

```
In [4]: import pandas as pd
from sklearn.cluster import KMeans
from sklearn.preprocessing import MinMaxScaler
from matplotlib import pyplot as plt
from sklearn.datasets import load_iris
%matplotlib inline
```

```
In [5]: iris=load_iris()
```

```
In [10]: df= pd.DataFrame(iris.data,columns=iris.feature_names)
df.head()
```

Out[10]:	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)
0	5.1	3.5	1.4	0.2
1	4.9	3.0	1.4	0.2
2	4.7	3.2	1.3	0.2
3	4.6	3.1	1.5	0.2
4	5.0	3.6	1.4	0.2

```
In [11]: df['flower']=iris.target
df.head()
```

Out[11]:	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)	flower
0	5.1	3.5	1.4	0.2	0
1	4.9	3.0	1.4	0.2	0
2	4.7	3.2	1.3	0.2	0
3	4.6	3.1	1.5	0.2	0
4	5.0	3.6	1.4	0.2	0

```
In [12]: df.drop(['sepal length (cm)', 'sepal width (cm)', 'flower'],axis='columns',inplace=True)
```

```
In [13]: df.head(3)
```

	petal length (cm)	petal width (cm)
<b>0</b>	1.4	0.2
<b>1</b>	1.4	0.2
<b>2</b>	1.3	0.2

```
In [14]: km=KMeans(n_clusters=3)
yp=km.fit_predict(df)
yp
```

```
Out[14]: array([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, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,  
                0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 2, 0, 0, 0, 0, 0, 2, 0, 0, 0, 0,  
                0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 2, 2, 2, 2, 2, 2, 0, 2, 2, 2, 2,  
                2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,  
                2, 2, 2, 2, 2, 2, 0, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2])
```

```
In [15]: df['cluster']=yp
df.head(2)
```

petal length (cm)	petal width (cm)	cluster	
0	1.4	0.2	1
1	1.4	0.2	1

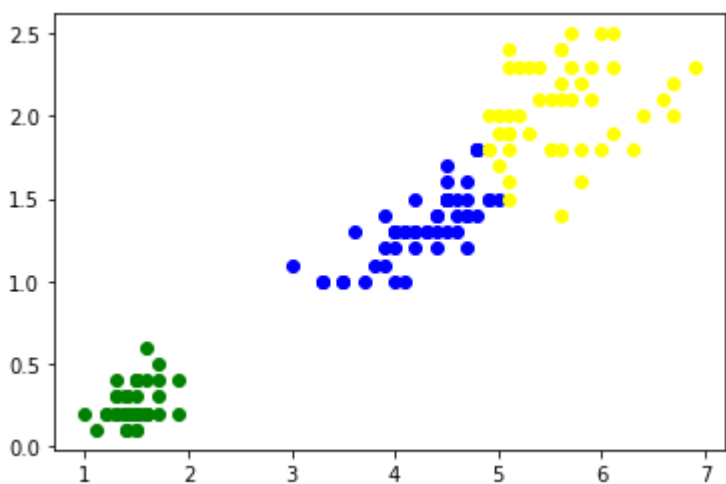
```
In [16]: df.cluster.unique()
```

```
Out[16]: array([1, 0, 2])
```

```
In [18]: df1=df[df.cluster==0]
df2=df[df.cluster==1]
df3=df[df.cluster==2]
```

```
In [20]: plt.scatter(df1['petal length (cm)'],df1['petal width (cm)'],color='blue')
plt.scatter(df2['petal length (cm)'],df2['petal width (cm)'],color='green')
plt.scatter(df3['petal length (cm)'],df3['petal width (cm)'],color='yellow')
```

```
Out[20]: <matplotlib.collections.PathCollection at 0x24f965b4f70>
```

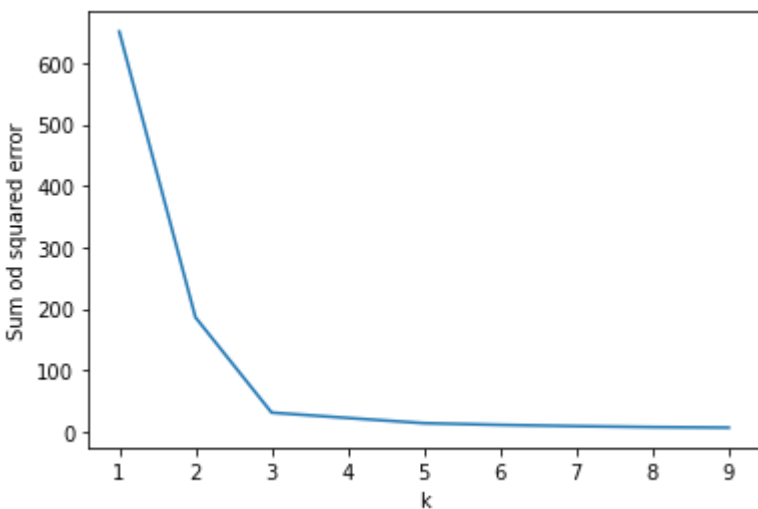


```
In [21]: sse = []
          k_rng = range(1,10)
          for k in k_rng:
              km=KMeans(n_clusters=k)
              km.fit(df)
              sse.append(km.inertia_)
```

```
C:\Users\HP\anaconda3\lib\site-packages\sklearn\cluster\_kmeans.py:881: 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(
```

```
In [22]: plt.xlabel('k')
plt.ylabel('Sum od squared error')
plt.plot(k_rng,sse)
```

```
Out[22]: []
```



In [ ]: