```
import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)

import os
for dirname, _, filenames in os.walk('/kaggle/input'):
    for filename in filenames:
        print(os.path.join(dirname, filename))
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
df=pd.read_csv('Mall_Customers.csv')

df.rename(columns={'Genre':'Gender'},inplace=True)
df.head()

df.describe()

df.isnull().sum()

df.drop(['CustomerID'],axis=1,inplace=True)

plt.figure(1,figsize=(15,6))
n = 0
for x in ['Age','Annual Income (k$)','Spending Score (1-100)']:
    n +=1
    plt.subplot(1,3,n)
    plt.subplots_adjust(hspace=0.5,wspace=0.5)
    sns.distplot(df[x],bins=20)
    plt.title('Distplot of {}'.format(x))
plt.show()

plt.figure(figsize=(15,5))
sns.countplot(y='Gender',data=df)
plt.show()

plt.figure(1,figsize=(15,6))
n = 0
for cols in ['Age','Annual Income (k$)','Spending Score (1-100)']:
    n +=1
    plt.subplot(1,3,n)
    sns.set(style="whitegrid")
    plt.subplots_adjust(hspace=0.5,wspace=0.5)
    sns.violinplot(x = cols,y = 'Gender',data=df)
    plt.ylabel('Gender' if n== 1 else '')
    plt.title('Violin Plot')
plt.show()
```

```
age_18_25 = df.Age[(df.Age >=18) & (df.Age <= 25)]
age_26_35 = df.Age[(df.Age >=26) & (df.Age <= 35)]
age_36_45 = df.Age[(df.Age >=36) & (df.Age <= 45)]
age_46_55 = df.Age[(df.Age >=46) & (df.Age <= 55)]
age_55_above = df.Age[(df.Age >= 56)]

age_x =["18-25","26-35","36-45","46-55","55+"]
age_y = [len(age_18_25.values),len(age_26_35.values),len(age_36_45),len
(age_46_55),len(age_55_above)]

plt.figure(figsize = (15,6))
sns.barplot(x=age_x, y=age_y,palette = "mako")
plt.title("Number of Customer and Ages")
plt.xlabel("Age")
plt.ylabel("Number of Customer")
plt.show()
```

```
sns.relplot(x="Annual Income (k$)",y = "Spending Score (1-100)",data=df
)
```

```
ss_1_20 = df["Spending Score (1-100)"][(df["Spending Score (1-100)"] >=
1) & (df["Spending Score (1-100)"] <= 20)]
ss_21_40 = df["Spending Score (1-100)"][(df["Spending Score (1-100)"] >
= 21) & (df["Spending Score (1-100)"] <= 40)]
ss_41_60 = df["Spending Score (1-100)"][(df["Spending Score (1-100)"] >
= 41) & (df["Spending Score (1-100)"] <= 60)]
ss_61_80 = df["Spending Score (1-100)"][(df["Spending Score (1-100)"] >
= 61) & (df["Spending Score (1-100)"] <= 80)]
ss_81_100 = df["Spending Score (1-100)"][(df["Spending Score (1-100)"]
>= 81) & (df["Spending Score (1-100)"] <= 100)]

ssx= ["1-20","21-40","41-60","61-80","81-100"]
ssy=[len(ss_1_20.values),len(ss_21_40.values),len(ss_41_60.values),len(
ss_61_80.values),len(ss_81_100.values)]

plt.figure(figsize=(15,6))
sns.barplot(x=ssx,y=ssy, palette="rocket")
plt.title("Spending Scores")
plt.xlabel("Score")
plt.ylabel("Number of Customer having the Score")
plt.show()
```

```
ai_0_30 = df["Annual Income (k$)"][(df["Annual Income (k$)"] >= 0) & (d
f["Annual Income (k$)"] <= 30)]
ai_31_60= df["Annual Income (k$)"][(df["Annual Income (k$)"] >=31)& (df
["Annual Income (k$)"] <=60)]
ai_61_90= df["Annual Income (k$)"][(df["Annual Income (k$)"] >=61)& (df
["Annual Income (k$)"] <=90)]
```

```
ai_61_90=df["Annual Income (k$)"][(df["Annual Income (k$)"] >=91)& (df[
"Annual Income (k$)"] <=120)]
ai_121_150 = df["Annual Income (k$)"][(df["Annual Income (k$)"]>=121) &
(df["Annual Income (k$)"] <=150)]

aix = ["$ 0 - 30,000","$ 30,001 - 60,000","$ 60,001 - 90,000","$ 90,001
- 120,000","$ 120,001 - 150,000"]
aiy = [len(ai_0_30.values),len(ai_31_60.values),len(ai_61_90.values),le
n(ai_61_90.values),len(ai_121_150.values)]

plt.figure(figsize=(15,6))
sns.barplot(x=aix,y=aiy,palette="Spectral")
plt.title("Annual Incomes")
plt.xlabel("Income")
plt.ylabel("Numer of Customer")
plt.show()
```

```
kmeans = KMeans(n_clusters=4)

label = kmeans.fit_predict(X1)

print(label)
```

```
print(kmeans.cluster_centers_)
```

```
plt.scatter(X1[:,0],X1[:,1],c=kmeans.labels_,cmap='rainbow')
plt.scatter(kmeans.cluster_centers_[:,0],kmeans.cluster_centers_[:,1],c
olor='black')
plt.title('Clusters of Customers')
plt.xlabel('Age')
plt.ylabel('Spending Score(1-100)')
plt.show
```

```
kmeans = KMeans(n_clusters=5)

label = kmeans.fit_predict(X2)

print(label)
```

```
print(kmeans.cluster_centers_)
```

```
plt.scatter(X2[:,0],X1[:,1],c=kmeans.labels_,cmap='rainbow')
plt.scatter(kmeans.cluster_centers_[:,0],kmeans.cluster_centers_[:,1],c
olor='black')
plt.title('Clusters of Customers')
plt.xlabel('Annual Income (k$)')
```

```
plt.ylabel('Spending Score(1-100)')
plt.show
```

Done by

Muhammed Midlaj R (72092144033)

Aswin K (72092144008)

Mohammad Safar R (72092144032)

Athul Krishnan VR (72092144009)