In [18]:
```
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans
import warnings
from sklearn.preprocessing import StandardScaler
warnings.filterwarnings('ignore')
```

In [2]:
```
df = pd.read_csv("sales_data_sample.csv", encoding="latin")
```

In [3]:
```
df.head()
```

Out[3]:

	ORDERNUMBER	QUANTITYORDERED	PRICEEACH	ORDERLINENUMBER	SAI
0	10107	30	95.70	2	2871
1	10121	34	81.35	5	2765
2	10134	41	94.74	2	3884
3	10145	45	83.26	6	3746
4	10159	49	100.00	14	5205

5 rows × 25 columns

In [4]:
```
df.info()
```

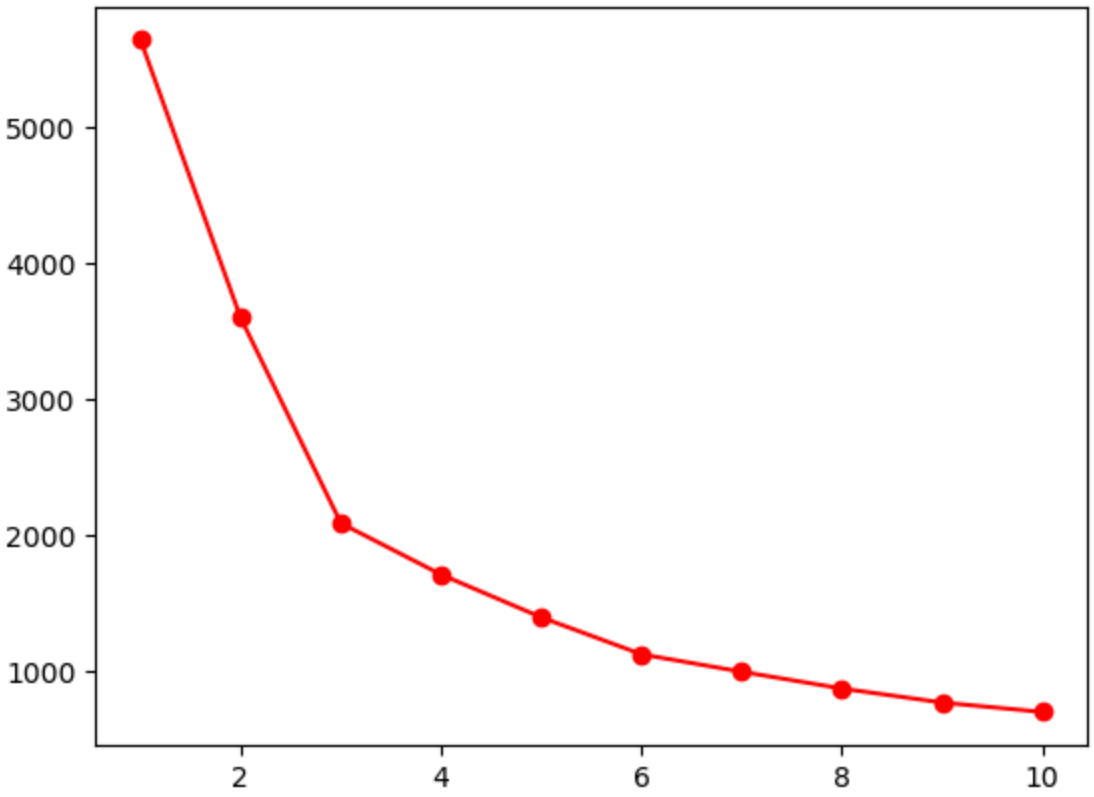
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2823 entries, 0 to 2822
Data columns (total 25 columns):
 #   Column            Non-Null Count  Dtype  
---  ------            --------------  -----  
 0   ORDERNUMBER       2823 non-null   int64  
 1   QUANTITYORDERED   2823 non-null   int64  
 2   PRICEEACH         2823 non-null   float64
 3   ORDERLINENUMBER   2823 non-null   int64  
 4   SALES             2823 non-null   float64
 5   ORDERDATE         2823 non-null   object 
 6   STATUS            2823 non-null   object 
 7   QTR_ID            2823 non-null   int64  
 8   MONTH_ID          2823 non-null   int64  
 9   YEAR_ID           2823 non-null   int64  
 10  PRODUCTLINE       2823 non-null   object 
 11  MSRP              2823 non-null   int64  
 12  PRODUCTCODE       2823 non-null   object 
 13  CUSTOMERNAME      2823 non-null   object 
 14  PHONE             2823 non-null   object 
 15  ADDRESSLINE1      2823 non-null   object 
 16  ADDRESSLINE2      302 non-null    object 
 17  CITY              2823 non-null   object 
 18  STATE             1337 non-null   object 
 19  POSTALCODE        2747 non-null   object 
 20  COUNTRY           2823 non-null   object 
 21  TERRITORY         1749 non-null   object 
 22  CONTACTLASTNAME   2823 non-null   object 
 23  CONTACTFIRSTNAME  2823 non-null   object 
 24  DEALSIZE          2823 non-null   object 
dtypes: float64(2), int64(7), object(16)
memory usage: 551.5+ KB
```

In [32]:
```
df = df[['ORDERLINENUMBER', 'SALES']]
```

In [33]:
```
scaler = StandardScaler()
scaled_values = scaler.fit_transform(df.values)
```

In [34]:
```
wcss = []
for i in range(1, 11):
    model = KMeans(n_clusters=i, init='k-means++')
    model.fit_predict(scaled_values)
    wcss.append(model.inertia_)
```

In [35]:
```
plt.plot(range(1, 11), wcss, 'ro-')
plt.show()
```



In [42]:
```
model = KMeans(n_clusters=7, init='k-means++')
clusters = model.fit_predict(scaled_values)
clusters
```

Out[42]:
```
array([3, 3, 0, ..., 4, 3, 6])
```

In [43]:
```
df['cluster'] = clusters
```

In [44]:
```
df
```

Out[44]:

	ORDERLINENUMBER	SALES	cluster
0	2	2871.00	3
1	5	2765.90	3
2	2	3884.34	0
3	6	3746.70	0
4	14	5205.27	5
...	...	...	...
2818	15	2244.40	1
2819	1	3978.51	0
2820	4	5417.57	4
2821	1	2116.16	3
2822	9	3079.44	6

2823 rows × 3 columns

In [45]:
```
model.inertia_
```

Out[45]: 993.4283577026391

In [46]:
```
plt.scatter(df['ORDERLINENUMBER'], df['SALES'], c=df['cluster'])
plt.show()
```