import pandas as pd

from sklearn.tree import DecisionTreeClassifier

from sklearn.model\_selection import train\_test\_split

from sklearn.metrics import accuracy\_score

# Sample dataset (you should replace this with real clinical data)

data = {

'Age': [28, 55, 35, 70, 45, 60, 33, 40],

'PainLevel': [7, 4, 8, 9, 5, 6, 2, 7], # 0-10 scale

'Fever': [1, 0, 1, 1, 0, 1, 0, 1], # 1 = yes, 0 = no

'WBC\_Count': [12000, 8000, 15000, 18000, 9000, 16000, 7000, 14000], # White blood cell count

'HeartRate': [98, 85, 110, 120, 88, 115, 80, 105],

'Need\_Colostomy': [1, 0, 1, 1, 0, 1, 0, 1] # 1 = Yes, 0 = No

}

# Convert to DataFrame

df = pd.DataFrame(data)

# Features and target

X = df[['Age', 'PainLevel', 'Fever', 'WBC\_Count', 'HeartRate']]

y = df['Need\_Colostomy']

# Split the data

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.3, random\_state=42)

# Train a Decision Tree Classifier

clf = DecisionTreeClassifier()

clf.fit(X\_train, y\_train)

# Predict

y\_pred = clf.predict(X\_test)

# Accuracy

print("Model Accuracy:", accuracy\_score(y\_test, y\_pred))

# Predict for a new patient

new\_patient = pd.DataFrame({

'Age': [50],

'PainLevel': [8],

'Fever': [1],

'WBC\_Count': [16000],

'HeartRate': [110]

})

prediction = clf.predict(new\_patient)

print("Predicted: Need Colostomy" if prediction[0] == 1 else "Predicted: No Colostomy Needed")