

Program:

```
import numpy as np

import pandas as pd

import seaborn as sns

import matplotlib.pyplot as plt

from sklearn.linear_model import LogisticRegression

from sklearn.model_selection import train_test_split

from sklearn.preprocessing import LabelEncoder, StandardScaler

from sklearn.metrics import confusion_matrix, ConfusionMatrixDisplay,
accuracy_score, classification_report


data = pd.read_csv("bmi.csv")

print(data.head())

print(data.columns)


data['gender'] = LabelEncoder().fit_transform(data['gender'])


x = data[['age', 'gender', 'bmi', 'blood_pressure', 'cholesterol']]
y = data['condition']


scaler = StandardScaler()

xscale = scaler.fit_transform(x)


xtr, xte, ytr, yte = train_test_split(xscale, y, test_size=0.2, random_state=42)


model = LogisticRegression()

model.fit(xtr, ytr)
```

```
ypr = model.predict(xte)
yprob = model.predict_proba(xte)[:, 1]

print("Accuracy:", accuracy_score(yte, ypr))
print("Classification Report:\n", classification_report(yte, ypr, zero_division=1))

cm = confusion_matrix(yte, ypr)
disp = ConfusionMatrixDisplay(confusion_matrix=cm)
disp.plot(cmap='Blues')
plt.title("Confusion Matrix")
plt.show()

new = pd.DataFrame([[60, 1, 27, 130, 200]], columns=['age', 'gender', 'bmi',
'blood_pressure', 'cholesterol'])
newscale = scaler.transform(new)
newcondition = model.predict_proba(newscale)[0][1]
print(f"Probability of developing the condition: {newcondition:.2f}")
```

Output:

