```
In [18]: import numpy as np
         import matplotlib.pyplot as plt
         import pandas as pd
In [19]: df=pd.read_csv('heart.csv')
         df.head()
         df=df.drop(columns=['thal'])
Out[19]:
              age sex cp trestbps chol fbs restecg thalach exang oldpeak slope ca target
           0 63
                   1 3
                             145
                                 233
                                       1
                                               0
                                                    150
                                                            0
                                                                  2.3
                                                                         0 0
                                                                                  1
           1 37
                   1 2
                             130 250
                                       0
                                                    187
                                                                  3.5
                                                                         0 0
                                                                                  1
           2
             41
                   0
                      1
                             130
                                 204
                                       0
                                               0
                                                    172
                                                            0
                                                                  1.4
                                                                         2 0
                                                                                  1
                                                                         2 0
           3
              56
                   1 1
                             120 236
                                       0
                                               1
                                                    178
                                                            0
                                                                  8.0
                                                                                  1
           4
              57
                   0
                       0
                             120 354
                                       0
                                               1
                                                    163
                                                            1
                                                                  0.6
                                                                         2 0
                                                                                  1
         298
              57
                   0
                      0
                             140 241
                                               1
                                                    123
                                                            1
                                                                  0.2
                                                                         1 0
                                                                                  0
         299
              45
                   1
                       3
                             110 264
                                               1
                                                    132
                                                            0
                                                                  1.2
                                                                         1 0
                                                                                  0
              68
                   1 0
                             144 193
                                       1
                                               1
                                                    141
                                                            0
                                                                  3.4
                                                                         1 2
                                                                                  0
         300
         301
              57
                   1
                       0
                             130 131
                                       0
                                                    115
                                                                  1.2
                                                                         1 1
                                                                                   0
         302 57
                   0
                      1
                             130 236
                                                    174
                                                            0
                                                                  0.0
                                                                         1 1
                                                                                  0
         303 rows × 13 columns
In [20]: independent=df[["age", "sex", "cp", "trestbps", "chol", "fbs", "restecg", "thalach", "exang", "oldpeak", "slope", "ca", ]]
         dependent=df[["target"]]
         from sklearn.model_selection import train_test_split
         X_train, X_test, Y_train, Y_test=train_test_split(independent, dependent, test_size=1/3, random_state=0)
In [21]: from sklearn import linear_model
         regressor=linear_model.RidgeCV(alphas=np.logspace(-6,6,13))
         regressor.fit(X_train,Y_train)
                                                     RidgeCV
         RidgeCV(alphas=array([1.e-06, 1.e-05, 1.e-04, 1.e-03, 1.e-02, 1.e-01, 1.e+00, 1.e+01,
                 1.e+02, 1.e+03, 1.e+04, 1.e+05, 1.e+06]))
         weight=regressor.coef_
         print("Weight of the model={}".format(weight))
         bais=regressor.intercept_
         print("Bails of the model={}".format(bais))
         Weight of the model=[[0.00094047 -0.22223953 0.09940261 -0.00152358 -0.00055783 -0.06438072]
            0.00106035 \quad 0.00360479 \quad -0.1472733 \quad -0.07205665 \quad 0.04252212 \quad -0.10744561]]
         Bails of the model=[0.49908277]
In [23]: y_pred=regressor.predict(X_test)
         from sklearn.metrics import r2_score
         r_score=r2_score(Y_test,y_pred)
         print("r_score=", r_score)
         r_score= 0.42773659320883317
In [24]: age=int(input("enter the age value"))
         sex=int(input("enter the sex value"))
         cp=int(input("enter the cp value"))
         trestbps=int(input("enter the trestbps value"))
         chol=int(input("enter the chol value"))
         fbs=int(input("enter the fbs value"))
         restecg=int(input("enter the restecg value"))
         thalach=int(input("enter the thalach value"))
         exang=int(input("enter the exang value"))
         oldpeak=float(input("enter the oldpeak value"))
         slope=int(input("enter the slope value"))
         ca=int(input("enter the ca value"))
         future_prediction=regressor.predict([[age,sex,cp,trestbps,chol,fbs,restecg,thalach,exang,oldpeak,slope,ca,]])
         print("future_prediction={Purchased=0, Non Purchased=1}", format(future_prediction))
         enter the age value57
         enter the sex value10
         enter the cp value10
         enter the trestbps value25
         enter the chol value21
         enter the fbs value210
         enter the restecg value25
         enter the thalach value1025
         enter the exang value1542
         enter the oldpeak value3.5
         enter the slope value20
         enter the ca value25
         future_prediction={Purchased=0, Non Purchased=1} [[-239.7073345]]
         C:\ProgramData\anaconda3\Lib\site-packages\sklearn\base.py:464: UserWarning: X does not have valid feature names, but RidgeCV was fitted with feature names
           warnings.warn(
```