

In [3]:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans
```

In [6]:

```
x = [ [2,10],[9, 0], [6, 5], [11, 0], [9, 699], [4, 220], [6, 550],[5, 553],
      [8, 450],
      [4, 150], [5, 120], [7, 100], [5, 200], [12, 0], [8, 0],[6, 120],
      [7, 404], [5, 388], [6, 225], [8, 350], [8, 236], [4, 167], [9, 400],
      [9, 817], [10, 1010], [7, 25],[11, 825], [11, 700],[10, 20],[8,921]]
```

In [7]:

```
x
```

Out[7]:

```
[[2, 10],
 [9, 0],
 [6, 5],
 [11, 0],
 [9, 699],
 [4, 220],
 [6, 550],
 [5, 553],
 [8, 450],
 [4, 150],
 [5, 120],
 [7, 100],
 [5, 200],
 [12, 0],
 [8, 0],
 [6, 120],
 [7, 404],
 [5, 388],
 [6, 225],
 [8, 350],
 [8, 236],
 [4, 167],
 [9, 400],
 [9, 817],
 [10, 1010],
 [7, 25],
 [11, 825],
 [11, 700],
 [10, 20],
 [8, 921]]
```

In [9]:

```
k=4
model = KMeans(n_clusters = k , n_init = 5, max_iter = 100)
model = model.fit(x)
```

Out[9]:

```
KMeans(max_iter=100, n_clusters=4, n_init=5)
```

**In a Jupyter environment, please rerun this cell to show the HTML representation or trust the notebook.
On GitHub, the HTML representation is unable to render, please try loading this page with
nbviewer.org.**

KMeans

```
KMeans(max_iter=100, n_clusters=4, n_init=5)
```

In [11]:

```
labels = model.predict(x)
```

In [12]:

```
labels
```

Out[12]:

```
array([3, 3, 3, 3, 2, 1, 0, 0, 0, 1, 1, 1, 1, 3, 3, 1, 0, 0, 1, 0, 1, 1,
       0, 2, 2, 3, 2, 2, 3, 2])
```

In [59]:

```
X = np.array(x)
cluster_points = {}
for cluster_label in range(0,k):
    points_in_cluster = X[labels == cluster_label]
    cluster_points[cluster_label] = points_in_cluster
cluster_points
```

Out[59]:

```
{0: array([[ 6, 550],
           [ 5, 553],
           [ 8, 450],
           [ 7, 404],
           [ 5, 388],
           [ 8, 350],
           [ 9, 400]]),
 1: array([[ 4, 220],
           [ 4, 150],
           [ 5, 120],
           [ 7, 100],
```

```

        [ 5, 200],
        [ 6, 120],
        [ 6, 225],
        [ 8, 236],
        [ 4, 167]]),
2: array([[ 9, 699],
        [ 9, 817],
        [10, 1010],
        [11, 825],
        [11, 700],
        [ 8, 921]]),
3: array([[ 2, 10],
        [ 9, 0],
        [ 6, 5],
        [11, 0],
        [12, 0],
        [ 8, 0],
        [ 7, 25],
        [10, 20]])}

```

In [41]:

```

# print(cluster_points)
centroids = model.cluster_centers_

```

In [49]:

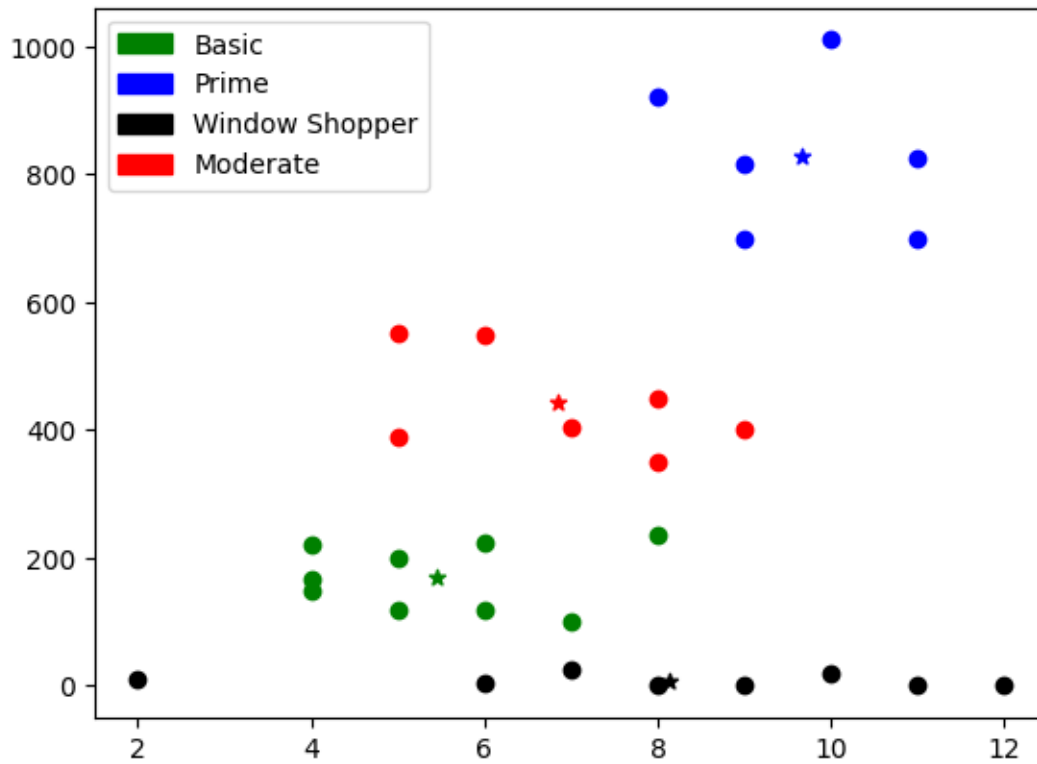
```

import matplotlib.pyplot as plt
import matplotlib.patches as mpatches
fig, ax = plt.subplots()
colors=['red','green','blue','black'];
clr = 0;
i = 0;
for cluster_label, points_array in cluster_points.items():
    print(f"Cluster {cluster_label} points:")
    ax.scatter(centroids[i][0],centroids[i][1],color = colors[clr],marker='*')
    i=i+1;
    for points in points_array:
        ax.scatter(points[0],points[1],color=colors[clr])
    clr=clr + 1

classes=['Window Shopper','Basic', 'Moderate', 'Prime']
colors_for_legends=['black','green','red','blue']
# Cluster Labelling in colors
black_patch = mpatches.Patch(color=colors_for_legends[0], label=classes[0])
green_patch = mpatches.Patch(color=colors_for_legends[1], label=classes[1])
red_patch = mpatches.Patch(color=colors_for_legends[2], label=classes[2])
blue_patch = mpatches.Patch(color=colors_for_legends[3], label=classes[3])
plt.legend(handles=[green_patch, blue_patch, black_patch, red_patch, ])
plt.show()

```

Cluster 0 points:
Cluster 1 points:
Cluster 2 points:
Cluster 3 points:



In [38]:

In [39]:

centeroids

Out[39]:

```
array([[ 6.85714286, 442.14285714],
       [ 5.44444444, 170.88888889],
       [ 9.66666667, 828.66666667],
       [ 8.125      ,   7.5       ]])
```

In [51]:

```
labels = model.predict(x)
```

In [52]:

labels

Out[52]:

```
array([3, 3, 3, 3, 2, 1, 0, 0, 0, 1, 1, 1, 1, 3, 3, 1, 0, 0, 1, 0, 1, 1,
       0, 2, 2, 3, 2, 2, 3, 2])
```

In [53]:

```
X = np.array(x)
```

In [64]:

```
X = np.array(x)
cluster_points = {}
for cluster_label in range(0,k):
    points_in_cluster = X[labels == cluster_label]
    cluster_points[cluster_label] = points_in_cluster
cluster_points
```

Out[64]:

```
{0: array([[ 6, 550],
           [ 5, 553],
           [ 8, 450],
           [ 7, 404],
           [ 5, 388],
           [ 8, 350],
           [ 9, 400]]),
 1: array([[ 4, 220],
           [ 4, 150],
           [ 5, 120],
           [ 7, 100],
           [ 5, 200],
           [ 6, 120],
           [ 6, 225],
           [ 8, 236],
           [ 4, 167]]),
 2: array([[ 9, 699],
           [ 9, 817],
           [10, 1010],
           [11, 825],
           [11, 700],
           [ 8, 921]]),
 3: array([[ 2, 10],
           [ 9,  0],
           [ 6,  5],
           [11,  0],
           [12,  0],
           [ 8,  0],
           [ 7, 25],
           [10, 20]])}
```

In [65]:

```

{0: array([[ 6, 550],
          [ 5, 553],
          [ 8, 450],
          [ 7, 404],
          [ 5, 388],
          [ 8, 350],
          [ 9, 400]]), 1: array([[ 4, 220],
          [ 4, 150],
          [ 5, 120],
          [ 7, 100],
          [ 5, 200],
          [ 6, 120],
          [ 6, 225],
          [ 8, 236],
          [ 4, 167]]), 2: array([[ 9, 699],
          [ 9, 817],
          [10, 1010],
          [11, 825],
          [11, 700],
          [ 8, 921]]), 3: array([[ 2, 10],
          [ 9, 0],
          [ 6, 5],
          [11, 0],
          [12, 0],
          [ 8, 0],
          [ 7, 25],
          [10, 20]])}

```

In [82]:

```

import matplotlib.patches as mpatch
fig,ax = plt.subplots()
centeriods = model.cluster_centers_
colors = ['black','green','red','blue'];
clr = 0;
i = 0;
for cluster_labels,points in cluster_points.items():
    print(cluster_labels,":")
    ax.scatter(centeriods[i][0],centeriods[i][1],color=colors[clr],marker="*")
    i=i+1
    for point in points:
        ax.scatter(point[0],point[1],)
    clr = clr+1;
classes = ['window shoper','basic','moderate','prime']
handles = [
    mpatch.Patch(color=colors[0], label=classes[0]),
    mpatch.Patch(color=colors[1], label=classes[1]),
    mpatch.Patch(color=colors[2], label=classes[2]),
    mpatch.Patch(color=colors[3], label=classes[3])
]

```

```
]

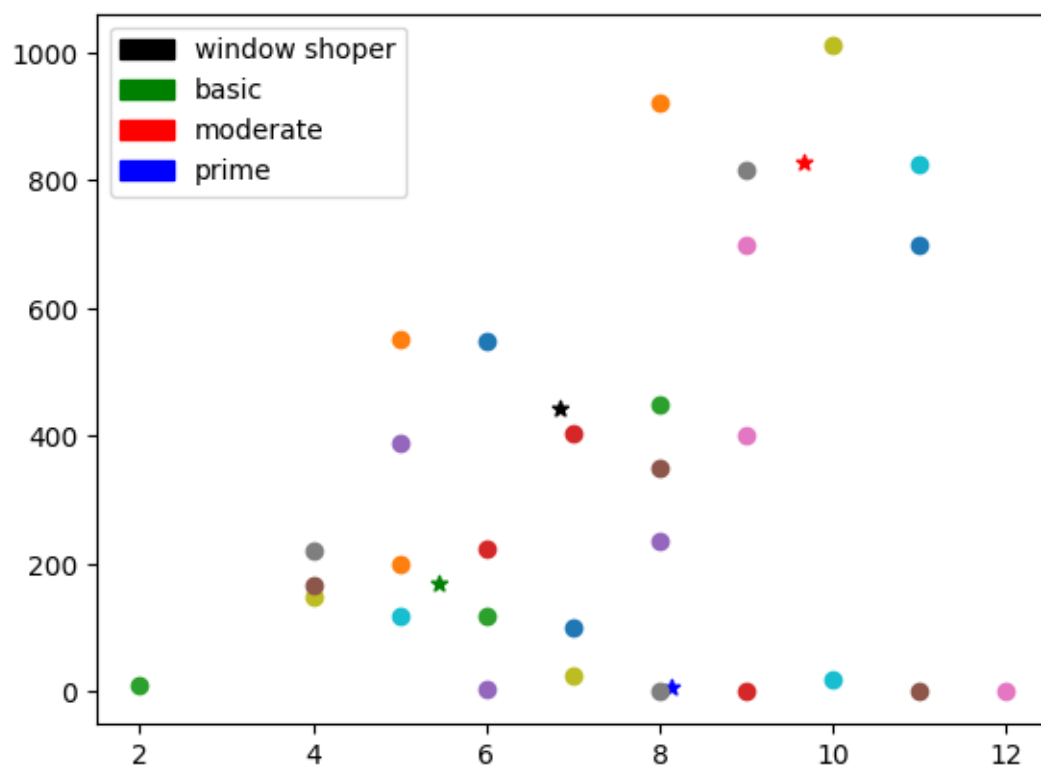
```

```
plt.legend(handles=handles)
plt.show()

```

```
0 :
1 :
2 :
3 :

```



```
In [83]:

```

```
model.inertia_

```

```
Out[83]:

```

```
135454.367063492

```

```
In [ ]:

```