import numpy as np

import pandas as pd

#Loading the DataFrame

heart\_disease\_dataset = pd.read\_csv('heart disease classification dataset.csv')

del heart\_disease\_dataset["Unnamed: 0"]

print(heart\_disease\_dataset.head(5))

#Checking missing values

print(heart\_disease\_dataset.isnull().sum())

#Handelling the Missing Values

from sklearn.impute import SimpleImputer

impute = SimpleImputer(missing\_values=np.nan, strategy='mean')

impute.fit(heart\_disease\_dataset[["trestbps"]])

heart\_disease\_dataset[["trestbps"]] = impute.transform(heart\_disease\_dataset[["trestbps"]])

impute.fit(heart\_disease\_dataset[["chol"]])

heart\_disease\_dataset[["chol"]] = impute.transform(heart\_disease\_dataset[["chol"]])

heart\_disease\_dataset["thalach"].fillna(int(np.mean(heart\_disease\_dataset["thalach"])), inplace = True)

#No null values are present

print(heart\_disease\_dataset.isnull().sum())

#Encoding categorical features of sex and target column

from sklearn.preprocessing import LabelEncoder

enc = LabelEncoder()

heart\_disease\_dataset["sex"] = enc.fit\_transform(heart\_disease\_dataset["sex"])

enc = LabelEncoder()

heart\_disease\_dataset["target"] = enc.fit\_transform(heart\_disease\_dataset["target"])

#male - 1

#female - 0

print(heart\_disease\_dataset[["sex"]])

#yes - 1

#no - 0

print(heart\_disease\_dataset[["target"]])

#Checking the sex and target columns

print(heart\_disease\_dataset)

#Scaling all the values between 0-1

from sklearn.preprocessing import MinMaxScaler

scaler = MinMaxScaler()

scaler.fit(heart\_disease\_dataset)

heart\_disease\_dataset\_scaled = scaler.transform(heart\_disease\_dataset)

heart\_disease\_dataset\_scaled = pd.DataFrame(heart\_disease\_dataset\_scaled, columns = ['age', 'sex', 'cp', 'trestbps', 'chol', 'fbs', 'restecg', 'thalach',

'exang', 'oldpeak', 'slope', 'ca', 'thal', 'target'])

print(heart\_disease\_dataset\_scaled)

#Separating features

features = heart\_disease\_dataset.loc[:,'trestbps':'thal']

print(features, end='\n\n\n')

print(labels, end='\n\n\n')