

```

import numpy as np
from sklearn.neighbors import KNeighborsClassifier
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score
import pandas as pd

data = {
    'fever': [1, 1, 0, 0, 1, 0, 1, 0, 1, 0],
    'cough': [1, 1, 0, 0, 1, 0, 1, 1, 1, 0],
    'fatigue': [1, 1, 1, 0, 1, 0, 0, 0, 1, 0],
    'headache': [1, 0, 0, 0, 1, 0, 1, 1, 0, 0],
    'shortness_of_breath': [1, 0, 0, 0, 1, 0, 1, 0, 1, 0],
    'condition': [1, 1, 0, 0, 1, 0, 1, 0, 1, 0]
}

df = pd.DataFrame(data)

X = df.drop('condition', axis=1)

y = df['condition']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)

knn = KNeighborsClassifier()

knn.fit(X_train, y_train)

y_pred = knn.predict(X_test)
print(f"Model Accuracy: {accuracy_score(y_test, y_pred) * 100:.2f}%")

print("\nEnter the symptoms for the new patient:")
fever = int(input("Fever (1 for yes, 0 for no): "))
cough = int(input("Cough (1 for yes, 0 for no): "))
fatigue = int(input("Fatigue (1 for yes, 0 for no): "))
headache = int(input("Headache (1 for yes, 0 for no): "))
shortness_of_breath = int(input("Shortness of breath (1 for yes, 0 for no): "))

new_patient = np.array([[fever, cough, fatigue, headache, shortness_of_breath]])

k = int(input("\nEnter the number of neighbors (k): "))

knn = KNeighborsClassifier(n_neighbors=k)

knn.fit(X, y)

prediction = knn.predict(new_patient)

if prediction[0] == 1:
    print("\nThe patient is predicted to have the medical condition (1).")
else:
    print("\nThe patient is predicted to NOT have the medical condition (0).")

```

Model Accuracy: 66.67%

Enter the symptoms for the new patient:

Fever (1 for yes, 0 for no): 1

Cough (1 for yes, 0 for no): 0

Fatigue (1 for yes, 0 for no): 1

Headache (1 for yes, 0 for no): 1

Shortness of breath (1 for yes, 0 for no): 0

Enter the number of neighbors (k): 3

The patient is predicted to have the medical condition (1).