

PROGRAM :

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
dataset=pd.read_csv("C:/Users/STUDENT/Desktop/diabetes.csv")
dataset.head()
dataset.shape
dataset.isnull().values.any()
dataset.info()
dataset.describe()
dataset.isnull().sum()
sns.countplot(x = 'Outcome',data =dataset)
sns.pairplot(data = dataset, hue = 'Outcome')
plt.show()
sns.heatmap(dataset.corr(), annot = True)
plt.show()
```

```
dataset_new = dataset
dataset_new[['Glucose', 'BloodPressure', 'SkinThickness',
'Insulin', 'BMI']] = dataset_new[['Glucose',
'BloodPressure', 'SkinThickness', 'Insulin',
'BMI']].replace(0,np.NaN)
dataset_new.isnull().sum()
dataset_new['Glucose'].fillna(dataset_new['Glucose'].mean(),inplace=True)
dataset_new['BloodPressure'].fillna(dataset_new['Blood Pressure'].mean(),inplace = True)
dataset_new['SkinThickness'].fillna(dataset_new['SkinThickness'].mean(),inplace = True)
dataset_new['Insulin'].fillna(dataset_new['Insulin'].mean(),inplace = True)
dataset_new['BMI'].fillna(dataset_new['BMI'].mean(),inplace = True)
dataset_new.isnull().sum()
y = dataset_new['Outcome']
X = dataset_new.drop('Outcome', axis=1)
from sklearn.model_selection import train_test_split
```

```
X_train, X_test, Y_train, Y_test = train_test_split(X, y,
test_size = 0.20, random_state = 42, stratify =
dataset_new['Outcome'] )

from sklearn.linear_model import LogisticRegression
Model = LogisticRegression()
Model.fit(X_train, Y_train)
Y_predict = Model.predict(X_test)
Y_predict

from sklearn.metrics import confusion_matrix
Cm = confusion_matrix(Y_test, Y_predict)
Cm

sns.heatmap(pd.DataFrame(Cm),annot=True)

from sklearn.metrics import accuracy_score
accuracy =accuracy_score(Y_test, Y_predict)
accuracy

Y_predict =
Model.predict([[1,148,72,35,79.799,33.6,0.627,50]])
print(Y_predict)

if Y_predict==1:
```

```
print('Diabetes')
```

```
else:
```

```
    print('Non Diabeties')
```

OUTPUT :

```
In [18]: runfile('C:/Users/STUDENT/Downloads/diabetes.py', wdir='C:/Users/STUDENT/Downloads')
<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
[1]
Diabetes
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\linear_model\_logistic.py:762: ConvergenceWarning: lbfgs failed to converge (status=1):
STOP: TOTAL NO. of ITERATIONS REACHED LIMIT.

Increase the number of iterations (max_iter) or scale the data as shown in:
    https://scikit-learn.org/stable/modules/preprocessing.html
Please also refer to the documentation for alternative solver options:
    https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression
n_iter_i = _check_optimize_result(
```