1. Import Libraries

In [29]:
```
import seaborn as sns
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans
import warnings
warnings.filterwarnings('ignore')
```

2. Import Datasets

In [10]:
```
sns.get_dataset_names()
```

Out[10]:
```
['anagrams',
 'anscombe',
 'attention',
 'brain_networks',
 'car_crashes',
 'diamonds',
 'dots',
 'exercise',
 'flights',
 'fmri',
 'gammas',
 'geyser',
 'iris',
 'mpg',
 'penguins',
 'planets',
 'taxis',
 'tips',
 'titanic']
```

In [12]:
```
data=sns.load_dataset('iris')
```

In [13]:
```
data.head()
```

Out[13]:

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa

3. Data Undestanding

In [ ]:
```
x=data.drop(['species'],axis=1)
y=data['species']
```

In [16]:
```
from sklearn.preprocessing import StandardScaler
```

In [41]:
```
scl=StandardScaler()
scaled_x=scl.fit_transform(x)
scaled_x=pd.DataFrame(scaled_x,columns=x.columns)
scaled_x.head()
```

Out[41]:

	sepal_length	sepal_width	petal_length	petal_width
0	-0.900681	1.019004	-1.340227	-1.315444
1	-1.143017	-0.131979	-1.340227	-1.315444
2	-1.385353	0.328414	-1.397064	-1.315444
3	-1.506521	0.098217	-1.283389	-1.315444
4	-1.021849	1.249201	-1.340227	-1.315444

4. Find best K

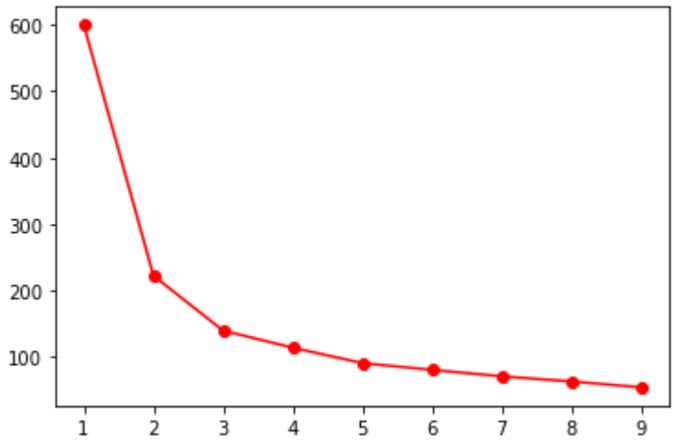
In [30]:
```
wcss=[]
for i in range(1,10):
    knn=KMeans(n_clusters=i)
    knn.fit(scaled_x)
    wcss.append(knn.inertia_)
print(wcss)
```

```
[599.9999999999999, 222.36170496502294, 139.82049635974968, 114.0925469040309,
90.85850278921471, 80.75886526941828, 71.00980160028563, 63.337004756179475, 5
4.44902636554481]
```

5.Elbow Curve

In [32]:
```
plt.plot(range(1,10),wcss,'ro-')
```

Out[32]:
```
[<matplotlib.lines.Line2D at 0x22585d47400>]
```



6. Data training

In [33]:
```
#Ideally value of k=3
```

In [40]:
```
knn=KMeans(n_clusters=3)
cluster=knn.fit(scaled_x)
cluster.labels_
cluster.get_params
```

Out[40]:
```
<bound method BaseEstimator.get_params of KMeans(n_clusters=3)>
```

In [37]:
```
data['cluster']=cluster.labels_
```

In [39]:
```
data.head()
```

Out[39]:

	sepal_length	sepal_width	petal_length	petal_width	species	cluster
0	5.1	3.5	1.4	0.2	setosa	1
1	4.9	3.0	1.4	0.2	setosa	1
2	4.7	3.2	1.3	0.2	setosa	1
3	4.6	3.1	1.5	0.2	setosa	1
4	5.0	3.6	1.4	0.2	setosa	1