```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

```
data=pd.read_csv("/content/sample_data/Iris.csv")
```

```
data.head()
```

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	2	4.9	3.0	1.4	0.2	Iris-setosa
2	3	4.7	3.2	1.3	0.2	Iris-setosa
3	4	4.6	3.1	1.5	0.2	Iris-setosa
4	5	5.0	3.6	1.4	0.2	Iris-setosa

```
data=data.drop(['Id'],axis=1)
```

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150 entries, 0 to 149
Data columns (total 5 columns):
 #   Column         Non-Null Count  Dtype  
---  ------         --------------  -----  
 0   SepalLengthCm  150 non-null    float64
 1   SepalWidthCm   150 non-null    float64
 2   PetalLengthCm  150 non-null    float64
 3   PetalWidthCm   150 non-null    float64
 4   Species        150 non-null    object 
dtypes: float64(4), object(1)
memory usage: 6.0+ KB
```

```
data.Species.value_counts()
```

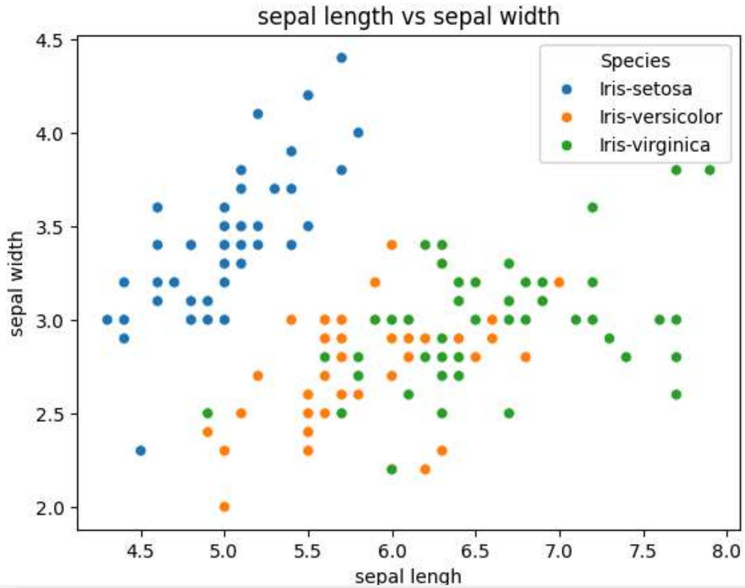
	count
Species	
Iris-setosa	50
Iris-versicolor	50
Iris-virginica	50

```
df=data['Species']
df
```

	Species
0	Iris-setosa
1	Iris-setosa
2	Iris-setosa
3	Iris-setosa
4	Iris-setosa
...	...
145	Iris-virginica
146	Iris-virginica
147	Iris-virginica
148	Iris-virginica
149	Iris-virginica

150 rows × 1 columns

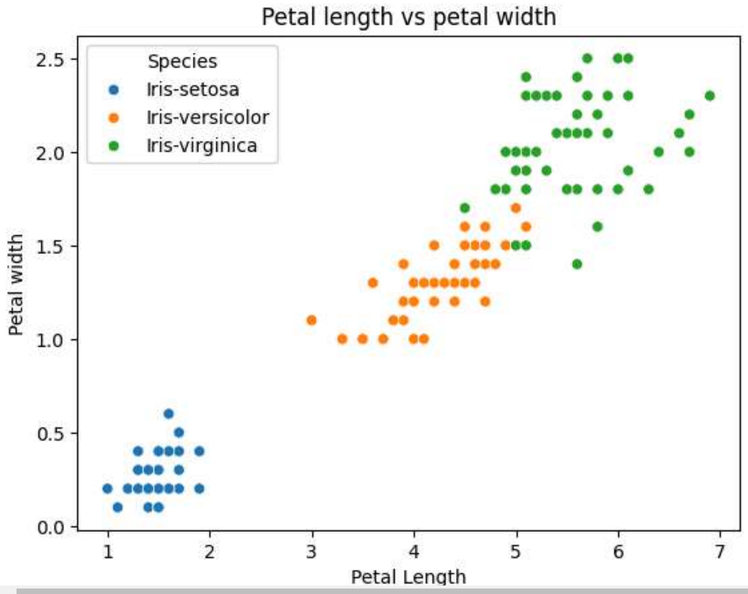
```
sns.scatterplot(x='SepalLengthCm',y='SepalWidthCm',data=data,hue='Species')
plt.xlabel('sepal lengh')
plt.ylabel('sepal width')
plt.title('sepal length vs sepal width')
plt.show()
```



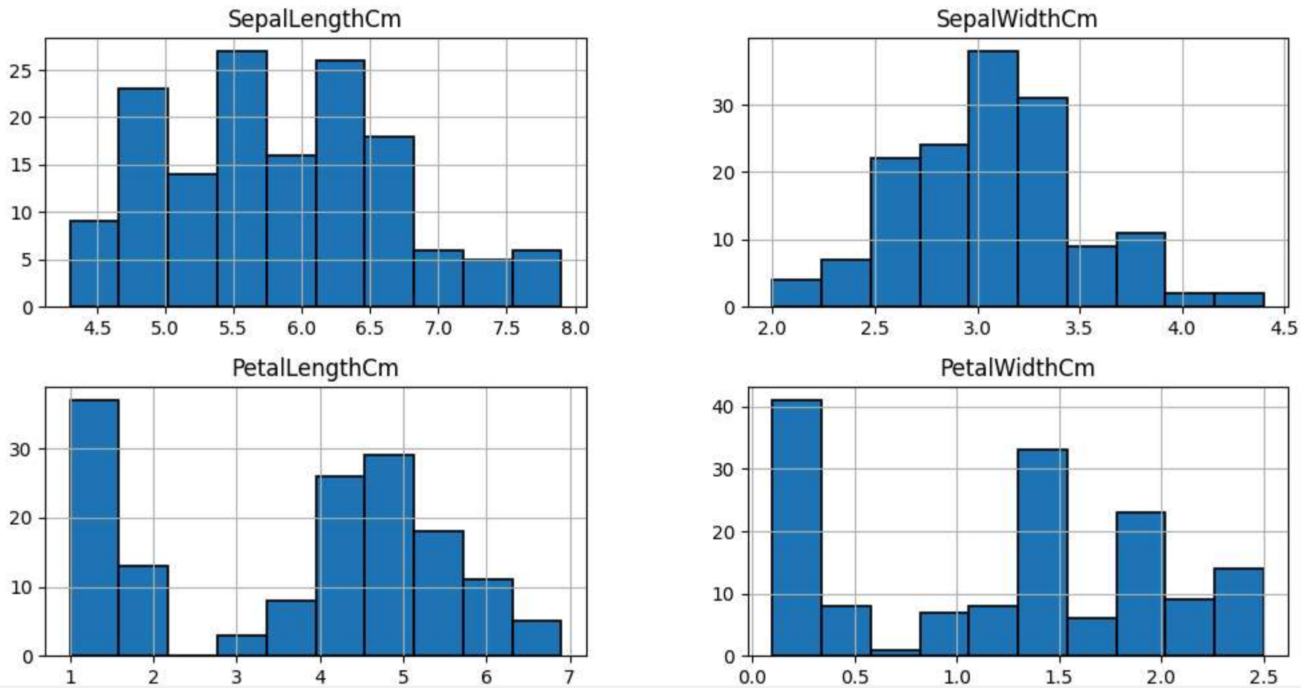
```
data.columns
```

```
Index(['SepalLengthCm', 'SepalWidthCm', 'PetalLengthCm', 'PetalWidthCm',
       'Species'],
      dtype='object')
```

```
sns.scatterplot(data=data,x='PetalLengthCm',y='PetalWidthCm',hue='Species')
plt.xlabel('Petal Length')
plt.ylabel('Petal width')
plt.title('Petal length vs petal width')
plt.show()
```



```
data.hist(edgecolor='black', linewidth=1.2)
fig=plt.gcf()
fig.set_size_inches(12,6)
plt.show()
```



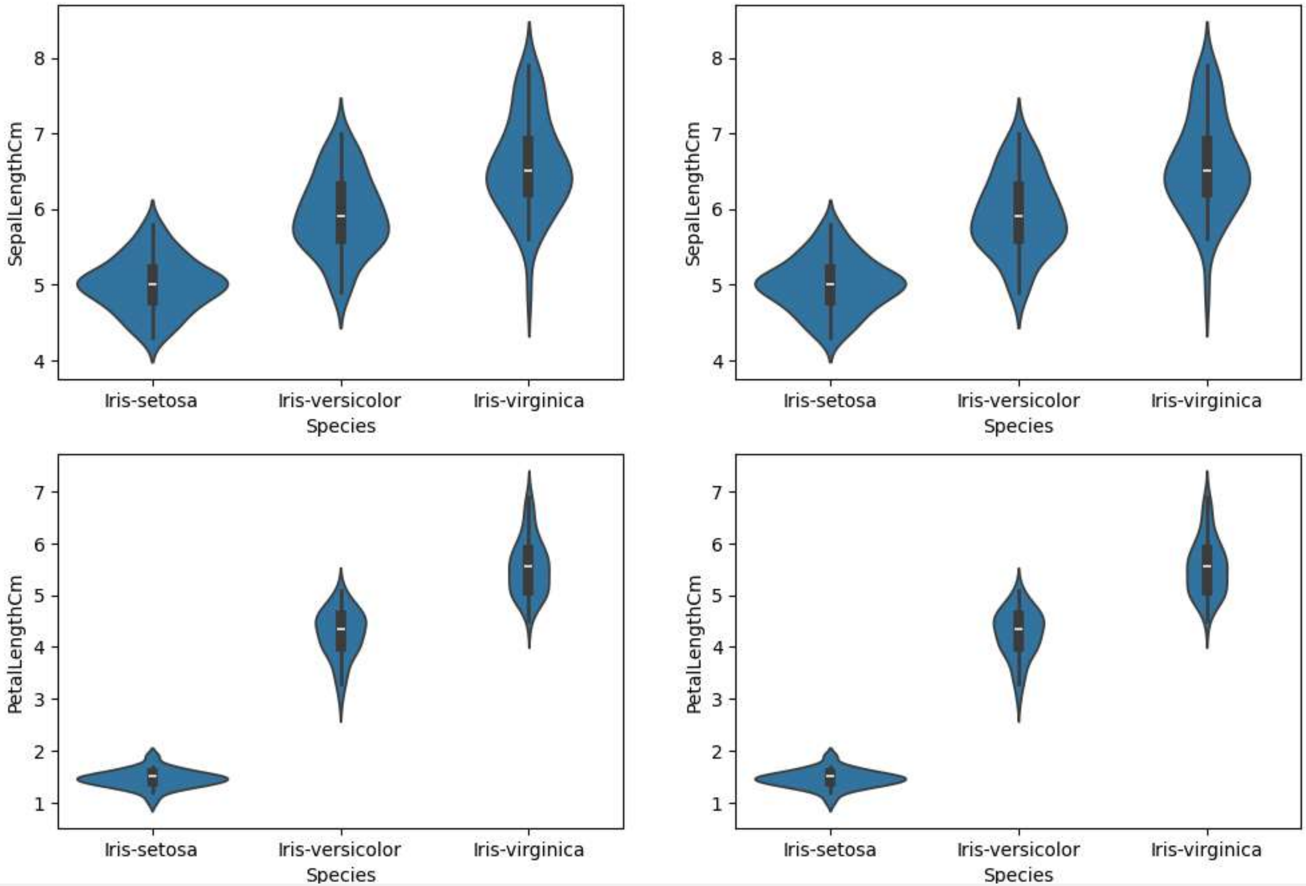
```
data.isna().sum()
```

```
                 0
SepalLengthCm    0
SepalWidthCm     0
PetalLengthCm    0
PetalWidthCm     0
Species          0

dtype: int64
```

```
plt.figure(figsize=(12,8))
plt.subplot(2,2,1)
sns.violinplot(x='Species',y='SepalLengthCm',data=data)
plt.subplot(2,2,2)
sns.violinplot(x='Species',y='SepalLengthCm',data=data)
plt.subplot(2,2,3)
sns.violinplot(x='Species',y='PetalLengthCm',data=data)
plt.subplot(2,2,4)
sns.violinplot(x='Species',y='PetalLengthCm',data=data)
```

```
<Axes: xlabel='Species', ylabel='PetalLengthCm'>
```



```
x=data.drop('Species',axis=1)
y=data['Species']
```

```
from sklearn.model_selection import train_test_split
```

```
xtrain,xtest,ytrain,ytest=train_test_split(x,y,test_size=0.3,random_state=42)
```

```
# logistice regression
from sklearn.linear_model import LogisticRegression
```

```
lr=LogisticRegression()
```

```
lr.fit(xtrain,ytrain)
```

```
▾ LogisticRegression
LogisticRegression()
```

```
ypred=lr.predict(xtest)
```

```
from sklearn.metrics import *
```

```
accuracy_score(ytest,ypred)
```

```
1.0
```

```
lr_df=pd.DataFrame()
lr_df['y_actual']=ytest
lr_df['y_model']=ypred
lr_df
```

	y_actual	y_model
73	Iris-versicolor	Iris-versicolor
18	Iris-setosa	Iris-setosa
118	Iris-virginica	Iris-virginica
78	Iris-versicolor	Iris-versicolor
76	Iris-versicolor	Iris-versicolor
31	Iris-setosa	Iris-setosa
64	Iris-versicolor	Iris-versicolor
141	Iris-virginica	Iris-virginica
68	Iris-versicolor	Iris-versicolor
82	Iris-versicolor	Iris-versicolor
110	Iris-virginica	Iris-virginica
12	Iris-setosa	Iris-setosa
36	Iris-setosa	Iris-setosa
9	Iris-setosa	Iris-setosa
19	Iris-setosa	Iris-setosa
56	Iris-versicolor	Iris-versicolor
104	Iris-virginica	Iris-virginica
69	Iris-versicolor	Iris-versicolor
55	Iris-versicolor	Iris-versicolor
132	Iris-virginica	Iris-virginica
29	Iris-setosa	Iris-setosa
127	Iris-virginica	Iris-virginica
26	Iris-setosa	Iris-setosa
128	Iris-virginica	Iris-virginica
131	Iris-virginica	Iris-virginica
145	Iris-virginica	Iris-virginica
108	Iris-virginica	Iris-virginica
143	Iris-virginica	Iris-virginica
45	Iris-setosa	Iris-setosa
30	Iris-setosa	Iris-setosa
22	Iris-setosa	Iris-setosa
15	Iris-setosa	Iris-setosa
65	Iris-versicolor	Iris-versicolor
11	Iris-setosa	Iris-setosa
42	Iris-setosa	Iris-setosa
146	Iris-virginica	Iris-virginica
51	Iris-versicolor	Iris-versicolor
27	Iris-setosa	Iris-setosa
4	Iris-setosa	Iris-setosa
32	Iris-setosa	Iris-setosa
142	Iris-virginica	Iris-virginica
85	Iris-versicolor	Iris-versicolor
86	Iris-versicolor	Iris-versicolor
16	Iris-setosa	Iris-setosa
10	Iris-setosa	Iris-setosa

```
#decision tree
from sklearn.tree import DecisionTreeClassifier
```

```
dt=DecisionTreeClassifier()
```

```
dt.fit(xtrain,ytrain)
```

```
▾ DecisionTreeClassifier
DecisionTreeClassifier()
```

```
ypred_dt=dt.predict(xtest)
```

```
accuracy_score(ytest,ypred_dt)
```

```
1.0
```

```
# random forest classifier
from sklearn.ensemble import RandomForestClassifier
rfc=RandomForestClassifier()
```

```
rfc.fit(xtrain,ytrain)
```

```
▾ RandomForestClassifier
RandomForestClassifier()
```

```
ypred_rfc=rfc.predict(xtest)
```

```
accuracy_score(ytest,ypred_rfc)
```

```
1.0
```

```
rfc_df=pd.DataFrame()
rfc_df['yactual']=ytest
rfc_df['ymodeled']=ypred_rfc
rfc_df
```

	yactual	ymodeled
73	Iris-versicolor	Iris-versicolor
18	Iris-setosa	Iris-setosa
118	Iris-virginica	Iris-virginica
78	Iris-versicolor	Iris-versicolor
76	Iris-versicolor	Iris-versicolor
31	Iris-setosa	Iris-setosa
64	Iris-versicolor	Iris-versicolor
141	Iris-virginica	Iris-virginica
68	Iris-versicolor	Iris-versicolor