Code:

```
# Importing the libraries
from pandas import read_csv as rc
from matplotlib import pyplot as plt
import numpy as np
from sklearn.cluster import KMeans

# Importing the dataset
data_set = rc(r'C:\Users\shrey\OneDrive\Desktop\Exposys Data Science Project\customer-segmentation-dataset\Mall_Customers - Usable.csv')
data_set.head(15)

# Converting the gender values to 1(Male) and 0(Female)
gender = {'Male' : 1, 'Female': 0}
data_set.Gender = [gender[item] for item in data_set.Gender]
data_set.head(15)

# Since customer ID is unusable for us we'll remove it
data_set = data_set.iloc[:, 1:]
data_set.head(15)

# Describing some common features of our dataset
data_set.describe()

# Visualization of Gender and Age
# Filtering out Gender and Age columns into gender_age variable
gender_age = data_set.iloc[:,0:2]
gender_age.head(10)

# Describing some common features of Gender and Age
gender_age.describe()

# Histogram distribution of age
gender_age.iloc[:,1].plot.hist(alpha= 0.5, stacked = True, bins = 25)

# Histogram distribution of Gender
gender_age.iloc[:,0].plot.hist(alpha= 0.5, stacked = True, bins = 100)

# Describing some common features of Gender 'Male'
male_gender = gender_age[gender_age['Gender'] == 1]
male_gender.describe()

# Describing some common features of Gender 'Female'
female_gender = gender_age[gender_age['Gender'] == 0]
female_gender.describe()

# Histogram distribution of Age with Male as a gender
male_gender.hist(column = "Age", bins = 25)
```

```
# Histogram distribution of Age with Female as a gender
female_gender.hist(column = "Age", bins = 25)

# Hexbin of Gender and Age
data_set.plot(kind = 'hexbin',x='Gender', y = 'Age',gridsize = 20,sharex=False)

# Scatterplot of Gender and Age
plt.scatter(data_set.iloc[:, 0], data_set.iloc[:, 1], c = 'white', marker = 'o', edgecolor='black',s=50)
plt.xlabel('Gender')
plt.ylabel('Age')
plt.title("Gender Age Scatterplot")
plt.show()

# Scatterplot of Annual Income(x - axis) and Spending Score(y-axis)
plt.scatter(data_set.iloc[:, 2], data_set.iloc[:, 3], c = 'white', marker = 'o', edgecolor='black',s=50)
plt.xlabel('Annual Income')
plt.ylabel('Spending Score')
plt.title('Annual Income Spending Score Scatterplot')
plt.show()

# Choosing the optimal number of clusters

# Checking the values of Inertia and Distortion using Elbow method
```

```
from scipy.spatial.distance import cdist
distortions = []
inertias = []
mapping1 = {}
mapping2 = {}
K = range(1,10)

for k in K:
    #Building and fitting the model
    kmeanModel = KMeans(n_clusters=k).fit(data_set)
    kmeanModel.fit(data_set)

    distortions.append(sum(np.min(cdist(data_set,
kmeanModel.cluster_centers_,
                      'euclidean'),axis=1)) / data_set.shape[0])
    inertias.append(kmeanModel.inertia_)

    mapping1[k] = sum(np.min(cdist(data_set,
kmeanModel.cluster_centers_,
                 'euclidean'),axis=1)) / data_set.shape[0]
    mapping2[k] = kmeanModel.inertia_

for key,val in mapping1.items():
    print(str(key)+' : '+str(val))
```

```
plt.plot(K, distortions, 'bx-')
plt.xlabel('Values of K')
plt.ylabel('Distortion')
plt.title('The Elbow Method using Distortion')
plt.show()

for key,val in mapping2.items():
    print(str(key)+' : '+str(val))

plt.plot(K, inertias, 'bx-')
plt.xlabel('Values of K')
plt.ylabel('Inertia')
plt.title('The Elbow Method using Inertia')
plt.show()



# Applying KMeans clustering algorithm to the dataset
kmeans = KMeans(n_clusters=5,
init='random',n_init=10,max_iter=300,tol=1e-04, random_state=0)
clustered_data = kmeans.fit_predict(data_set)
print(clustered_data)

# Concatenating this cluster label to the original data_set
from pandas import DataFrame as df
from pandas import concat as cc
```

```
clusters = df(kmeans.labels_)
data_set = cc((data_set, clusters), axis = 1)
data_set = data_set.rename({0:'Clusters'}, axis = 1)
data_set.head()

# Importing the seaborn library to visualize the graphs
import seaborn as sns

# Visualizing the Gender vs Age clusters
sns.lmplot(x = "Gender", y = "Age", data = data_set,
hue='Clusters',fit_reg=False)

# Visualizing the Annual Income vs Spending Scores clusters
sns.lmplot(x='Annual Income (k$)', y ='Spending Score (1-
100)',data=data_set,hue='Clusters',fit_reg=False)

# Plotting pairplot to get pairwise relationships in a dataset
sns.pairplot(data_set,hue='Clusters')
```