**SAMPLE CODE:**

import numpy as n

import pandas as p

import seaborn as sns

d=p.read\_csv("Churn\_Modelling.csv")

d.drop(['RowNumber','CustomerId','Surname'], axis=1,inplace=True)

geo = p.get\_dummies(d['Geography'], drop\_first=True)

gen = p.get\_dummies(d['Gender'], drop\_first=True)

d=p.concat([d,geo,gen], axis=1)

d.drop(['Geography','Gender'], axis=1, inplace=True)

x=d.drop('Exited',axis=1)

y=d['Exited']

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

xtr, xts, ytr, yts = train\_test\_split(x,y,test\_size=0.2,random\_state=0)

print("X's train size: {},X's test size: {}".format(xtr.shape,xts.shape))

from sklearn.preprocessing import StandardScaler

sc = StandardScaler()

xtr = sc.fit\_transform(xtr)

xts = sc.transform(xts)

import keras

from keras.models import Sequential

from keras.layers import Dense

classifier = Sequential()

classifier.add(Dense(units=6, kernel\_initializer='he\_uniform', activation='relu', input\_dim=11))

classifier.add(Dense(units=6, kernel\_initializer='he\_uniform', activation='relu'))

classifier.add(Dense(units=1, kernel\_initializer='glorot\_uniform', activation='sigmoid'))

classifier.compile(optimizer='adam', loss='binary\_crossentropy', metrics=['accuracy'])

model\_history = classifier.fit(xtr, ytr, batch\_size=10, validation\_split=0.33, epochs=100)

yprd = classifier.predict(xts)

yprd = (yprd > 0.5)