In [1]:

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

In [2]:

```
my=pd.read_excel('Global.xls')
my.head()
```

Out[2]:

| | 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 | Nev |
| **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 | Queer |
| **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 × 24 columns

In [3]:

```
#check the null value from dataset
my.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
my=my.drop(columns=['Postal Code'])
my.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 | Nev |
| **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 | Queer |
| **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
my.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
my.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
my.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
my['Segment'].value_counts()
```

Out[8]:

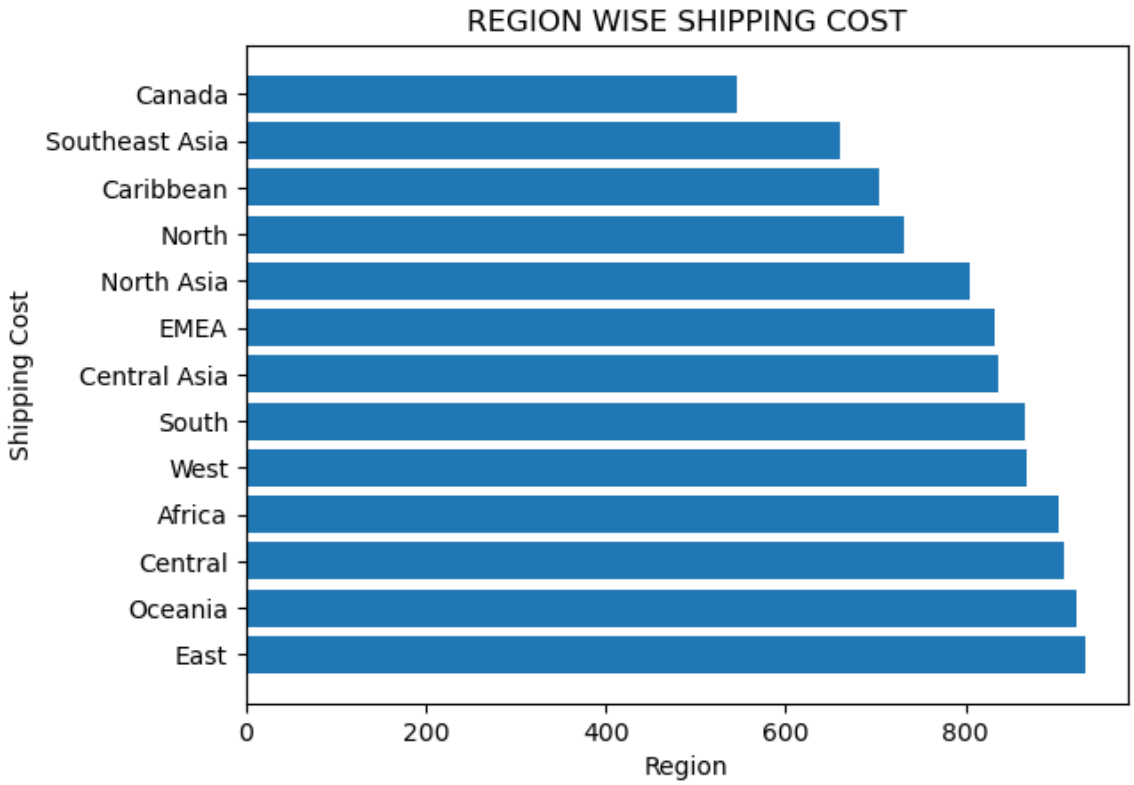
```
Consumer       26518
Corporate      15429
Home Office     9343
Name: Segment, dtype: int64
```

In [9]:

```
#BAR CHART
a=my['Region']
b=my['Shipping Cost']
plt.title('REGION WISE SHIPPING COST')
plt.xlabel('Region')
plt.ylabel('Shipping Cost')
plt.barh(a,b)
```

Out[9]:

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



In [10]:

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

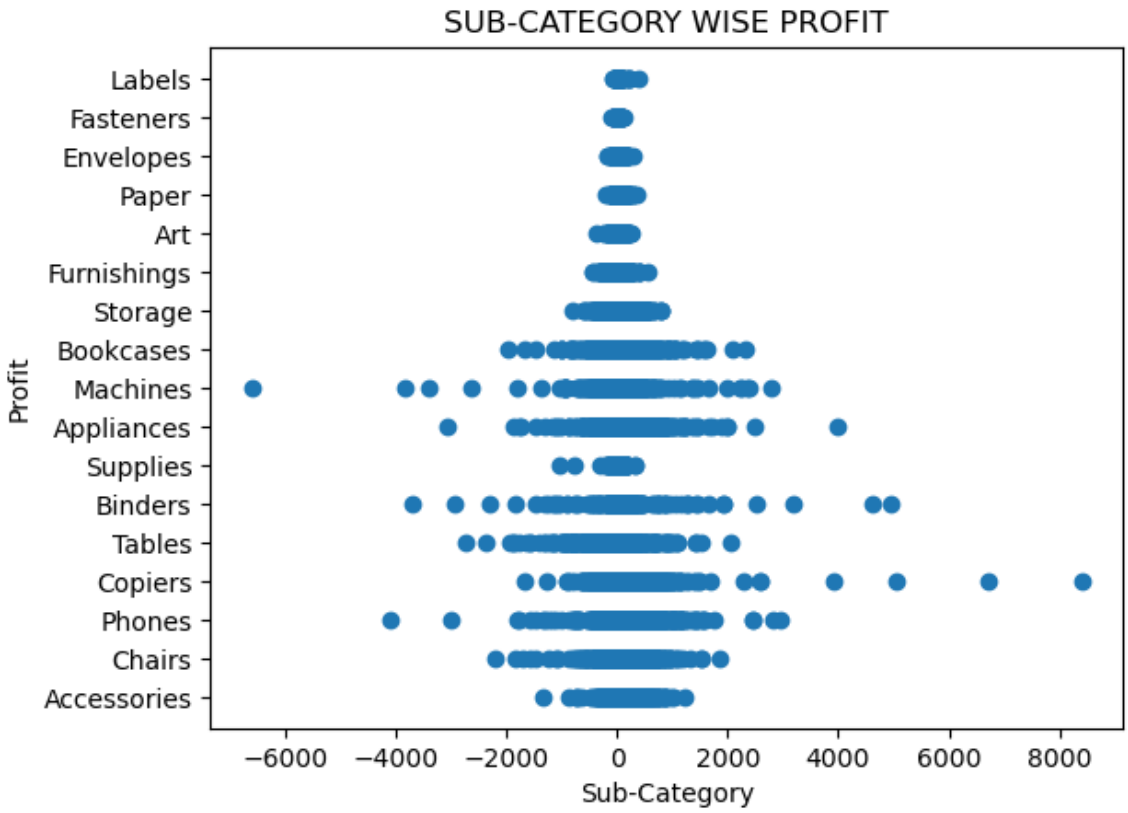
In [11]:

```
#scatter plot
f=my['Sub-Category']
my['Profit'] = my['Profit'].astype(int)
display(my.dtypes)
g=my['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 [12]:

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

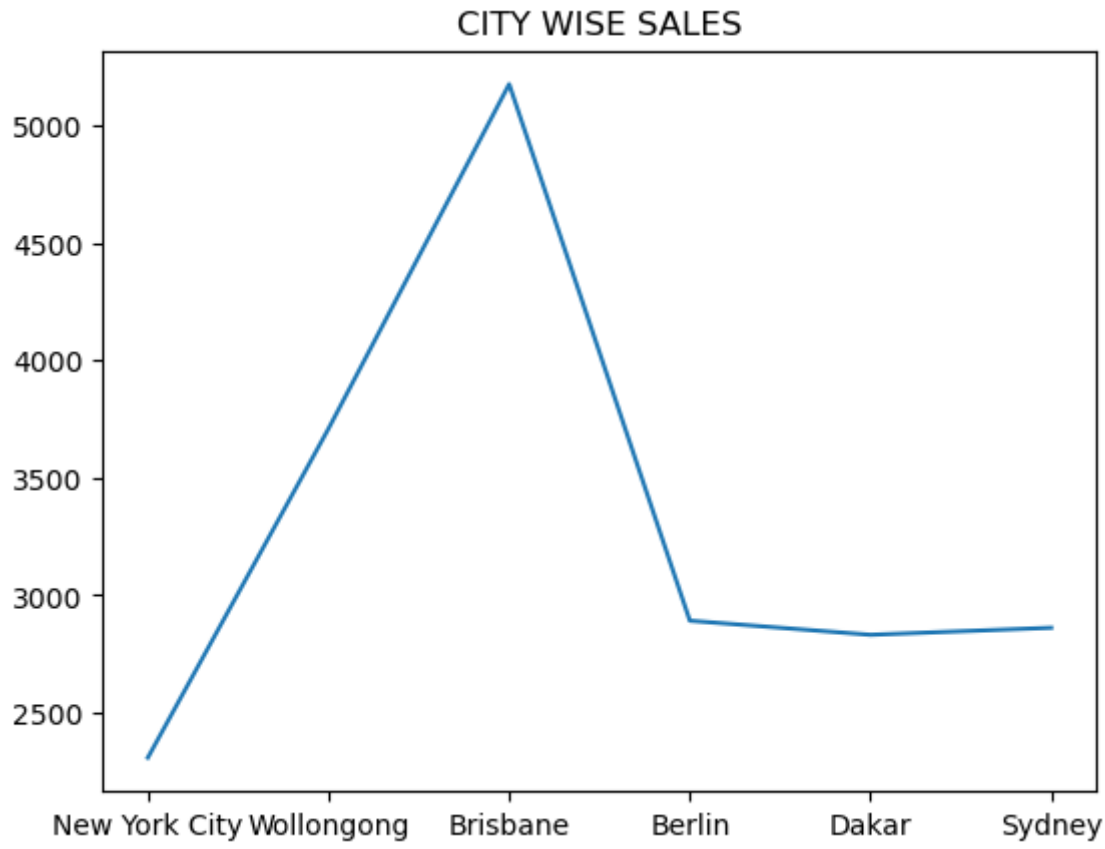


In [13]:

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

Out[13]:

```
Text(0.5, 1.0, 'CITY WISE SALES')
```




In [14]:

```
ab=my['Sales']
w=ab.head(n=3)
w
```

Out[14]:

```
0    2309
1    3709
2    5175
Name: Sales, dtype: int32
```

In [15]:

```
cd=my['Segment']
l=cd.head(n=3)
l
```

Out[15]:

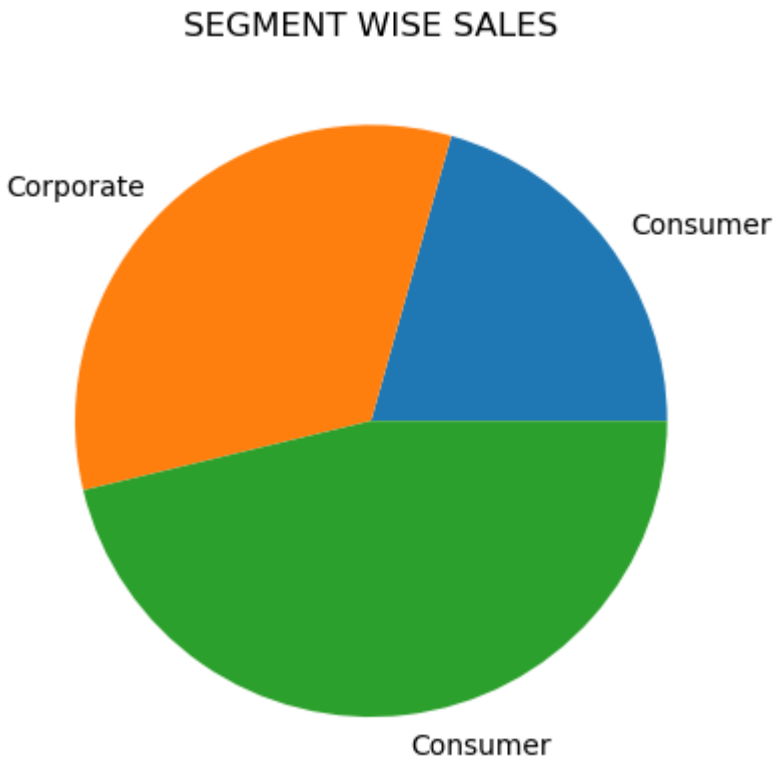
```
0     Consumer
1    Corporate
2     Consumer
Name: Segment, dtype: object
```

In [16]:

```
#PIE CHART
plt.pie(w,labels=l)
plt.title('SEGMENT WISE SALES')
```

Out[16]:

```
Text(0.5, 1.0, 'SEGMENT WISE SALES')
```




In [ ]: