Problem 1 [15 marks]

In [1]:
```
def pyramid_pattern (n):
    counter = 0
    for i in range (0,n):
        string = ""
        if i % 2 ==0 :
            print (' '*(n-i),end="")
            counter +=1
            for j in range (counter,1,-1):
                string +=str(j)
                string +="-"
            for k in range (1,counter+1):
                string += str(k)
                if k!=  counter :
                    string+="-"
        else :
            print (' '*(n-i),end="")
            string += "-"
            for j in range (counter,1,-1):
                string +=str(j)
                string +="-"
            for k in range (1,counter+1):
                string += str(k)
                string += "-"

        print (string)
```

In [2]:
```
pyramid_pattern(19)
```

```
                  1
                 -1-
                2-1-2
               -2-1-2-
              3-2-1-2-3
             -3-2-1-2-3-
            4-3-2-1-2-3-4
           -4-3-2-1-2-3-4-
          5-4-3-2-1-2-3-4-5
         -5-4-3-2-1-2-3-4-5-
        6-5-4-3-2-1-2-3-4-5-6
       -6-5-4-3-2-1-2-3-4-5-6-
      7-6-5-4-3-2-1-2-3-4-5-6-7
     -7-6-5-4-3-2-1-2-3-4-5-6-7-
    8-7-6-5-4-3-2-1-2-3-4-5-6-7-8
   -8-7-6-5-4-3-2-1-2-3-4-5-6-7-8-
  9-8-7-6-5-4-3-2-1-2-3-4-5-6-7-8-9
 -9-8-7-6-5-4-3-2-1-2-3-4-5-6-7-8-9-
10-9-8-7-6-5-4-3-2-1-2-3-4-5-6-7-8-9-10
```

Problem 2 [15 marks]

In [3]:
```
#A
def Triplesort(a,b,c):
    unsort_list = [a,b,c]
    sorted_list = []

    while unsort_list:
        min_value = unsort_list[0]
        for i in unsort_list:
            if i < min_value:
                min_value = i
        sorted_list.append(min_value)
        unsort_list.remove(min_value)
    return  sorted_list

print(Triplesort(5,9,2))
```

```
[2, 5, 9]
```

In [4]:
```
#b
def ListSort(list):
    ordered = False
    while ordered==False:
        ordered=True
        for i in range(0, len(list)-1):
            if list[i]>list[i+1]:
                ordered=False
        if ordered==True:
            break
        else:
            for i in range(0, len(list)-2):
                sorted_three = Triplesort(list[i], list[i+1], list[i+2])
                list[i] = sorted_three[0]
                list[i+1] = sorted_three[1]
                list[i+2] = sorted_three[2]
    print(list)
```

In [5]:
```
#C
ListSort([5,2,1,6,8,0,4,9,18,-8,-100,100,26,9,18,28,30])
```

```
[-100, -8, 0, 1, 2, 4, 5, 6, 8, 9, 9, 18, 18, 26, 28, 30, 100]
```

In [6]:
```
#D
import numpy as np
np.random.seed(90)

p2= np.random.randint(-5000, 5000,(3000))

ListSort(p2)
%timeit ()
```

```
[-4996 -4996 -4980 ...  4997  4998  4998]
7.26 ns ± 0.0659 ns per loop (mean ± std. dev. of 7 runs, 100000000 loops each)
```

How Efficiency Can be improved :

In TripleSort, it compares the min value against itself.

unsort_list[0] is compared with itself when i= unsort_list[0] and min_value = unsort_list[0]

after the first pars of the while loop, it will have the first element in correct position in list and doesnt need to be checked again , to improve this can check for the max_value in addition to min value

In regards to ListSort : after the first pass of the list sort the final elementwill be the largest and to improve this could use a constant that increases by 1 after each pass and is subtracted from the max_value the the for loop goes up to.

Problem 3 [20 marks]

In [7]:
```
import numpy as np
#A
np.random.seed(120)
#B
b= np.random.randint(-10, 200, (50, 6))
print(b)
print()
```

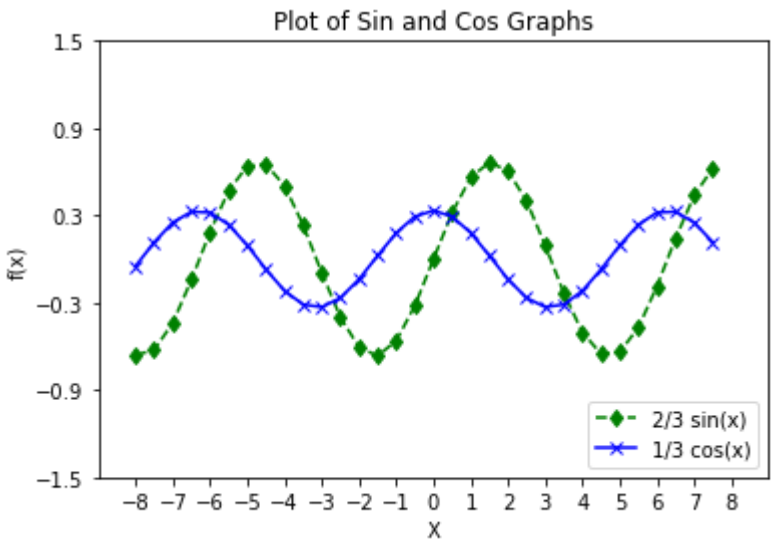
```
[[157 -10 148 116   0  37]
 [ 46  74 167  34 141 117]
 [ 23 137  78  95  74  95]
 [ 95 132  67  79  87  58]
 [ 27 125 163 168  88 120]
 [ 86  18 102 144  69  58]
 [ 48  24 131 152  -2 171]
 [124  42 151 111   5  40]
 [118 157 155 141  51  78]
 [ 42 199  44 191 145  11]
 [ 84 188 167  27  -6  62]
 [  1 176 127  61 174  60]
 [118 155  62  76 128  14]
 [107  48 183 116 175 196]
 [  4 186  97 193  85 171]
 [157  47 119 164 179  84]
 [ 48  75  50 146  62 113]
 [ 51  15 184  39 140 167]
 [ 17  76 186  26 130 173]
 [ 62 110  17  16 144 115]
 [165   0  50  13 185  90]
 [ 68  30  32 172  87  12]
 [177   1 139  34  16 169]
 [145  53 140 161 147 132]
 [108  62  78 155 188 105]
 [ 21 185 189 174 198 100]
 [  6  81  14 183  62 102]
 [ 77 196  55  97 121 177]
 [ 20  51  59   5  47 172]
 [121 107 158  38 136 168]
 [116 151 129 141 172  34]
 [ 61   7 168 140  98  -6]
 [  3 167 128 117  39  36]
 [160  73  28 120 197 154]
 [ 63  12 102 147 154  32]
 [ 53  58 155  53 198 136]
 [130 189 195  32   5 183]
 [ 83  42  52 143  88  21]
 [188 186 116  51 136 106]
 [ 14  35  74 190  69  96]
 [145  31 138  65 165 159]
 [ 99 145 128 109  -2 106]
 [ 79 128  60   4 109  60]
 [143 117 190 114 169 146]
 [168  98 156 138 137 110]
 [141 149 193  81  34 137]
 [198 180 121 138  94  90]
 [ 80   0 182  81 183  88]
 [ 32  59  29 160 177  92]
 [ 27  83  84 137 103  70]]
```

In [8]:
```
#C
for i in range (0,6):
    print("collumn" ,i+1,",number of elements>50:", np.count_nonzero(b > 50, axis=0)[i])
print()


#D

d=((b[np.any(b <0, axis=1)]))
rows,collumns = d.shape
print("number of rows that contain negative values:", rows )
```

```
collumn 1 ,number of elements>50: 34
collumn 2 ,number of elements>50: 34
collumn 3 ,number of elements>50: 42
collumn 4 ,number of elements>50: 39
collumn 5 ,number of elements>50: 40
collumn 6 ,number of elements>50: 40

number of rows that contain negative values: 5
```

Problem 4 [20 marks]

In [9]:
```
import matplotlib as mpl
import matplotlib.pyplot as plt
%matplotlib inline
import numpy as np

x= np.arange (-8,8,0.5)
#lines

plt.plot(x,2/3*np.sin(x), label='2/3 sin(x)',linestyle='dashed',marker = 'd', color='green');

plt.plot(x,1/3*np.cos(x),  label='1/3 cos(x)',marker = 'x', color='blue')

#Graph set up
plt.xlim(-9,9) # x-axis paramaters
plt.ylim(-1.5,1.5) #y-axis paramaters
plt.title("Plot of Sin and Cos Graphs") # Graph title
plt.xlabel("X") #X xis title
plt.ylabel("f(x)") #y axis title
plt.yticks(np.arange(-1.5, 2, 0.6))
plt.xticks(np.arange(-8, 9 , 1))
plt.legend(loc='lower right');

#setting up graph


plt.savefig('P4.png')
```



Problem 5 [20 marks]

In [11]:
```
import pandas as pd
import numpy as np

#A
df = pd.read_csv (r'/Users/elliottoates/OneDrive - University of Exeter/Data Science-Elliott's MacBook Pro/Assi
df = df.drop('Rank', axis=1)
display(df)
print()
print('Number of rows:',len(df))
print()
```

	Username	Owner	Followers(millions)	Likes(millions)	Description	Country	Brand_Account
0	@charlidamelio	Charli D'Amelio	125.1	9900.0	Dancer and social media personality	United States	NaN
1	@khaby.lame	Khabane Lame	113.5	1700.0	Social media personality	Italy	NaN
2	@addisonre	Addison Rae	84.6	5400.0	Dancer and social media personality	United States	NaN
3	@bellapoarch	Bella Poarch	82.8	1900.0	Social media personality	United States	NaN
4	@zachking	Zach King	65.2	808.5	Filmmaker and social media personality	United States	NaN
5	@willsmith	Will Smith	62.1	388.9	Actor	United States	NaN
6	@tiktok	TikTok	57.0	258.3	Social media platform	United States	Yes
7	@dixiedamelio	Dixie D'Amelio	55.1	3000.0	Singer and social media personality	United States	NaN
8	@spencerx	Spencer Polanco Knight	54.8	1300.0	Beatboxer and social media personality	United States	NaN
9	@lorengray	Loren Gray	54.1	2900.0	Singer, dancer, and social media personality	United States	NaN
10	@kimberly.loaiza	Kimberly Loaiza	51.4	2900.0	Singer and social media personality	Mexico	NaN
11	@justmaiko	Michael Le	50.3	1400.0	Dancer and social media personality	United States	NaN
12	@jasonderulo	Jason Derulo	49.7	1100.0	Singer, songwriter, and dancer	United States	NaN
13	@cznburak	Burak Özdemir	48.2	893.0	Social media personality	Turkey	NaN
14	@riyaz.14	Riyaz Aly	44.3	2100.0	Social media personality	India	NaN
15	@bts_official_bighit	BTS	41.9	696.1	Band	South Korea	Yes
16	@domelipa	Dominik Lipa	41.2	2300.0	Social media personality	Mexico	NaN
17	@brentrivera	Brent Rivera	40.8	1300.0	Actor and social media personality	United States	NaN
18	@therock	The Rock	40.4	216.8	Actor and former wrestler	United States	NaN
19	@youneszarou	Younes Zarou	38.7	781.5	Social media personality	Germany	NaN
20	@junya1gou	Junya Gou	37.9	645.9	Social media personality	Japan	NaN
21	@avani	Avani Gregg	36.8	2500.0	Social media personality	United States	NaN
22	@itsjojosiwa	JoJo Siwa	36.4	1200.0	Singer, actress, dancer, and social media pers...	United States	NaN
23	@elrodcontreras	Rod Contreras	36.2	2100.0	Social media personality	Mexico	NaN
24	@kallmekris	Kris Collins	36.1	1600.0	Social media personality	Canada	NaN
25	@kyliejenner	Kylie Jenner	35.9	423.6	Singer, Model and social media personality	United States	NaN
26	@jamescharles	James Charles	35.8	906.3	Social media personality	United States	NaN
27	@babyariel	Baby Ariel	35.6	1900.0	Singer, actress, and social media personality	United States	NaN
28	@selenagomez	Selena Gomez	35.4	181.8	Singer and actress	United States	NaN
29	@joealbanese	Joe Albanese	35.3	1300.0	Social media personality	United States	NaN
30	@billieeilish	Billie Eilish	33.8	175.3	Singer and songwriter	United States	NaN
31	@gilmhercroes	Gilmher Croes	33.6	821.0	Social media personality	Aruba	NaN
32	@thekiryalife	Kirya Kolesnikov	33.6	709.0	Social media personality	Russia	NaN
33	@carlosferiag	Carlos Feria	32.3	1500.0	Social media personality	Colombia	NaN
34	@lilhuddy	Chase Hudson	32.2	1600.0	Social media personality	United States	NaN
35	@mr_faisu_07	Faisal Shaikh	32.2	2000.0	Social media personality	India	NaN
36	@scottsreality	Scott Hentzepeter	31.5	612.2	Social media personality	Netherlands	NaN
37	@stokestwins	Stokes Twins	31.4	809.3	Social media personalities	United States	NaN
38	@homm9k	Alina Kim	30.8	530.6	Social media personality	Kazakhstan	NaN
39	@dobretwins	Lucas and Marcus	30.6	846.2	Social media personalities	United States	NaN
40	@noahbeck	Noah Beck	30.4	1900.0	Social media personality	United States	NaN
41	@kylethomas	Kyle Thomas	30.2	1400.0	Social media personality	United Kingdom	NaN
42	@ox_zung	Won Jeong	30.1	757.3	Social media personality	Korea	NaN

	Username	Owner	Followers(millions)	Likes(millions)	Description	Country	Brand_Account
43	@wigofellas	WigoFellas	29.3	746.4	Social media personality	United States	Yes
44	@briandadeyanara	Brianda Deyanara	29.3	2000.0	Social media personality	Mexico	NaN
45	@savv.labrant	Savannah LaBrant	29.1	1500.0	Social media personality	United States	NaN
46	@darianrojasc	Darian Rojas	28.8	1100.0	Social media personality	Mexico	NaN
47	@anokhinalz	Liza Anokhina	28.7	907.3	Social media personality	Russia	NaN
48	@lizzza	Liza Koshy	28.6	357.0	Social media personality	United States	NaN
49	@_arishfakhan_	Arishfa Khan	28.6	1100.0	Actress and social media personality	India	NaN

```
Number of rows: 50
```

In [12]:
```
#B
unique_countries = df['Country'].unique()
print('List:Unique Countries:',unique_countries)
```

```
List:Unique Countries: [' United States' ' Italy' ' Mexico' ' Turkey' ' India' ' South Korea'
 ' Germany' ' Japan' ' Canada' ' Aruba' ' Russia' ' Colombia'
 ' Netherlands' ' Kazakhstan' ' United Kingdom' ' Korea']
```

In [13]:
```
#C
df['Country'] = df['Country'].str.lstrip()
unique_countries = df['Country'].unique()
print('List: Unique Countries(no leading space):',unique_countries)
```

```
List: Unique Countries(no leading space): ['United States' 'Italy' 'Mexico' 'Turkey' 'India' 'South Korea' 'G
ermany'
 'Japan' 'Canada' 'Aruba' 'Russia' 'Colombia' 'Netherlands' 'Kazakhstan'
 'United Kingdom' 'Korea']
```

In [14]:
```
#D
df['American Account'] = [1 if x == 'United States' else 0 for x in df['Country']]

#E
def conditions(i):
    if (i['American Account'] == 1) and (i['Brand_Account'] != 'Yes'):
        return True
    else:
        return False
df['American Person'] = df.apply(conditions, axis=1)
```

In [15]:
```
 American Accounts / total number of accounts
nts = df.apply(lambda x : True
 x['American Account'] == 1 else False, axis = 1)
ows = len(df[american_accounts == True].index)
ion of accounts that are american =',round(100*(num_american_rows/len(df)),2),'%')


e of American persons among all persons
ns = df.apply(lambda x : True
 x['American Person'] == True else False, axis = 1)
 df.apply(lambda x : True
 x['Brand_Account'] != 'Yes' else False, axis = 1)
ersons_rows = len(df[american_persons == True].index)
ons = len(df[total_persons == True].index)
ion of Personal accounts that are american =',round(100*(num_american_persons_rows/num_total_persons),2),'%')



ericans, % with > 1 billion likes
2(i):
rican Person'] == True) and (i['Likes(millions)'] > 1000):
 True

 False

l_likes= df.apply(conditions2, axis=1)

lbill_likes=len(df[americans_1bill_likes == True].index)
lbill_likes
ion of American Personal accounts with more than 1 billion likes =',round(100*(num_americans_1bill_likes/num_amer
```

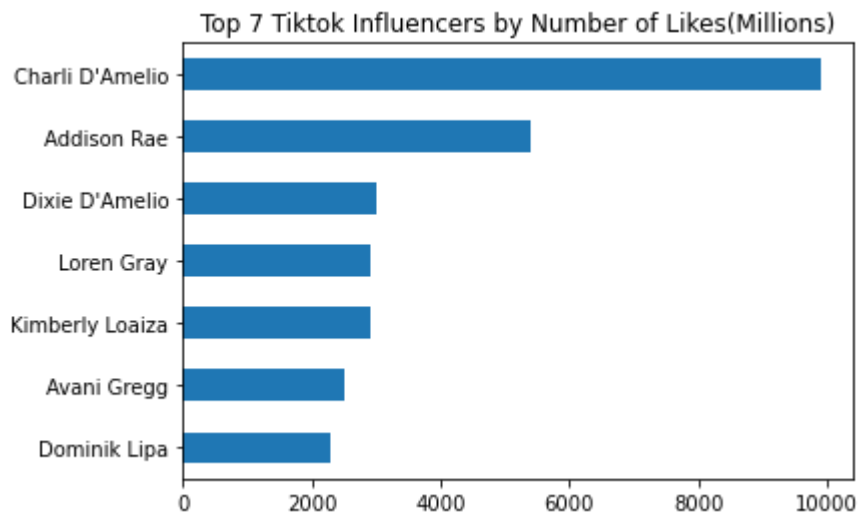
```
Proportion of accounts that are american = 56.0 %

Proportion of Personal accounts that are american = 55.32 %

Proportion of American Personal accounts with more than 1 billion likes = 61.54 %
```

In [16]:
```
#G
df_persons = df.loc[(df['Brand_Account'] != 'Yes')]
df_persons = df_persons.drop('Brand_Account', axis=1)
display(df_persons)
```

	Username	Owner	Followers(millions)	Likes(millions)	Description	Country	American Account	American Person
0	@charlidamelio	Charli D'Amelio	125.1	9900.0	Dancer and social media personality	United States	1	True
1	@khaby.lame	Khabane Lame	113.5	1700.0	Social media personality	Italy	0	False
2	@addisonre	Addison Rae	84.6	5400.0	Dancer and social media personality	United States	1	True
3	@bellapoarch	Bella Poarch	82.8	1900.0	Social media personality	United States	1	True
4	@zachking	Zach King	65.2	808.5	Filmmaker and social media personality	United States	1	True
5	@willsmith	Will Smith	62.1	388.9	Actor	United States	1	True
7	@dixiedamelio	Dixie D'Amelio	55.1	3000.0	Singer and social media personality	United States	1	True
8	@spencerx	Spencer Polanco Knight	54.8	1300.0	Beatboxer and social media personality	United States	1	True
9	@lorengray	Loren Gray	54.1	2900.0	Singer, dancer, and social media personality	United States	1	True
10	@kimberly.loaiza	Kimberly Loaiza	51.4	2900.0	Singer and social media personality	Mexico	0	False
11	@justmaiko	Michael Le	50.3	1400.0	Dancer and social media personality	United States	1	True
12	@jasonderulo	Jason Derulo	49.7	1100.0	Singer, songwriter, and dancer	United States	1	True
13	@cznburak	Burak Özdemir	48.2	893.0	Social media personality	Turkey	0	False
14	@riyaz.14	Riyaz Aly	44.3	2100.0	Social media personality	India	0	False
16	@domelipa	Dominik Lipa	41.2	2300.0	Social media personality	Mexico	0	False
17	@brentrivera	Brent Rivera	40.8	1300.0	Actor and social media personality	United States	1	True
18	@therock	The Rock	40.4	216.8	Actor and former wrestler	United States	1	True
19	@youneszarou	Younes Zarou	38.7	781.5	Social media personality	Germany	0	False
20	@junya1gou	Junya Gou	37.9	645.9	Social media personality	Japan	0	False
21	@avani	Avani Gregg	36.8	2500.0	Social media personality	United States	1	True
22	@itsjojosiwa	JoJo Siwa	36.4	1200.0	Singer, actress, dancer, and social media pers...	United States	1	True
23	@elrodcontreras	Rod Contreras	36.2	2100.0	Social media personality	Mexico	0	False
24	@kallmekris	Kris Collins	36.1	1600.0	Social media personality	Canada	0	False
25	@kyliejenner	Kylie Jenner	35.9	423.6	Singer, Model and social media personality	United States	1	True
26	@jamescharles	James Charles	35.8	906.3	Social media personality	United States	1	True
27	@babyariel	Baby Ariel	35.6	1900.0	Singer, actress, and social media personality	United States	1	True
28	@selenagomez	Selena Gomez	35.4	181.8	Singer and actress	United States	1	True
29	@joealbanese	Joe Albanese	35.3	1300.0	Social media personality	United States	1	True
30	@billieeilish	Billie Eilish	33.8	175.3	Singer and songwriter	United States	1	True
31	@gilmhercroes	Gilmher Croes	33.6	821.0	Social media personality	Aruba	0	False
32	@thekiryalife	Kirya Kolesnikov	33.6	709.0	Social media personality	Russia	0	False
33	@carlosferiag	Carlos Feria	32.3	1500.0	Social media personality	Colombia	0	False
34	@lilhuddy	Chase Hudson	32.2	1600.0	Social media personality	United States	1	True
35	@mr_faisu_07	Faisal Shaikh	32.2	2000.0	Social media personality	India	0	False
36	@scottsreality	Scott Hentzepeter	31.5	612.2	Social media personality	Netherlands	0	False
37	@stokestwins	Stokes Twins	31.4	809.3	Social media personalities	United States	1	True
38	@homm9k	Alina Kim	30.8	530.6	Social media personality	Kazakhstan	0	False
39	@dobretwins	Lucas and Marcus	30.6	846.2	Social media personalities	United States	1	True
40	@noahbeck	Noah Beck	30.4	1900.0	Social media personality	United States	1	True

	Username	Owner	Followers(millions)	Likes(millions)	Description	Country	American Account	American Person
41	@kylethomas	Kyle Thomas	30.2	1400.0	Social media personality	United Kingdom	0	False
42	@ox_zung	Won Jeong	30.1	757.3	Social media personality	Korea	0	False
44	@briandadeyanara	Brianda Deyanara	29.3	2000.0	Social media personality	Mexico	0	False
45	@savv.labrant	Savannah LaBrant	29.1	1500.0	Social media personality	United States	1	True
46	@darianrojasc	Darian Rojas	28.8	1100.0	Social media personality	Mexico	0	False
47	@anokhinalz	Liza Anokhina	28.7	907.3	Social media personality	Russia	0	False
48	@lizzza	Liza Koshy	28.6	357.0	Social media personality	United States	1	True
49	@_arishfakhan_	Arishfa Khan	28.6	1100.0	Actress and social media personality	India	0	False

In [17]:
```
#H
import matplotlib.pyplot as plt
%matplotlib inline


df_ranked = df.sort_values(by='Likes(millions)',ascending=True)
df_ranked=df_ranked.loc[:,['Owner','Likes(millions)',]]
df_ranked=df_ranked.iloc[-7:]

df_ranked.plot.barh(x='Owner', y='Likes(millions)', title = 'Top 7 Tiktok Influencers by Number of Likes(Millio

plt.savefig('P5.png')
```



Problem 6 [10 marks]

Due to the monetization and increasing commercialisation of social media platforms for individuals and brands i think and area of compelling extra analysis exploring the strength of different accounts community engagement. A metric for community engagement would accordingly be the ratio of likes:followers

It is important to note that since the majority of tiktok accounts are public, including all the ones below, the likes:follower ratio is not suggestive of the attribution of an accounts likes per follower since public accounts can recceive likes from users who dont follow them

The ratio is important as it indicates which accounts retain a high engagement rate in proportion to their followers which is perhaps a quality that determines the monetability of different accounts.

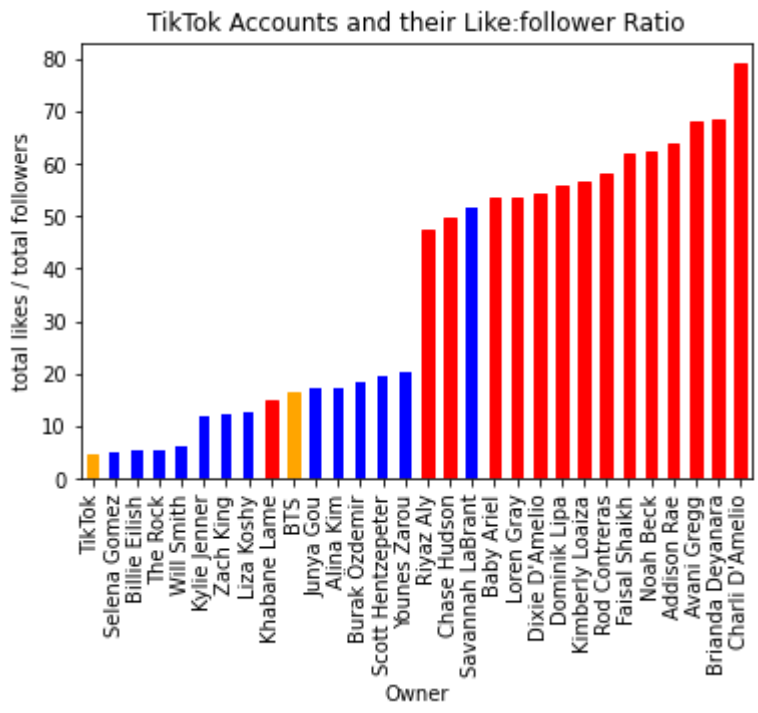
The data frame below now includes a collumn with the like:follower ratios. For example "Charliee DAmelio" has a ratio of 79.14 which suggests that every ~79 likes on her account is proportional to ~1 follower.

In [18]:
```
#Create new df with follower:likes ratios
df6= df.sort_values(by='Likes(millions)',ascending=False)
df6['Likes_followers_Ratio'] =   round((df6['Likes(millions)']/df6['Followers(millions)']),2 )
df6= df6.sort_values(by='Likes_followers_Ratio',ascending=True)

display(df6)
```

	Username	Owner	Followers(millions)	Likes(millions)	Description	Country	Brand_Account	American Account	American Person	Likes_followers_R
6	@tiktok	TikTok	57.0	258.3	Social media platform	United States	Yes	1	False	
28	@selenagomez	Selena Gomez	35.4	181.8	Singer and actress	United States	NaN	1	True	
30	@billieeilish	Billie Eilish	33.8	175.3	Singer and songwriter	United States	NaN	1	True	
18	@therock	The Rock	40.4	216.8	Actor and former wrestler	United States	NaN	1	True	
5	@willsmith	Will Smith	62.1	388.9	Actor	United States	NaN	1	True	
25	@kyliejenner	Kylie Jenner	35.9	423.6	Singer, Model and social media personality	United States	NaN	1	True	1
4	@zachking	Zach King	65.2	808.5	Filmmaker and social media personality	United States	NaN	1	True	1
48	@lizzza	Liza Koshy	28.6	357.0	Social media personality	United States	NaN	1	True	1
1	@khaby.lame	Khabane Lame	113.5	1700.0	Social media personality	Italy	NaN	0	False	1
15	@bts_official_bighit	BTS	41.9	696.1	Band	South Korea	Yes	0	False	1
20	@junya1gou	Junya Gou	37.9	645.9	Social media personality	Japan	NaN	0	False	1
38	@homm9k	Alina Kim	30.8	530.6	Social media personality	Kazakhstan	NaN	0	False	1
13	@cznburak	Burak Özdemir	48.2	893.0	Social media personality	Turkey	NaN	0	False	1
36	@scottsreality	Scott Hentzepeter	31.5	612.2	Social media personality	Netherlands	NaN	0	False	1
19	@youneszarou	Younes Zarou	38.7	781.5	Social media personality	Germany	NaN	0	False	2
32	@thekiryalife	Kirya Kolesnikov	33.6	709.0	Social media personality	Russia	NaN	0	False	2
12	@jasonderulo	Jason Derulo	49.7	1100.0	Singer, songwriter, and dancer	United States	NaN	1	True	2
3	@bellapoarch	Bella Poarch	82.8	1900.0	Social media personality	United States	NaN	1	True	2
8	@spencerx	Spencer Polanco Knight	54.8	1300.0	Beatboxer and social media personality	United States	NaN	1	True	2
31	@gilmhercroes	Gilmher Croes	33.6	821.0	Social media personality	Aruba	NaN	0	False	2
42	@ox_zung	Won Jeong	30.1	757.3	Social media personality	Korea	NaN	0	False	2
26	@jamescharles	James Charles	35.8	906.3	Social media personality	United States	NaN	1	True	2
43	@wigofellas	WigoFellas	29.3	746.4	Social media personality	United States	Yes	1	False	2
37	@stokestwins	Stokes Twins	31.4	809.3	Social media personalities	United States	NaN	1	True	2

	Username	Owner	Followers(millions)	Likes(millions)	Description	Country	Brand_Account	American Account	American Person	Likes_followers_R
39	@dobretwins	Lucas and Marcus	30.6	846.2	Social media personalities	United States	NaN	1	True	2
11	@justmaiko	Michael Le	50.3	1400.0	Dancer and social media personality	United States	NaN	1	True	2
47	@anokhinalz	Liza Anokhina	28.7	907.3	Social media personality	Russia	NaN	0	False	3
17	@brentrivera	Brent Rivera	40.8	1300.0	Actor and social media personality	United States	NaN	1	True	3
22	@itsjojosiwa	JoJo Siwa	36.4	1200.0	Singer, actress, dancer, and social media pers...	United States	NaN	1	True	3
29	@joealbanese	Joe Albanese	35.3	1300.0	Social media personality	United States	NaN	1	True	3
46	@darianrojasc	Darian Rojas	28.8	1100.0	Social media personality	Mexico	NaN	0	False	3
49	@_arishfakhan_	Arishfa Khan	28.6	1100.0	Actress and social media personality	India	NaN	0	False	3
24	@kallmekris	Kris Collins	36.1	1600.0	Social media personality	Canada	NaN	0	False	4
41	@kylethomas	Kyle Thomas	30.2	1400.0	Social media personality	United Kingdom	NaN	0	False	4
33	@carlosferiag	Carlos Feria	32.3	1500.0	Social media personality	Colombia	NaN	0	False	4
14	@riyaz.14	Riyaz Aly	44.3	2100.0	Social media personality	India	NaN	0	False	4
34	@lilhuddy	Chase Hudson	32.2	1600.0	Social media personality	United States	NaN	1	True	4
45	@savv.labrant	Savannah LaBrant	29.1	1500.0	Social media personality	United States	NaN	1	True	5
27	@babyariel	Baby Ariel	35.6	1900.0	Singer, actress, and social media personality	United States	NaN	1	True	5
9	@lorengray	Loren Gray	54.1	2900.0	Singer, dancer, and social media personality	United States	NaN	1	True	5
7	@dixiedamelio	Dixie D'Amelio	55.1	3000.0	Singer and social media personality	United States	NaN	1	True	5
16	@domelipa	Dominik Lipa	41.2	2300.0	Social media personality	Mexico	NaN	0	False	5
10	@kimberly.loaiza	Kimberly Loaiza	51.4	2900.0	Singer and social media personality	Mexico	NaN	0	False	5
23	@elrodcontreras	Rod Contreras	36.2	2100.0	Social media personality	Mexico	NaN	0	False	5
35	@mr_faisu_07	Faisal Shaikh	32.2	2000.0	Social media personality	India	NaN	0	False	6
40	@noahbeck	Noah Beck	30.4	1900.0	Social media personality	United States	NaN	1	True	6
2	@addisonre	Addison Rae	84.6	5400.0	Dancer and social media personality	United States	NaN	1	True	6
21	@avani	Avani Gregg	36.8	2500.0	Social media personality	United States	NaN	1	True	6
44	@briandadeyanara	Brianda Deyanara	29.3	2000.0	Social media personality	Mexico	NaN	0	False	6
0	@charlidamelio	Charli D'Amelio	125.1	9900.0	Dancer and social media personality	United States	NaN	1	True	7

In [19]:
```
#Graph top 5 and bottom 5 Ratios

df7 = df6.loc[:,['Owner','Likes_followers_Ratio','Brand_Account','Likes(millions)']]
df8=(df7.drop(df7.index[15:-15]))
df8


ax = df8.plot(x='Owner', y='Likes_followers_Ratio', kind='bar', color='b',  legend = False, title='TikTok Accour

buisness_accounts  = (df8['Brand_Account']=='Yes')
for i,b in enumerate(buisness_accounts):
    if b:
        ax.patches[i].set_color('orange')

high_likes = (df8['Likes(millions)']>1500)
for i,b in enumerate(high_likes):
    if b:
        ax.patches[i].set_color('red')


plt.savefig('P6.png')
```



In the above bar graph, different bars have different coloured columns. those that are orange are brand accounts and those that are red are accounts with over 1 billion account likes.

From the above graph one can see that the majority of accounts with a very high like to follower ratio are also accounts with a massive amount of likes. This suggests that at such a scale of likes, the ratio is less indicitave of the strength of community engagements.

However at the other end of the scale, it is clear that Brand accounts are able to retain a lower ratio. This potentially suggests that tik tok is a viable platform for brand accounts to advertise with good engagementlevels.