In [1]:
```
import pandas as pd
```

In [2]:
```
df = pd.read_csv('diabetes.csv')
```

In [3]:
```
df.head()
```

Out[3]:

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeF
0	6	148	72	35	0	33.6	
1	1	85	66	29	0	26.6	
2	8	183	64	0	0	23.3	
3	1	89	66	23	94	28.1	
4	0	137	40	35	168	43.1	

In [4]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 768 entries, 0 to 767
Data columns (total 9 columns):
 #   Column                    Non-Null Count  Dtype  
---  ------                    --------------  -----  
 0   Pregnancies               768 non-null    int64  
 1   Glucose                   768 non-null    int64  
 2   BloodPressure             768 non-null    int64  
 3   SkinThickness             768 non-null    int64  
 4   Insulin                   768 non-null    int64  
 5   BMI                       768 non-null    float64
 6   DiabetesPedigreeFunction  768 non-null    float64
 7   Age                       768 non-null    int64  
 8   Outcome                   768 non-null    int64  
dtypes: float64(2), int64(7)
memory usage: 54.1 KB
```

In [5]:
```
df.shape
```

Out[5]:
```
(768, 9)
```

In [6]:
```
# removing Outcome and storing it in different variable
x = df.drop('Outcome',axis=1)
```

In [7]:
```
x.head()
```

Out[7]:

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeF
0	6	148	72	35	0	33.6	
1	1	85	66	29	0	26.6	
2	8	183	64	0	0	23.3	
3	1	89	66	23	94	28.1	
4	0	137	40	35	168	43.1	

In [8]:
```
# storing the colmn Outcome in y(target)
y = df['Outcome']
```

In [9]:
```
y.head()
```

Out[9]:
```
0    1
1    0
2    1
3    0
4    1
Name: Outcome, dtype: int64
```

In [10]:
```
from sklearn.model_selection import train_test_split
```

In [11]:
```
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.3) # setting te
```

In [12]:
```
x_train.shape
```

Out[12]:
```
(537, 8)
```

In [13]:
```
x_test.shape
```

Out[13]:
```
(231, 8)
```

In [14]:
```
from sklearn.tree import DecisionTreeClassifier
```

In [18]:
```
model = DecisionTreeClassifier()
```

In [19]:
```
model.fit(x_train, y_train)
```

Out[19]:
```
DecisionTreeClassifier()
```

In [20]:
```
y_predict = model.predict(x_test)
```

In [21]:
```
y_test
```

Out[21]:
```
626    0
583    0
700    0
314    1
504    0
      ..
577    1
518    0
104    0
223    0
63     0
Name: Outcome, Length: 231, dtype: int64
```

In [22]:
```
from sklearn import metrics
```

In [23]:
```
print(metrics.accuracy_score(y_test,y_predict))
```

```
0.6926406926406926
```

In [24]:
```
df.head()
```

Out[24]:

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeF
0	6	148	72	35	0	33.6	
1	1	85	66	29	0	26.6	
2	8	183	64	0	0	23.3	
3	1	89	66	23	94	28.1	
4	0	137	40	35	168	43.1	

In [37]:
```
print("Enter your following report data: ")
print()
data=[]
l=["Pregnancies","Glucose","BloodPressure","SkinThickness","Insulin","BMI","Diab
i=0
for c in l:
    if i==5 or i==6:
        user=float(input(f"Enter {l[i]}: \n"))
        i+=1
    else:
        user=int(input(f"Enter {l[i]}: \n"))
        i+=1
    data.append(user)
if model.predict([data])[0] == 1:
  print("\n\nHaving diabetes")
else:
  print("\n\nNot having diabetes")
```

```
Enter your following report data:


Not having diabetes
```

```
C:\Users\Abhishek Raj\AppData\Local\Programs\Python\Python312\Lib\site-packages\s
klearn\base.py:493: UserWarning: X does not have valid feature names, but Decisio
nTreeClassifier was fitted with feature names
  warnings.warn(
```

In [ ]:

In [ ]: