```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans
```

```
data = {
    'CustomerID': list(range(1, 21)),
    'Gender': ['Male', 'Male', 'Female', 'Female', 'Female', 'Female', 'Female', 'Female', 'Male', 'Female', 'Male', 'Female', 'Fem
    'Age': [19,21,20,23,31,22,35,23,64,30,67,35,58,24,37,22,35,20,52,35],
    'Annual Income (k$)': [15,15,16,16,17,17,18,18,19,19,19,19,20,20,20,20,21,21,23,23],
    'Spending Score (1-100)': [39,81,6,77,40,76,6,94,3,72,14,99,15,77,13,79,35,66,29,98]
}
```

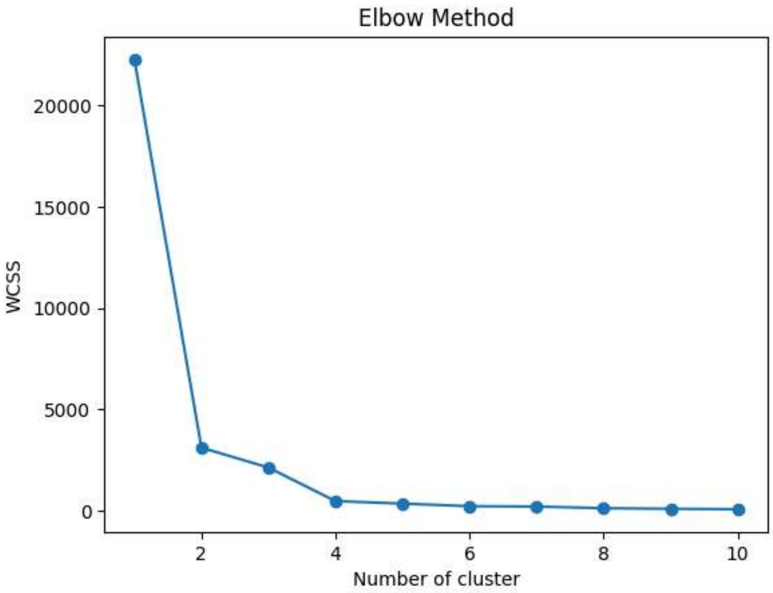
```
df = pd.DataFrame(data)
```

```
X = df[['Annual Income (k$)', 'Spending Score (1-100)']]
```

```
wcss = []
for i in range(1, 11):
    kmeans = KMeans(n_clusters=i, init = 'k-means++', random_state=42)
    kmeans.fit(X)
    wcss.append(kmeans.inertia_)
```

```
plt.plot(range(1, 11), wcss, marker='o')
plt.title('Elbow Method')
plt.xlabel('Number of cluster')
plt.ylabel('WCSS')
plt.show
```

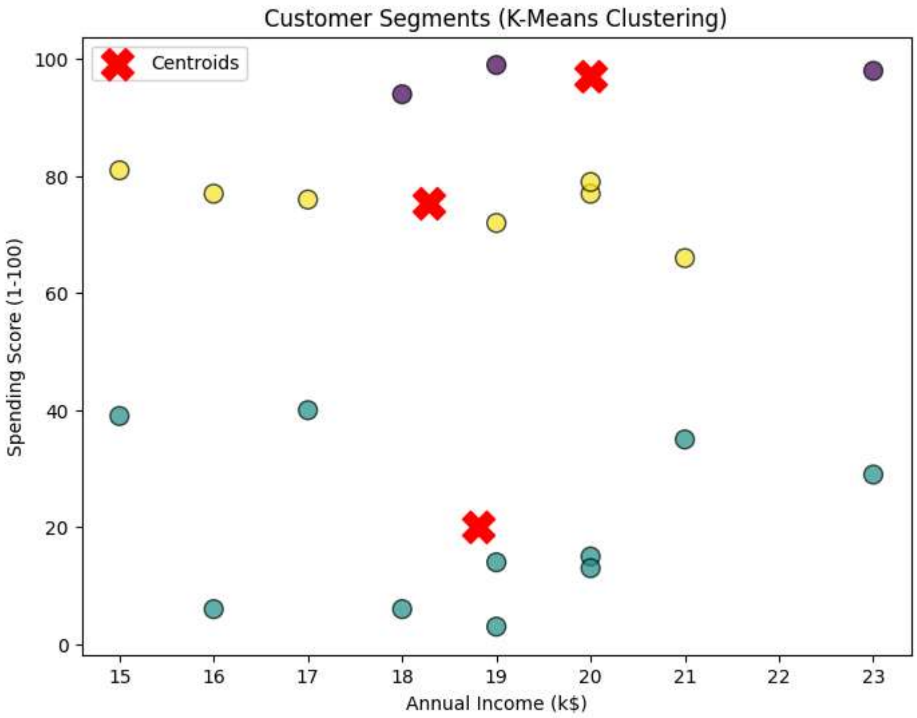
```
matplotlib.pyplot.show
def show(*args, **kwargs) -> None

/usr/local/lib/python3.12/dist-packages/matplotlib/pyplot.py
Display all open figures.

Parameters
----------
block : bool, optional
```



```
kmeans = KMeans(n_clusters=3, init='k-means++', random_state=42)
df['Cluster'] = kmeans.fit_predict(X)
```

```
plt.figure(figsize=(8,6))
plt.scatter(X['Annual Income (k$)'], X['Spending Score (1-100)'],
           c=df['Cluster'], cmap='viridis', s=100, alpha=0.7, edgecolors='k')
plt.scatter(kmeans.cluster_centers_[:,0], kmeans.cluster_ ✦ ers_[:,1],
           s=300, c='red', marker='X', label='Centroids')
```

```
plt.title('Customer Segments (K-Means Clustering)')
plt.xlabel('Annual Income (k$)')
plt.ylabel('Spending Score (1-100)')
plt.legend()
plt.show()
```



```
print(df[['CustomerID', 'Annual Income (k$)', 'Spending Score (1-100)', 'Cluster']])
```

```
    CustomerID  Annual Income (k$)  Spending Score (1-100)  Cluster
0            1                  15                      39        1
1            2                  15                      81        2
2            3                  16                       6        1
3            4                  16                      77        2
4            5                  17                      40        1
5            6                  17                      76        2
6            7                  18                       6        1
7            8                  18                      94        0
8            9                  19                       3        1
9           10                  19                      72        2
10          11                  19                      14        1
11          12                  19                      99        0
12          13                  20                      15        1
13          14                  20                      77        2
14          15                  20                      13        1
15          16                  20                      79        2
16          17                  21                      35        1
17          18                  21                      66        2
18          19                  23                      29        1
19          20                  23                      98        0
```

Start coding or generate with AI.