Home Work 4

1. sugar

In [3]:
```
import numpy as np
import pandas as pd
raw_data = pd.read_csv('cereal.csv')
raw_data.head()
```

Out[3]:

	name	mfr	type	calories	protein	fat	sodium	fiber	carbo	sugars	potass	vitamins	sh
0	100% Bran	N	C	70	4	1	130	10.0	5.0	6	280	25	
1	100% Natural Bran	Q	C	120	3	5	15	2.0	8.0	8	135	0	
2	All-Bran	K	C	70	4	1	260	9.0	7.0	5	320	25	
3	All-Bran with Extra Fiber	K	C	50	4	0	140	14.0	8.0	0	330	25	
4	Almond Delight	R	C	110	2	2	200	1.0	14.0	8	-1	25	

In [6]:
```
# let extract the feature names
feature_names = raw_data.columns
print(feature_names)
```

```
Index(['name', 'mfr', 'type', 'calories', 'protein', 'fat', 'sodium', 'fibe
r',
       'carbo', 'sugars', 'potass', 'vitamins', 'shelf', 'weight', 'cups',
       'rating'],
      dtype='object')
```

In [19]:
```
#let extrat list of sugar per serving
raw_data['sugars']
```

Out[19]:
```
0      6
1      8
2      5
3      0
4      8
      ..
72     3
73    12
74     3
75     3
76     8
Name: sugars, Length: 77, dtype: int64
```

In [20]:
```
# list of ounce per serving
raw_data['weight']
```

Out[20]:
```
0     1.0
1     1.0
2     1.0
3     1.0
4     1.0
     ...
72    1.0
73    1.0
74    1.0
75    1.0
76    1.0
Name: weight, Length: 77, dtype: float64
```

In [18]:
```
# we can caluculate list of sugar per once
raw_data['sugar_per_ounce'] = raw_data['sugars'] / raw_data['weight']
raw_data
```

Out[18]:

	name	mfr	type	calories	protein	fat	sodium	fiber	carbo	sugars	potass	vitamins
0	100% Bran	N	C	70	4	1	130	10.0	5.0	6	280	25
1	100% Natural Bran	Q	C	120	3	5	15	2.0	8.0	8	135	0
2	All-Bran	K	C	70	4	1	260	9.0	7.0	5	320	25
3	All-Bran with Extra Fiber	K	C	50	4	0	140	14.0	8.0	0	330	25
4	Almond Delight	R	C	110	2	2	200	1.0	14.0	8	-1	25
...	...	...	...	...	...	...	...	...	...	...	...	...
72	Triples	G	C	110	2	1	250	0.0	21.0	3	60	25
73	Trix	G	C	110	1	1	140	0.0	13.0	12	25	25
74	Wheat Chex	R	C	100	3	1	230	3.0	17.0	3	115	25
75	Wheaties	G	C	100	3	1	200	3.0	17.0	3	110	25
76	Wheaties Honey Gold	G	C	110	2	1	200	1.0	16.0	8	60	25

77 rows × 17 columns

In [22]:
```
# product has a least of amount of sugar
idx = raw_data['sugar_per_ounce'].argmin()
Name_least = raw_data.loc[idx,'name']
print(idx)
print(Name_least)
```

```
57
Quaker Oatmeal
```

In [23]:
```
# average sugar per ounce
print(raw_data['sugar_per_ounce'].mean())
```

```
6.555489623158796
```

2.calories

In [24]:
```
# calorie per gram of each cereal
print(raw_data['calories'])
```

```
0      70
1     120
2      70
3      50
4     110
     ...
72    110
73    110
74    100
75    100
76    110
Name: calories, Length: 77, dtype: int64
```

In [25]:
```
# highest calorie per gram
print(raw_data['calories'].max())
```

```
160
```

In [26]:
```
# index highest calories
print(raw_data['calories'].argmax())
```

```
46
```

In [27]:
```
# name of the product of highest calorie
print(raw_data.loc[46,'name'])
```

```
Mueslix Crispy Blend
```

In [28]:
```
# product with the lowest calories
# index lowest calories
# name of the product
print(raw_data['calories'].min())
print(raw_data['calories'].argmin())
```

```
50
3
```

In [29]:
```
print(raw_data.loc[3,'name'])
```

```
All-Bran with Extra Fiber
```

In [ ]:
```

```