In [23]:
```
import math
import pandas as pd
from collections import Counter
from sklearn.datasets import load_iris
```

In [24]:
```
#Load iris dataset
iris = load_iris()
X = iris.data.tolist()
y = iris.target.tolist()
```

In [25]:
```
# Combine features
data = [X[i] + [y[i]] for i in range(len(y))]
```

In [26]:
```
df = pd.DataFrame(data)
df.head()
```

Out[26]:

	0	1	2	3	4
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 [27]:
```
def euclidean_distance(point1, point2):
    distance = 0
    for i in range(len(point1)):
        distance += (point1[i] - point2[i]) ** 2
    return math.sqrt(distance)
```

In [28]:
```
def knn(train_data, test_point, k=3):
    distances = []
    for data_point in train_data:
        features = data_point[:-1]
        label = data_point[-1]
        distance = euclidean_distance(features, test_point)
        distances.append((distance, label))

    distances.sort(key=lambda x: x[0])
    neighbors_labels = [label for _, label in distances[:k]]
    most_common = Counter(neighbors_labels).most_common(1)
    return most_common[0][0]
```

In [29]:
```
#Test new data
test_point = X[10]
predicted_label = knn(data, test_point, k=3)

print(f"Test point: {test_point}")
```

```
print(f"Actual label: {y[10]}")
print(f"Predicted label: {predicted_label}")
```

```
Test point: [5.4, 3.7, 1.5, 0.2]
Actual label: 0
Predicted label: 0
```

In [ ]: