import pandas as pd

from sklearn import preprocessing

data=pd.read\_csv("laptopPrice.csv")

print(data.head(6))

data.isna().sum()

data.isnull().sum()

features=data.drop("rating",axis=1)

target=data["rating"]

label\_encoder=preprocessing.LabelEncoder()

features['brand']=label\_encoder.fit\_transform(features['brand'])

features['brand'].unique()

features['processor\_brand']=label\_encoder.fit\_transform(features['processor\_brand'])

features['processor\_brand'].unique()

features['processor\_name']=label\_encoder.fit\_transform(features['processor\_name'])

features['processor\_name'].unique()

features['processor\_gnrtn']=label\_encoder.fit\_transform(features['processor\_gnrtn'])

features['processor\_gnrtn'].unique()

features['ram\_gb']=label\_encoder.fit\_transform(features['ram\_gb'])

features['ram\_gb'].unique()

features['ram\_type']=label\_encoder.fit\_transform(features['ram\_type'])

features['ram\_type'].unique()

features['ssd']=label\_encoder.fit\_transform(features['ssd'])

features['ssd'].unique()

features['hdd']=label\_encoder.fit\_transform(features['hdd'])

features['hdd'].unique()

features['os']=label\_encoder.fit\_transform(features['os'])

features['os'].unique()

features['os\_bit']=label\_encoder.fit\_transform(features['os\_bit'])

features['os\_bit'].unique()

features['graphic\_card\_gb']=label\_encoder.fit\_transform(features['graphic\_card\_gb'])

features['graphic\_card\_gb'].unique()

features['weight']=label\_encoder.fit\_transform(features['weight'])

features['weight'].unique()

features['warranty']=label\_encoder.fit\_transform(features['warranty'])

features['warranty'].unique()

features['Touchscreen']=label\_encoder.fit\_transform(features['Touchscreen'])

features['Touchscreen'].unique()

features['msoffice']=label\_encoder.fit\_transform(features['msoffice'])

features['msoffice'].unique()

data['rating']=label\_encoder.fit\_transform(data['rating'])

data['rating'].unique()

target=data['rating']

import numpy as np

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

x\_train,x\_test,y\_train,y\_test=train\_test\_split(features,target,test\_size=0.2,random\_state=42)

print(x\_train.shape)

print(x\_test.shape)

print(y\_train.shape)

print(y\_test.shape)

from sklearn.ensemble import RandomForestClassifier

rf=RandomForestClassifier(n\_estimators=42)

rf\_1=rf.fit(x\_train,y\_train)

y\_pred=rf.predict(x\_test)

from sklearn.metrics import accuracy\_score,confusion\_matrix

import seaborn as sns

import matplotlib.pyplot as plt

accuracy= accuracy\_score(y\_pred,y\_test)

cm=confusion\_matrix(y\_pred,y\_test)

plt.figure(figsize=(5,5))

sns.heatmap(cm,annot=True)

plt.show()