

```
import numpy as np
from sklearn.datasets import load_iris
#Importamos la funcion que nos ayudara a separar nuestro dataset en entrenamiento y prueba
from sklearn.model_selection import train_test_split
#Importamos el modelo de knn
from sklearn.neighbors import KNeighborsClassifier
```

```
#Cargamos el dataset de iris
iris = load_iris()
#Imprimimos los key para acceder a los datos del dataset
iris.keys()
```

```
↳ dict_keys(['data', 'target', 'target_names', 'DESCR', 'feature_names'])
```

```
#Separamos el dataset en entrenamiento y prueba
X_train, X_test, Y_train, Y_test = train_test_split(iris['data'], iris['target'])
```

```
#Definimos el modelo con una kernel de 2
knn = KNeighborsClassifier(n_neighbors=2)
#Entrenamos el modelo con los datos de entrenamiento
knn.fit(X_train, Y_train)
#Mostramos la presicion del modelo
knn.score(X_test, Y_test)
```

```
↳ 0.9736842105263158
```

```
print(X_test[3])
print("Los datos anteriores pertenecen a la clase: ",Y_test[3])
```

```
#Hacemos una prueba con un dato al azar del dataset
p = knn.predict([X_test[3]])
print("El modelo predijo: ", p).
```

```
↳ [6.4 2.7 5.3 1.9]
Los datos anteriores pertenecen a la clase: 2
El modelo predijo: [2]
```

