Adhish Bahl

2347203

1MCA B

Python Lab Exam

ESE Component 3

Importing Libraries

```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
```

Q1) Importing Dataset

```
df = pd.read_csv("./UScereal.csv")
```

Q2) Finding Summary of the Dataset

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 65 entries, 0 to 64
Data columns (total 12 columns):
 #   Column     Non-Null Count  Dtype  
---  ------     --------------  -----  
 0   Name       65 non-null     object 
 1   mfr        65 non-null     object 
 2   calories   65 non-null     float64
 3   protein    65 non-null     float64
 4   fat        65 non-null     float64
 5   sodium     65 non-null     float64
 6   fibre      65 non-null     float64
 7   carbo      65 non-null     float64
 8   sugars     65 non-null     float64
 9   shelf      65 non-null     int64  
 10  potassium  65 non-null     float64
 11  vitamins   65 non-null     object 
dtypes: float64(8), int64(1), object(3)
memory usage: 6.2+ KB
```

Observation: From the above report generated using info() function, we can observe that there are 65 rows in the dataset and 12 coulumns. Datatype of each column is given in the report. Like, Datatype of "name" is "object", "calaries" is "float" and "shelf" is "int" and so on for all the columns.

We can also see that there are no missing data i the dataaset because it clearly says for all the columns that there are "65 non-null" values. This concludes that there are no missing data in the dataset since there are only 65 rows in it.

Head and Tail of the dataset

```
df.head(10)
```

```
                        Name mfr  calories  protein   fat  sodium
fibre  \
0                  100% Bran   N    212.12    12.12  3.03  393.94
30.30
1                   All-Bran   K    212.12    12.12  3.03  787.88
27.27
2  All-Bran with Extra Fiber   K    100.00     8.00  0.00  280.00
28.00
3    Apple Cinnamon Cheerios   G    146.67     2.67  2.67  240.00
2.00
4                 Apple Jacks   K    110.00     2.00  0.00  125.00
1.00
5                   Basic 4   G    173.33     4.00  2.67  280.00
2.67
6                 Bran Chex   R    134.33     2.99  1.49  298.51
5.97
7                Bran Flakes   P    134.33     4.48  0.00  313.43
7.46
8               Cap'n'Crunch   Q    160.00     1.33  2.67  293.33
0.00
9                  Cheerios   G     88.00     4.80  1.60  232.00
1.60

   carbo  sugars  shelf  potassium  vitamins
0  15.15   18.18      3     848.48  enriched
1  21.21   15.15      3     969.70  enriched
2  16.00    0.00      3     660.00  enriched
3  14.00   13.33      1      93.33  enriched
4  11.00   14.00      2      30.00  enriched
5  24.00   10.67      3     133.33  enriched
6  22.39    8.96      1     186.57  enriched
7  19.40    7.46      3     283.58  enriched
8  16.00   16.00      2      46.67  enriched
9  13.60    0.80      1      84.00  enriched
```

```
df.tail(10)
```

```
                    Name mfr  calories  protein   fat  sodium  fibre
carbo  \
55               Smacks   K    146.67     2.67  1.33   93.33   1.33
12.00
56            Special K   K    110.00     6.00  0.00  230.00   1.00
16.00
57    Total Corn Flakes   G    110.00     2.00  1.00  200.00   0.00
21.00
58    Total Raisin Bran   G    140.00     3.00  1.00  190.00   4.00
15.00
```

```
59    Total Whole Grain   G    100.00     3.00  1.00  200.00   3.00
16.00
60              Triples   G    146.67     2.67  1.33  333.33   0.00
28.00
61                 Trix   G    110.00     1.00  1.00  140.00   0.00
13.00
62           Wheat Chex   R    149.25     4.48  1.49  343.28   4.48
25.37
63              Wheaties   G    100.00     3.00  1.00  200.00   3.00
17.00
64  Wheaties Honey Gold   G    146.67     2.67  1.33  266.67   1.33
21.33

    sugars  shelf  potassium  vitamins
55   20.00      2      53.33  enriched
56    3.00      1      55.00  enriched
57    3.00      3      35.00      100%
58   14.00      3     230.00      100%
59    3.00      3     110.00      100%
60    4.00      3      80.00  enriched
61   12.00      2      25.00  enriched
62    4.48      1     171.64  enriched
63    3.00      1     110.00  enriched
64   10.67      1      80.00  enriched
```

Q3) Average protein value of each manufacturer

```
df.groupby('mfr')['protein'].mean()

mfr
G    2.885000
K    3.919048
N    7.026667
P    4.698889
Q    3.460000
R    2.604000
Name: protein, dtype: float64
```

Observation: Here all the Manufactureres are listed with their mean/average value of protein in their products. groupby() function is used to group the Manufactureres in the dataset and then that group is passed to find their mean/average value by the function mean().

Q4) Name of the Cereal with high sugar from Manufacturer G.

```
df2 = df[df["mfr"] == "G"]

df2 = df2.sort_values(by=["sugars"], ascending=False)

print(df2)
```

```
                        Name mfr  calories  protein   fat  sodium
fibre  \
43     Oatmeal Raisin Crisp   G    260.00     6.00  4.00  340.00
3.00
49          Raisin Nut Bran   G    200.00     6.00  4.00  280.00
5.00
58        Total Raisin Bran   G    140.00     3.00  1.00  190.00
4.00
11                Clusters   G    220.00     6.00  4.00  280.00
4.00
3   Apple Cinnamon Cheerios   G    146.67     2.67  2.67  240.00
2.00
19   Crispy Wheat & Raisins   G    133.33     2.67  1.33  186.67
2.67
33       Honey Nut Cheerios   G    146.67     4.00  1.33  333.33
2.00
12              Cocoa Puffs   G    110.00     1.00  1.00  180.00
0.00
16            Count Chocula   G    110.00     1.00  1.00  180.00
0.00
61                    Trix   G    110.00     1.00  1.00  140.00
0.00
38             Lucky Charms   G    110.00     2.00  1.00  180.00
0.00
28           Golden Grahams   G    146.67     1.33  1.33  373.33
0.00
10    Cinnamon Toast Crunch   G    160.00     1.33  4.00  280.00
0.00
5                   Basic 4   G    173.33     4.00  2.67  280.00
2.67
64      Wheaties Honey Gold   G    146.67     2.67  1.33  266.67
1.33
40     Multi-Grain Cheerios   G    100.00     2.00  1.00  220.00
2.00
60                  Triples   G    146.67     2.67  1.33  333.33
0.00
57        Total Corn Flakes   G    110.00     2.00  1.00  200.00
0.00
59        Total Whole Grain   G    100.00     3.00  1.00  200.00
3.00
63                Wheaties   G    100.00     3.00  1.00  200.00
3.00
36                     Kix   G     73.33     1.33  0.67  173.33
0.00
9                   Cheerios   G     88.00     4.80  1.60  232.00
1.60

    carbo  sugars  shelf  potassium  vitamins
43  27.00   20.00      3     240.00  enriched
49  21.00   16.00      3     280.00  enriched
```

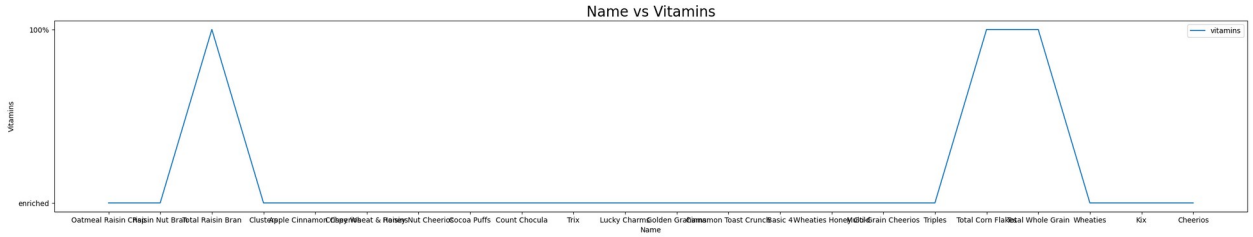
58	15.00	14.00	3	230.00	100%
11	26.00	14.00	3	210.00	enriched
3	14.00	13.33	1	93.33	enriched
19	14.67	13.33	3	160.00	enriched
33	15.33	13.33	1	120.00	enriched
12	12.00	13.00	2	55.00	enriched
16	12.00	13.00	2	65.00	enriched
61	13.00	12.00	2	25.00	enriched
38	12.00	12.00	2	55.00	enriched
28	20.00	12.00	2	60.00	enriched
10	17.33	12.00	2	60.00	enriched
5	24.00	10.67	3	133.33	enriched
64	21.33	10.67	1	80.00	enriched
40	15.00	6.00	1	90.00	enriched
60	28.00	4.00	3	80.00	enriched
57	21.00	3.00	3	35.00	100%
59	16.00	3.00	3	110.00	100%
63	17.00	3.00	1	110.00	enriched
36	14.00	2.00	2	26.67	enriched
9	13.60	0.80	1	84.00	enriched

Observation: In the above list we can see that all the products from the manufacturer "G" are listed with descending order of their sugar value. We can see that the product with the highest sugar value is "Oatmeal Raisin Crisp" with sugar value being "20". The product with the lowest sugar value is "Cheerios" with sugar value being "0.80".

Generate any three different suitable plots using matplotlib library for the following. Use appropriate formatting like title, legend, x-value, y-value ect.

a)Using only shelf or vitamins

```
df3 = df.head(10)
plt.figure().set_figwidth(30)
plt.plot(df2["Name"], df2["vitamins"], label = "vitamins")
plt.xlabel('Name')
plt.ylabel('Vitamins')
plt.title('Name vs Vitamins', fontsize = 20)
plt.legend()
plt.show()
```



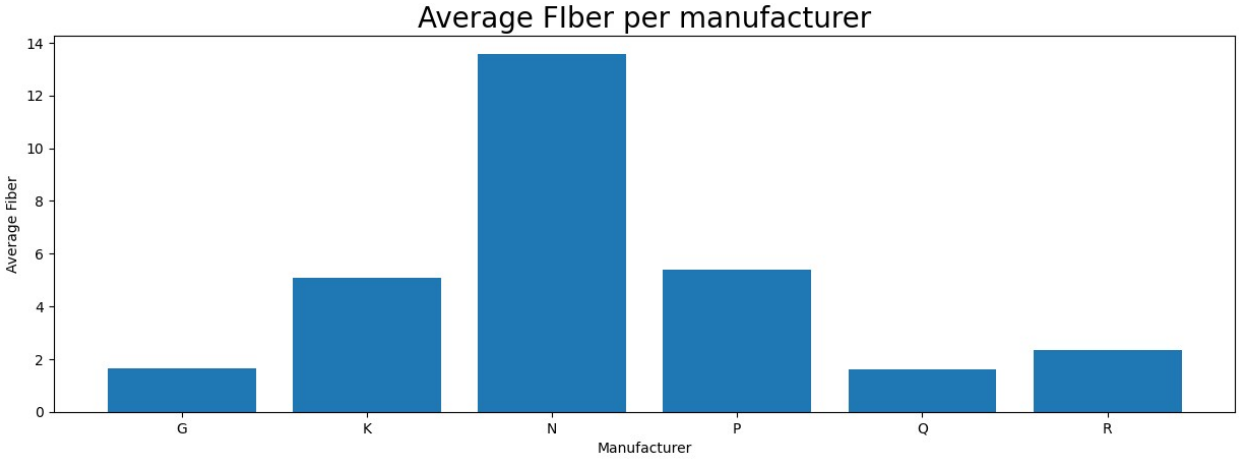
Observation: We can observe from the above line graph to see the vitamins value for the fisrt 10 recordds in the dataset with their names.

b) Using mfr and fiber

```
df3 = df.groupby('mfr')['fibre'].mean()

mfrDict = {
    "mfr": ["G","K", "N", "P", "Q", "R"]
}
x =  pd.DataFrame(mfrDict)
y =  pd.DataFrame(df.groupby('mfr')['fibre'].mean())

plt.figure().set_figwidth(15)
plt.bar(x['mfr'], y['fibre'])
plt.xlabel('Manufacturer')
plt.ylabel('Average Fiber')
plt.title('Average FIber per manufacturer', fontsize = 20)
plt.show()
```

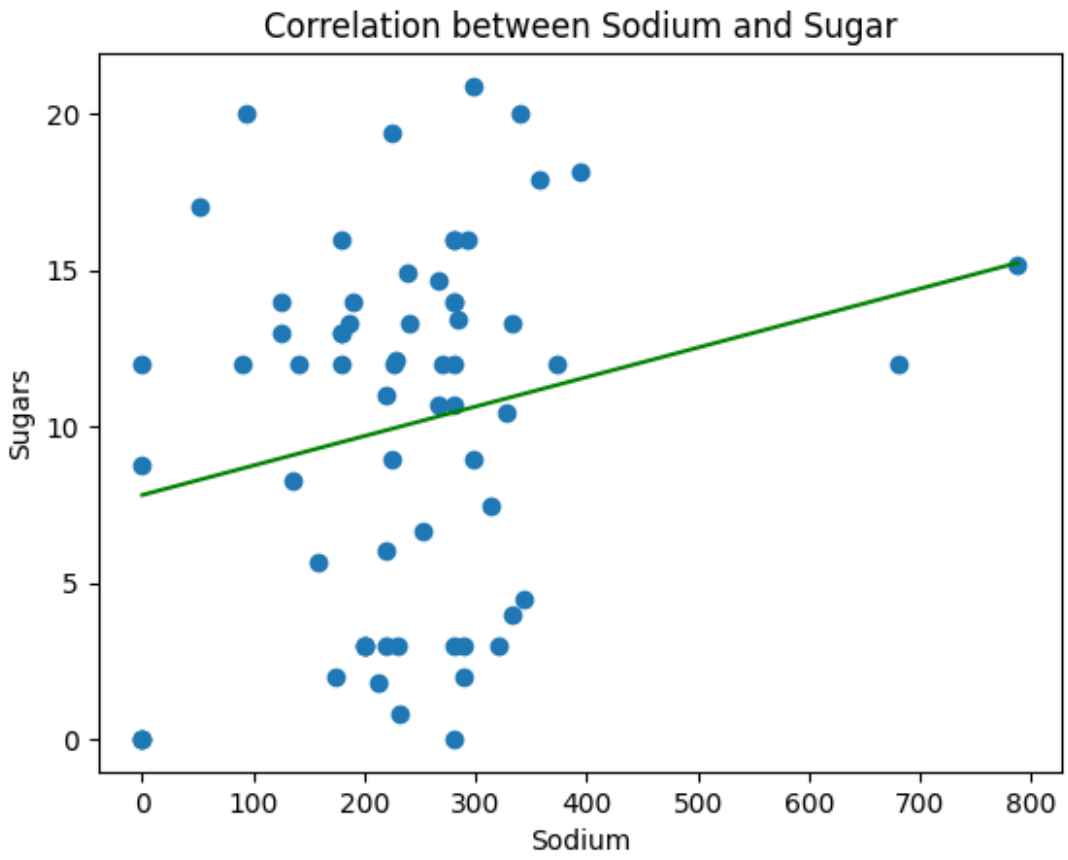


Observation: We can observe from the above bar graph that the manufacturer "N" has the height average "fibre" value for the products, while Q being the lowest. From the above bar graph we can observe the average "fibre" value for the differnt "Manufacturer".

c) Using sodium and sugars

```
plt.title ('Correlation between Sodium and Sugar')
plt.scatter (df['sodium'], df['sugars'])
plt.plot (np.unique(df['sodium']), np.poly1d (np.polyfit(df['sodium'],
df['sugars'], 1))(np.unique (df['sodium'])), color = 'green')
plt.xlabel ('Sodium')
plt.ylabel ('Sugars')

Text(0, 0.5, 'Sugars')
```



Observation: In the above graph we can see that the correlation between "Sodium" and "Sugars" in all the products of the dataset is POSITIVE.This means that as anyone value increases abome the two, the second value increases by itself.