

NAME OF THE PROJECT

**Flight Price Prediction**

Submitted by:

**Sujay Bharmal**

**ACKNOWLEDGMENT**

This includes mentioning all the references, research papers, data sources, professionals, and other resources that helped you and guided you in the completion of the project.

**Introduction:- Problem Definition**

Anyone who has booked a flight ticket knows how unexpectedly the prices vary. The cheapest available ticket on a given flight gets more and less expensive over time. This usually happens as an attempt to maximize revenue based on - 1. Time of purchase patterns (making sure last-minute purchases are expensive) 2. Keeping the flight as full as they want it (raising prices on a flight that is filling up to reduce sales and hold back inventory for those expensive last-minute expensive purchases) So, you have to work on a project where you collect data on flight fares with other features and work to make a model to predict fares of flights.

**Data Analysis:-**

In this project, we have 6 inputs and 1 output that are available according to my information I see that this case is a supervised learning case.

First, we import all the necessary libraries. Like pandas, NumPy seaborn, and matplotlib, and then run that cell. After that, we read the all data and save it into the pdf folder. There are 1440 rows and 7 columns. There are 7 columns of the input and price is the output variable.

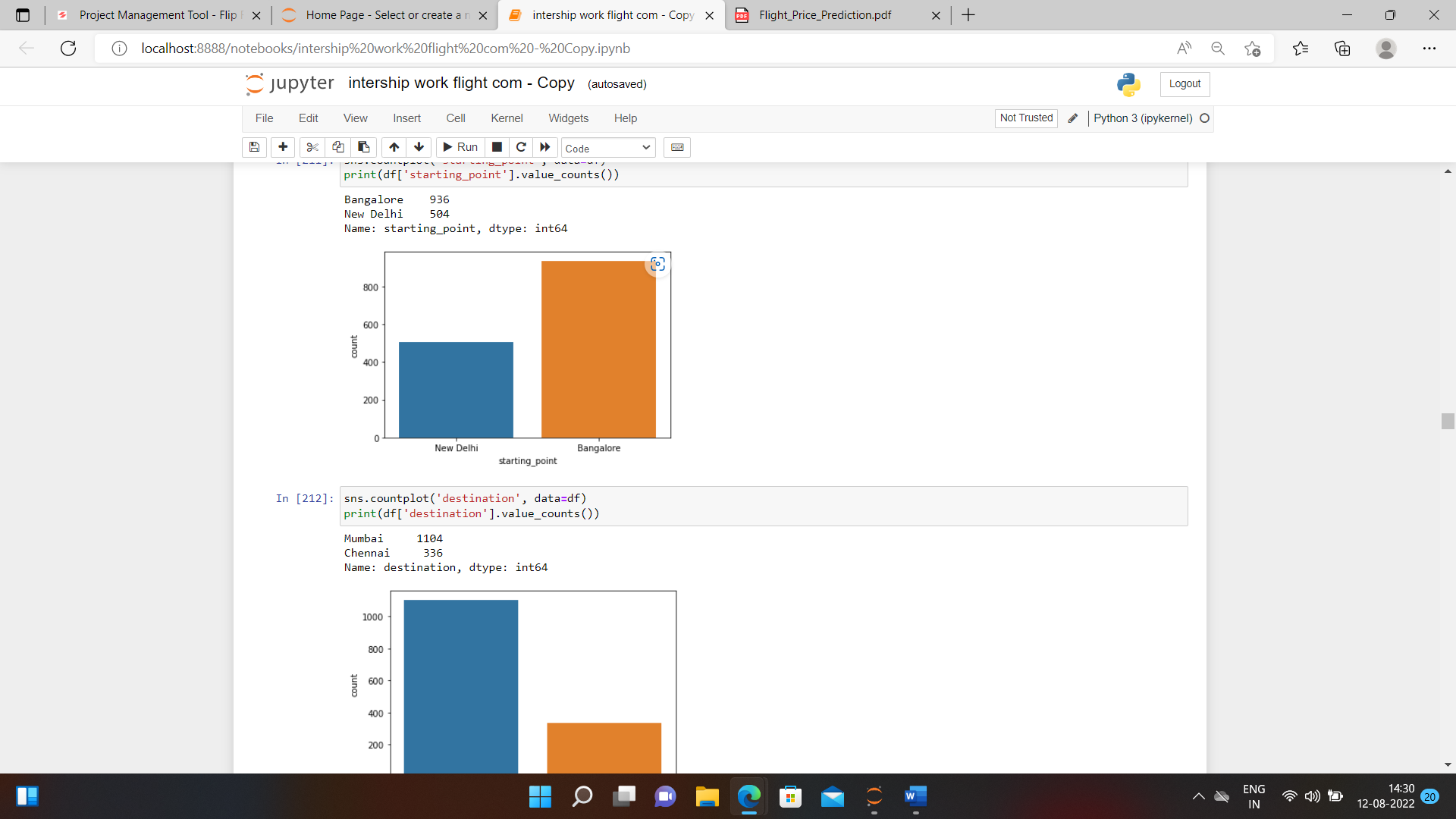
First, we see that there are in this dataset’s null values are present or not, we give df. IsNull().sum() code to run and we see that there is no null value present.

Then we check all column datatypes by using the df.info() code.

We see that there are no null values present so we get the further process.

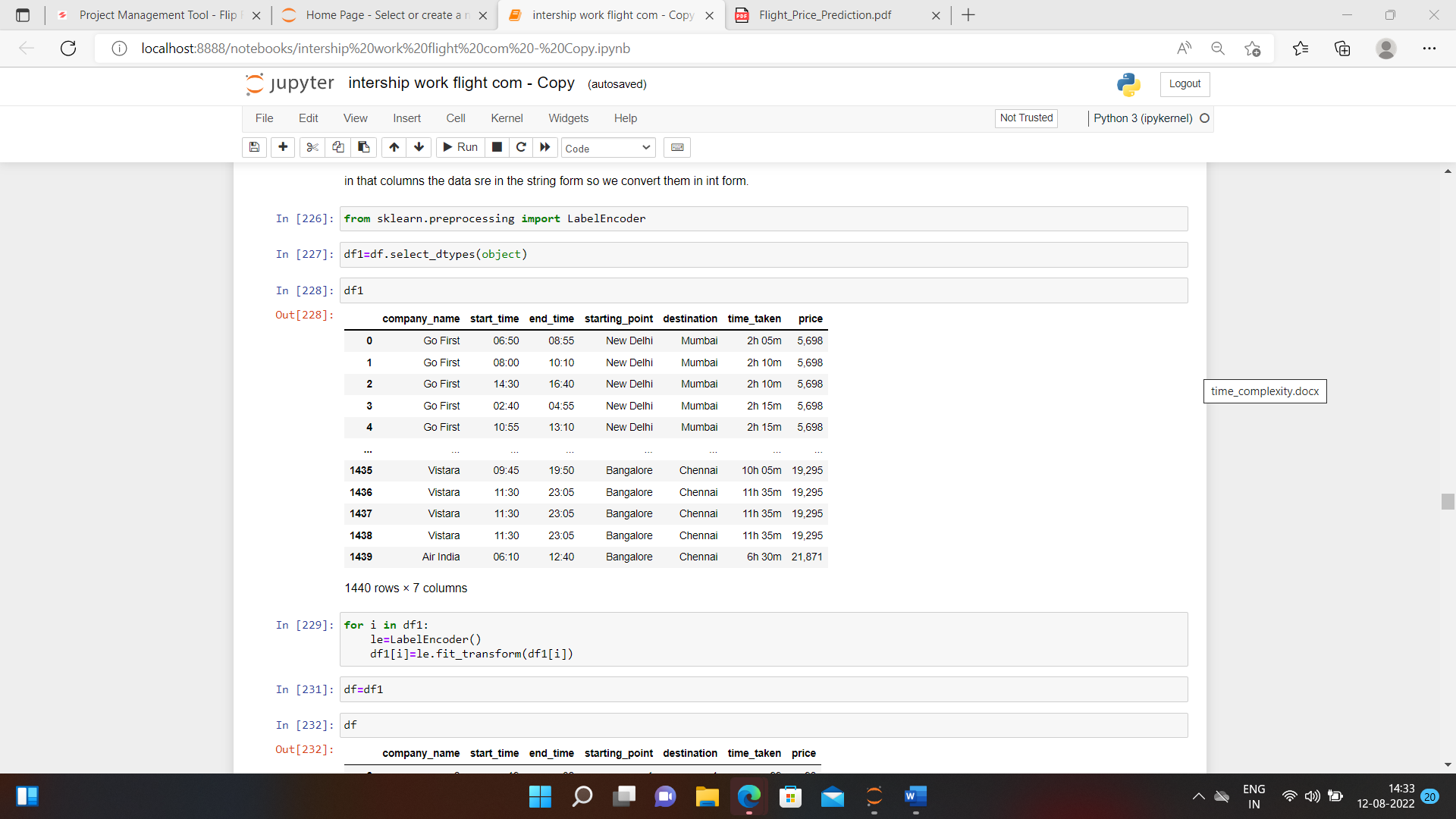
After that, we install the necessary libraries like seaborn and matplotlib for the data analysis purpose.

sns.countplot('company\_name', data=df)print(df['company\_name'].value\_counts()) by using this shell we see the company name of the flights.



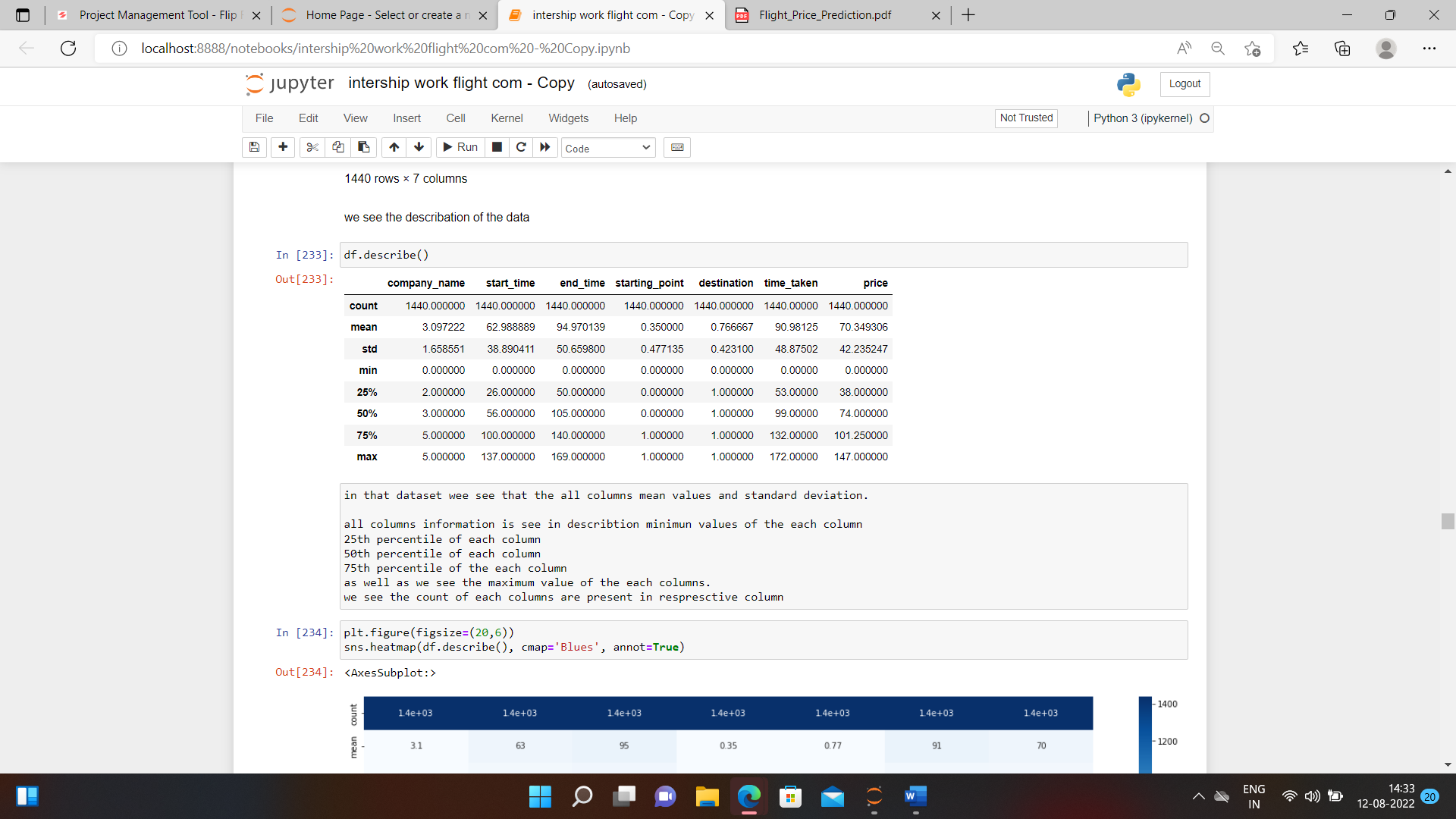
We see the different counts of the projects.

Use the label encoding method,



Using this method we convert all the data into numeric form.

Description of the dataset:-



in that dataset, we see all column mean values and standard deviation.

all cocolumnnformation is seen in the description minimum values of each column

25th percentile of each column

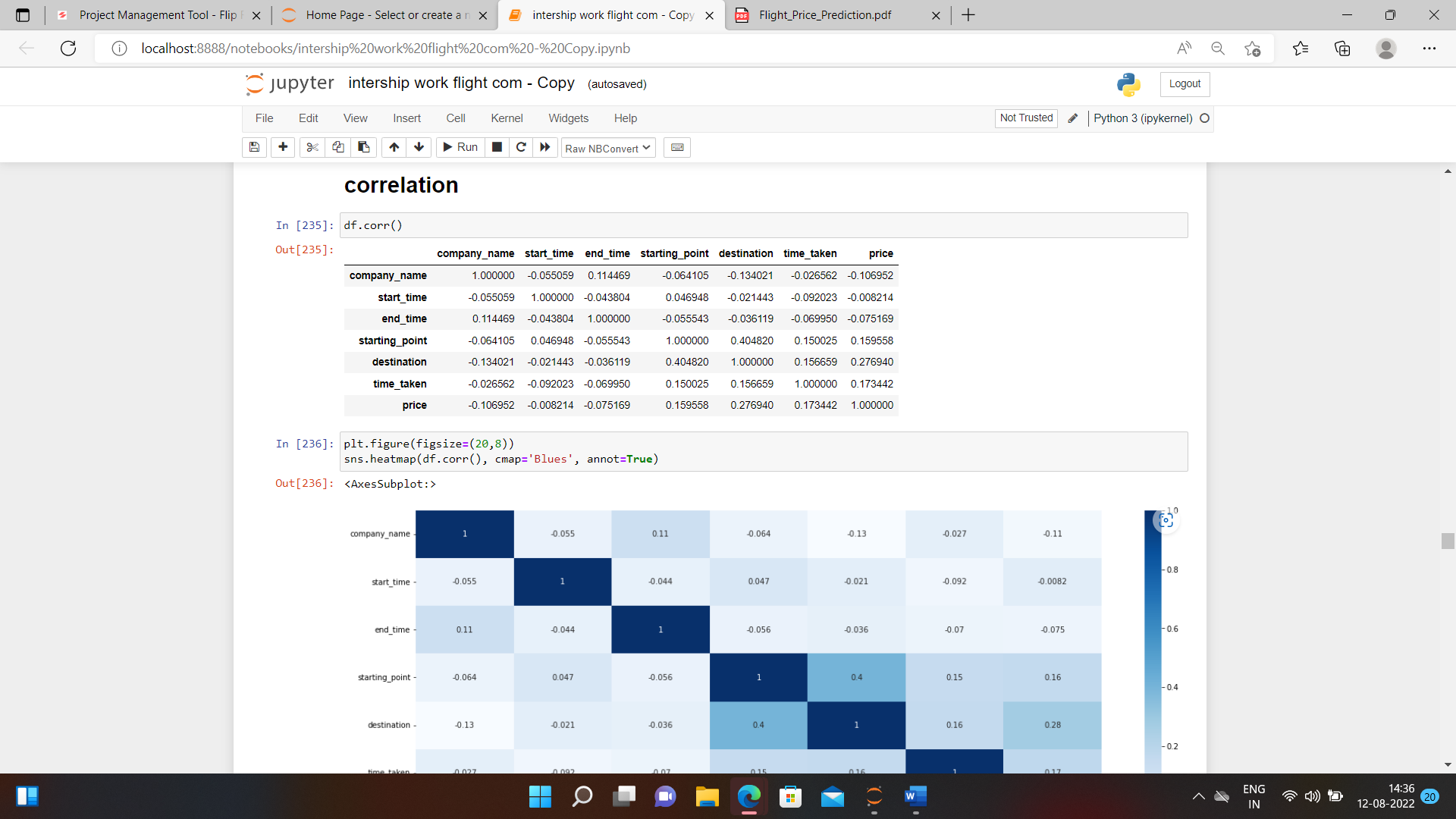
50th percentile of each column

75th percentile of each column

as well as see the maximum value of each column.

we see the count of each column is present in the respective column

we see the correlation of the each columns with each other.



In this image we see the correlation of all the columns.

After that we see the skewness of the all the columns and specifically the output variable of price also.

After that we get other visualization like distplot as well as histplot and see the results. Using the box plot we see the outliers are presents or not presents that we see.

After that we use the zscore for the outliers removing and get the clean data.

Then we seprate the data set by x and y , x are all columns expect price and y is price coumn.

Using the power transform method too x variable.

Then standrize the data set using standardscaler method.

Machine Learning model building:-

from sklearn.linear\_model import LinearRegression

lr=LinearRegression()

from sklearn.metrics import r2\_score

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

using the methods we see the model building.

Using range method we see the resukts of all test then using linearregression method wesee the result.

from sklearn.model\_selection import GridSearchCV

from sklearn.model\_selection import cross\_val\_score

from sklearn.linear\_model import Lasso

parameters={'alpha':[.0001, .001, .01,.1,1,10],'random\_state':list(range(0,10))}

ls=Lasso()

clf=GridSearchCV(ls,parameters)

clf.fit(x\_train, y\_train)

print(clf.best\_params\_)

we see the lasso effect on that model.

cv\_score=cross\_val\_score(ls,x,y,cv=5)

cv\_mean=cv\_score.mean()

cv\_mean

see the mean cv score.

from sklearn.model\_selection import GridSearchCV

from sklearn.ensemble import RandomForestRegressor

parameters={'criterion':['mse','mae'], 'max\_features':['auto','sqrt','log2']}

rf=RandomForestRegressor()

clt=GridSearchCV(rf,parameters)

clt.fit(x\_train,y\_train)

print(clt.best\_params\_)

we select the best fit model and that model we use {'criterion': 'mse', 'max\_features': 'log2'}

then,

rf=RandomForestRegressor(criterion='mse',max\_features='log2')

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

rf.score(x\_train, y\_train)

pred\_decision=rf.predict(x\_test)

rfs=r2\_score(y\_test,pred\_decision)

print('R2 score', rfs\*100)

rfscore=cross\_val\_score(rf,x,y,cv=5)

rfc=rfscore.mean()

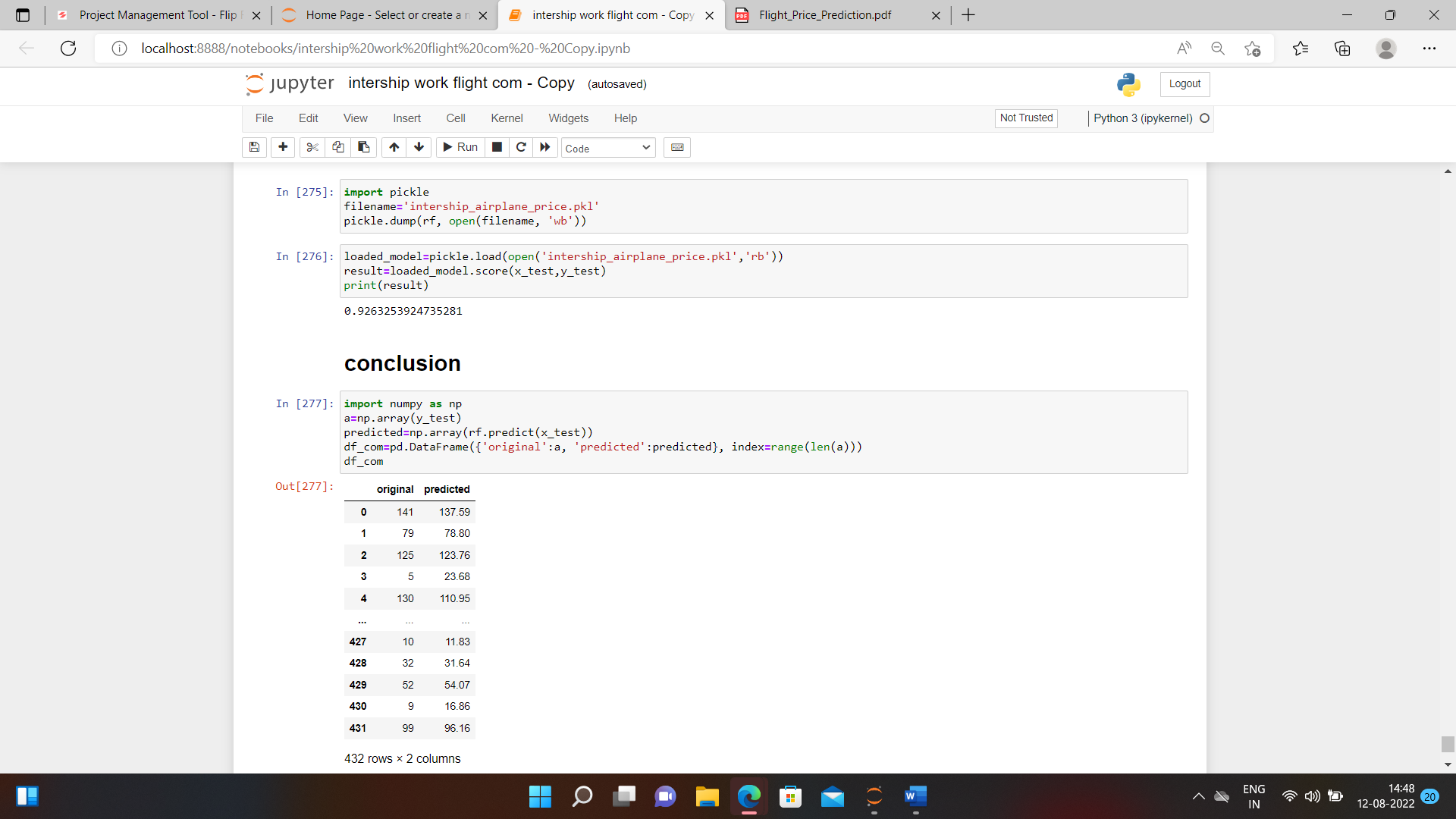
print('Cross Val Score', rfc\*100)

we see the result,

R2 score 92.63253924735281

Cross Val Score 87.29092666103678

After that we save the model,



Saving the model and conclusion tell that the original price and the predicted price of the flight.