M Abdullah 19F-916 Zainab Ijaz 19F-0917

SE (7A)

Applied Machine Learning

Assignment 2

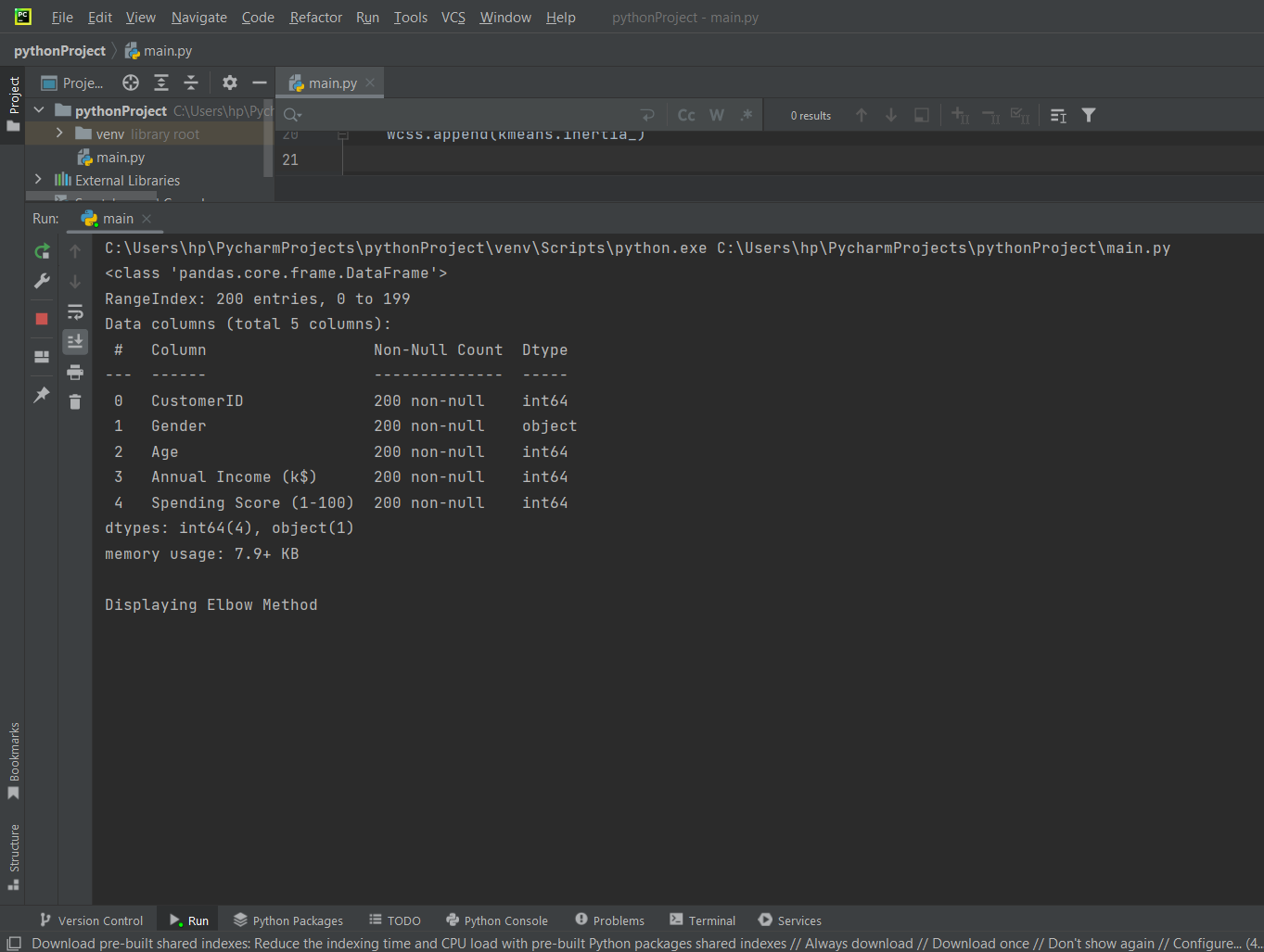
**Question # 1**

**Solution**

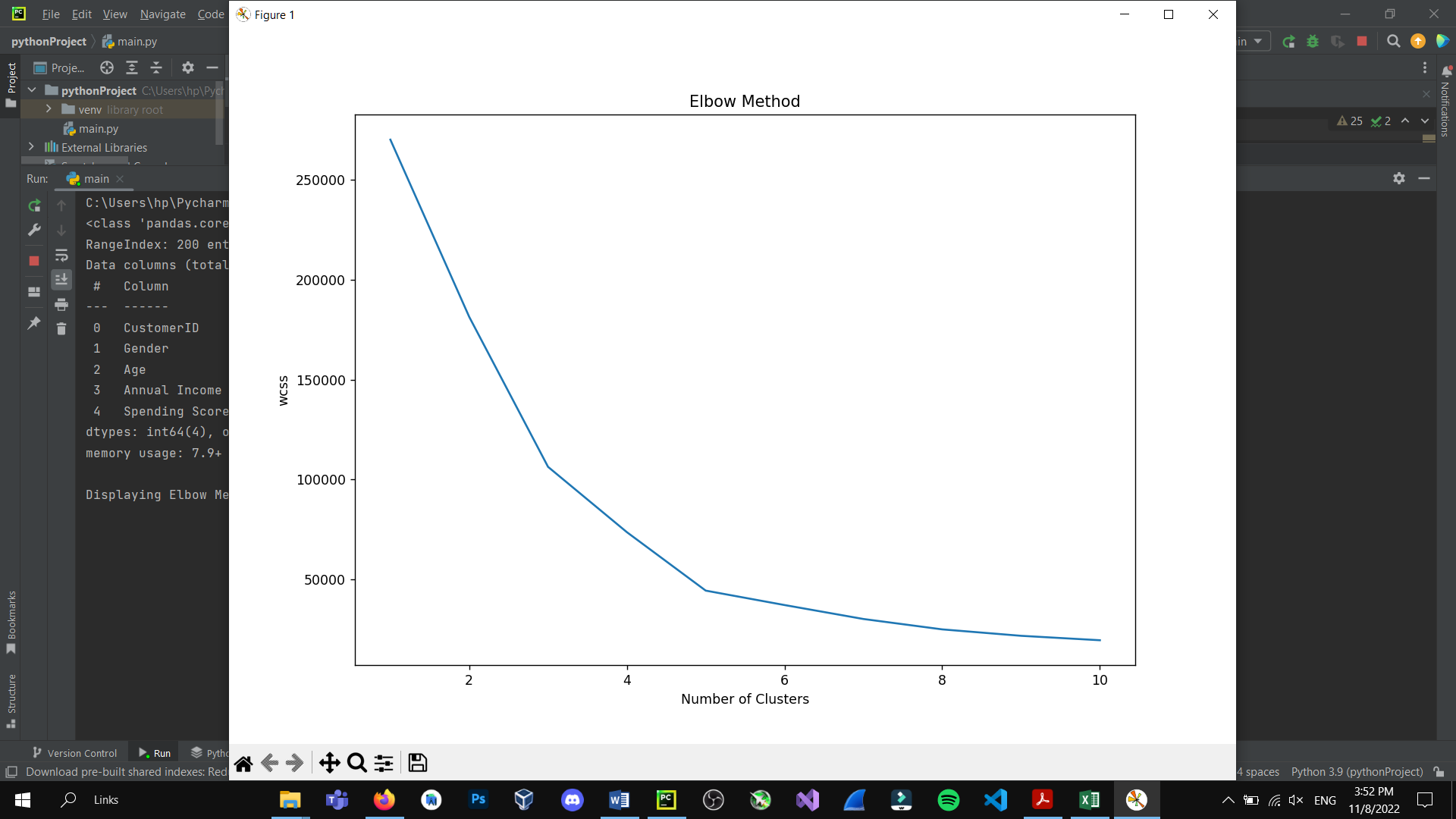
**Python Code:**

import pandas as pd  
import matplotlib.pyplot as plt  
from sklearn.cluster import KMeans  
  
dataset = pd.read\_csv('C:/Users/hp/Desktop/ML Assignment 2/data.csv')  
temp = dataset.copy()  
temp.head()  
  
temp.isnull().sum()  
temp.describe()  
temp.info()  
  
X = temp.iloc[:, [3, 4]].values  
temp = pd.get\_dummies(temp, columns = ['Gender'], prefix = ['Gender'])  
  
wcss = []  
for i in range(1, 11):  
 kmeans = KMeans(n\_clusters = i, init = 'k-means++', max\_iter = 300, n\_init = 10)  
 kmeans.fit(X)  
 wcss.append(kmeans.inertia\_)  
  
def plotElbow():  
 plt.plot(range(1, 11), wcss)  
 plt.title('Elbow Method')  
 plt.xlabel('Number of Clusters')  
 plt.ylabel('wcss')  
 plt.show()  
  
  
kmeans = KMeans(n\_clusters = 5, init = 'k-means++', max\_iter = 300, n\_init = 10)  
y\_kmeans = kmeans.fit\_predict(X)  
  
def plotClusters():  
 plt.scatter(X[y\_kmeans == 0, 0], X[y\_kmeans == 0, 1], s=100, c='black', label='1st Cluster')  
 plt.scatter(X[y\_kmeans == 1, 0], X[y\_kmeans == 1, 1], s=100, c='red', label='2nd Cluster')  
 plt.scatter(X[y\_kmeans == 2, 0], X[y\_kmeans == 2, 1], s=100, c='blue', label='3rd Cluster')  
 plt.scatter(X[y\_kmeans == 3, 0], X[y\_kmeans == 3, 1], s=100, c='green', label='4th Cluster')  
 plt.scatter(X[y\_kmeans == 4, 0], X[y\_kmeans == 4, 1], s=100, c='cyan', label='5th CLuster')  
 plt.title('K Mean Clusters Formation')  
 plt.xlabel('Annual Income')  
 plt.ylabel('Spending Score')  
 plt.legend()  
 plt.show()  
  
  
if \_\_name\_\_ == "\_\_main\_\_":  
 print("\nDisplaying Elbow Method")  
 plotElbow()  
 print("\nDisplaying K Mean Clusters")  
 plotClusters()

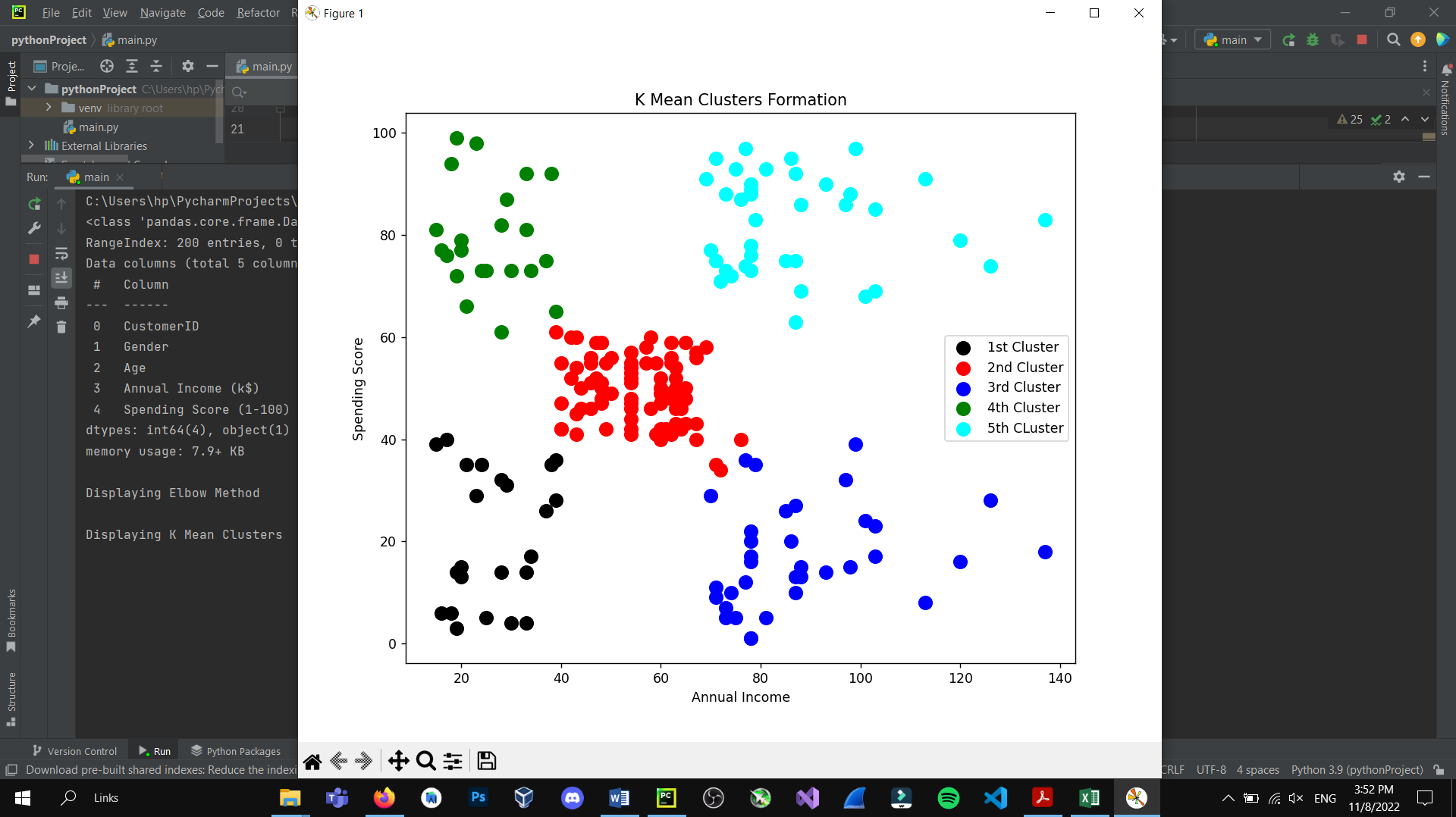
**Output:**



**Using Elbow Method:**



**Using K-Means Clustering:**



**Question # 2**

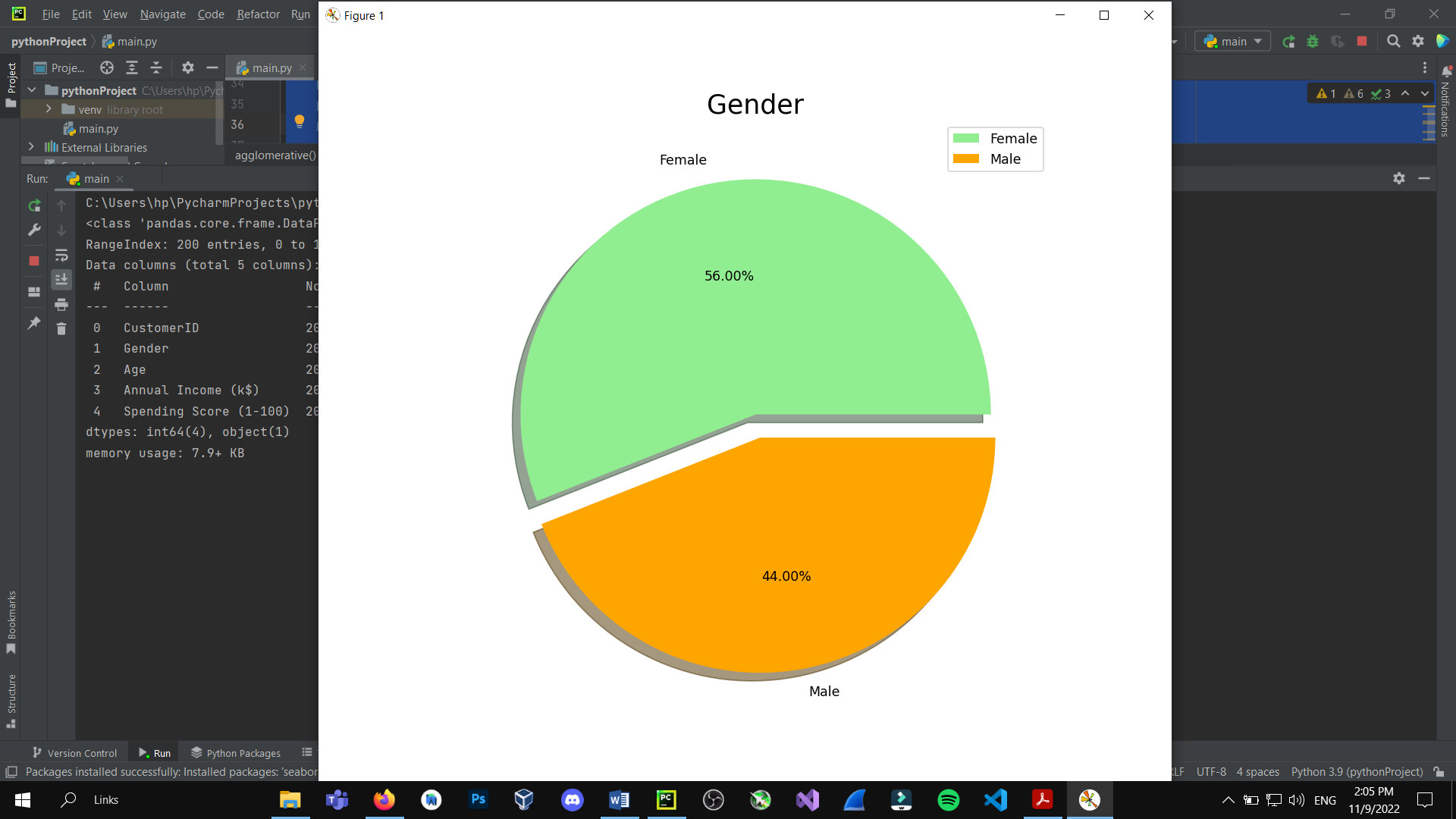
**Solution**

**Python Code for Agglomerative Clustering:**

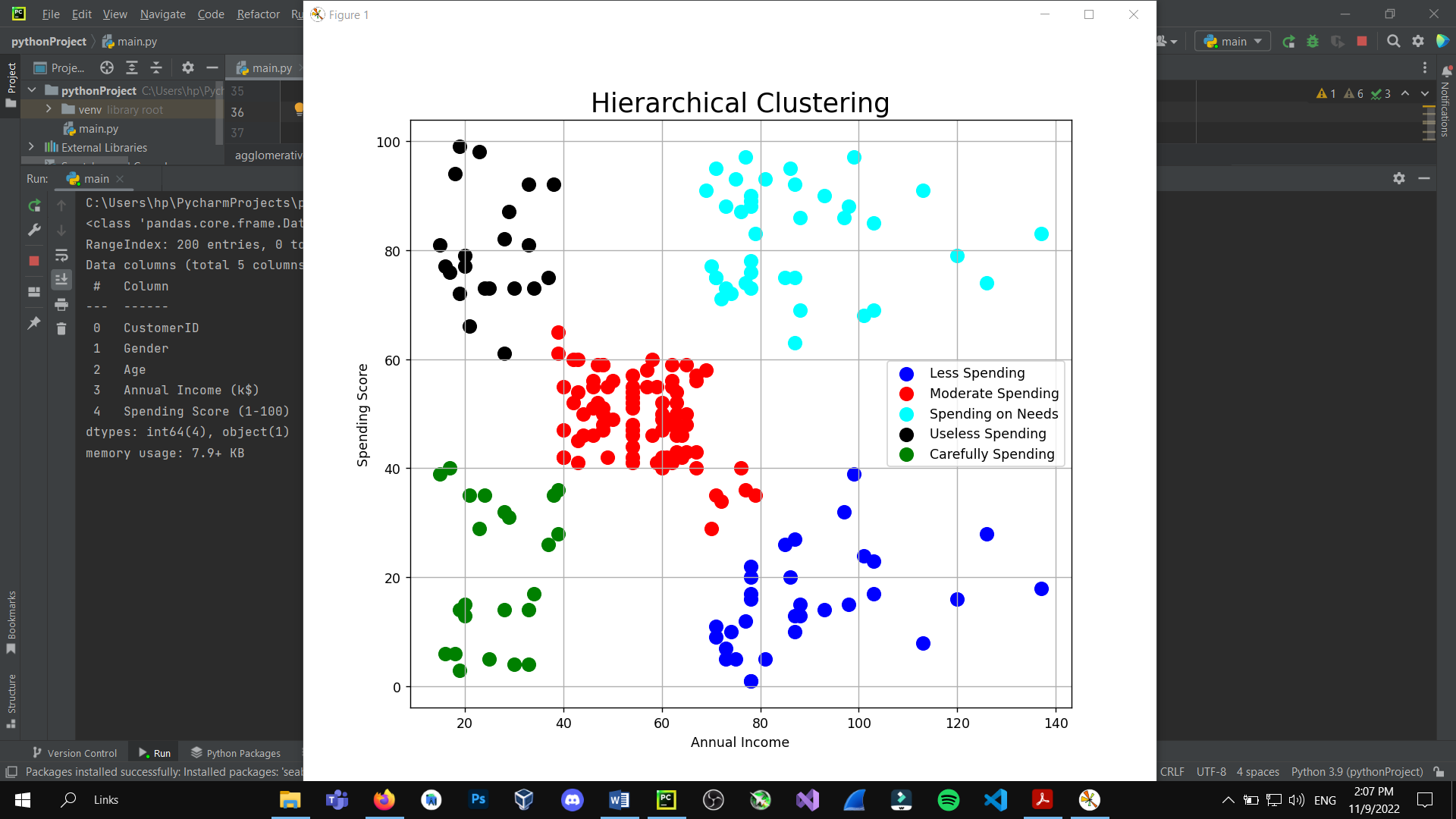
import numpy as np  
import pandas as pd  
import matplotlib.pyplot as plt  
import scipy.cluster.hierarchy as sch  
  
dataset = pd.read\_csv('C:/Users/hp/Desktop/ML Assignment 2/data.csv')  
temp = dataset.copy()  
temp.head()  
  
temp.isnull().sum()  
temp.describe()  
temp.info()  
x = temp.iloc[:, [3, 4]].values  
  
def ratio():  
 labels = ['Female', 'Male']  
 size = temp['Gender'].value\_counts()  
 colors = ['lightgreen', 'orange']  
 explode = [0, 0.1]  
 plt.rcParams['figure.figsize'] = (9, 9)  
 plt.pie(size, colors=colors, explode=explode, labels=labels, shadow=True, autopct='%.2f%%')  
 plt.title('Gender', fontsize=20)  
 plt.axis('off')  
 plt.legend()  
 plt.show()  
  
def agglomerative():  
 km = KM(n\_clusters=5, init='k-means++', max\_iter=300, n\_init=10, random\_state=0)  
 hc = AgglomerativeClustering(n\_clusters=5, affinity='euclidean',linkage='ward')  
 y\_hc = hc.fit\_predict(x)  
 plt.scatter(x[y\_hc == 0, 0], x[y\_hc == 0, 1], s=100, c='pink', label='Less Spending')  
 plt.scatter(x[y\_hc == 1, 0], x[y\_hc == 1, 1], s=100, c='yellow', label='Moderate Spending')  
 plt.scatter(x[y\_hc == 2, 0], x[y\_hc == 2, 1], s=100, c='cyan', label='Spending on Needs')  
 plt.scatter(x[y\_hc == 3, 0], x[y\_hc == 3, 1], s=100, c='magenta', label='Useless Spending')  
 plt.scatter(x[y\_hc == 4, 0], x[y\_hc == 4, 1], s=100, c='orange', label='Carefully Spending')  
 plt.title('Hierarchical Clustering', fontsize=20)  
 plt.xlabel('Annual Income')  
 plt.ylabel('Spending Score')  
 plt.legend()  
 plt.grid()  
 plt.show()  
  
def dendrogram():  
  
 sch.dendrogram(sch.linkage(x, method='ward'))  
 plt.title('Dendrogram', fontsize=20)  
 plt.xlabel('Customers')  
 plt.ylabel('Euclidean Distance')  
 plt.show()  
  
  
if \_\_name\_\_ == "\_\_main\_\_":  
 ratio()  
 dendrogram()  
 agglomerative()

**Output:**

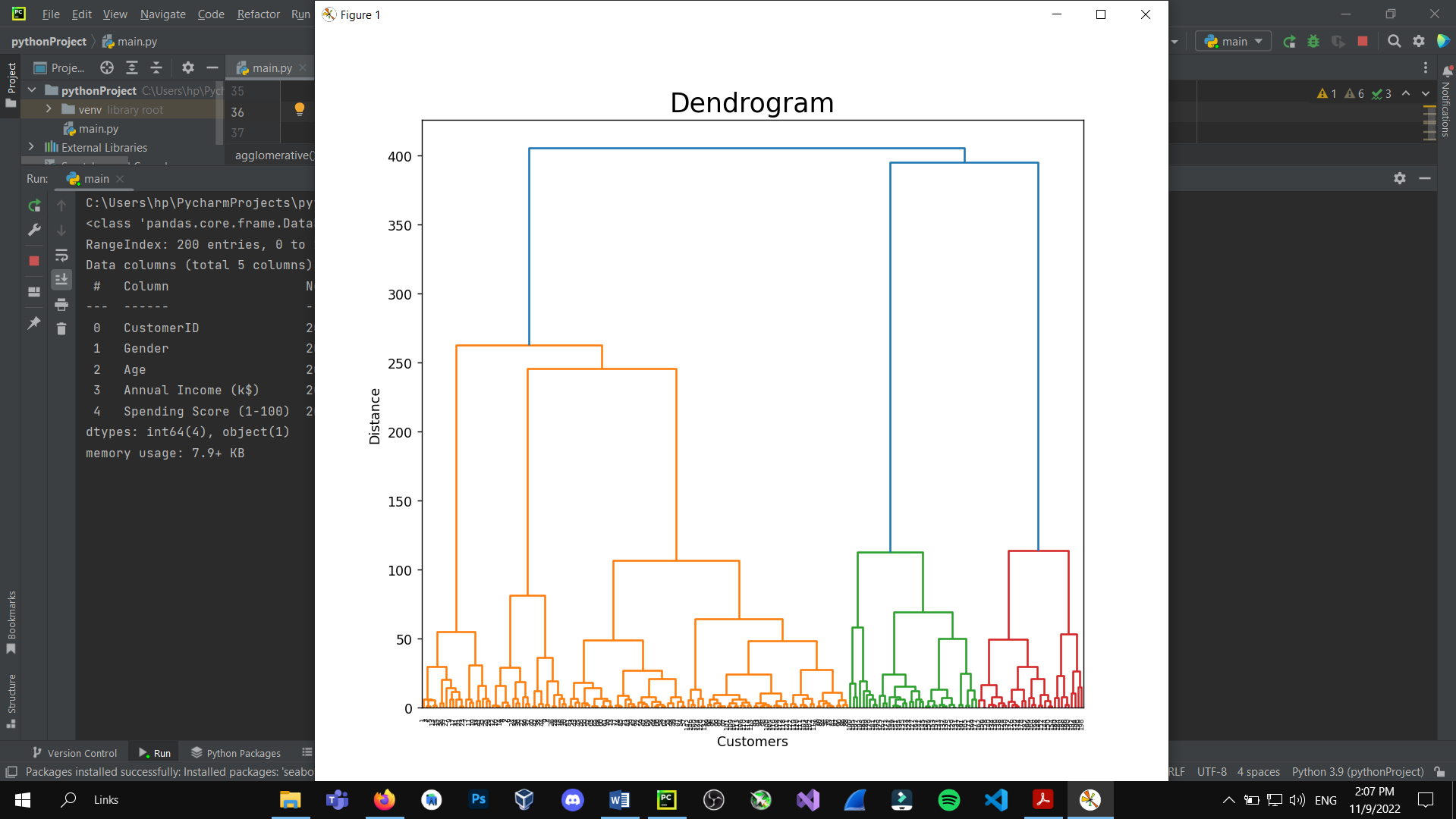
**Spending Ratio between Male and Female**



**Hierarchical Clustering:**



**Dendrogram:**

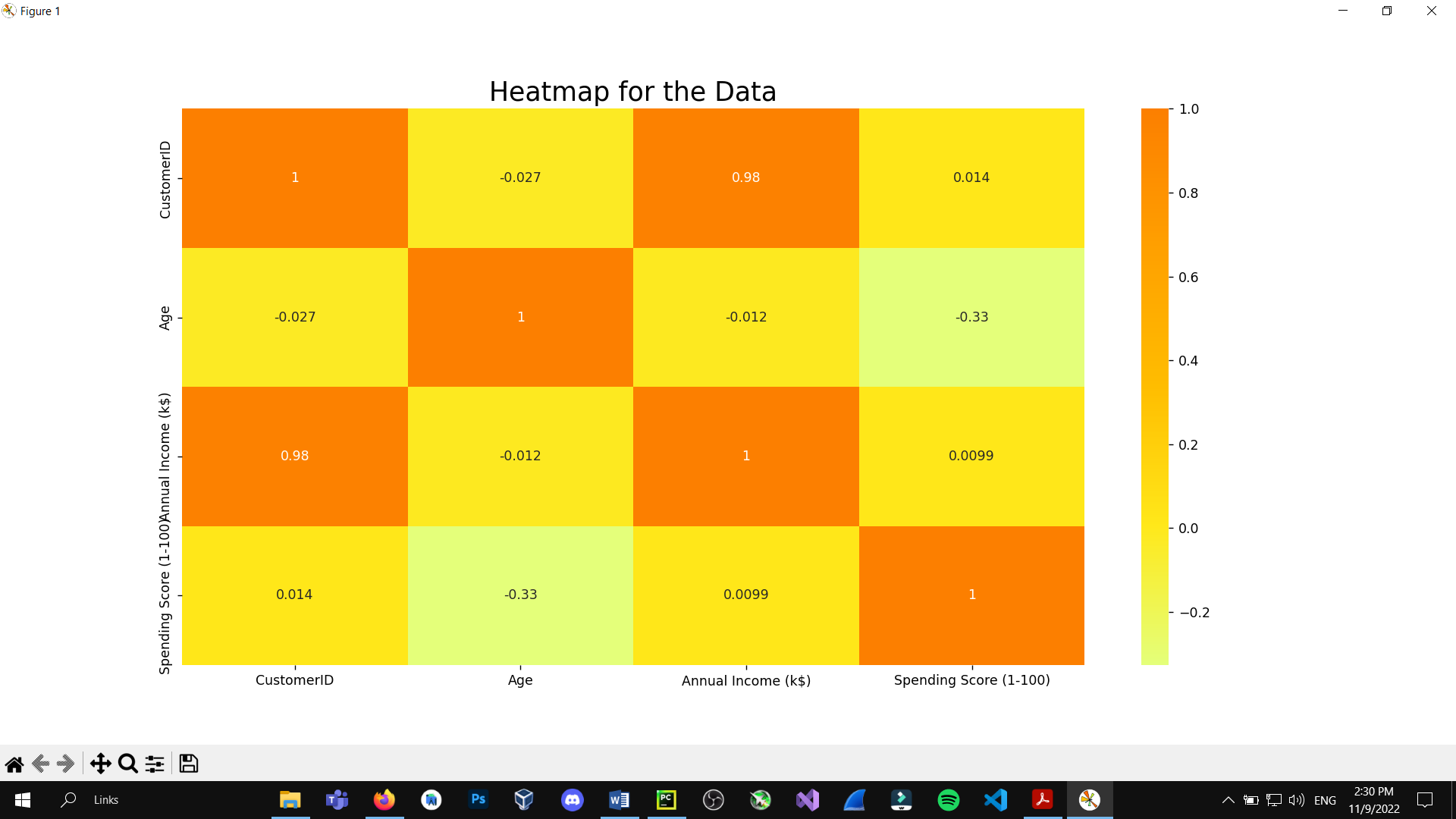


**Python Code for K-Medoid Clustering:**

import numpy as np  
import pandas as pd  
import matplotlib.pyplot as plt  
import seaborn as sns  
  
dataset = pd.read\_csv('C:/Users/hp/Desktop/ML Assignment 2/data.csv')  
temp = dataset.copy()  
temp.head()  
  
temp.isnull().sum()  
temp.describe()  
temp.info()  
x = temp.iloc[:, [3, 4]].values  
  
def ratio():  
 labels = ['Female', 'Male']  
 size = temp['Gender'].value\_counts()  
 colors = ['lightgreen', 'orange']  
 explode = [0, 0.1]  
 plt.rcParams['figure.figsize'] = (9, 9)  
 plt.pie(size, colors=colors, explode=explode, labels=labels, shadow=True, autopct='%.2f%%')  
 plt.title('Gender', fontsize=20)  
 plt.axis('off')  
 plt.legend()  
 plt.show()  
  
def kMedoids():  
  
 wcss = []  
 for i in range(1, 11):km = KMedoids(n\_clusters = i, init = 'k-means++', max\_iter = 300, n\_init = 10, random\_state = 0)  
 km.fit(x)  
 wcss.append(km.inertia\_)  
 km = KMedoids n\_clusters=5, init='k-means++', max\_iter=300, n\_init=10, random\_state=0)  
 y\_means = km.fit\_predict(x)  
 plt.scatter(x[y\_means == 0, 0], x[y\_means == 0, 1], s=100, c='pink', label='Less Spending')  
 plt.scatter(x[y\_means == 1, 0], x[y\_means == 1, 1], s=100, c='yellow', label='Moderate Spending')  
 plt.scatter(x[y\_means == 2, 0], x[y\_means == 2, 1], s=100, c='cyan', label='Spending on Needs')  
 plt.scatter(x[y\_means == 3, 0], x[y\_means == 3, 1], s=100, c='magenta', label='Useless Spending')  
 plt.scatter(x[y\_means == 4, 0], x[y\_means == 4, 1], s=100, c='orange', label='Carefully Spending')  
 plt.scatter(km.cluster\_centers\_[:, 0], km.cluster\_centers\_[:, 1], s=50, c='black', label='Clusters Center')  
 plt.title('K Medoids Clustering', fontsize=20)  
 plt.xlabel('Annual Income')  
 plt.ylabel('Spending Score')  
 plt.legend()  
 plt.grid()  
 plt.show()  
  
def heatMap():  
 plt.rcParams['figure.figsize'] = (15, 8)  
 sns.heatmap(temp.corr(), cmap = 'Wistia', annot = True)  
 plt.title('Heatmap for the Data', fontsize = 20)  
 plt.show()  
  
if \_\_name\_\_ == "\_\_main\_\_":  
 ratio()  
 heatMap()  
 kMedoids()

**Output:**

**HeatMap for the Dataset**



**K-Medoids Clustering**

