```
import numpy as np
import matplotlib.pyplot as plt
from sklearn.cluster import KMeans
import seaborn as sns
import pandas as pd
%matplotlib inline
```

```
df = sns.load_dataset('iris')
df.head()
```

	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

```
df['species'],categories =pd.factorize(df['species'])
df.head()
```

	sepal_length	sepal_width	petal_length	petal_width	species
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

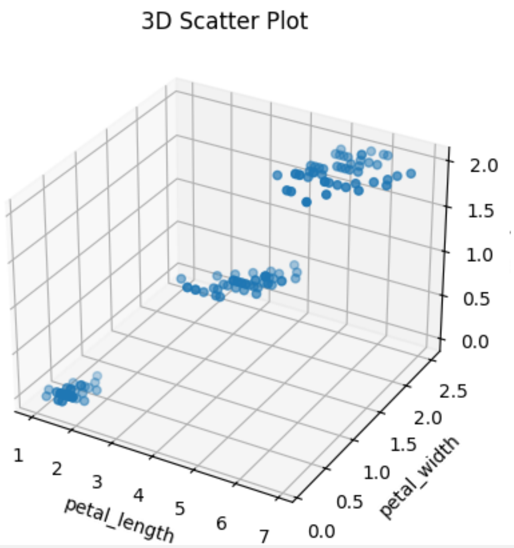
```
df.describe()
```

	sepal_length	sepal_width	petal_length	petal_width	species
count	150.000000	150.000000	150.000000	150.000000	150.000000
mean	5.843333	3.057333	3.758000	1.199333	1.000000
std	0.828066	0.435866	1.765298	0.762238	0.819232
min	4.300000	2.000000	1.000000	0.100000	0.000000
25%	5.100000	2.800000	1.600000	0.300000	0.000000
50%	5.800000	3.000000	4.350000	1.300000	1.000000
75%	6.400000	3.300000	5.100000	1.800000	2.000000
max	7.900000	4.400000	6.900000	2.500000	2.000000

```
df.isna().sum()
```

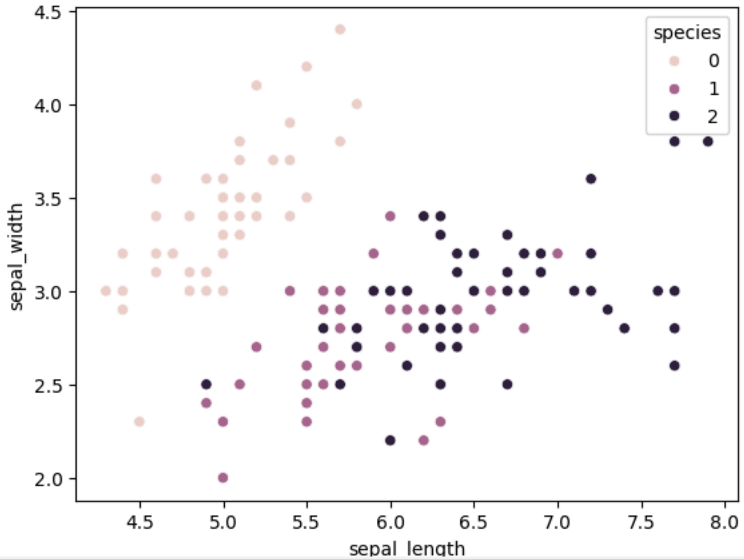
```
sepal_length    0
sepal_width     0
petal_length    0
petal_width     0
species         0
dtype: int64
```

```
from mpl_toolkits.mplot3d import Axes3D
fig = plt.figure()
ax = fig.add_subplot(111, projection='3d')
ax.scatter(df.petal_length, df.petal_width, df.species)
ax.set_xlabel('petal_length')
ax.set_ylabel('petal_width')
ax.set_zlabel('species')
plt.title('3D Scatter Plot')
plt.show()
```



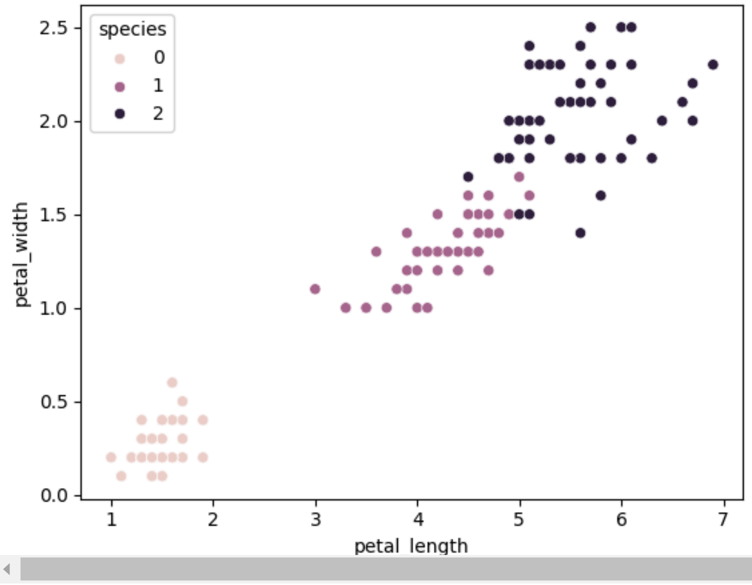
```
sns.scatterplot(data=df, x='sepal_length', y='sepal_width', hue='species')
```

```
<Axes: xlabel='sepal_length', ylabel='sepal_width'>
```



```
sns.scatterplot(data=df, x='petal_length', y='petal_width', hue='species')
```

```
<Axes: xlabel='petal_length', ylabel='petal_width'>
```



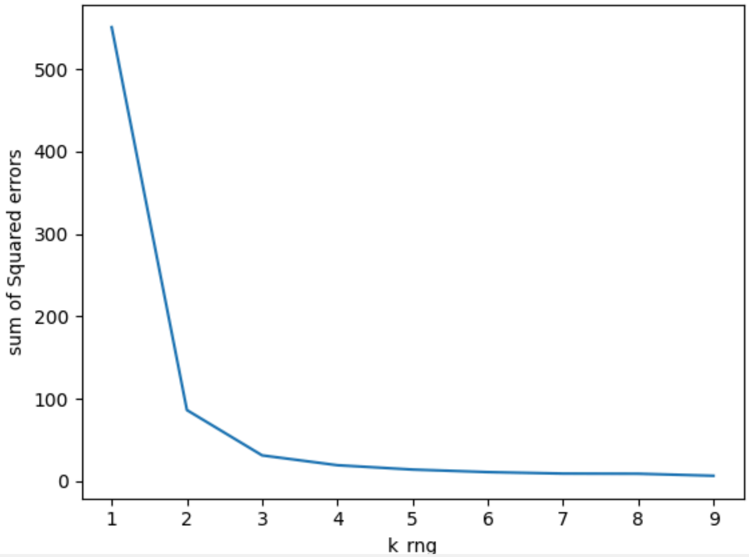
```
k_rng = range(1,10)
sse=[]


for k in k_rng:
    km = KMeans(n_clusters=k)
    km.fit(df[['petal_length', 'petal_width']])
    sse.append(km.inertia_)
```

```
sse
```

```
[550.8953333333334,
 86.39021984551395,
 31.412885668276978,
 19.483000899685116,
 14.200320553539019,
 11.13014340156125,
 9.328327985739753,
 9.149253588128271,
 6.676325163398694]
```

```
plt.xlabel('k_rng')
plt.ylabel("sum of Squared errors")
plt.plot(k_rng,sse)
```

```
[<matplotlib.lines.Line2D at 0x1cae4efbfb0>]
```



```
km = KMeans(n_clusters=3,random_state=0)
y_predicted = km.fit_predict(df[['petal_length', 'petal_width']])
y_predicted
```

```
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, 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, 0, 2, 0, 0, 0, 0, 0, 2, 0, 0, 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, 0, 2, 2, 2, 2, 2, 2, 0, 2, 2, 2, 2, 2,
       2, 2, 2, 2, 2, 2, 0, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2])
```

```
df['cluster'] = y_predicted
df.head(150)
```

	sepal_length	sepal_width	petal_length	petal_width	species	cluster
0	5.1	3.5	1.4	0.2	0	1
1	4.9	3.0	1.4	0.2	0	1
2	4.7	3.2	1.3	0.2	0	1
3	4.6	3.1	1.5	0.2	0	1
4	5.0	3.6	1.4	0.2	0	1
...	...	...	...	...	...	...
145	6.7	3.0	5.2	2.3	2	2
146	6.3	2.5	5.0	1.9	2	2
147	6.5	3.0	5.2	2.0	2	2
148	6.2	3.4	5.4	2.3	2	2
149	5.9	3.0	5.1	1.8	2	2

150 rows × 6 columns

```
from sklearn.metrics import confusion_matrix
cm = confusion_matrix(df['species'],df['cluster'])
cm
```

```
array([[ 0, 50,  0],
       [48,  0,  2],
       [ 4,  0, 46]], dtype=int64)
```

```
true_labels = df['species']
predicted_labels = df['cluster']
cm = confusion_matrix(true_labels, predicted_labels)
class_labels = ['Setosa', 'versicolor','virginica']
plt.imshow(cm, interpolation='nearest', cmap=plt.cm.Blues)
plt.title('Confusion Matrix')
plt.colorbar
Tick_marks = np.arange(len(class_labels))
plt.xticks(Tick_marks, class_labels, rotation=45)
plt.yticks(Tick_marks, class_labels)
plt.xlabel('Predicted Labels')
plt.ylabel('True Labels')
```