```
import numpy as np
import pandas as pd
from sklearn.neighbors import KNeighborsClassifier
from google.colab import files
from sklearn.model_selection import train_test_split
from sklearn import metrics
import io


names = ['sepal-length', 'sepal-width', 'petal-length', 'petal-width', 'Class']


uploaded = files.upload()
```

```
Choose Files 8-dataset.csv
• 8-dataset.csv(text/csv) - 4558 bytes, last modified: 2/8/2025 - 100% done
Saving 8-dataset.csv to 8-dataset.csv
```

```
dataset = pd.read_csv(io.BytesIO(uploaded['8-dataset.csv']))
X = dataset.iloc[:, :-1]
y = dataset.iloc[:, -1]
print(X.head())
Xtrain, Xtest, ytrain, ytest = train_test_split(X, y, test_size=0.10)
```

```
   5.1  3.5  1.4  0.2
0  4.9  3.0  1.4  0.2
1  4.7  3.2  1.3  0.2
2  4.6  3.1  1.5  0.2
3  5.0  3.6  1.4  0.2
4  5.4  3.9  1.7  0.4
```

```
classifier = KNeighborsClassifier(n_neighbors=5).fit(Xtrain, ytrain)


ypred = classifier.predict(Xtest)
i = 0


print ("\n-------------------------------------------------------------------------")
print ('%-25s %-25s %-25s' % ('Original Label', 'Predicted Label', 'Correct/Wrong'))
print ("-------------------------------------------------------------------------")
for label in ytest:
    print ('%-25s %-25s' % (label, ypred[i]), end="")
    if (label == ypred[i]):
        print (' %-25s' % ('Correct'))
    else:
        print (' %-25s' % ('Wrong'))
    i = i + 1
```

```
-------------------------------------------------------------------------
Original Label            Predicted Label           Correct/Wrong
-------------------------------------------------------------------------
Iris-setosa               Iris-setosa               Correct
Iris-versicolor           Iris-versicolor           Correct
Iris-setosa               Iris-setosa               Correct
Iris-virginica            Iris-virginica            Correct
Iris-setosa               Iris-setosa               Correct
Iris-versicolor           Iris-versicolor           Correct
Iris-versicolor           Iris-virginica            Wrong
Iris-versicolor           Iris-versicolor           Correct
Iris-versicolor           Iris-versicolor           Correct
Iris-versicolor           Iris-versicolor           Correct
Iris-virginica            Iris-virginica            Correct
Iris-virginica            Iris-virginica            Correct
Iris-versicolor           Iris-versicolor           Correct
Iris-setosa               Iris-setosa               Correct
Iris-versicolor           Iris-versicolor           Correct
```

```
print ("-------------------------------------------------------------------------")
print("\nConfusion Matrix:\n",metrics.confusion_matrix(ytest, ypred))
print ("-------------------------------------------------------------------------")
print("\nClassification Report:\n",metrics.classification_report(ytest, ypred))
print ("-------------------------------------------------------------------------")
print('Accuracy of the classifer is %0.2f' % metrics.accuracy_score(ytest,ypred))
print ("-------------------------------------------------------------------------")
```

```
--------------------------------------------------------------------------

Confusion Matrix:
 [[4 0 0]
 [0 7 1]
 [0 0 3]]
--------------------------------------------------------------------------

Classification Report:
                  precision    recall  f1-score   support

    Iris-setosa       1.00      1.00      1.00         4
Iris-versicolor       1.00      0.88      0.93         8
 Iris-virginica       0.75      1.00      0.86         3

       accuracy                           0.93        15
      macro avg       0.92      0.96      0.93        15
   weighted avg       0.95      0.93      0.94        15

--------------------------------------------------------------------------
Accuracy of the classifer is 0.93
--------------------------------------------------------------------------
```