Feynn Labs

PROJECT REPORT ON

"CUSTOMER SEGMENTATION"

DATA SCIENCE PROJECT

Submitted by – Ashutosh Mishra

ACKNOWLEDGEMENTS

We, take this opportunity to thank one and all involved in helping us build this project. Firstly, we would like to thank the college for providing us an opportunity to work on this project.

We thank the management of **Feynn Labs** for providing all the resources required for the project.

We wish to acknowledge my sincere gratitude to their faculties for their constant encouragement and for providing us with all the facilities required for the accomplishment of this project.

We also, are highly grateful to the guidance offered By staff members who have been very generous in assisting and supporting, to do this project named **"CUSTOMER SEGMENTATION"**, which formally started as just a rough idea and now has resulted in the form of this project.

We also would like to thank all the other teaching and non-teaching staff members who had extended their hand for support and co-operation while bringing up this project.

ASHUTOSH MISHRA (1JS19CS035)

ABSTRACT

Customer segmentation is an effective tool for businesses to closely align their strategy and tactics with, and better target, their customers. Every customer is different and every customer journey is different so a single approach often isn't going to work for all. This is where customer segmentation becomes a valuable process. Let's begin with understanding exactly what customer segmentation is

Customer segmentation is the process by which you divide your customers up based on common characteristics – such as demographics or behaviours, so you can market to those customers more effectively.

These customer segmentation groups can also be used to begin discussions of building a marketing persona. This is because customer segmentation is typically used to inform a brand's messaging, positioning and to improve how a business sells – so marketing personas need to be closely aligned to those customer segments in order to be effective.

The marketing "persona" is by definition a personification of a customer segment, and it is not uncommon for businesses to create several personas to match their different customer segments.

But for that to happen, a business needs a robust set of customer segments off of which to base it. Which leads us to the next section, distinguishing the difference between customer segmentation and market segmentation, so that your segmentation is as accurate as possible.

CODE SNIPPET

(976) Customer Segmentation Tu | Downloads/customer-segmentat | customer segmentation - Jupyter | (10) WhatsApp

localhost:8888/notebooks/customer%20segmentation.ipynb

jupyter **customer segmentation** Last Checkpoint: 16 hours ago (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help

Trusted | Python 3 (ipykernel)

Run | Code

In [ ]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [4]:
```
df = pd.read_csv("Mall_Customers.csv")
```

In [5]:
```
df.head()
```

Out[5]:

	CustomerID	Gender	Age	Annual Income (k$)	Spending Score (1-100)
0	1	Male	19	15	39
1	2	Male	21	15	81
2	3	Female	20	16	6
3	4	Female	23	16	77
4	5	Female	31	17	40

In [6]:
```
df.columns
```

Out[6]:
```
Index(['CustomerID', 'Gender', 'Age', 'Annual Income (k$)',
       'Spending Score (1-100)'],
      dtype='object')
```

In [7]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 200 entries, 0 to 199
```

(976) Customer Segmentation Tu | Downloads/customer-segmentat | customer segmentation - Jupyter | (10) WhatsApp

localhost:8888/notebooks/customer%20segmentation.ipynb

jupyter **customer segmentation** Last Checkpoint: 16 hours ago (autosaved)

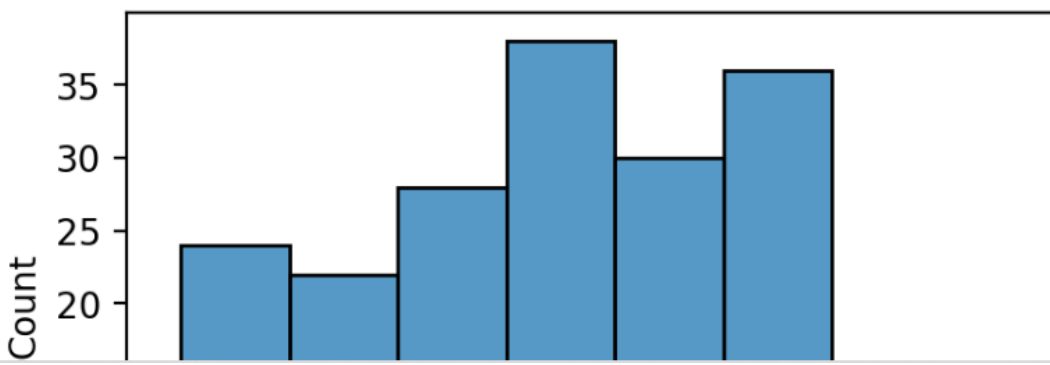
File Edit View Insert Cell Kernel Widgets Help

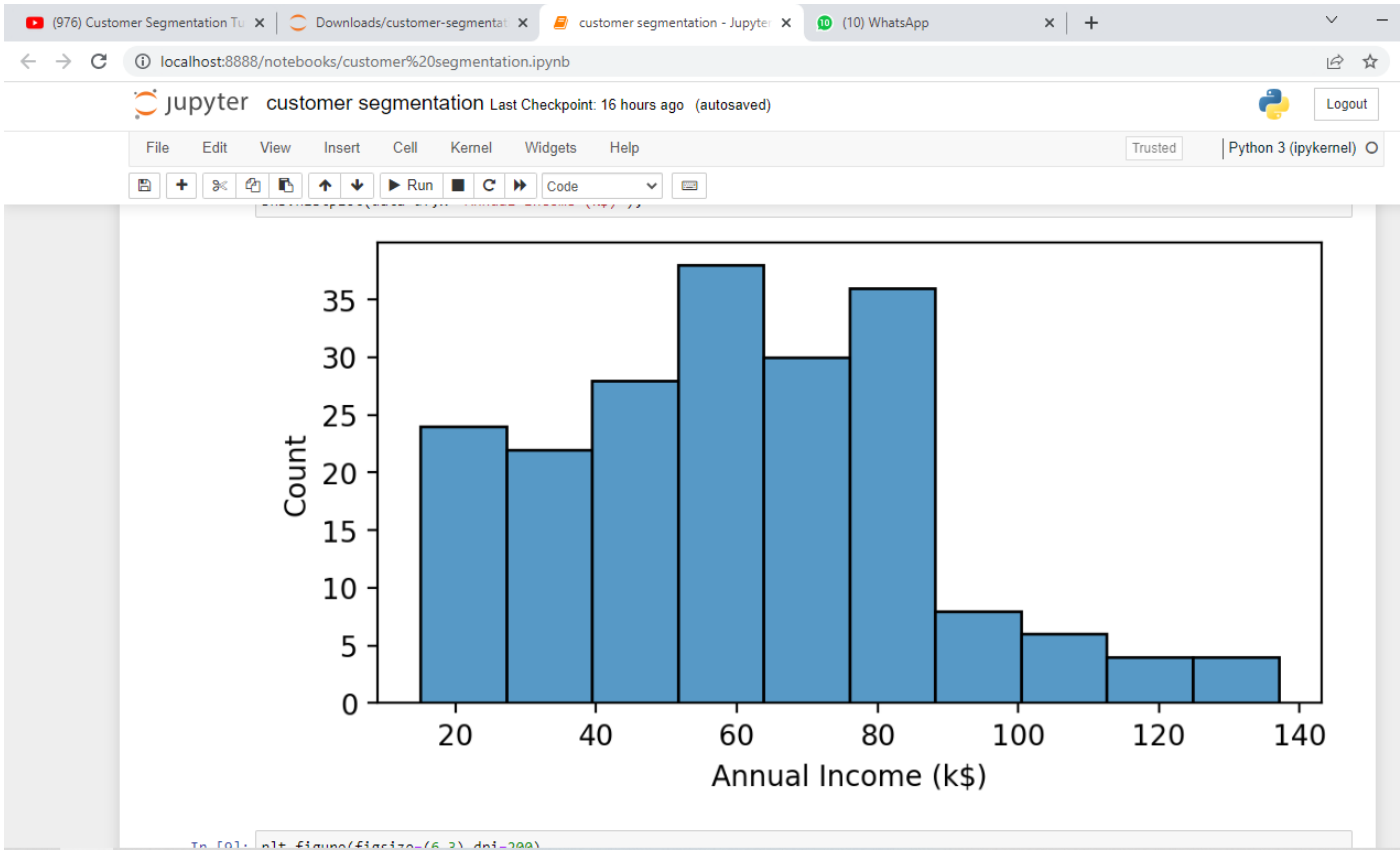
Run | Code

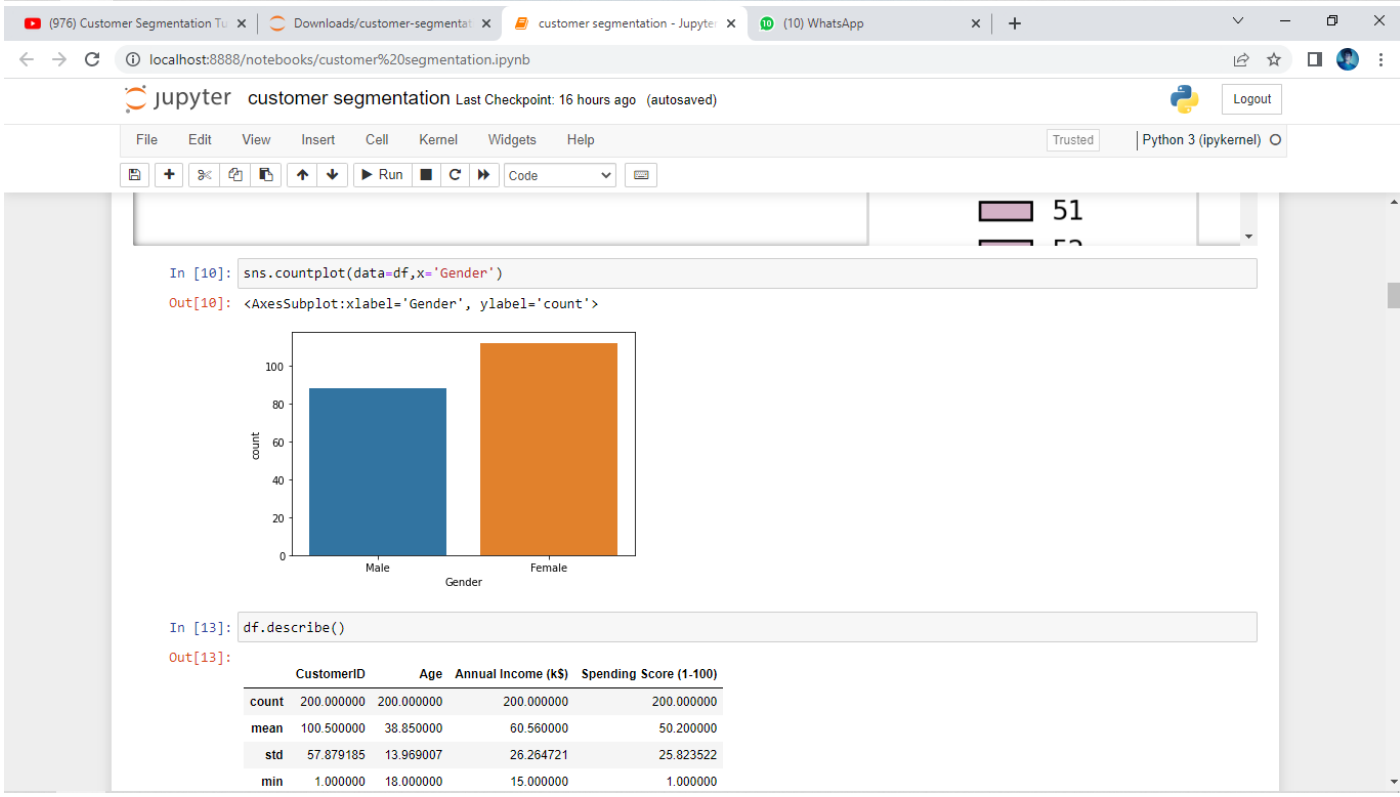
In [7]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 200 entries, 0 to 199
Data columns (total 5 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   CustomerID              200 non-null    int64
 1   Gender                  200 non-null    object
 2   Age                     200 non-null    int64
 3   Annual Income (k$)      200 non-null    int64
 4   Spending Score (1-100)  200 non-null    int64
dtypes: int64(4), object(1)
memory usage: 7.9+ KB
```

In [8]:
```
plt.figure(figsize=(6,3),dpi=200)
sns.histplot(data=df,x='Annual Income (k$)');
```







	CustomerID	Age	Annual Income (k$)	Spending Score (1-100)
count	200.000000	200.000000	200.000000	200.000000
mean	100.500000	38.850000	60.560000	50.200000
std	57.879185	13.969007	26.264721	25.823522
min	1.000000	18.000000	15.000000	1.000000

(976) Customer Segmentation Tu | Downloads/customer-segmenta | customer segmentation - Jupyter | (10) WhatsApp

localhost:8888/notebooks/customer%20segmentation.ipynb

jupyter **customer segmentation** Last Checkpoint: 16 hours ago (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help

Trusted | Python 3 (ipykernel)

Run | Code

In [14]:
```
df.dtypes
```

Out[14]:
```
CustomerID                int64
Gender                   object
Age                       int64
Annual Income (k$)        int64
Spending Score (1-100)    int64
dtype: object
```

In [20]:
```
df.isnull().sum()
```

Out[20]:
```
CustomerID                0
Gender                    0
Age                       0
Annual Income (k$)        0
Spending Score (1-100)    0
dtype: int64
```

In [23]:
```
df.drop(["CustomerID"],axis=1, inplace=True)
```

In [24]:
```
df.head()
```

Out[24]:

	Gender	Age	Annual Income (k$)	Spending Score (1-100)
0	Male	19	15	39
1	Male	21	15	81
2	Female	20	16	6
3	Female	23	16	77
4	Female	31	17	40

(976) Customer Segmentation Tu | Downloads/customer-segmenta | customer segmentation - Jupyter | (10) WhatsApp

localhost:8888/notebooks/customer%20segmentation.ipynb

jupyter **customer segmentation** Last Checkpoint: 16 hours ago (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help

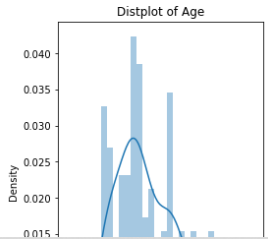
Trusted | Python 3 (ipykernel)

Run | Code

2	Female	20	16	6
3	Female	23	16	77
4	Female	31	17	40

In [25]:
```
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()
```

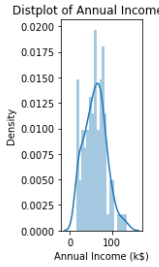
```
C:\Users\Ashutosh\Downloads\New folder\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated function and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```



Age

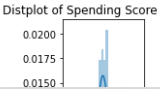
```
C:\Users\Ashutosh\Downloads\New folder\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated function and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

Distplot of Annual Income (k$)



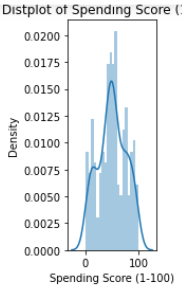

```
C:\Users\Ashutosh\Downloads\New folder\lib\site-packages\seaborn\distributions.py:2619: FutureWarning: `distplot` is a deprecated function and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

Distplot of Spending Score (1-100)

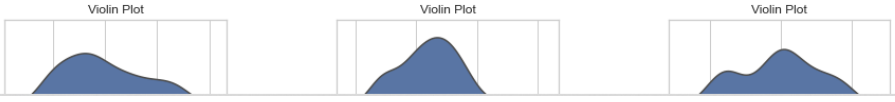


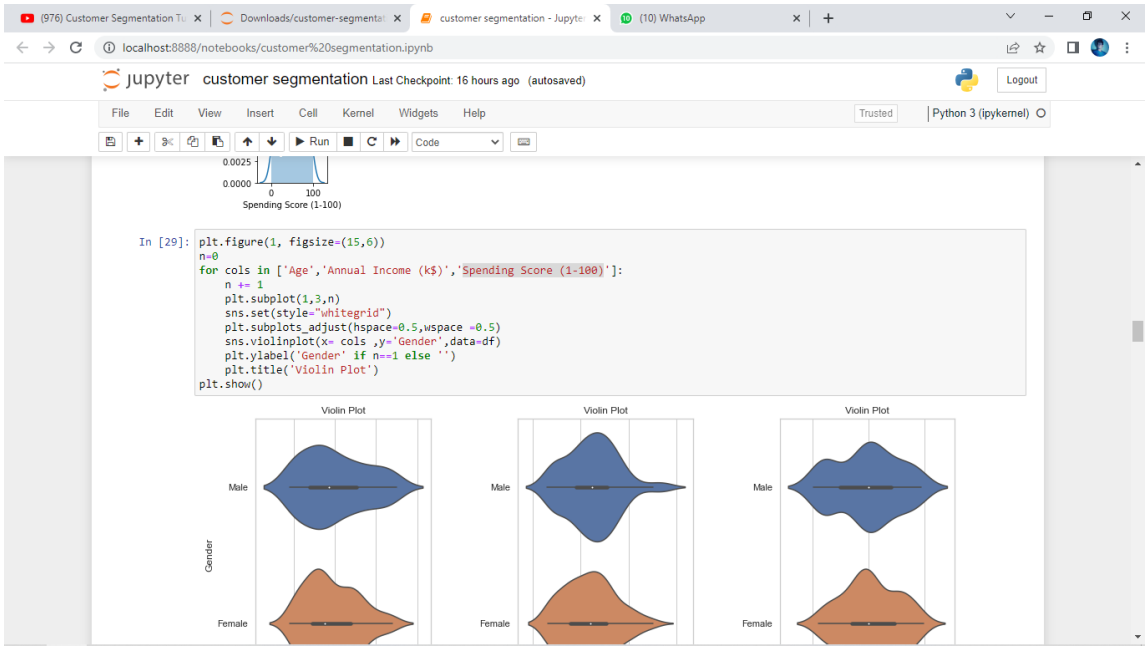
(976) Customer Segmentation Tu | Downloads/customer-segmentat | customer segmentation - Jupyter | (10) WhatsApp
localhost:8888/notebooks/customer%20segmentation.ipynb
jupyter customer segmentation Last Checkpoint: 16 hours ago (autosaved) Logout
File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel)
Run Code

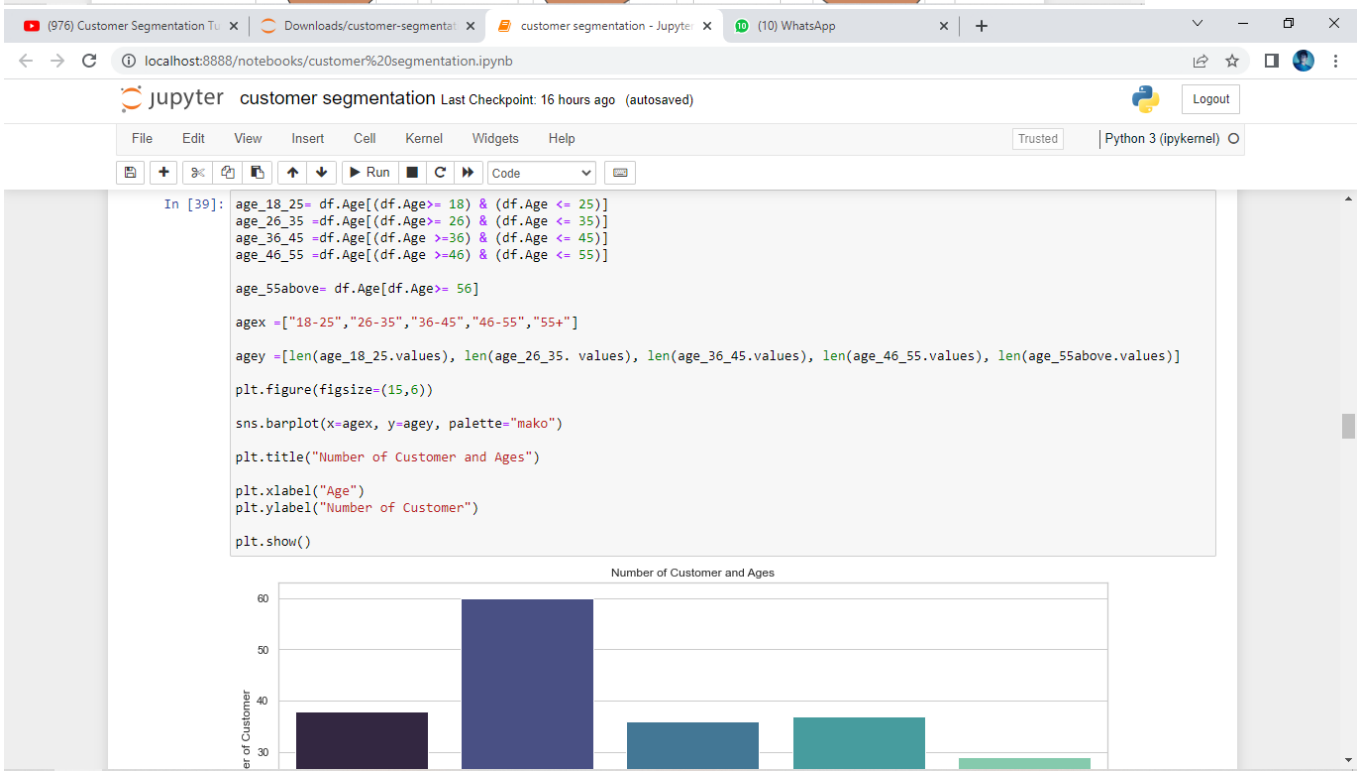
Distplot of Spending Score (1-100)



In [29]:
```
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()
```




jupyter customer segmentation Last Checkpoint: 16 hours ago (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3 (ipykernel)
Run
Code
0.0025
0.0000
0
100
Spending Score (1-100)
In [29]:
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()
Violin Plot
Violin Plot
Violin Plot
Male
Female
Gender


jupyter customer segmentation Last Checkpoint: 16 hours ago (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3 (ipykernel)
Run
Code
In [39]:
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_55above= df.Age[df.Age>= 56]
agex =["18-25","26-35","36-45","46-55","55+"]
agey =[len(age_18_25.values), len(age_26_35. values), len(age_36_45.values), len(age_46_55.values), len(age_55above.values)]
plt.figure(figsize=(15,6))
sns.barplot(x=agex, y=agey, palette="mako")
plt.title("Number of Customer and Ages")
plt.xlabel("Age")
plt.ylabel("Number of Customer")
plt.show()
Number of Customer and Ages
60
50
40
30

(976) Customer Segmentation Tu | Downloads/customer-segmentat | customer segmentation - Jupyter | (10) WhatsApp

localhost:8888/notebooks/customer%20segmentation.ipynb

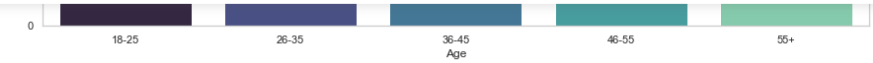
jupyter customer segmentation Last Checkpoint: 16 hours ago (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help

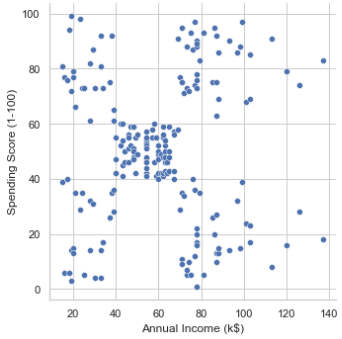
Trusted | Python 3 (ipykernel)

Run | Code



In [41]:
```
sns.relplot(x="Annual Income (k$)", y="Spending Score (1-100)",data=df)
```

Out[41]: <seaborn.axisgrid.FacetGrid at 0x201c29f46d0>



In [48]:
```
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)]
```

(976) Customer Segmentation Tu | Downloads/customer-segmentat | customer segmentation - Jupyter | (10) WhatsApp

localhost:8888/notebooks/customer%20segmentation.ipynb

jupyter customer segmentation Last Checkpoint: 16 hours ago (autosaved)

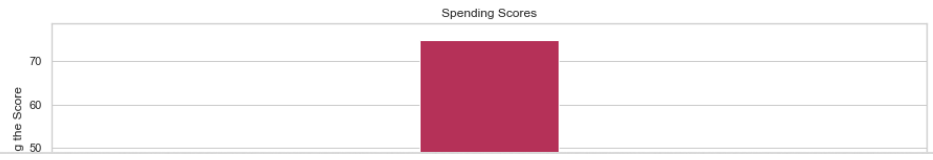
Logout

File Edit View Insert Cell Kernel Widgets Help

Trusted | Python 3 (ipykernel)

Run | Code

In [48]:
```
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()
```



(976) Customer Segmentation Tu | Downloads/customer-segmentat | customer segmentation - Jupyter | (10) WhatsApp

localhost:8888/notebooks/customer%20segmentation.ipynb

jupyter customer segmentation Last Checkpoint: 16 hours ago (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help

Trusted | Python 3 (ipykernel)

Run | Code

```
plt.show()
```

In [57]:

```
ai0_30=df["Annual Income (k$)"][(df["Annual Income (k$)"] >= 0) & (df["Annual Income (k$)"] <= 30)]
ai31_60=df["Annual Income (k$)"][(df["Annual Income (k$)"] >= 31) & (df["Annual Income (k$)"] <= 60)]

ai61_90=df["Annual Income (k$)"][(df["Annual Income (k$)"]>=61) & (df["Annual Income (k$)"] <= 90)]
ai91_120= df["Annual Income (k$)"][(df["Annual Income (k$)"]>= 91) & (df["Annual Income (k$)"] <= 120)]
ai121_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"]
```

(976) Customer Segmentation Tu | Downloads/customer-segmentat | customer segmentation - Jupyter | (10) WhatsApp

localhost:8888/notebooks/customer%20segmentation.ipynb

jupyter customer segmentation Last Checkpoint: 16 hours ago (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help

Trusted | Python 3 (ipykernel)

Run | Code

Score

In [57]:

```
ai0_30=df["Annual Income (k$)"][(df["Annual Income (k$)"] >= 0) & (df["Annual Income (k$)"] <= 30)]
ai31_60=df["Annual Income (k$)"][(df["Annual Income (k$)"] >= 31) & (df["Annual Income (k$)"] <= 60)]

ai61_90=df["Annual Income (k$)"][(df["Annual Income (k$)"]>=61) & (df["Annual Income (k$)"] <= 90)]
ai91_120= df["Annual Income (k$)"][(df["Annual Income (k$)"]>= 91) & (df["Annual Income (k$)"] <= 120)]
ai121_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(ai0_30.values), len(ai31_60.values), len(ai61_90.values), len(ai91_120.values), len(ai121_150.values)]
plt.figure(figsize=(15,6))

sns.barplot(x=aix, y=aiy, palette="Spectral")

plt.title("Annual Incomes")

plt.xlabel("Income")

plt.ylabel("Number of Customer")

plt.show()
```

(976) Customer Segmentation Tu | Downloads/customer-segmenta | customer segmentation - Jupyter | (10) WhatsApp

localhost:8888/notebooks/customer%20segmentation.ipynb

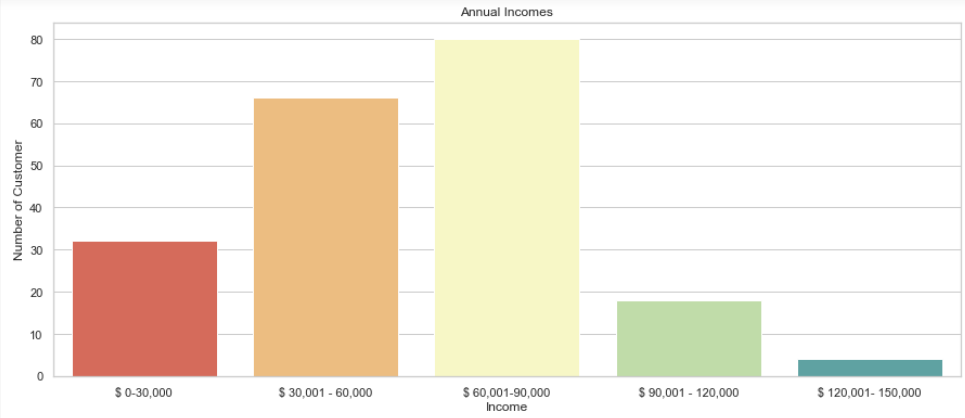
jupyter customer segmentation Last Checkpoint: 16 hours ago (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help

Trusted | Python 3 (ipykernel)

Run | Code



In [69]:
```
X1=df.loc[:, ["Age", "Spending Score (1-100)"]].values

from sklearn.cluster import KMeans
wcss =[]
for k in range(1,11):
    kmeans= KMeans(n_clusters=k, init="k-means++")
    kmeans.fit(X1)
    wcss.append(kmeans.inertia_)
plt.figure(figsize=(12,6))
plt.grid()
```

(976) Customer Segmentation Tu | Downloads/customer-segmenta | customer segmentation - Jupyter | (10) WhatsApp

localhost:8888/notebooks/customer%20segmentation.ipynb

jupyter customer segmentation Last Checkpoint: 16 hours ago (autosaved)

Logout

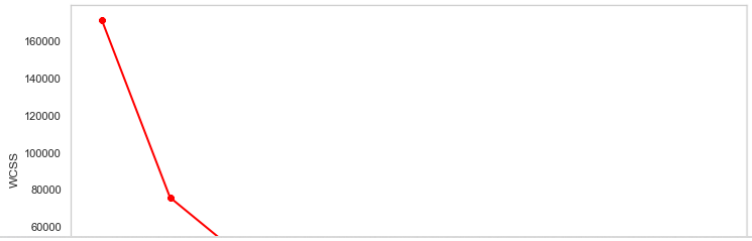
File Edit View Insert Cell Kernel Widgets Help

Trusted | Python 3 (ipykernel)

Run | Code

In [69]:
```
X1=df.loc[:, ["Age", "Spending Score (1-100)"]].values

from sklearn.cluster import KMeans
wcss =[]
for k in range(1,11):
    kmeans= KMeans(n_clusters=k, init="k-means++")
    kmeans.fit(X1)
    wcss.append(kmeans.inertia_)
plt.figure(figsize=(12,6))
plt.grid()
plt.plot(range(1,11),wcss, linewidth=2, color="red", marker="8")
plt.xlabel("K Value")
plt.ylabel("WCSS")
plt.show()
```

```
C:\Users\Ashutosh\Downloads\New folder\lib\site-packages\sklearn\cluster\_kmeans.py:1036: UserWarning: KMeans is known to have a memory leak on Windows with MKL, when there are less chunks than available threads. You can avoid it by setting the environment variable OMP_NUM_THREADS=1.
  warnings.warn(
```



(976) Customer Segmentation Tu | Downloads/customer-segmentat | customer segmentation - Jupyter | (10) WhatsApp

localhost:8888/notebooks/customer%20segmentation.ipynb

jupyter **customer segmentation** Last Checkpoint: 16 hours ago (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help | Trusted | Python 3 (ipykernel)

Run | Code

K Value

In [71]:
```
kmeans = KMeans(n_clusters=4)
label=kmeans.fit_predict(X1)
print(label)
```

```
[3 1 0 1 3 1 0 1 0 1 0 1 0 1 3 3 0 1 3 1 0 1 0 1 0 3 0 1 0 1 0 1 0
 1 0 1 2 1 2 3 0 3 2 3 3 3 2 3 3 2 2 2 2 2 3 2 2 3 2 2 2 3 2 2 3 3 2 2 2 2
 2 3 2 3 3 2 2 3 2 2 3 2 2 3 3 2 2 3 2 3 3 3 2 3 2 3 3 2 2 2 3 2 2 2 2 2
 3 3 3 3 3 2 2 2 2 3 3 3 1 3 1 2 1 0 1 0 1 3 1 0 1 0 1 0 1 0 1 3 1 0 1 2 1
 0 1 0 1 0 1 0 1 0 1 0 1 2 1 0 1 0 1 0 1 0 3 0 1 0 1 0 1 0 1 0 1 0 1 0 1 3
 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1]
```

In [72]:
```
print(kmeans.cluster_centers_)
```

```
[[43.29166667 15.02083333]
 [30.1754386  82.35087719]
 [55.70833333 48.22916667]
 [27.61702128 49.14893617]]
```

In [74]:
```
plt.scatter(X1[:,0],X1[:,1],c=kmeans.labels_,cmap='rainbow')
plt.scatter(kmeans.cluster_centers_[:,0],kmeans.cluster_centers_[:,1], color="black")

plt.title('Clusters of Customers')

plt.xlabel('age')

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

plt.show()
```

(976) Customer Segmentation Tu | Downloads/customer-segmentat | customer segmentation - Jupyter | (10) WhatsApp

localhost:8888/notebooks/customer%20segmentation.ipynb

jupyter **customer segmentation** Last Checkpoint: 16 hours ago (autosaved)

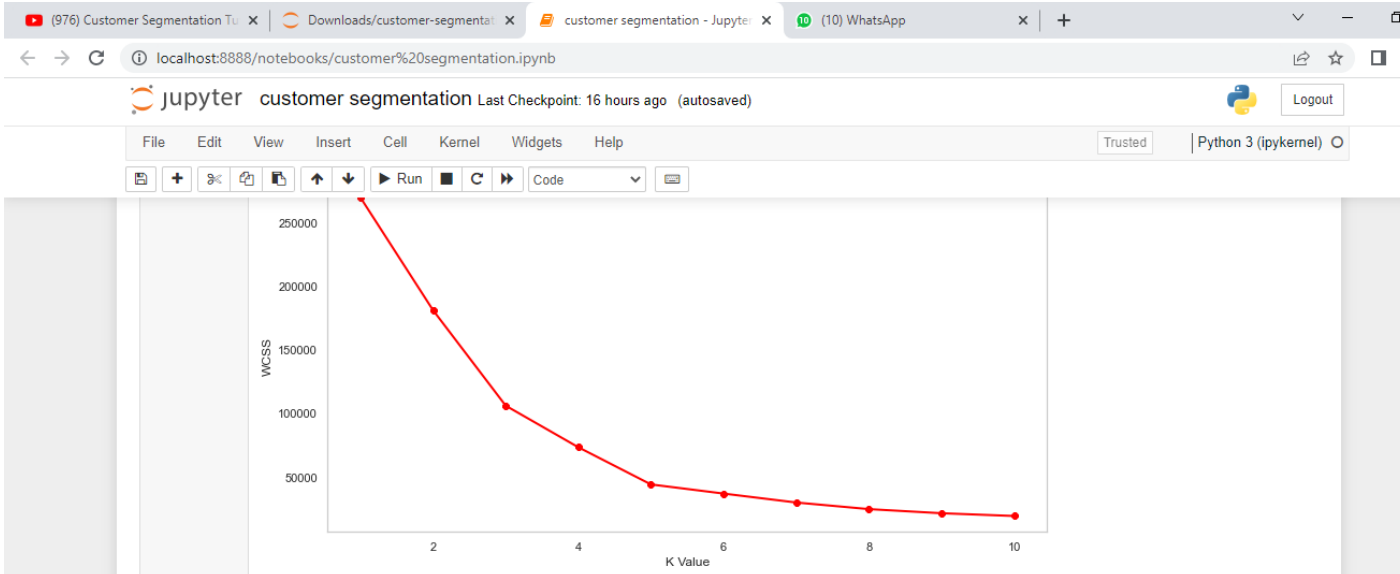
Logout

File Edit View Insert Cell Kernel Widgets Help | Trusted | Python 3 (ipykernel)

Run | Code

```
plt.show()
```

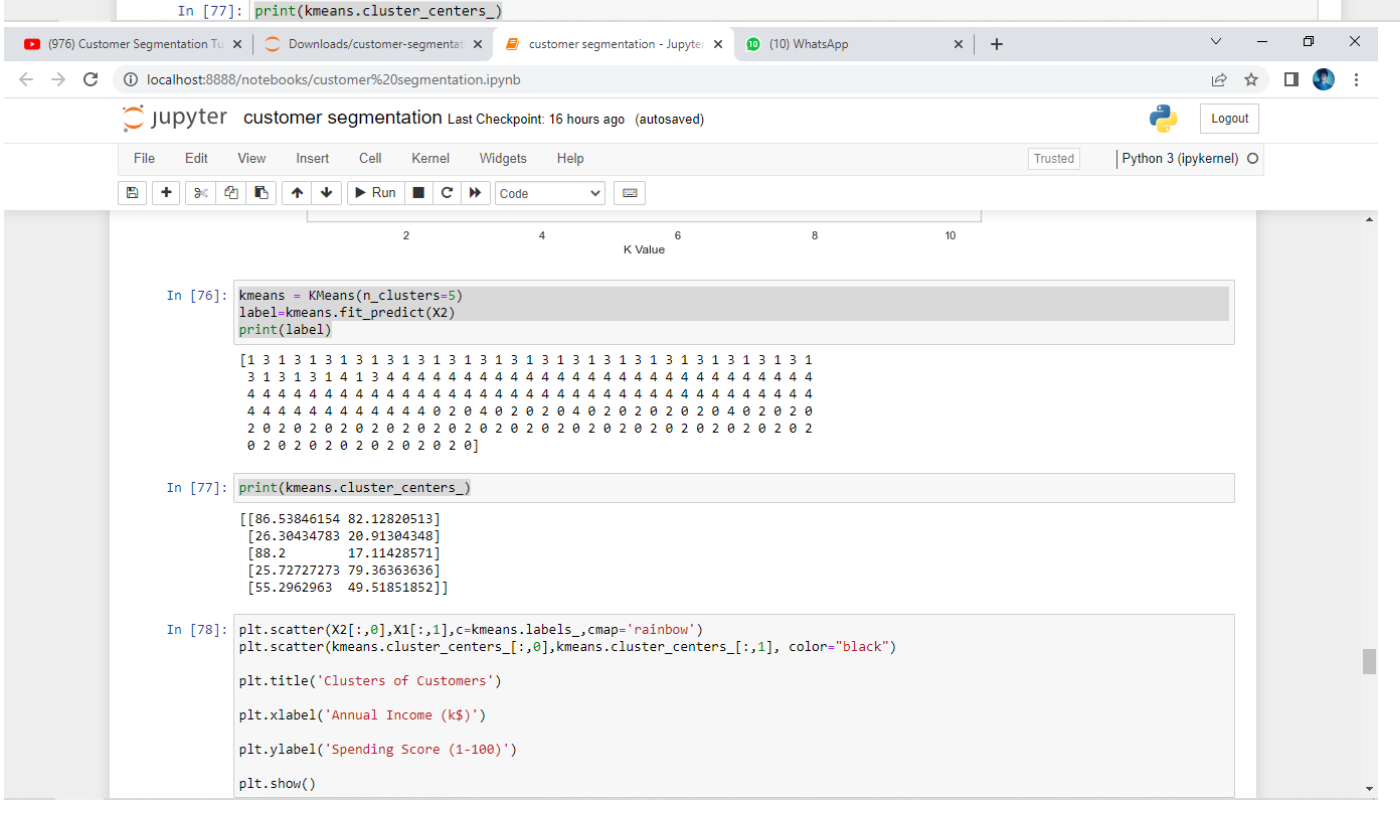
In [75]:
```
X2=df.loc[:, ["Annual Income (k$)", "Spending Score (1-100)"]].values

from sklearn.cluster import KMeans
wcss =[]
for k in range(1,11):
    kmeans= KMeans(n_clusters=k, init="k-means++")
    kmeans.fit(X2)
    wcss.append(kmeans.inertia_)
plt.figure(figsize=(12,6))
plt.grid()
plt.plot(range(1,11),wcss, linewidth=2, color="red", marker="8")
plt.xlabel("K Value")
plt.ylabel("WCSS")
plt.show()
```



In [76]:
```
kmeans = KMeans(n_clusters=5)
label=kmeans.fit_predict(X2)
print(label)
```

```
[1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1
 3 1 3 1 3 1 4 1 3 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4
 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4
 4 4 4 4 4 4 4 4 4 4 4 4 0 2 0 4 0 2 0 2 0 4 0 2 0 2 0 2 0 2 0 4 0 2 0 2 0
 2 0 2 0 2 0 2 0 2 0 2 0 2 0 2 0 2 0 2 0 2 0 2 0 2 0 2 0 2 0 2 0 2 0 2 0 2
 0 2 0 2 0 2 0 2 0 2 0 2 0 2 0]
```

In [77]:
```
print(kmeans.cluster_centers_)
```



In [76]:
```
kmeans = KMeans(n_clusters=5)
label=kmeans.fit_predict(X2)
print(label)
```

```
[1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1 3 1
 3 1 3 1 3 1 4 1 3 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4
 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4
 4 4 4 4 4 4 4 4 4 4 4 4 0 2 0 4 0 2 0 2 0 4 0 2 0 2 0 2 0 2 0 4 0 2 0 2 0
 2 0 2 0 2 0 2 0 2 0 2 0 2 0 2 0 2 0 2 0 2 0 2 0 2 0 2 0 2 0 2 0 2 0 2 0 2
 0 2 0 2 0 2 0 2 0 2 0 2 0 2 0]
```

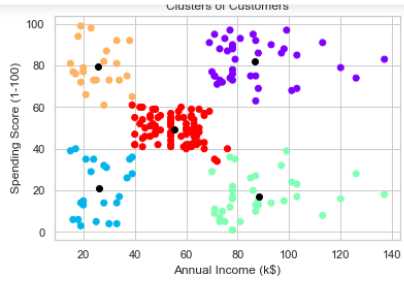
In [77]:
```
print(kmeans.cluster_centers_)
```

```
[[86.53846154 82.12820513]
 [26.30434783 20.91304348]
 [88.2        17.11428571]
 [25.72727273 79.36363636]
 [55.2962963  49.51851852]]
```

In [78]:
```
plt.scatter(X2[:,0],X1[:,1],c=kmeans.labels_,cmap='rainbow')
plt.scatter(kmeans.cluster_centers_[:,0],kmeans.cluster_centers_[:,1], color="black")

plt.title('Clusters of Customers')

plt.xlabel('Annual Income (k$)')

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

plt.show()
```

jupyter customer segmentation Last Checkpoint: 16 hours ago (autosaved)

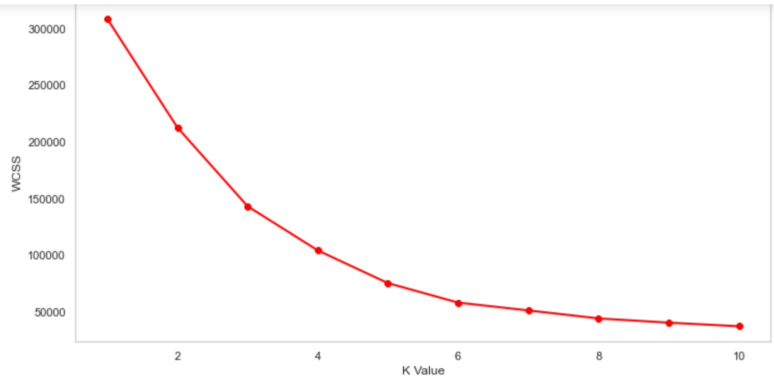
File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel)



In [79]:
```python
X3=df.iloc[:,1:]
wcss =[]
for k in range(1,11):
    kmeans= KMeans(n_clusters=k, init="k-means++")
    kmeans.fit(X3)
    wcss.append(kmeans.inertia_)
plt.figure(figsize=(12,6))
plt.grid()
plt.plot(range(1,11),wcss, linewidth=2, color="red", marker="8")
plt.xlabel("K Value")
plt.ylabel("WCSS")
plt.show()
```

```
C:\Users\Ashutosh\Downloads\New folder\lib\site-packages\sklearn\cluster\_kmeans.py:1036: UserWarning: KMeans is known to have
a memory leak on Windows with MKL, when there are less chunks than available threads. You can avoid it by setting the environme
nt variable OMP_NUM_THREADS=1.
  warnings.warn(
```

(976) Customer Segmentation Tu × | Downloads/customer-segmentat × | customer segmentation - Jupyter × | (10) WhatsApp ×
localhost:8888/notebooks/customer%20segmentation.ipynb
jupyter customer segmentation Last Checkpoint: 16 hours ago (autosaved)
File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel)



In [80]:
```python
kmeans = KMeans(n_clusters=5)
label=kmeans.fit_predict(X3)
print(label)
```

```
[0 4 0 4 0 4 0 4 0 4 0 4 0 4 0 4 0 4 0 4 0 4 0 4 0 4 0 4 0 4 0 4 0 4 0 4 0
 4 0 4 0 4 0 4 0 4 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 2 3 2 1 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2
 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3
 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2]
```

(976) Customer Segmentation Tu | Downloads/customer-segmentat | customer segmentation - Jupyter | (10) WhatsApp

localhost:8888/notebooks/customer%20segmentation.ipynb

jupyter customer segmentation Last Checkpoint: 16 hours ago (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help

Trusted | Python 3 (ipykernel)

Run | Code

```
[[45.2173913  26.30434783 20.91304348]
 [43.28205128 55.02564103 49.69230769]
 [32.69230769 86.53846154 82.12820513]
 [40.32432432 87.43243243 18.18918919]
 [25.52173913 26.30434783 78.56521739]]
```

In [84]:

```
clusters=kmeans.fit_predict(X3)
df["label"]= clusters

from mpl_toolkits.mplot3d import Axes3D

fig=plt.figure(figsize=(20,10))

ax=fig.add_subplot(111, projection='3d')

ax.scatter(df.Age[df.label==0], df["Annual Income (k$)"][df.label== 0], df["Spending Score (1-100)"][df.label== 0], c='blue', s=6
ax.scatter(df.Age[df.label==1], df["Annual Income (k$)"][df.label== 1], df["Spending Score (1-100)"][df.label== 1], c='red', s=60
ax.scatter(df.Age[df.label==2], df["Annual Income (k$)"][df.label== 2], df["Spending Score (1-100)"][df.label== 2], c='green', s=
ax.scatter(df.Age[df.label==3], df["Annual Income (k$)"][df.label== 3], df["Spending Score (1-100)"][df.label== 3], c='orange', s
ax.scatter(df.Age[df.label==4], df["Annual Income (k$)"][df.label== 4], df["Spending Score (1-100)"][df.label== 4], c='purple', s

ax.view_init(30, 185)

plt.xlabel("Age")

plt.ylabel("Annual Income (k$)")
ax.set_zlabel('Spending Score (1-100)')

plt.show()
```

(976) Customer Segmentation Tu | Downloads/customer-segmentat | customer segmentation - Jupyter | (10) WhatsApp

localhost:8888/notebooks/customer%20segmentation.ipynb

jupyter customer segmentation Last Checkpoint: 16 hours ago (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help

Trusted | Python 3 (ipykernel)

Run | Code