Using Credit Card Dataset develop a customer segmentation using KMeans to define marketing strategy.

```
The Dataset summarizes the usage behavior of about 9000 active credit card
holders during the last 6 months.

The file is at a customer level with 18 behavioral variables.

Following is the Data Dictionary for Credit Card dataset :-

CUST_ID : Identification of Credit Card holder (Categorical)
BALANCE : Balance amount left in their account to make purchases (
BALANCE_FREQUENCY : How frequently the Balance is updated, score between 0
and 1 (1 = frequently updated, 0 = not frequently updated)
PURCHASES : Amount of purchases made from account
ONEOFF_PURCHASES : Maximum purchase amount done in one-go
INSTALLMENTS_PURCHASES : Amount of purchase done in installment
CASH_ADVANCE : Cash in advance given by the user
PURCHASES_FREQUENCY : How frequently the Purchases are being made, score
between 0 and 1 (1 = frequently purchased, 0 = not frequently purchased)
ONEOFFPURCHASESFREQUENCY : How frequently Purchases are happening in one-go
(1 = frequently purchased, 0 = not frequently purchased)
PURCHASESINSTALLMENTSFREQUENCY : How frequently purchases in installments
are being done (1 = frequently done, 0 = not frequently done)
CASHADVANCEFREQUENCY : How frequently the cash in advance being paid
CASHADVANCETRX : Number of Transactions made with "Cash in Advanced"
PURCHASES_TRX : Numbe of purchase transactions made
CREDIT_LIMIT : Limit of Credit Card for user
PAYMENTS : Amount of Payment done by user
MINIMUM_PAYMENTS : Minimum amount of payments made by user
PRCFULLPAYMENT : Percent of full payment paid by user
TENURE : Tenure of credit card service for user
```

In [1]:

```python
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

In [2]:

```python
dataset = pd.read_csv('CC_GENERAL.csv')
X = dataset.iloc[:, 1:].values
```

In [13]:

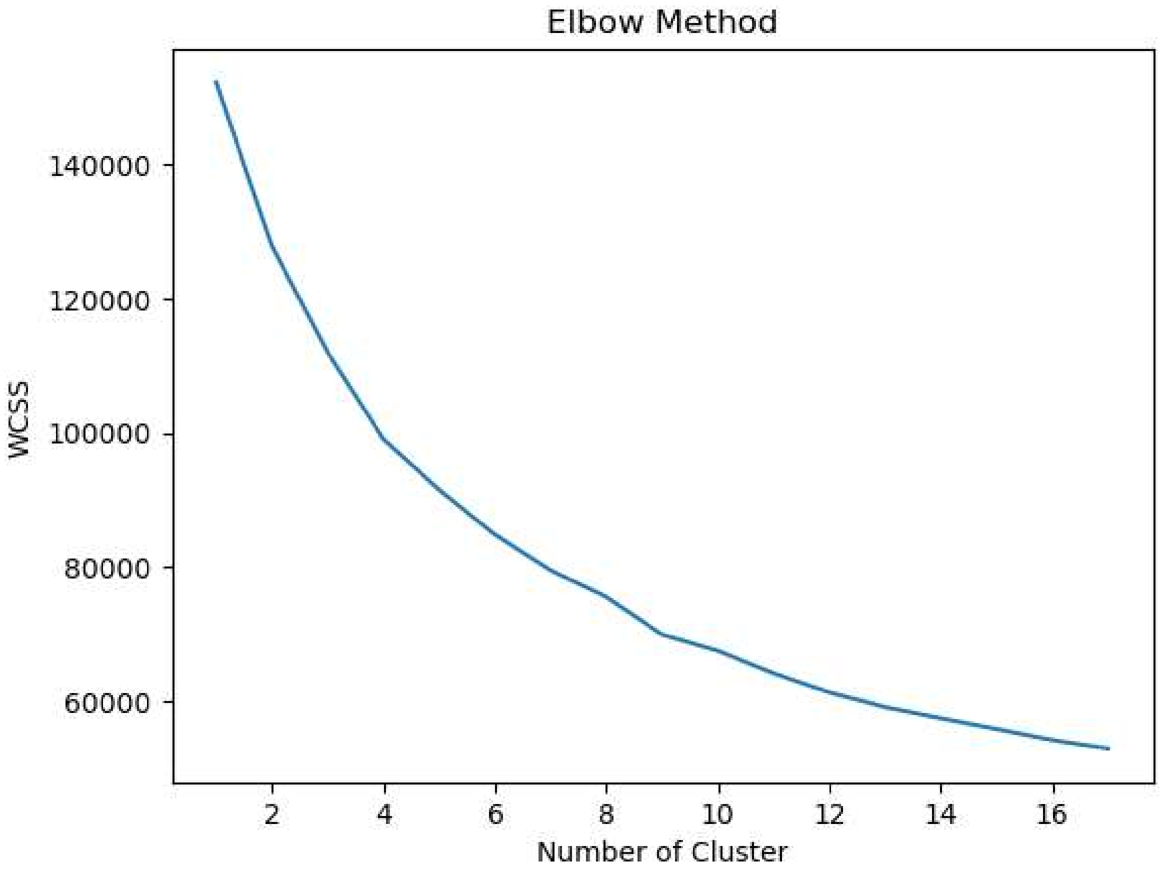
```python
from sklearn.impute import SimpleImputer
imputer = SimpleImputer(missing_values= np.NAN, strategy= 'mean', fill_value=N
imputer = imputer.fit(X)
X = imputer.transform(X)
```

```
C:\Users\Lenovo\anaconda3\lib\site-packages\sklearn\impute\_base.py:382: Fut
ureWarning: The 'verbose' parameter was deprecated in version 1.1 and will b
e removed in 1.3. A warning will always be raised upon the removal of empty
columns in the future version.
  warnings.warn(
```

In [14]:
```
from sklearn.preprocessing import StandardScaler
sc_X = StandardScaler()
X = sc_X.fit_transform(X)
```

In [15]:
```
from sklearn.cluster import KMeans
wcss = []
for i in range(1,18):
    kmeans = KMeans(n_clusters=i, init='k-means++', max_iter=300, n_init=10, r
    wcss.append(kmeans.inertia_)

plt.plot(range(1,18), wcss)
plt.title('Elbow Method')
plt.xlabel('Number of Cluster')
plt.ylabel('WCSS')
plt.show()
```



In [16]:
```
kmeans = KMeans(n_clusters=8, init='k-means++', max_iter=300, n_init=10, rando
y_kmeans = kmeans.fit_predict(X)
```

In [17]:
```
dataset['Cluster'] = y_kmeans
dataset.head()
```

Out[17]:

	CUST_ID	BALANCE	BALANCE_FREQUENCY	PURCHASES	ONEOFF_PURCHASES	INSTAL
0	C10001	40.900749	0.818182	95.40	0.00	
1	C10002	3202.467416	0.909091	0.00	0.00	
2	C10003	2495.148862	1.000000	773.17	773.17	
3	C10004	1666.670542	0.636364	1499.00	1499.00	
4	C10005	817.714335	1.000000	16.00	16.00	

In [ ]: