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

import pandas as pd

from sklearn.linear\_model import LogisticRegression

from sklearn.preprocessing import StandardScaler, LabelEncoder

from sklearn.metrics import accuracy\_score, classification\_report

from sklearn.metrics import confusion\_matrix, ConfusionMatrixDisplay

import matplotlib.pyplot as plt

df = pd.read\_csv("ex3.csv")

df['gender'] = LabelEncoder().fit\_transform(df['gender'])

X = df[['age', 'gender', 'bmi', 'blood\_pressure', 'cholestrol']]

y = df['condition']

scaler = StandardScaler()

X\_scaled = scaler.fit\_transform(X)

X\_train, X\_test, y\_train, y\_test = train\_test\_split(

    X\_scaled, y, test\_size=0.2, random\_state=42

)

model = LogisticRegression()

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

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

y\_prob = model.predict\_proba(X\_test)[:, 1]

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

print("Classification report:\n", classification\_report(y\_test, y\_pred, zero\_division=1))

cm = confusion\_matrix(y\_test, y\_pred)

disp = ConfusionMatrixDisplay(confusion\_matrix=cm)

disp.plot(cmap='Blues')

plt.title("Confusion Matrix")

plt.show()

new\_data = pd.DataFrame([[60, 1, 27, 130, 200]], columns=['age', 'gender', 'bmi', 'blood\_pressure', 'cholesterol'])

new\_scaled = scaler.transform(new\_data)

new\_prob = model.predict\_proba(new\_scaled)[0][1]

print(f"Probability of developing the condition: {new\_prob:.2f}")