

17	Track Name	Artist Name	Genre	Beats.Per.Minute	Energy	Danceability	Loudness..dB	Liveness	Valence	Length
	On the Spider-Verse									
18	Lalala	Y2K	canadian hip hop	130	39	84	-8	14	50	
19	Truth Hurts	Lizzo	escape room	158	62	72	-3	12	41	
20	Piece Of Your Heart	MEDUZA	pop house	124	74	68	-7	7	63	
21	Panini	Lil Nas X	country rap	154	59	70	-6	12	48	
22	No Me Conoce - Remix	Jhay Cortez	reggaeton flow	92	79	81	-4	9	58	
23	Soltera - Remix	Lunay	latin	92	78	80	-4	44	80	
24	bad guy (with Justin Bieber)	Billie Eilish	electropop	135	45	67	-11	12	68	
25	If I Can't Have You	Shawn Mendes	canadian pop	124	82	69	-4	13	87	
26	Dance Monkey	Tones and I	australian pop	98	59	82	-6	18	54	
27	It's You	Ali Gatie	canadian hip hop	96	46	73	-7	19	40	
28	Con Calma	Daddy Yankee	latin	94	86	74	-3	6	66	
29	QUE PRETENDES	J Balvin	latin	93	79	64	-4	36	94	
30	Takeaway	The Chainsmokers	edm	85	51	29	-8	10	36	
31	7 rings	Ariana Grande	dance pop	140	32	78	-11	9	33	
32	0.958333333	Maluma	reggaeton	96	71	78	-5	9	68	
33	The London (feat. J. Cole & Travis Scott)	Young Thug	atl hip hop	98	59	80	-7	13	18	
34	Never Really Over	Katy Perry	dance pop	100	88	77	-5	32	39	
35	Summer Days (feat. Macklemore & Patrick Stump ...	Martin Garrix	big room	114	72	66	-7	14	32	
36	Otro Trago	Sech	panamanian pop	176	70	75	-5	11	62	
37	Antisocial (with Travis Scott)	Ed Sheeran	pop	152	82	72	-5	36	91	
38	Sucker	Jonas Brothers	boy band	138	73	84	-5	11	95	
39	fuck, i'm lonely (with Anne-Marie) - from "13 ...	Lauv	dance pop	95	56	81	-6	6	68	
40	Higher Love	Kygo	edm	104	68	69	-7	10	40	
41	You Need To Calm Down	Taylor Swift	dance pop	85	68	77	-6	7	73	
42	Shallow	Lady Gaga	dance pop	96	39	57	-6	23	32	
43	-	-	...	

43	Talk	Khalid	pop	136	40	90	-9	6	35	
	Track.Name	Artist.Name	Genre	Beats.Per.Minute	Energy	Danceability	Loudness..dB..	Liveness	Valence.	Len
44	Con Altura	ROSALÍA	r&b en espanol	98	69	88	-4	5	75	
45	One Thing Right	Marshmello	brostep	88	62	66	-2	58	44	
46	Te Robaré	Nicky Jam	latin	176	75	67	-4	8	80	
47	Happier	Marshmello	brostep	100	79	69	-3	17	67	
48	Call You Mine	The Chainsmokers	edm	104	70	59	-6	41	50	
49	Cross Me (feat. Chance the Rapper & PnB Rock)	Ed Sheeran	pop	95	79	75	-6	7	61	



In [10]:

```
s=data.to_csv('top50.csv')
```

In [59]:

```
#f=d['Energy'].head(10).mean()
#g=d['Length.'].head(10).mean()
#f,g
f=d[['Energy','Length.']].head(10).mean()
f
```

Out[59]:

```
Energy      65.1
Length.     195.6
dtype: float64
```

In [81]:

```
print((d.groupby('Genre')['Length.'].sum().sort_values(axis=0,ascending=False)))
```

```
Genre
dance pop      1621
pop            1368
latin          1126
edm             656
reggaeton flow  611
canadian hip hop 579
panamanian pop  514
reggaeton      427
brostep        396
electropop     389
canadian pop   382
dfw rap        333
country rap    272
australian pop 210
atl hip hop    200
boy band       181
escape room    173
big room       164
r&b en espanol 162
pop house      153
trap music     131
Name: Length., dtype: int64
```

In [95]:

```
d.groupby('Artist.Name')['Genre'].count()
d[d['Artist.Name']=='Ed Sheeran']
```

Out[95]:

	Track.Name	Artist.Name	Genre	Beats.Per.Minute	Energy	Danceability	Loudness..dB..	Liveness	Valence.	Length.	Acc
3	Beautiful People (feat. Khalid)	Ed Sheeran	pop	93	65	64	-8	8	55	198	
5	I Don't Care (with Justin Bieber)	Ed Sheeran	pop	102	68	80	-5	9	84	220	
37	Antisocial (with Travis Scott)	Ed Sheeran	pop	152	82	72	-5	36	91	162	
49	Cross Me (feat. Chance the Rapper & PnB Rock)	Ed Sheeran	pop	95	79	75	-6	7	61	206	



In []:

PROGRAM2

In [1]:

```
import numpy as np
import pandas as pd

data={'English':{'Sam':60,'Jackson':74,'Ahree':85},
      'History':{'Gloria':83,'Sam':65,'Isla':78,'Aron':72,'Gray':61},
      'Geography':{'Jackson':92,'Gloria':95,'Isla':82,'Aron':75,'Ahree':76},
      'Mathematics':{'Sam':99,'Gloria':74,'Jackson':89,'Ahree':85,'Gray':95},
      'Science':{'Sam':89,'Aron':82,'Gray':78,'Isla':93,'Ahree':87}}
print(pd.Series(data))
```

```
English          {'Sam': 60, 'Jackson': 74, 'Ahree': 85}
History          {'Gloria': 83, 'Sam': 65, 'Isla': 78, 'Aron': ...
Geography        {'Jackson': 92, 'Gloria': 95, 'Isla': 82, 'Aro...
Mathematics      {'Sam': 99, 'Gloria': 74, 'Jackson': 89, 'Ahre...
Science          {'Sam': 89, 'Aron': 82, 'Gray': 78, 'Isla': 93...
dtype: object
```

In [12]:

```
g=pd.DataFrame(data,columns=['English','History','Geography','Mathematics','Science'])
print(g.fillna(0))
```

	English	History	Geography	Mathematics	Science
Sam	60.0	65.0	0.0	99.0	89.0
Jackson	74.0	0.0	92.0	89.0	0.0
Ahree	85.0	0.0	76.0	85.0	87.0
Gloria	0.0	83.0	95.0	74.0	0.0
Isla	0.0	78.0	82.0	0.0	93.0
Aron	0.0	72.0	75.0	0.0	82.0
Gray	0.0	61.0	0.0	95.0	78.0

In [13]:

```
p=g.T.fillna(0)
p['Average']=p['Sam']+p['Jackson']+p['Ahree']+p['Gloria']+p['Isla']+p['Aron']+p['Gray']
print(p)
```

	Sam	Jackson	Ahree	Gloria	Isla	Aron	Gray	Average
English	60.0	74.0	85.0	0.0	0.0	0.0	0.0	219.0
History	65.0	0.0	0.0	83.0	78.0	72.0	61.0	359.0
Geography	0.0	92.0	76.0	95.0	82.0	75.0	0.0	420.0
Mathematics	99.0	89.0	85.0	74.0	0.0	0.0	95.0	442.0
Science	89.0	0.0	87.0	0.0	93.0	82.0	78.0	429.0

In []:

PROGRAM3

In [2]:

```
import pandas as pd
import numpy as np

t=np.arange(1,10001)
a=pd.Series(t)
a
```

Out[2]:

```
0          1
1          2
2          3
3          4
4          5
...
9995      9996
9996      9997
9997      9998
9998      9999
9999     10000
Length: 10000, dtype: int32
```

In [27]:

```
for i in range(0,1000):
    if(a[i]%7==0 and a[i]%17==0):
        print(i," ",a[i])
```

```
118    119
237    238
356    357
475    476
594    595
713    714
832    833
951    952
```

In []:

In [47]:

```
# PROGRAM4
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
data=pd.read_csv('cereal.csv')
d=pd.DataFrame(data)
d
```

Out[47]:

	name	mfr	type	calories	protein	fat	sodium	fiber	carbo	sugars	potass	vitamins	shelf	weight	cups	rating
0	100% Bran	N	C	70	4	1	130	10.0	5.0	6	280.0	25	3	1.0	0.33	68.402973
1	100% Natural Bran	Q	C	120	3	5	15	2.0	8.0	8	135.0	0	3	1.0	1.00	33.983679
2	All-Bran	K	C	70	4	1	260	9.0	7.0	5	320.0	25	3	1.0	0.33	59.425505
3	All-Bran with Extra Fiber	K	C	50	4	0	140	14.0	8.0	0	330.0	25	3	1.0	0.50	93.704912
4	Almond Delight	R	C	110	2	2	200	1.0	14.0	8	NaN	25	3	1.0	0.75	34.384843
...
72	Triples	G	C	110	2	1	250	0.0	21.0	3	60.0	25	3	1.0	0.75	39.106174
73	Trix	G	C	110	1	1	140	0.0	13.0	12	25.0	25	2	1.0	1.00	27.753301
74	Wheat Chex	R	C	100	3	1	230	3.0	17.0	3	115.0	25	1	1.0	0.67	49.787445
75	Wheaties	G	C	100	3	1	200	3.0	17.0	3	110.0	25	1	1.0	1.00	51.592193
76	Wheaties Honey Gold	G	C	110	2	1	200	1.0	16.0	8	60.0	25	1	1.0	0.75	36.187559

77 rows x 16 columns



In [21]:

```
mfr=d['mfr']
mfr
```

Out[21]:

```
0      N
1      Q
2      K
3      K
4      R
..
72     G
73     G
74     R
75     G
76     G
```

Name: mfr, Length: 77, dtype: object

In [48]:

```
rt=d['rating']
```

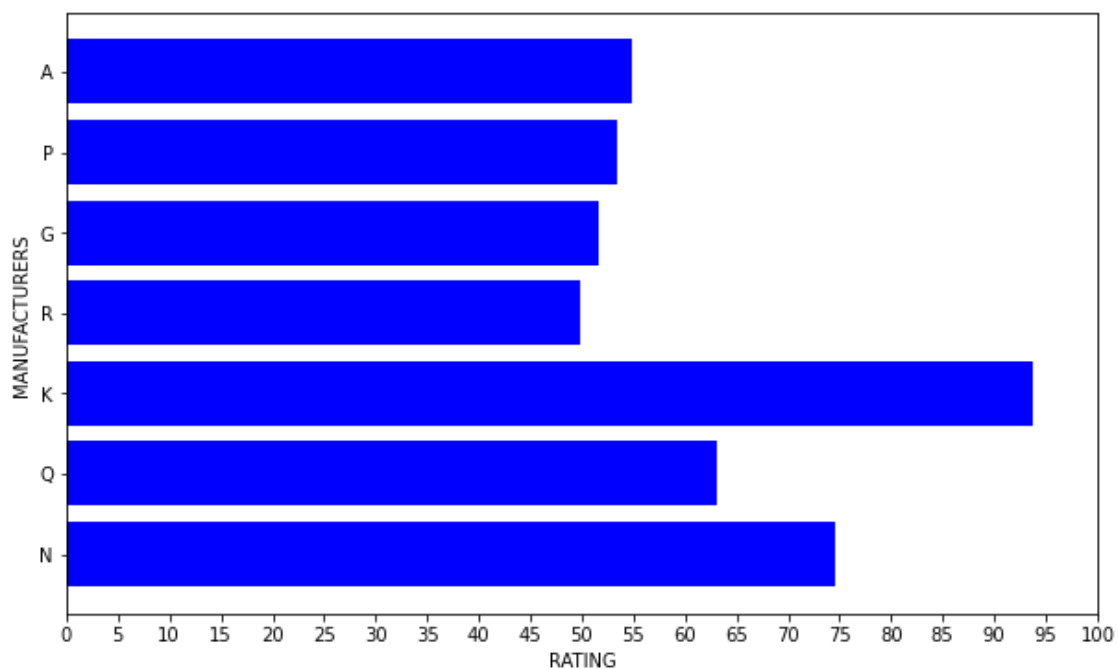
```
rt
```

```
Out[48]:
```

```
0      68.402973
1      33.983679
2      59.425505
3      93.704912
4      34.384843
...
72     39.106174
73     27.753301
74     49.787445
75     51.592193
76     36.187559
Name: rating, Length: 77, dtype: float64
```

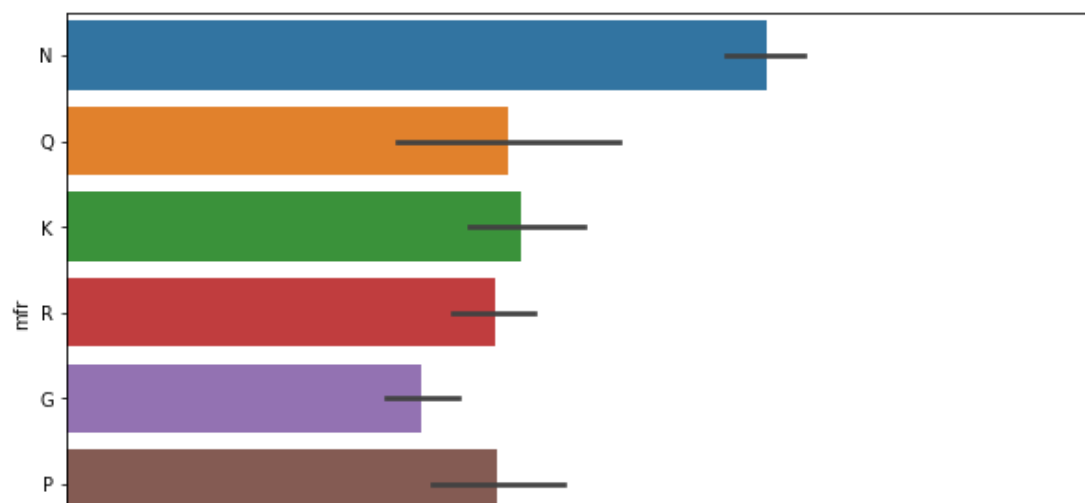
```
In [29]:
```

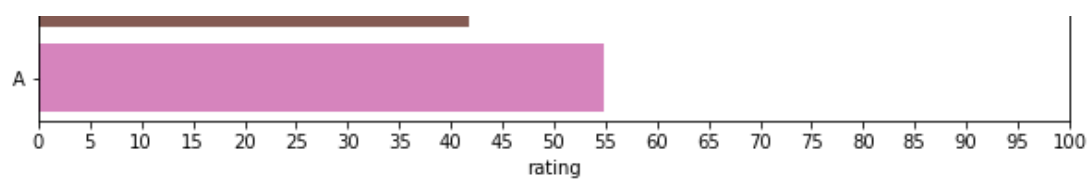
```
plt.figure(figsize=(10,6))
plt.barh(mfr,rt,color='b')
plt.xlabel('RATING')
plt.ylabel('MANUFACTURERS')
plt.xticks(np.arange(0,105,5))
plt.show()
```



```
In [50]:
```

```
plt.figure(figsize=(10,6))
sns.barplot(data=d,x="rating",y="mfr")
plt.xticks(np.arange(0,105,5))
plt.show()
```





In []: