tweets_clean.favorite_count.groupby([archive_clean['timestamp'].dt.year, archive_clean['timestamp'].dt.month]).mean().plot(kind=
tweets_clean.retweet_count.groupby([archive_clean['timestamp'].dt.year, archive_clean['timestamp'].dt.month]).mean().plot(kind='
```

Out[179]: <matplotlib.axes._subplots.AxesSubplot at 0x1111f23a948>



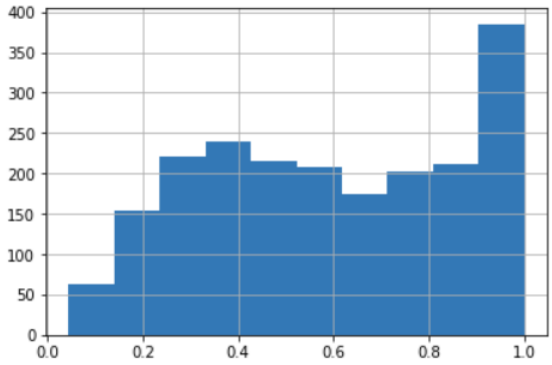
In [180]:
```
#visualize source trend
plt.title('twitter source', size=15)
plt.xlabel('Source')
plt.ylabel('Number of tweets')
plt.savefig('tweets_source')
tweets_stat_fin['source_x'].value_counts().plot(kind='bar')
```

Out[180]: <matplotlib.axes._subplots.AxesSubplot at 0x1111f2155c8>



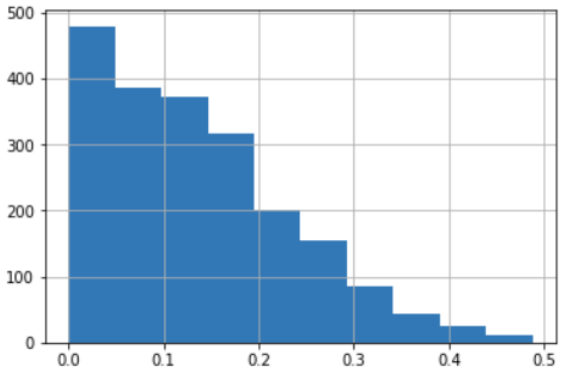
In [182]:
```
#checking prediction1_confidence
tweets_stat_fin.prediction1_confidence.hist()
```

Out[182]: <matplotlib.axes._subplots.AxesSubplot at 0x1111f571d08>



In [183]:
```
tweets_stat_fin.prediction2_confidence.hist()
```

Out[183]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x1111f6fe2c8>
```



In [184]:
```
#checking prediction3_confidence
tweets_stat_fin.prediction3_confidence.hist()
```

Out[184]:
```
<matplotlib.axes._subplots.AxesSubplot at 0x1111f980548>
```