In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

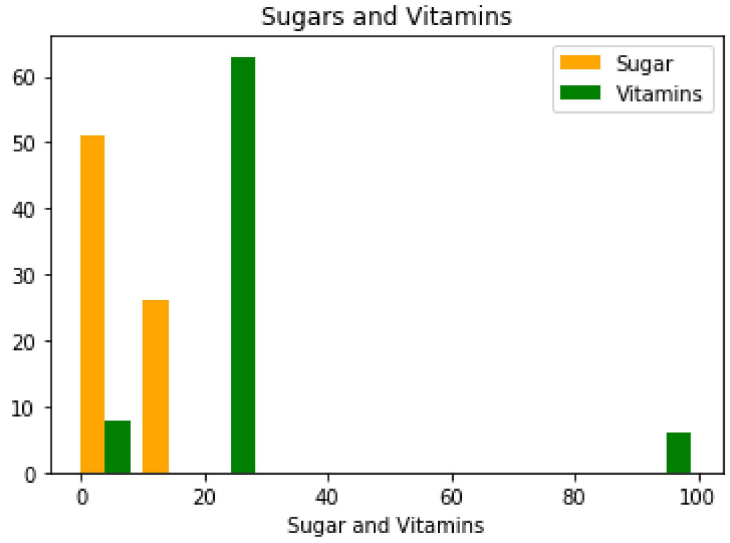
In [2]:
```
# 1.Load the data from “cereal.csv” and plot histograms of sugar and vitamin c
ontent across different cereals
df_cereal = pd.read_csv("cereal.csv")
df_cereal.head()
```

Out[2]:

	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 [5]:
```
plt.hist([df_cereal["sugars"], df_cereal["vitamins"]], color=['orange','green'
])
plt.title("Sugars and Vitamins")
plt.xlabel("Sugar and Vitamins")
plt.legend(["Sugar", "Vitamins"])
```

Out[5]:
```
<matplotlib.legend.Legend at 0x1c2c3dc8ec8>
```



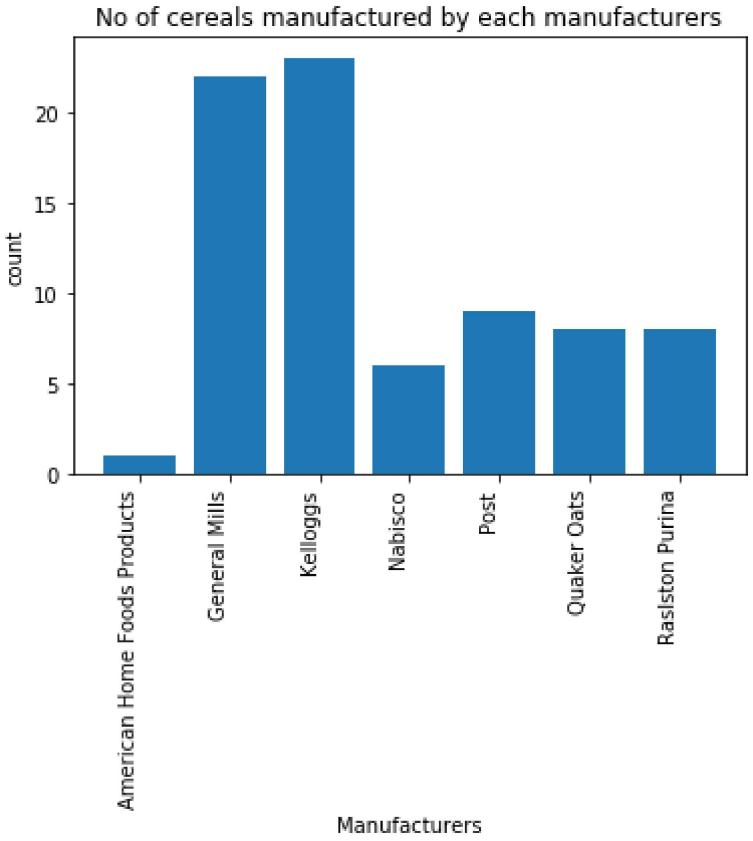
In [6]:
```
# 2.The names of the manufactures are coded using alphabets, create a new colu
mn with their fullname using the below mapping
dict_mfr = {'N': 'Nabisco',
            'Q': 'Quaker Oats',
            'K': 'Kelloggs',
            'R': 'Raslston Purina',
            'G': 'General Mills' ,
            'P' :'Post' ,
            'A':'American Home Foods Products'}
df_cereal["manufacturers"] = [dict_mfr[mfr] for mfr in df_cereal["mfr"]]
df_cereal.head()
```

Out[6]:

	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 [13]:
```
# Create a bar plot where each manufacturer is on the y axis and the
# height of the bars depict the number of cereals manufactured by them.
grouped_mfr = df_cereal.groupby(["manufacturers"], as_index=False).count()

x=  grouped_mfr["manufacturers"]
y=  grouped_mfr["mfr"]

plt.bar(x,y)
plt.setp(plt.gca().get_xticklabels(), rotation=90, horizontalalignment='right'
)
plt.xlabel("Manufacturers")
plt.ylabel("count")
plt.title("No of cereals manufactured by each manufacturers")
```

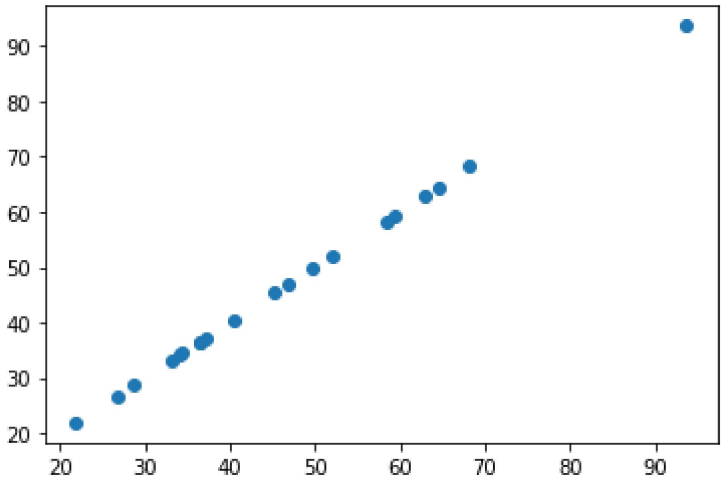
Out[13]: Text(0.5, 1.0, 'No of cereals manufactured by each manufacturers')



In [14]:
```
# 3.Extract the rating as your target variable ‘y’ and all numerical parameter
s as your predictors ‘x’. Separate 25% of your data as test set
from sklearn.model_selection import train_test_split
X = df_cereal.iloc[:,3:15]
Y = df_cereal["rating"]

x_train, x_test, y_train, y_test = train_test_split(X, Y, test_size=0.25, rand
om_state=10)
```

In [17]:
```
# 4.Fit a linear regression module and measure the mean squared error on test
 dataset.
from sklearn.linear_model import LinearRegression
from sklearn.metrics import accuracy_score

linear_model = LinearRegression()
linear_model.fit(x_train, y_train)

predicted_ratings = linear_model.predict(x_test)
plt.scatter(np.array(predicted_ratings), np.array(y_test))
```

Out[17]: <matplotlib.collections.PathCollection at 0x1c2c6c53208>



In [ ]: