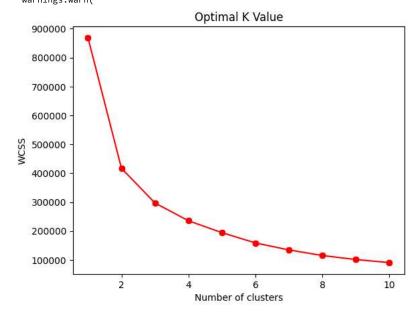
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from google.colab import files
uploaded = files.upload()
     Choose Files dataset.csv
     • dataset.csv(text/csv) - 2429 bytes, last modified: 4/20/2023 - 100% done
     Saving dataset.csv to dataset.csv
dataset = pd.read_csv('dataset.csv')
print(dataset.shape)
print(dataset.describe())
print(dataset.head(5))
 [→ (303, 2)
                INCOME
                              SPEND
     count 303.000000 303.000000
     mean 245.273927 149.646865
            48.499412 22.905161
126.000000 71.000000
     std
     25% 211.000000 133.500000
          240.000000 153.000000
274.000000 166.000000
     50%
     75%
          417.000000 202.000000
     max
       INCOME SPEND
     0
           233
                 150
           250
                 187
                172
     2
           204
     3
           236
                  178
     4
           354
                 163
Income = dataset['INCOME'].values
Spend = dataset['SPEND'].values
X = np.array(list(zip(Income, Spend)))
```



```
[20/, 130],
[311, 120],
[204, 162],
[232, 164],
[335, 143],
[205, 130],
[203, 161],
[318, 140],
[225, 146],
[212, 150],
[169, 144],
[187, 144],
[197, 136],
[176, 90],
[241, 123],
[264, 132],
[193, 141],
[131, 115],
[236, 174]])
```

```
from sklearn.cluster import KMeans
wcss = []
for i in range(1,11):
    km=KMeans(n_clusters=i, random_state=0)
    km.fit(X)
    wcss.append(km.inertia_)
plt.plot(range(1,11),wcss,color="red", marker ="8")
plt.title('Optimal K Value')
plt.xlabel('Number of clusters')
plt.ylabel('WCSS')
plt.show()
```

```
/usr/local/lib/python3.9/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: Th
     warnings.warn(
/usr/local/lib/python3.9/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: Th
     warnings.warn(
/usr/local/lib/python3.9/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: Th
     warnings.warn(
/usr/local/lib/python3.9/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: Th
     warnings.warn(
/usr/local/lib/python3.9/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The control of the contro
     warnings.warn(
/usr/local/lib/python3.9/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: Th
     warnings.warn(
```



```
*Alt+Q
```

```
model=KMeans(n_clusters=4, random_state=0)
y_means = model.fit_predict(X)

/usr/local/lib/python3.9/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarning: The default value of `n_init` will change from 10
    warnings.warn(
```

plt.scatter(X[y_means==0,0],X[y_means==0,1],s=50, c='brown',label='1')
plt.scatter(X[y_means==1,0],X[y_means==1,1],s=50, c='blue',label='2')
plt.scatter(X[y_means==2,0],X[y_means==2,1],s=50, c='green',label='3')
plt.scatter(X[v_means==3.0],X[v_means==3.1],s=50, c='cvan',label='4')

plt.scatter(X[y_means==2,0],X[y_means==2,1],s=50, c= green ,label= 5)
plt.scatter(X[y_means==3,0],X[y_means==3,1],s=50, c='cyan',label='4')
plt.scatter(model.cluster_centers_[:,0], model.cluster_centers_[:,1],s=100,marker='s', c='red', label='Centroids')
plt.title('Income Spent Analysis')
plt.xlabel('Income')

plt.ylabel('Spent')
plt.legend()

plt.show()

