In [25]:

```
#import packages in python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import os
```

In [2]:

```
#read globalsuperstore dataset
file=pd.read_excel('Global superstore.xls')
file.head()
```

1	26341	IN-2013-77878	2013-02-05	2013-02-07	Second Class	JR-16210	Justin Ritter	Corporate	Wollongong	New South Wales	...	F 10
2	25330	IN-2013-71249	2013-10-17	2013-10-18	First Class	CR-12730	Craig Reiter	Consumer	Brisbane	Queensland	...	T 10
3	13524	ES-2013-1579342	2013-01-28	2013-01-30	First Class	KM-16375	Katherine Murray	Home Office	Berlin	Berlin	...	T 10
4	47221	SG-2013-4320	2013-11-05	2013-11-06	Same Day	RH-9495	Rick Hansen	Consumer	Dakar	Dakar	...	10

In [3]:

```
#check the null value from dataset
file.isnull().sum()
```

Out[3]:

```
Row ID               0
Order ID             0
Order Date           0
Ship Date            0
Ship Mode            0
Customer ID          0
Customer Name        0
Segment              0
City                 0
State                0
Country              0
Postal Code      41296
Market               0
Region               0
Product ID           0
Category             0
Sub-Category         0
Product Name         0
Sales                0
Quantity             0
Discount             0
Profit               0
Shipping Cost        0
Order Priority       0
dtype: int64
```

In [4]:

```
#delete the null value column postal code
file=file.drop(columns=['Postal Code'])
file.head()
```

Out[4]:

	Row ID	Order ID	Order Date	Ship Date	Ship Mode	Customer ID	Customer Name	Segment	City	
0	32298	CA-2012-124891	2012-07-31	2012-07-31	Same Day	RH-19495	Rick Hansen	Consumer	New York City	Ne
1	26341	IN-2013-77878	2013-02-05	2013-02-07	Second Class	JR-16210	Justin Ritter	Corporate	Wollongong	New
2	25330	IN-2013-71249	2013-10-17	2013-10-18	First Class	CR-12730	Craig Reiter	Consumer	Brisbane	Quee
3	13524	ES-2013-1579342	2013-01-28	2013-01-30	First Class	KM-16375	Katherine Murray	Home Office	Berlin	
4	47221	SG-2013-4320	2013-11-05	2013-11-06	Same Day	RH-9495	Rick Hansen	Consumer	Dakar	

5 rows × 23 columns

In [5]:

```
#check the null value column delete or not
file.isnull().sum()
```

Out[5]:

```
Row ID            0
Order ID          0
Order Date        0
Ship Date         0
Ship Mode         0
Customer ID       0
Customer Name     0
Segment           0
City              0
State             0
Country           0
Market            0
Region            0
Product ID        0
Category          0
Sub-Category      0
Product Name      0
Sales             0
Quantity          0
Discount          0
Profit            0
Shipping Cost     0
Order Priority    0
dtype: int64
```

In [6]:

```
# to display statistics about data
file.describe()
```

Out[6]:

	Row ID	Sales	Quantity	Discount	Profit	Shipping Cost
count	51290.00000	51290.000000	51290.000000	51290.000000	51290.000000	51290.000000
mean	25645.50000	246.490581	3.476545	0.142908	28.610982	26.375818
std	14806.29199	487.565361	2.278766	0.212280	174.340972	57.296810
min	1.00000	0.444000	1.000000	0.000000	-6599.978000	0.002000
25%	12823.25000	30.758625	2.000000	0.000000	0.000000	2.610000
50%	25645.50000	85.053000	3.000000	0.000000	9.240000	7.790000
75%	38467.75000	251.053200	5.000000	0.200000	36.810000	24.450000
max	51290.00000	22638.480000	14.000000	0.850000	8399.976000	933.570000

In [7]:

```
# to display basic info datatype
file.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 51290 entries, 0 to 51289
Data columns (total 23 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   Row ID          51290 non-null  int64
 1   Order ID        51290 non-null  object
 2   Order Date      51290 non-null  datetime64[ns]
 3   Ship Date       51290 non-null  datetime64[ns]
 4   Ship Mode       51290 non-null  object
 5   Customer ID     51290 non-null  object
 6   Customer Name   51290 non-null  object
 7   Segment         51290 non-null  object
 8   City            51290 non-null  object
 9   State           51290 non-null  object
 10  Country         51290 non-null  object
 11  Market          51290 non-null  object
 12  Region          51290 non-null  object
 13  Product ID      51290 non-null  object
 14  Category        51290 non-null  object
 15  Sub-Category    51290 non-null  object
 16  Product Name    51290 non-null  object
 17  Sales           51290 non-null  float64
 18  Quantity        51290 non-null  int64
 19  Discount        51290 non-null  float64
 20  Profit          51290 non-null  float64
 21  Shipping Cost   51290 non-null  float64
 22  Order Priority  51290 non-null  object
dtypes: datetime64[ns](2), float64(4), int64(2), object(15)
memory usage: 9.0+ MB
```

In [8]:

```
#to display no of sample on each class
file['Segment'].value_counts()
```

Out[8]:

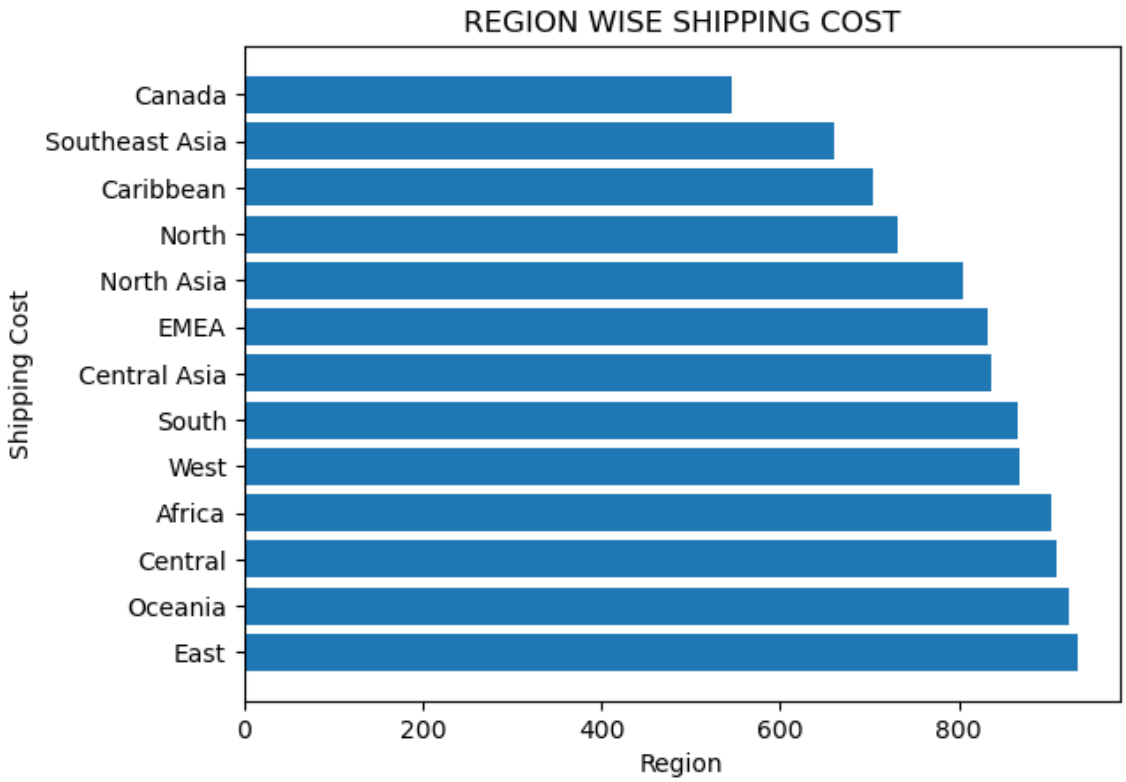
```
Consumer       26518
Corporate      15429
Home Office     9343
Name: Segment, dtype: int64
```

In [21]:

```
#to display
a=file['Region']
b=file['Shipping Cost']
plt.xlabel('Region')
plt.ylabel('Shipping Cost')
plt.title("REGION WISE SHIPPING COST")
plt.barh(a,b)
```

Out[21]:

```
<BarContainer object of 51290 artists>
```



In [10]:

```
#pie chart
#change sales datatype float to int
file['Sales'] = file['Sales'].astype(int)
display(file.dtypes)
```

```
Row ID                     int64
Order ID                  object
Order Date        datetime64[ns]
Ship Date         datetime64[ns]
Ship Mode                 object
Customer ID               object
Customer Name             object
Segment                   object
City                      object
State                     object
Country                   object
Market                    object
Region                    object
Product ID                object
Category                  object
Sub-Category              object
Product Name              object
Sales                      int32
Quantity                   int64
Discount                 float64
Profit                   float64
Shipping Cost            float64
Order Priority            object
dtype: object
```

In [19]:

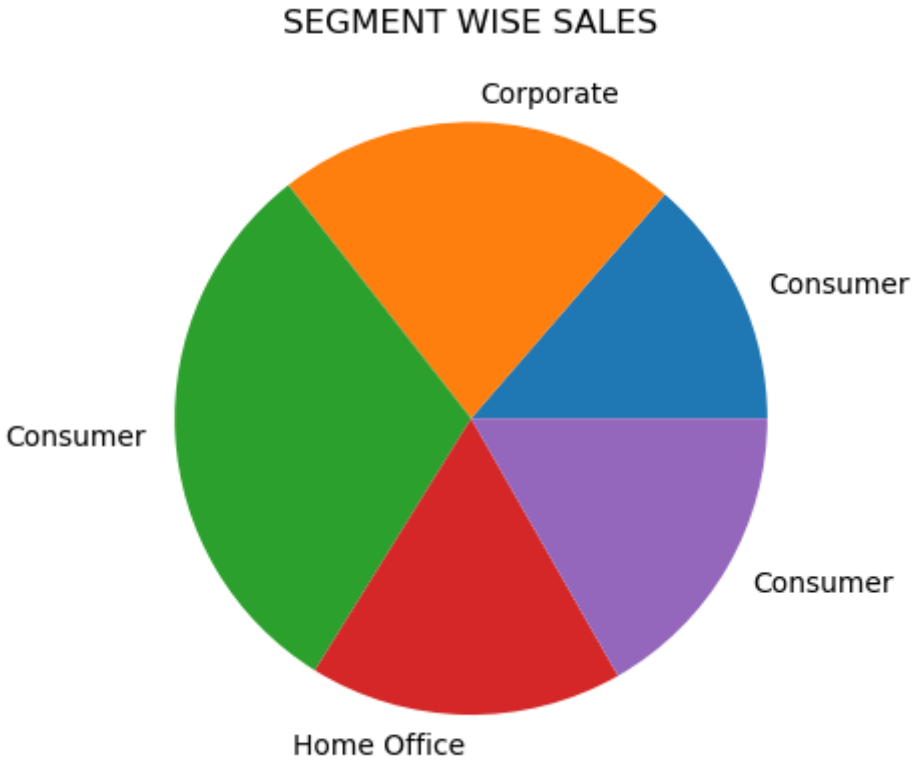
```
#PIE CHART
d=file['Sales']
e= d.head(n=5)
f=file['Segment']
g=f.head(n=5)
```

In [20]:

```
plt.title("SEGMENT WISE SALES")
plt.pie(e,labels=g)
```

Out[20]:

```
([<matplotlib.patches.Wedge at 0x1a88f63bdc0>,
  <matplotlib.patches.Wedge at 0x1a88f647370>,
  <matplotlib.patches.Wedge at 0x1a88f647850>,
  <matplotlib.patches.Wedge at 0x1a88f647d30>,
  <matplotlib.patches.Wedge at 0x1a88f653250>],
 [Text(1.0004138158326534, 0.45735347062326964, 'Consumer'),
  Text(0.02685970345934129, 1.0996720221639162, 'Corporate'),
  Text(-1.0983609076990308, -0.06002763062591273, 'Consumer'),
  Text(-0.01991583219119784, -1.099819694144514, 'Home Office'),
  Text(0.9513486874808749, -0.552209810514461, 'Consumer')])
```



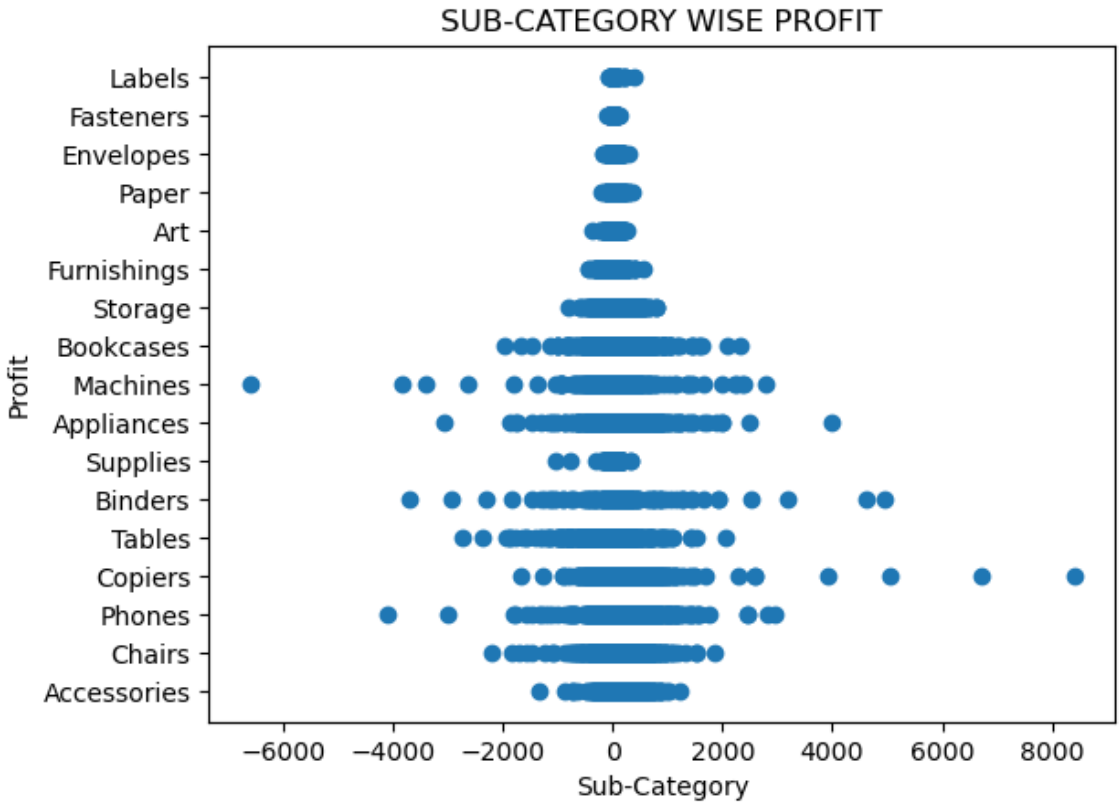
In [22]:

```
#scatter plot
#change Profit datatype float to int
f=file['Sub-Category']
file['Profit'] = file['Profit'].astype(int)
display(file.dtypes)
g=file['Profit']
```

```
Row ID                     int64
Order ID                  object
Order Date        datetime64[ns]
Ship Date         datetime64[ns]
Ship Mode                 object
Customer ID               object
Customer Name             object
Segment                   object
City                      object
State                     object
Country                   object
Market                    object
Region                    object
Product ID                object
Category                  object
Sub-Category              object
Product Name              object
Sales                      int32
Quantity                   int64
Discount                 float64
Profit                     int32
Shipping Cost            float64
Order Priority            object
dtype: object
```

In [14]:

```
plt.scatter(g,f)
plt.title("SUB-CATEGORY WISE PROFIT")
plt.xlabel("Sub-Category")
plt.ylabel("Profit")
plt.show()
```

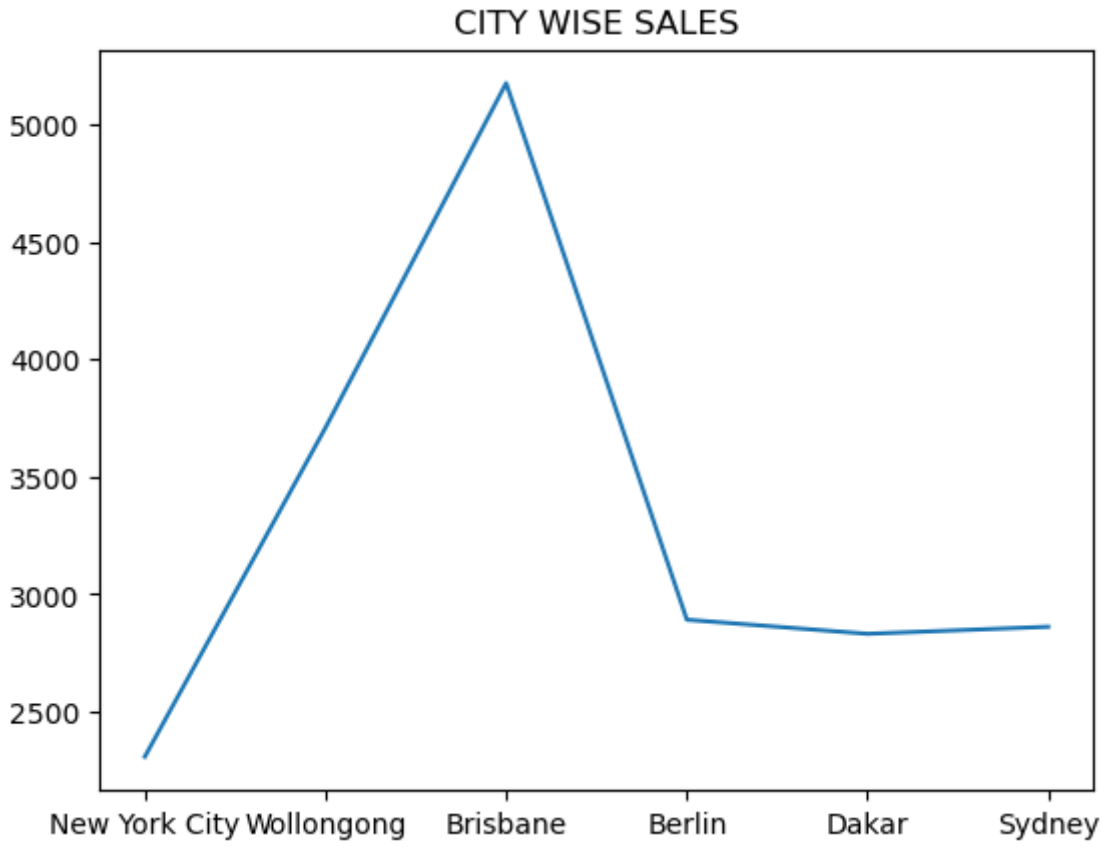


In [17]:

```
#Line chart
x=file['Sales']
b= x.head(n = 6)
y=file['City']
a=y.head(n=6)
plt.title("CITY WISE SALES")
plt.plot(a,b)
```

Out[17]:

```
[<matplotlib.lines.Line2D at 0x1a88f359970>]
```



In [24]:

```
z=("THANK YOU")
print(z)
```

```
THANK YOU
```

In [ ]:

```

```