**Blog-2**

**Flight\_Ticket\_Price\_prediction**

**Author**

**Neha Dixit**

**Flight Price Prediction**

**Problem Definition**

The variation in prices of flight tickets has always been very confusing for customers and is very difficult to guess. The airline companies implement dynamic strategies to assign pricing for airfare tickets to increase demand for their seats and maximize their revenue, hence it becomes difficult for consumers to buy tickets at a minimum price. The closely connected terms with the airline prices such as commercial, financial, social factors and marketing is under consideration while practicing these dynamic strategies. The airline companies are trying their best to keep their revenue high and increase their profit. The travellers often find the flight prices unpredictable as the flight prices tomorrow will not be the same as the flight prices of today. The system is complicated because each flight has a limited number of seats to be sold. In case the demand for air tickets is high, then the prices will increase and on the other hand if the seats are left unsold then the cost of air tickets might decrease as it represents a loss of revenue. To solve this problem of predicting flight prices, Machine Learning is a great idea to learn from historical data of the past flight prices and build logic on the given data. We will use Linear Regression which will help us in predicting the flight prices on the basis of certain factors which will involve data extracting, data analysing and data interpretation.

**Dataset Information**

The prices of flight tickets for various airlines are between the months of March and June of 2019 and between various cities.

We have two datasets i.e. Train Data and Test Data.

Size of training set: 10683 records

Size of test set: 2671 records

The size of the **Training Set** is 10,683 records which consists of both categorical and numeric data. Some special characters are also seen within the data to which we will apply data transformation before using it on the Model.

The **features** considered initially for each flight are:

Airline: The name of the airline.

Date\_of\_Journey: The date of the journey

Source: The source from which the service begins.

Destination: The destination where the service ends.

Route: The route taken by the flight to reach the destination.

Dep\_Time: The time when the journey starts from the source.

Arrival\_Time: Time of arrival at the destination.

Duration: Total duration of the flight.

Total\_Stops: Total stops between the source and destination.

Additional\_Info: Additional information about the flight

Price: The price of the ticket

The size of the **Testing Set** is 2671 records. The testing data is similar to the training data, except for the “Price” column which will be predicted using the model.

Link to find the .csv file is on Github is :

* <https://github.com/dsrscientist/Data-Science-ML-Capstone-Projects>

**This notebook covers following contents –**

Reading the data

Overview of data's structure - how various features and their respective values look like?

Finding and handling missing values into both training and test dataset

Dealing with categorical attributes and change into numberical numbers

Identifying higher correlation features (with the target)

Generating relevant insights about values of these high correlation features for churned customers (

Preparing data for models

Model generation and performance evaluation

**Data Collection**

From numerous sources the data was collected. The flight ticket detailed information is retrieved from an online data source (github.com). We took out this data from the website which is in the form of a csv record. The file consists of the information with input features and its target variable required for analyzing data. We have retrieved additional features from the existing variables to get more accuracy in the results.

**Importing the data:**

We need to import all the relevant libraries:

import numpy as np

import pandas as pd

import sklearn

import seaborn as sns

import matplotlib.pyplot as plt

from sklearn.preprocessing import LabelEncoder

from sklearn.linear\_model import LinearRegression

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

from sklearn.metrics import mean\_squared\_error,r2\_score,mean\_absolute\_error

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

from sklearn.linear\_model import Lasso, LassoCV, Ridge, RidgeCV

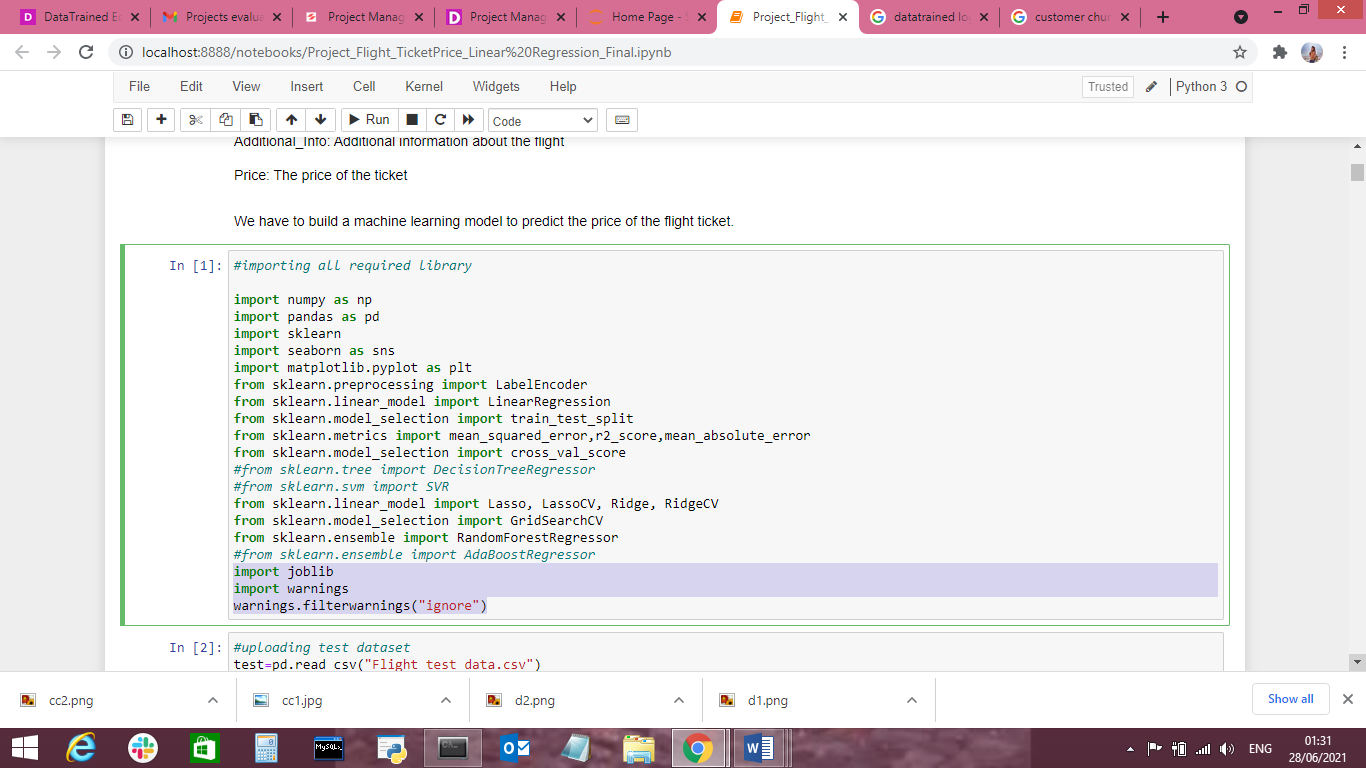
from sklearn.model\_selection import GridSearchCV

from sklearn.ensemble import RandomForestRegressor

import joblib

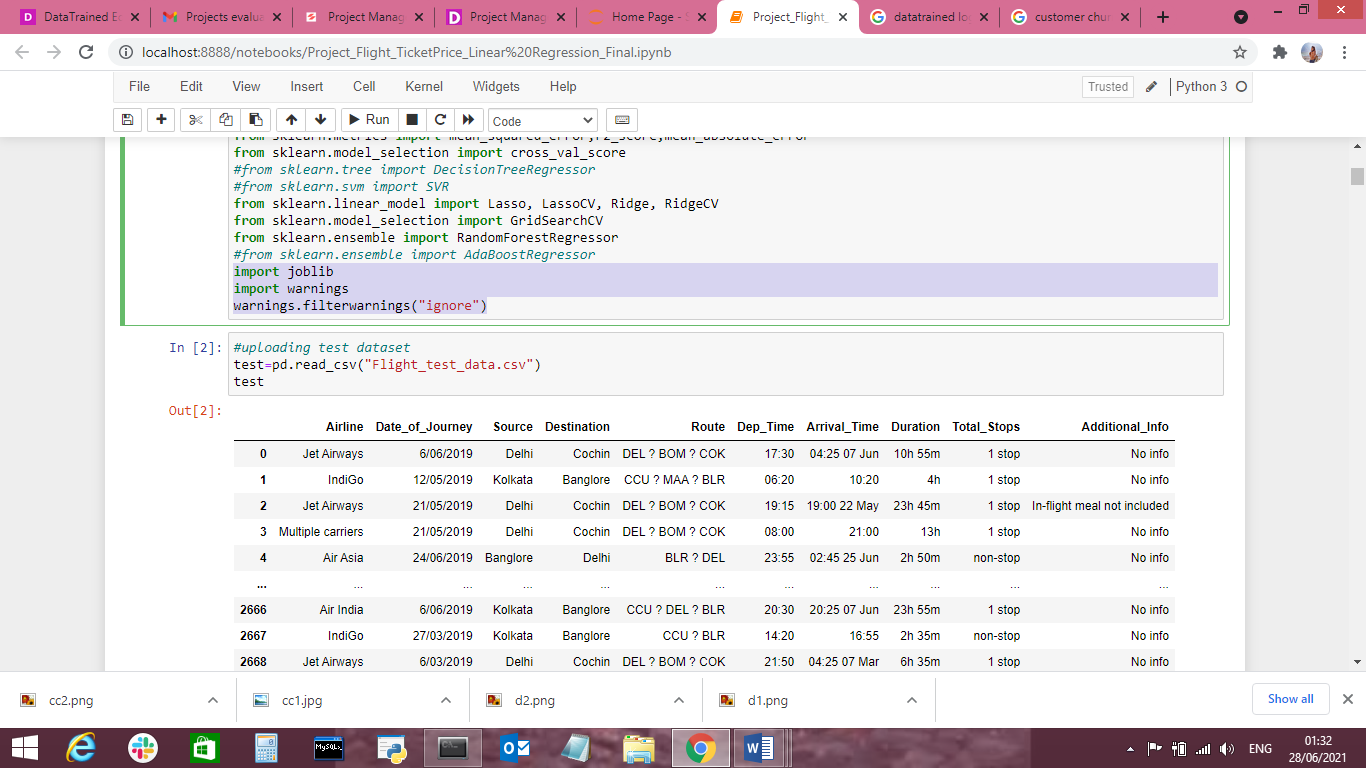
import warnings

warnings.filterwarnings("ignore")



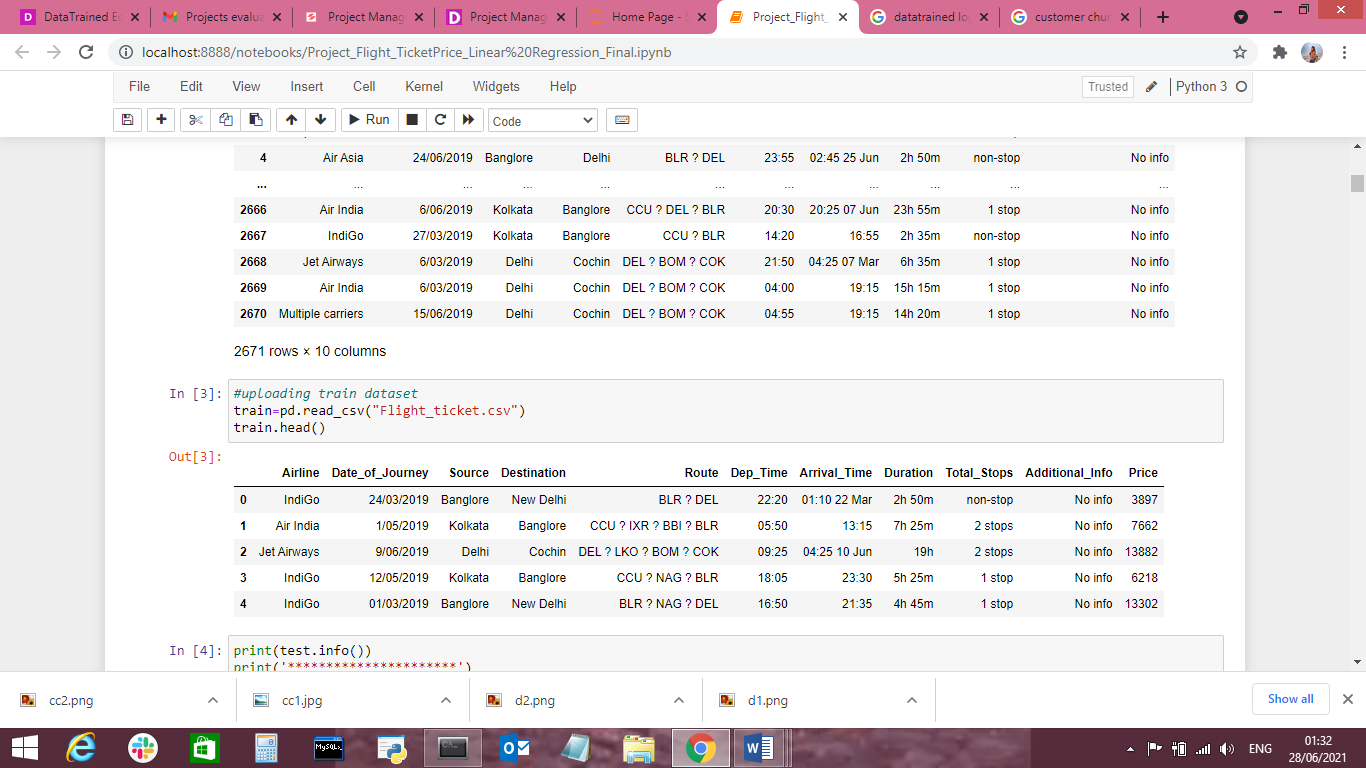
We have now imported all the important libraries that we will be needing during analysis.

We need to import the .csv file into the Jupyter notebook as shown below.



This data set contain only Independent (target) variables.

Independent variable: They are also known as Input variables. These are the input for a process that is being analysed.



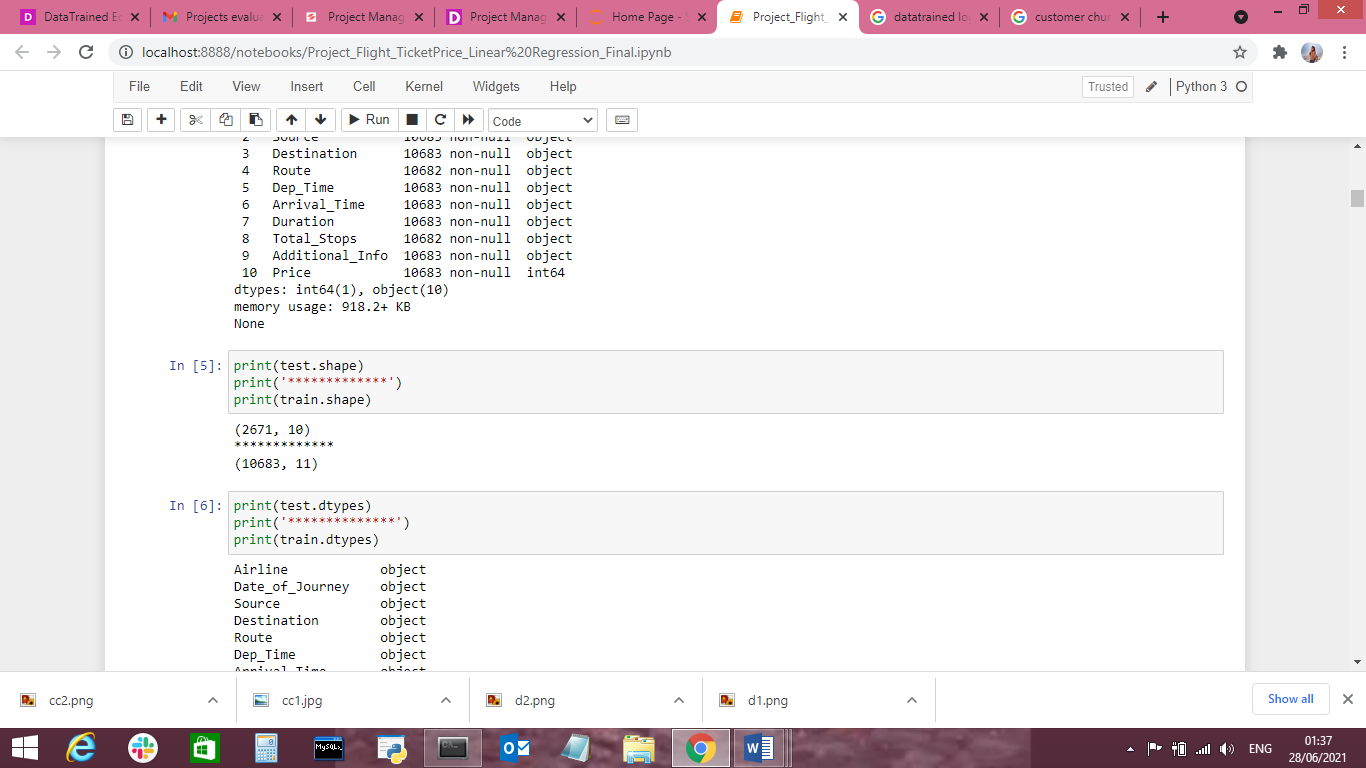
This data set contains both Independent and Dependent (target) variables.

Independent variable: They are also known as Input variables. These are the input for a process that is being analysed.

Dependent variable: They are also known as Output or Target variables. They are dependent on Independent variables for their outcome.

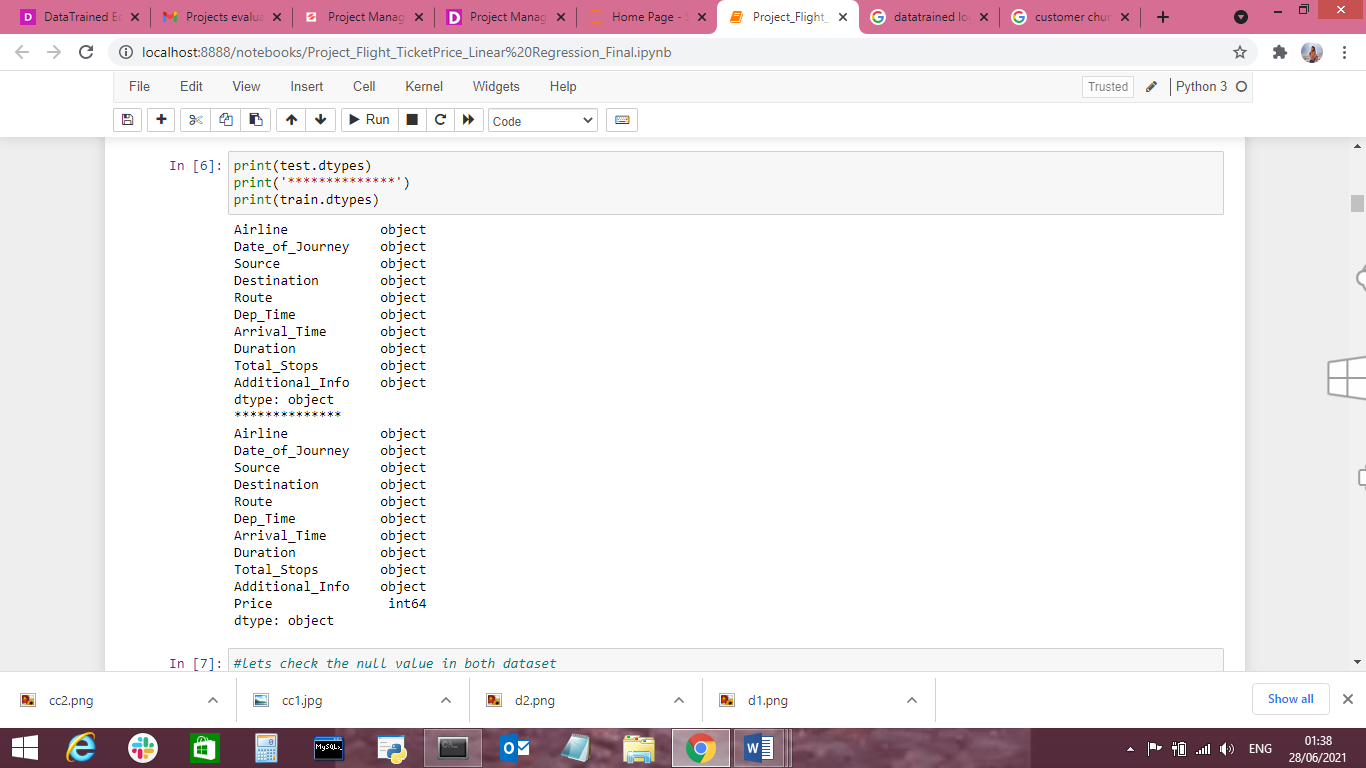
**Data Analysis**

After importing the dataset, display a sample of data. The variables in the dataset are as follows:



Test dataset has 271 columns and 10 rows

Train dataset has 10683 columns ad 11 rows



Both dataset has object values so we have to work on it and change into numerical numbers.

Features:

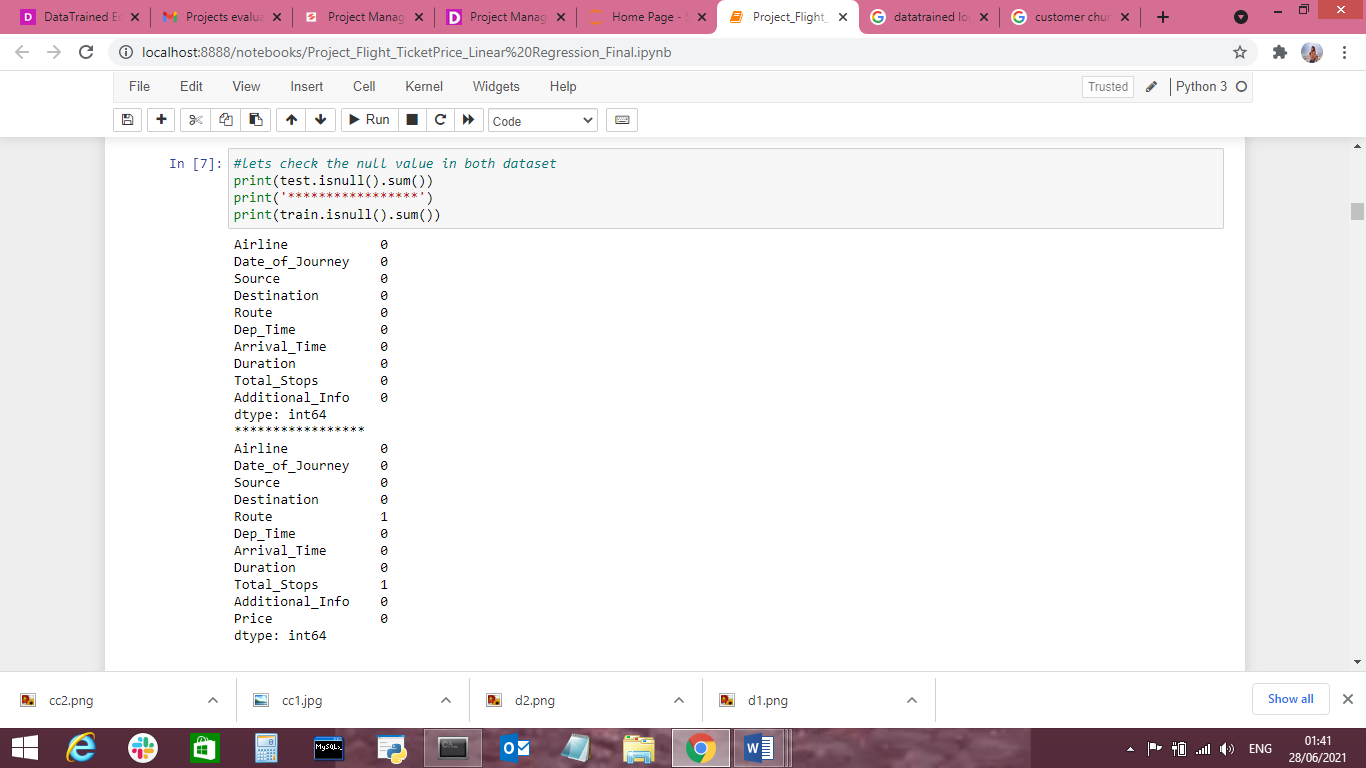
Such as “Arrival\_Time”, “Arrival\_Date”, “Arrival\_Month”, “Day”, “Month” & “Year”

are generated to make analysis of data.

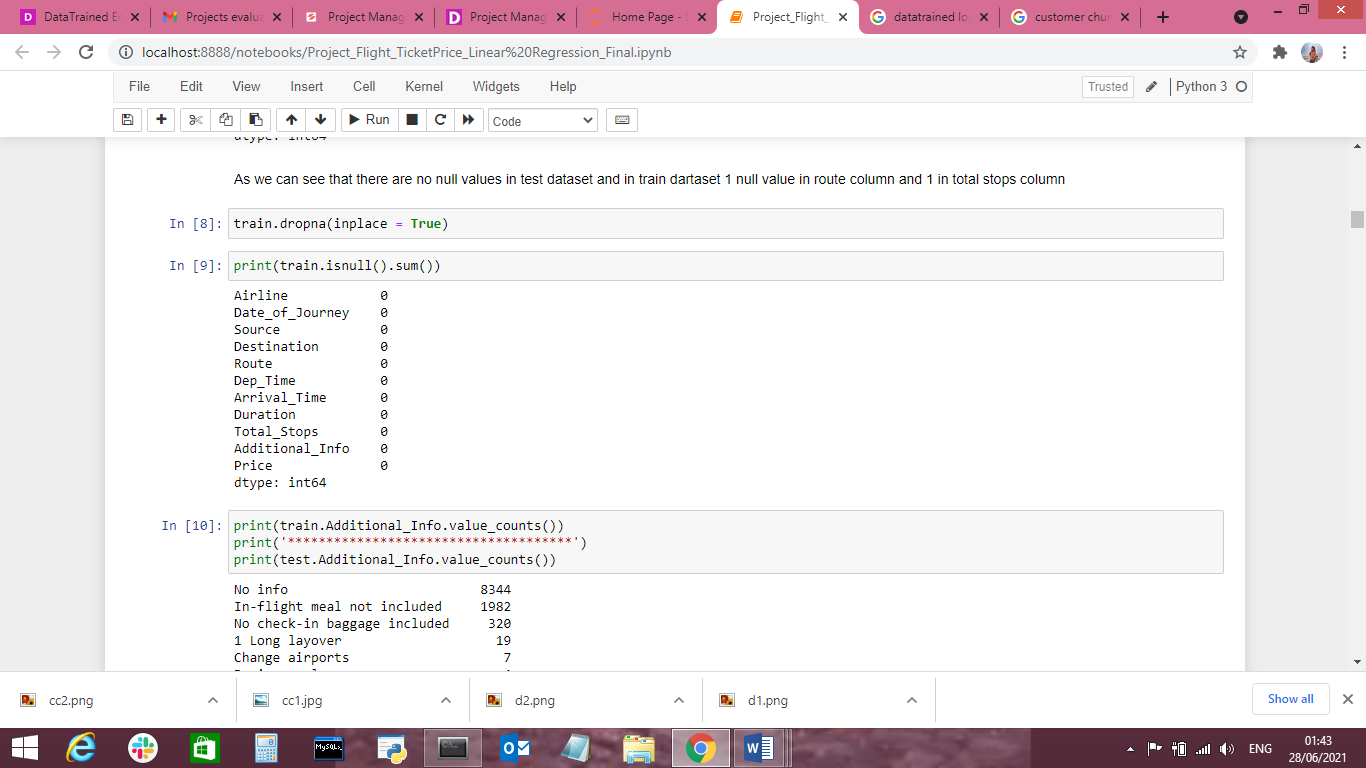
An important perspective is to wisely choose the necessary features required for calculating the flight prices as per expectations. All the existing variables and the retrieved features for each flight may not be required in making an accurate analysis of data, hence we only select variables that are significant and remove features with less importance for analyzing data accurately.

**Cleaning and Data Preparation**

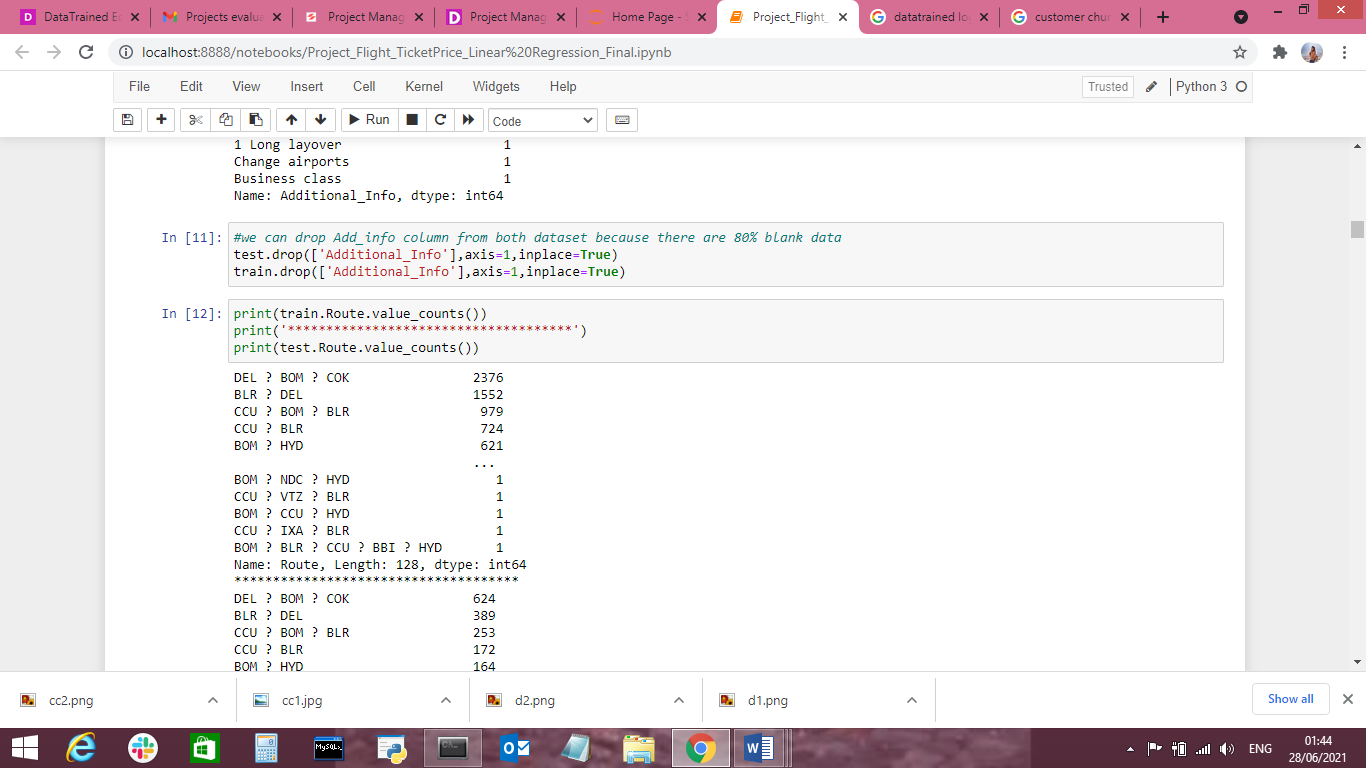
All the accumulated data needs a great deal of work. So after information gathering, all the irrelevant data (features such as “Arrival\_Date” & “Arrival\_Month”), duplicate features and invalid qualities (features such as “Additional\_Info” & “Year”) are deleted. As the dataset contains missing values in variables such as “Route” and “Total\_Stops”,



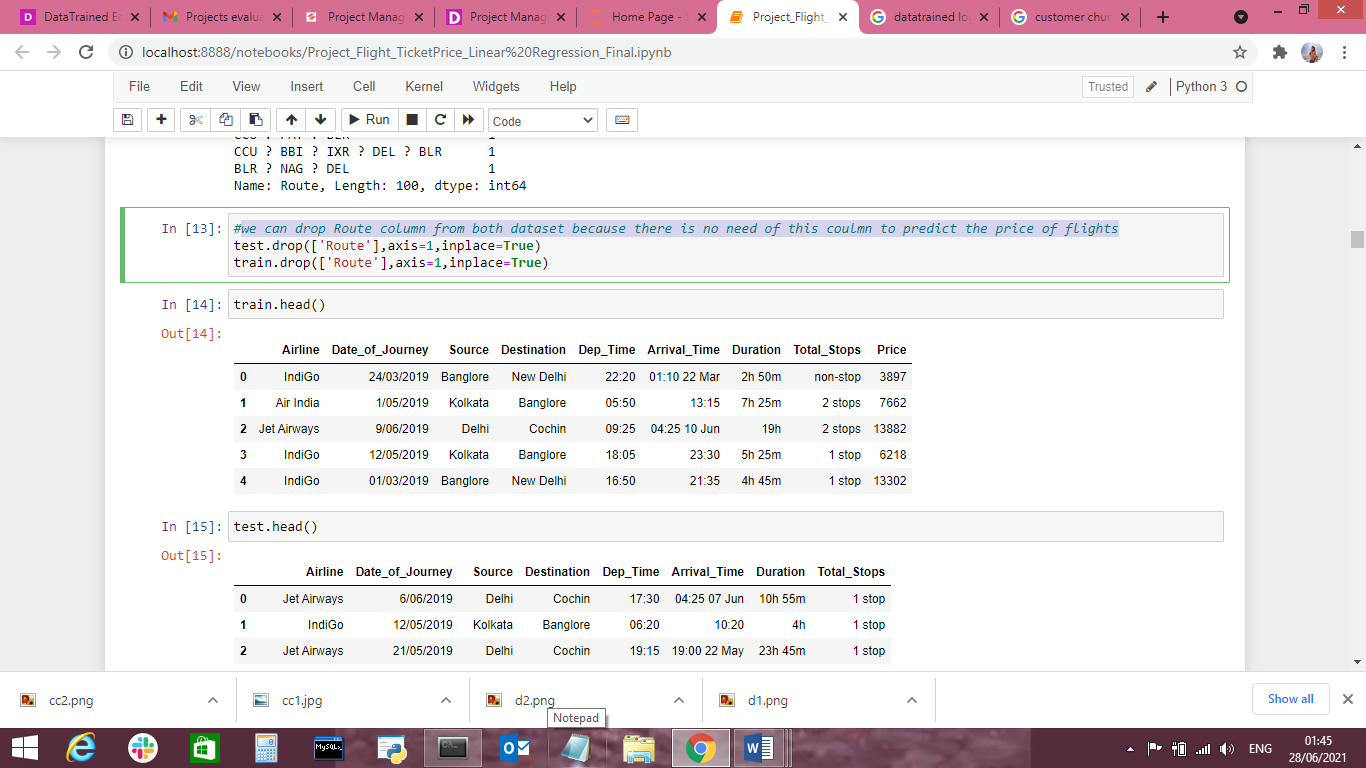
We will remove null value



We can drop “Additional\_Info” column as there is no need of it.



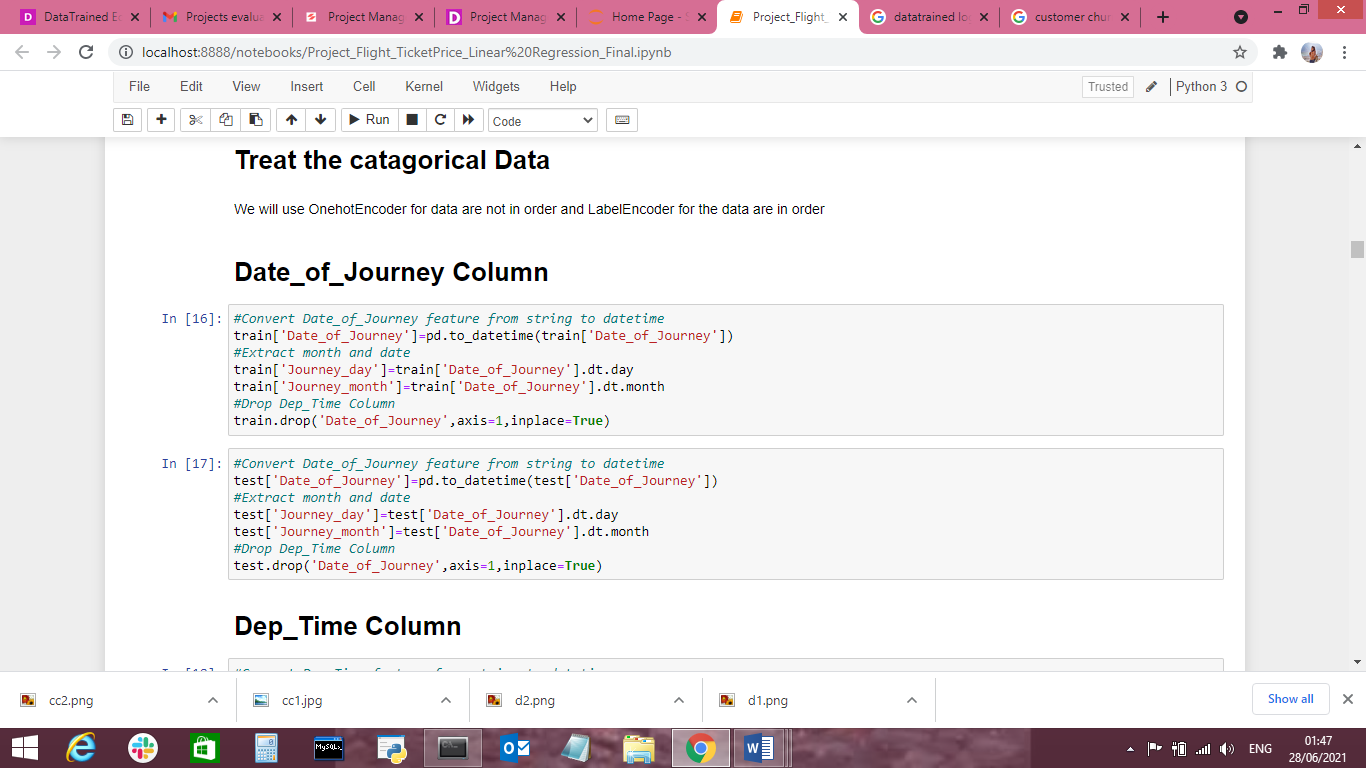
we can drop Route column from both dataset because there is no need of this column to predict the price of flights



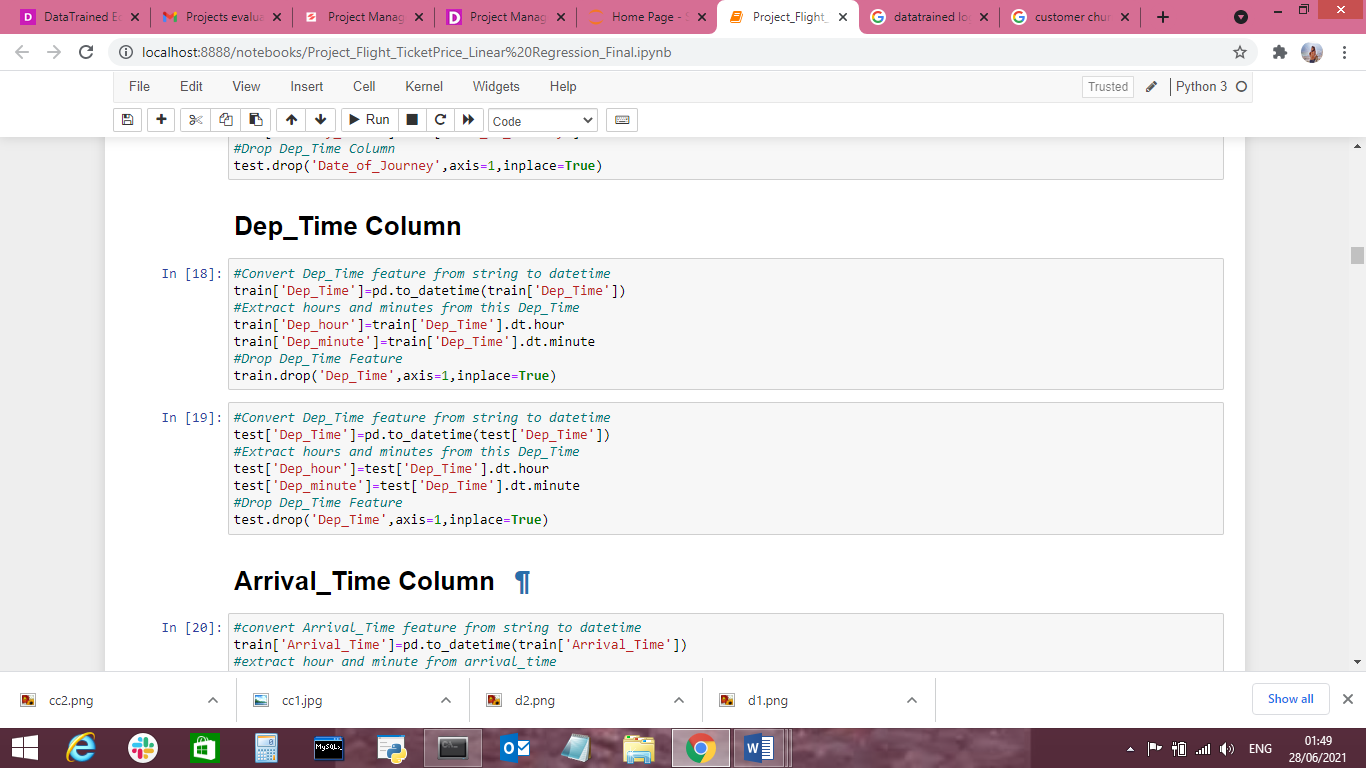
**Treat Catagorical Data:**

**We treat all object type of data in numerical numbers of both dataset**

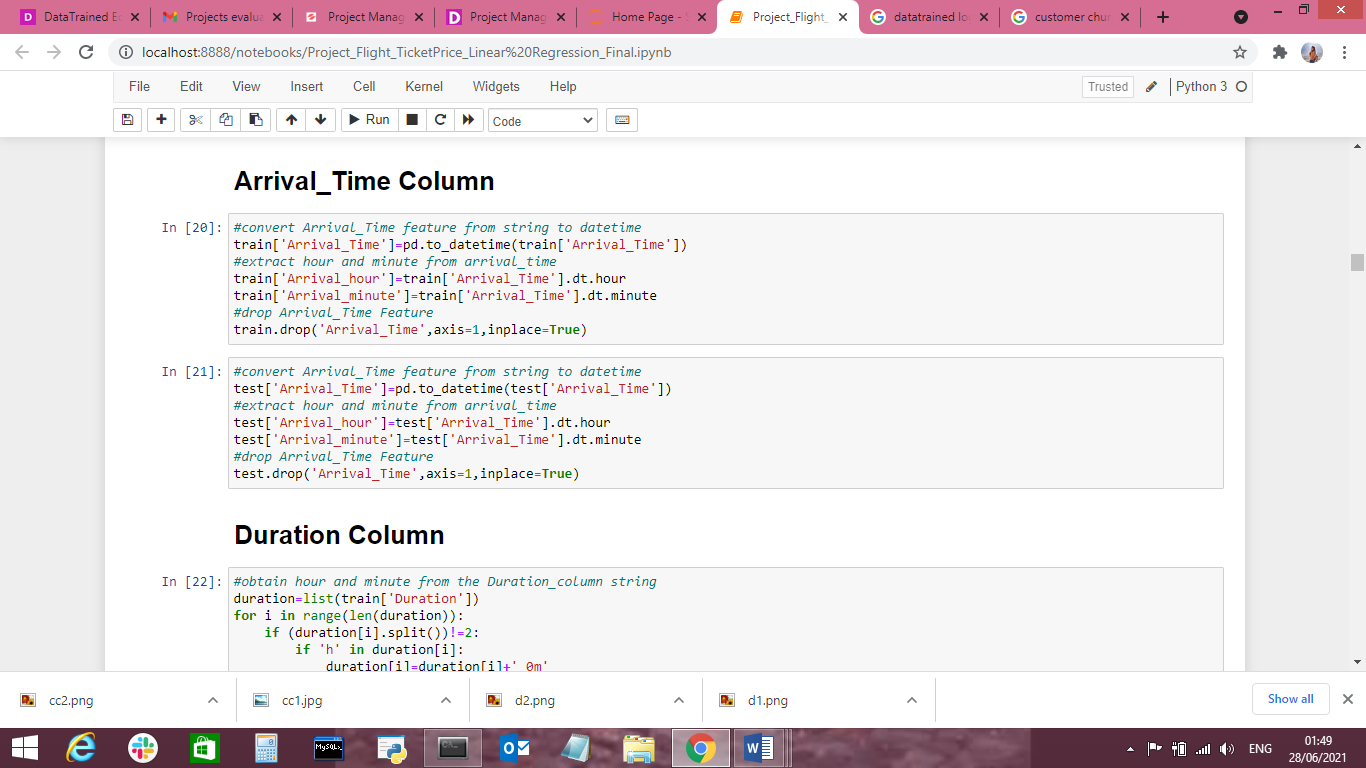
We splitted new column named Journey\_Day and Journey\_Month and then dropped column Date\_of a\_Journey



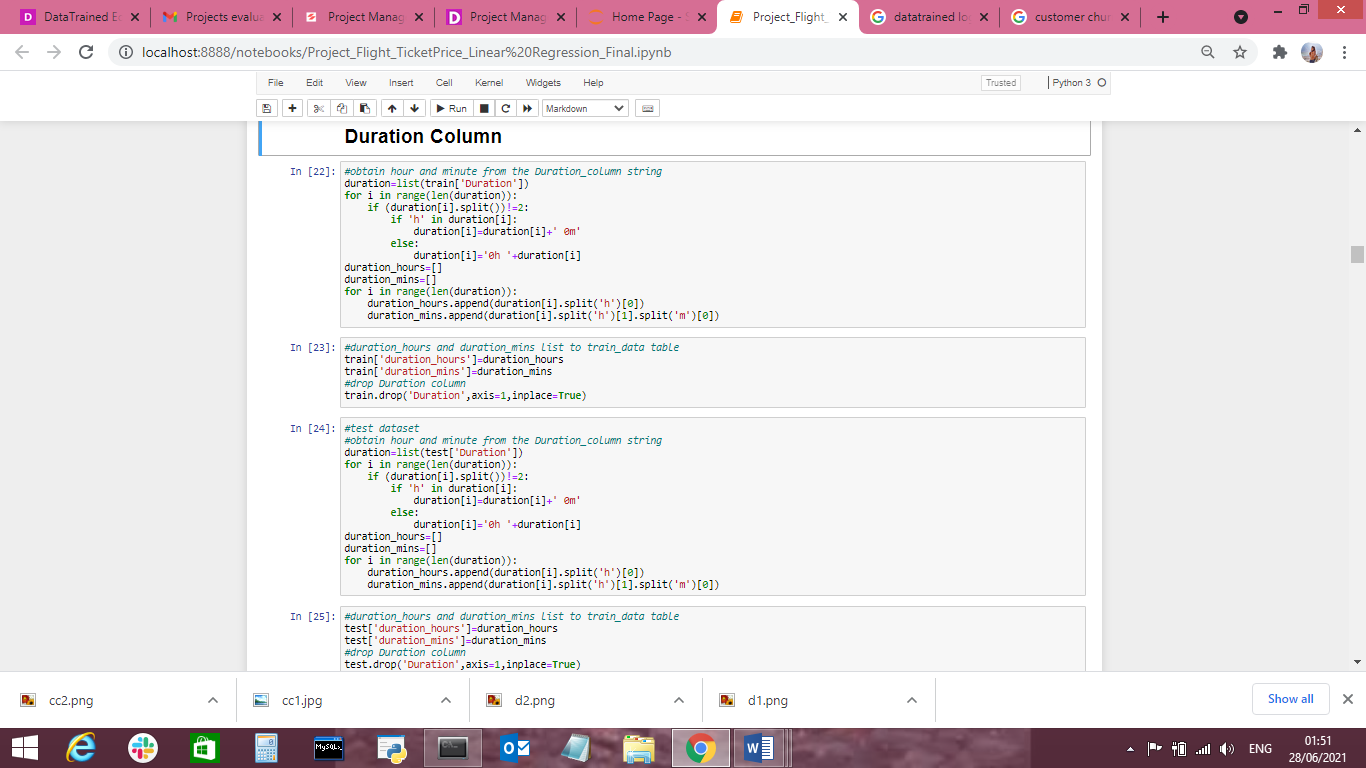
We splitted new columns named Dep\_hour, Dep\_minute and then dropped Dep\_Time Column



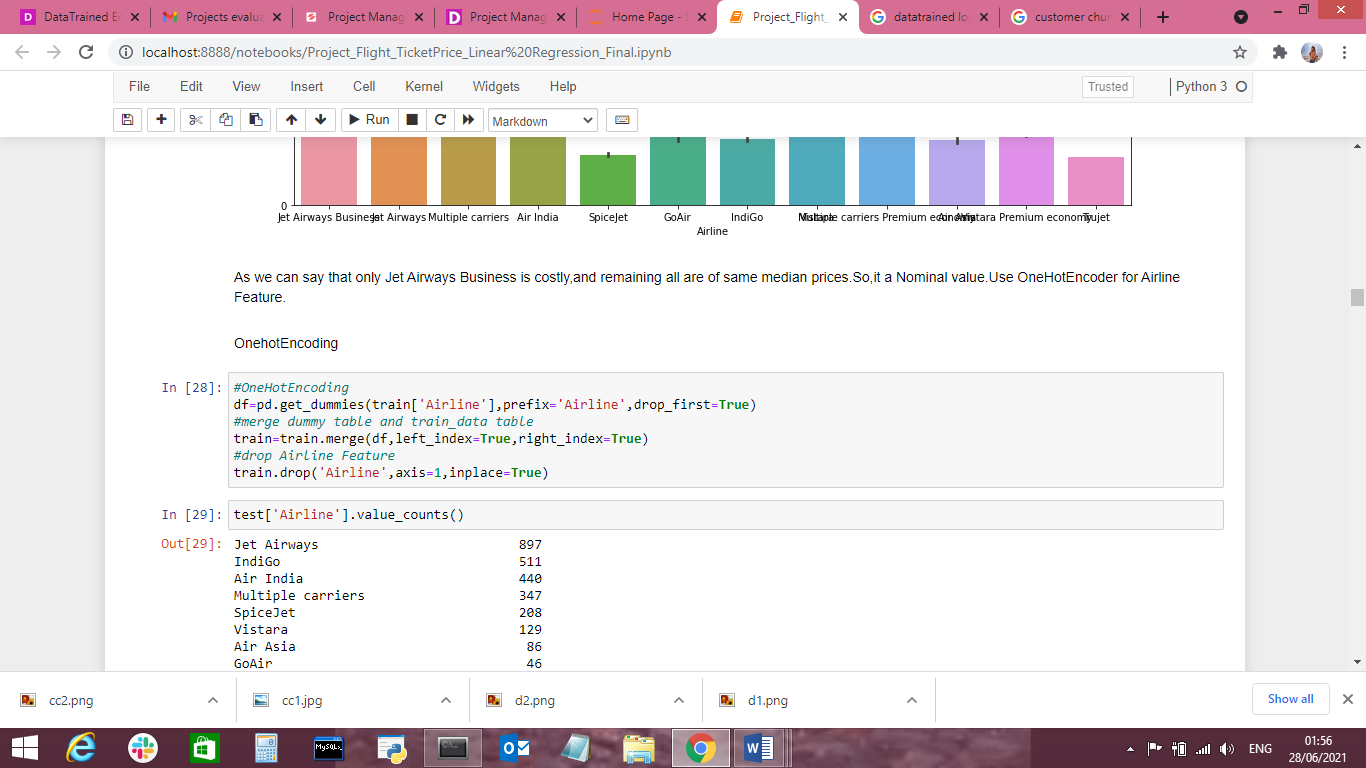
We spiltted column named; Arrival\_hour, Arrival\_minute and then dropped column

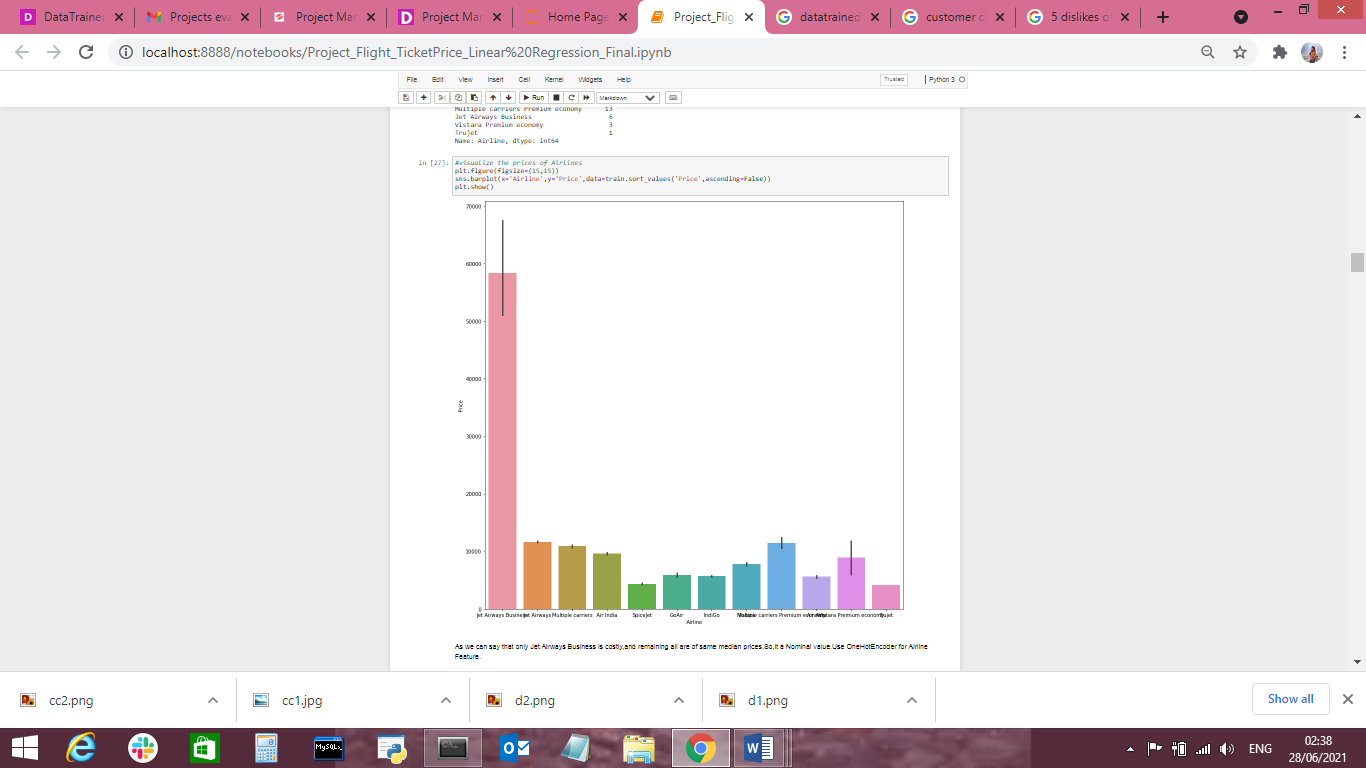


Duration Column spilled into duration into duration\_hours,duration\_mins

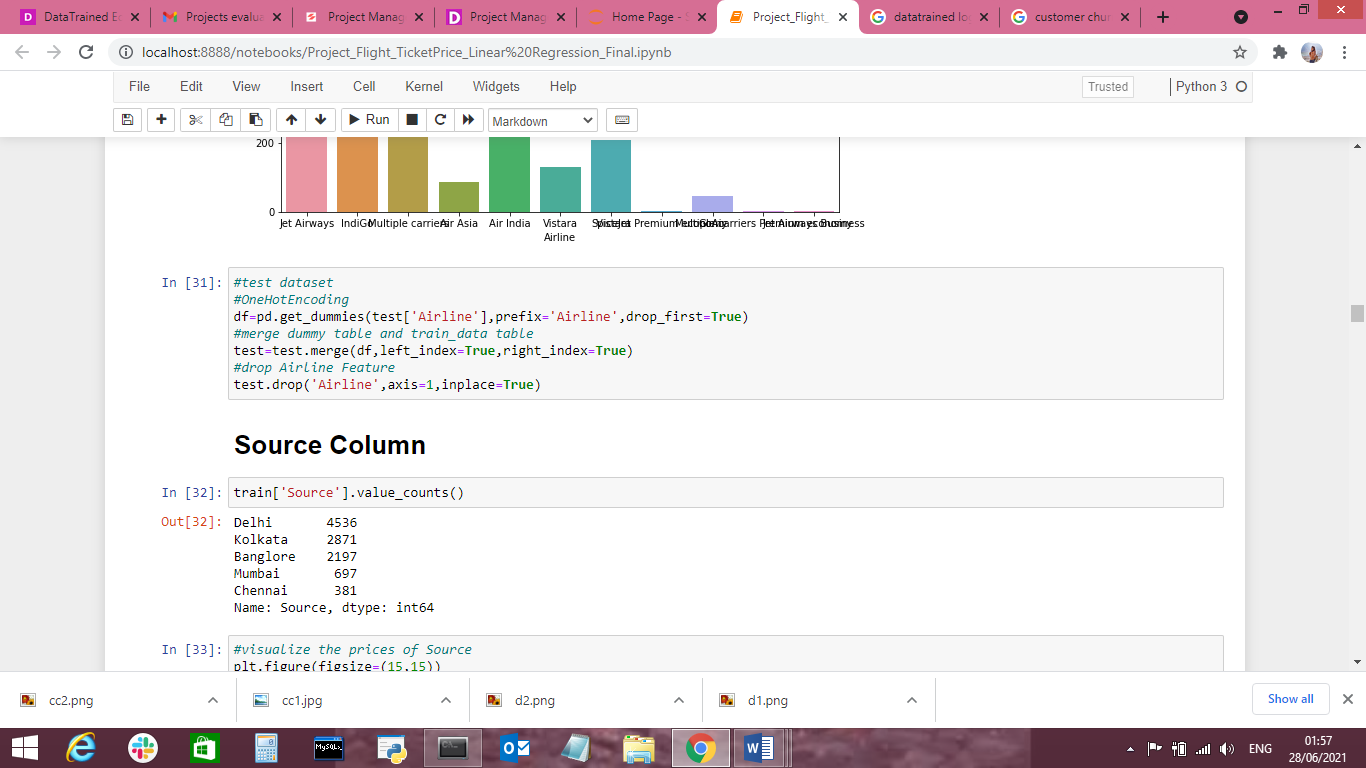


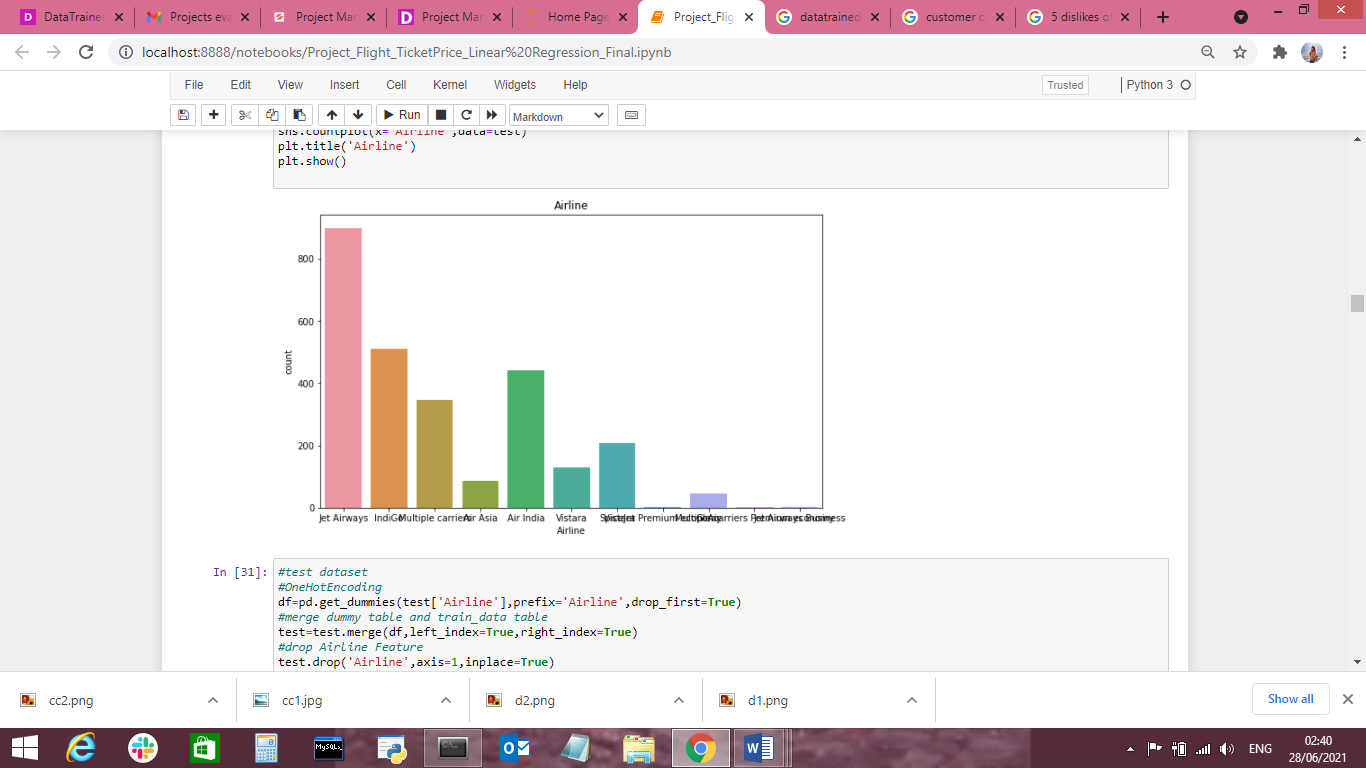
**Airline Column**



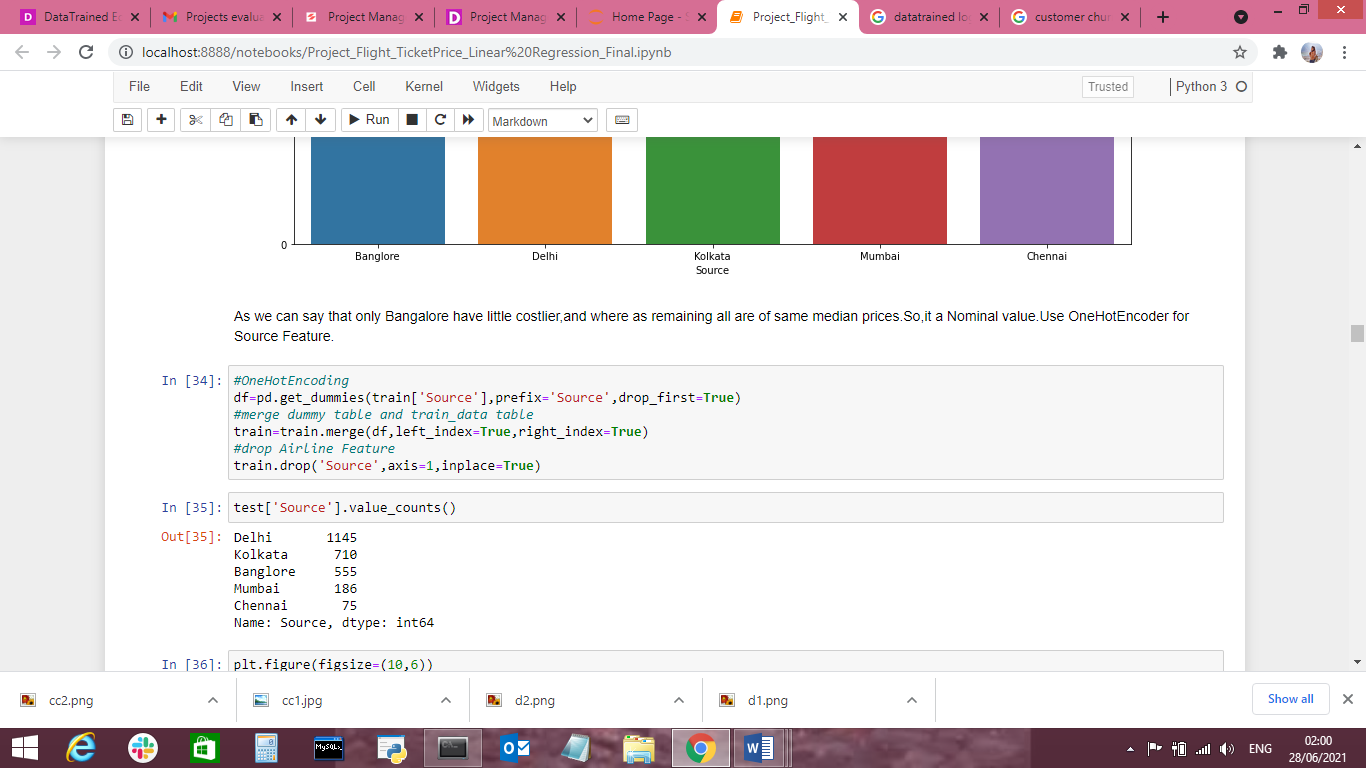


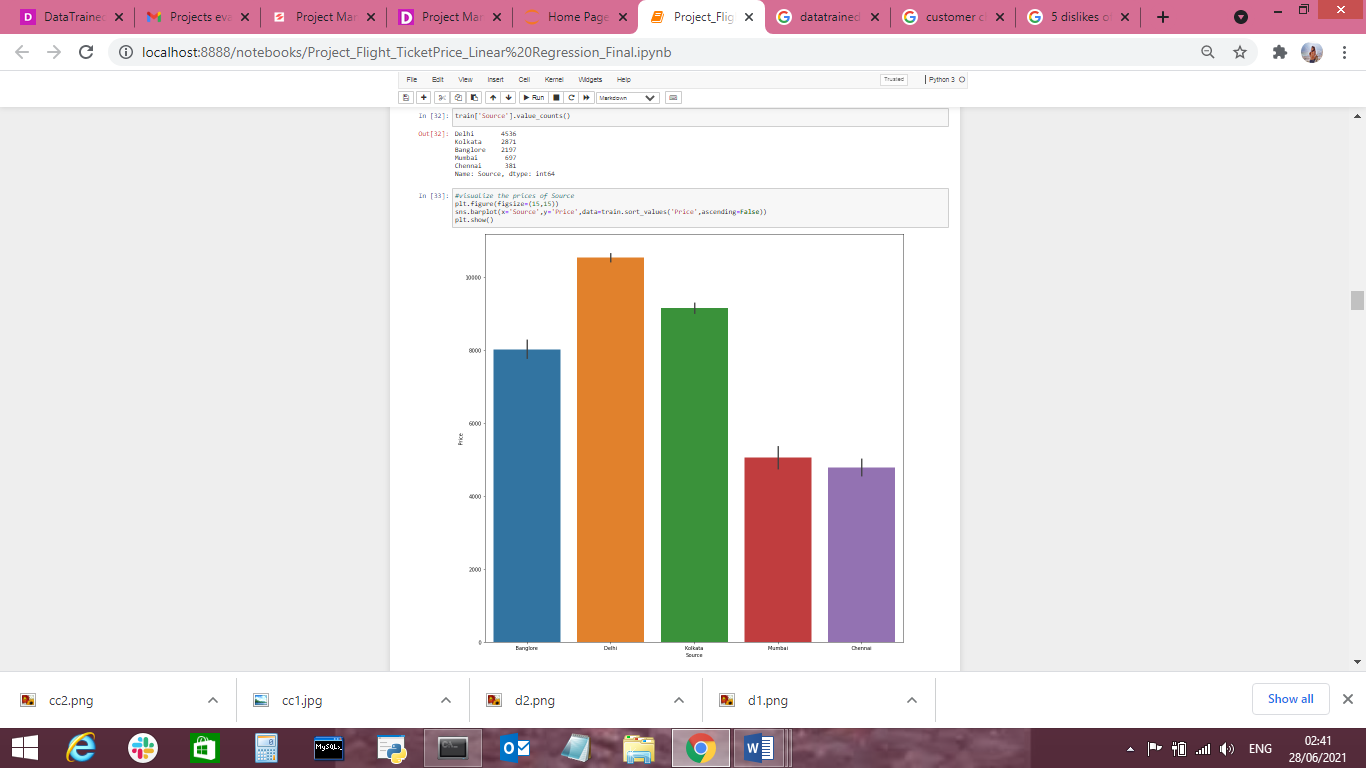
As we can say that only Jet Airways Business is costly,and remaining all are of same median prices.So,it a Nominal value.Use OneHotEncoder for Airline Feature



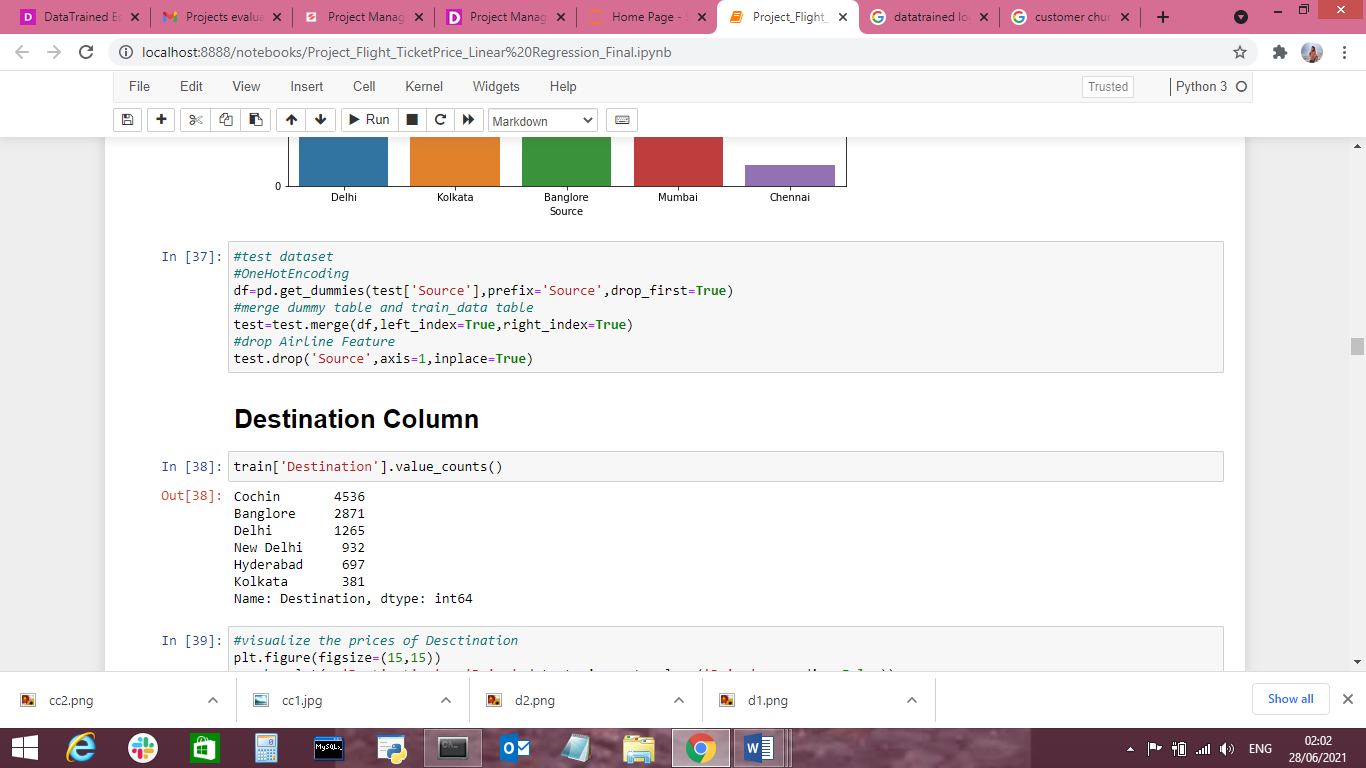


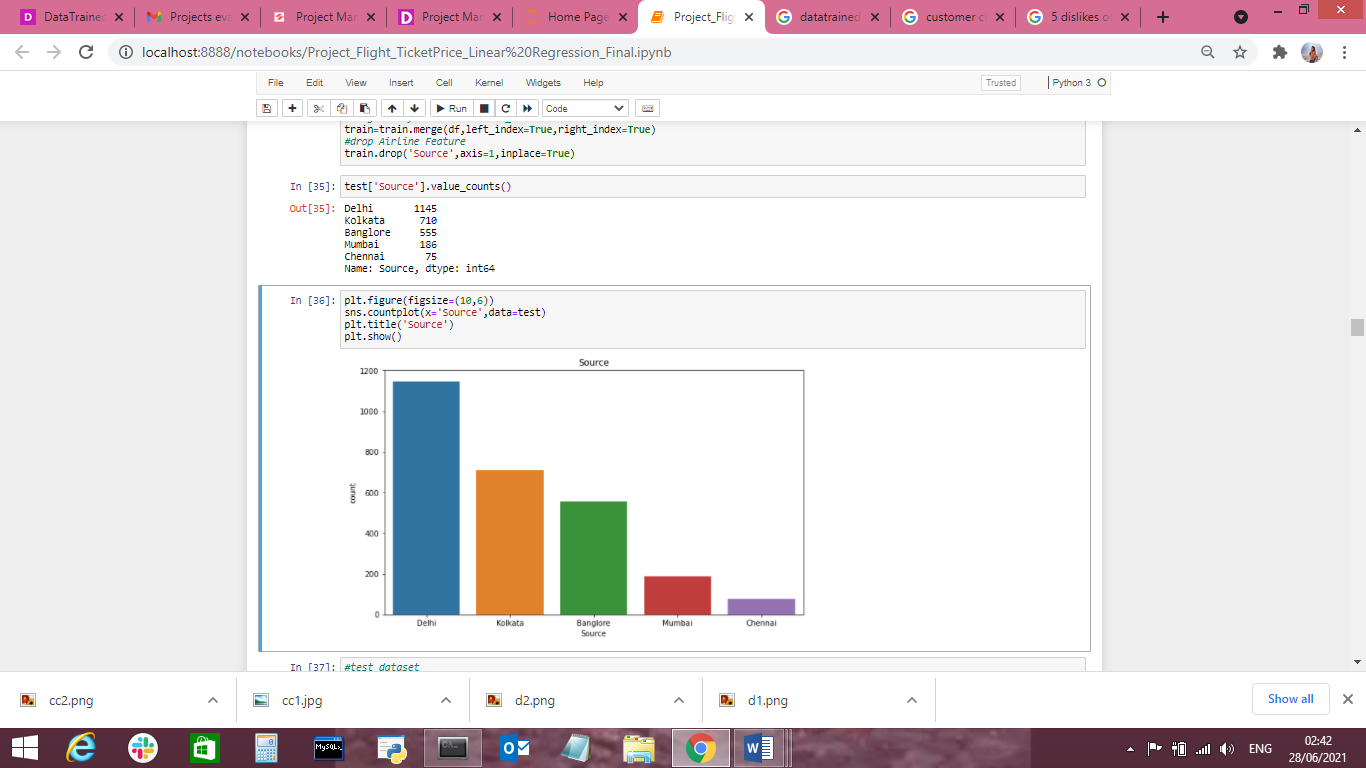
**Source Column**



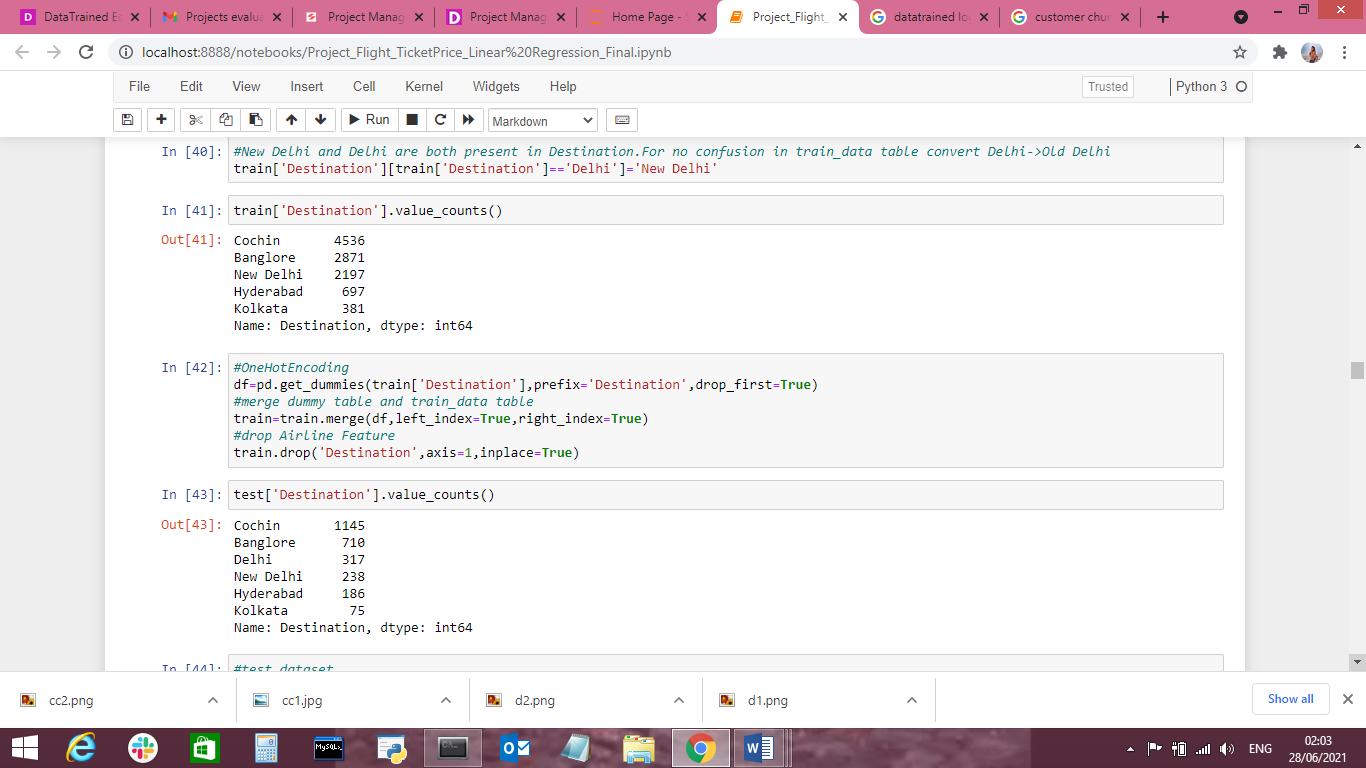


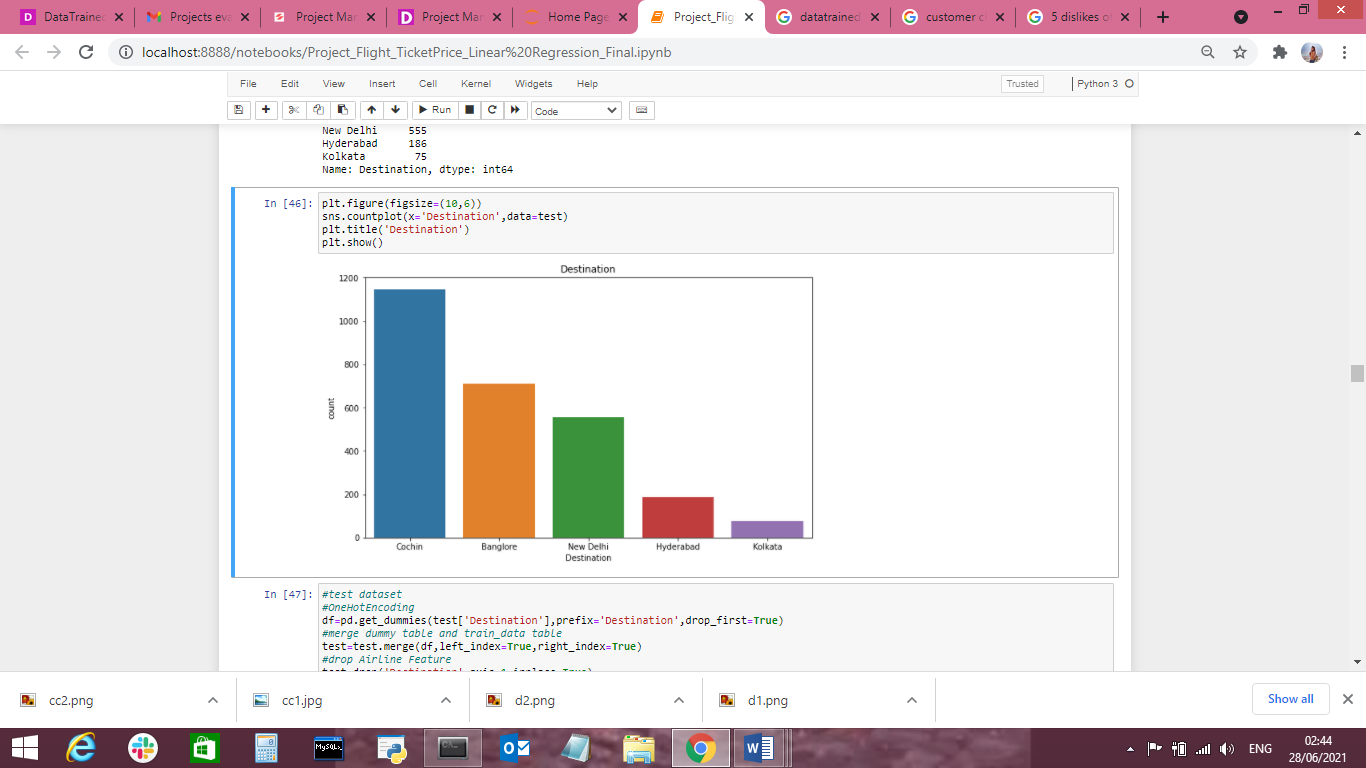
As we can say that only Bangalore have little costlier,and where as remaining all are of same median prices.So,it a Nominal value.Use OneHotEncoder for Source Feature.



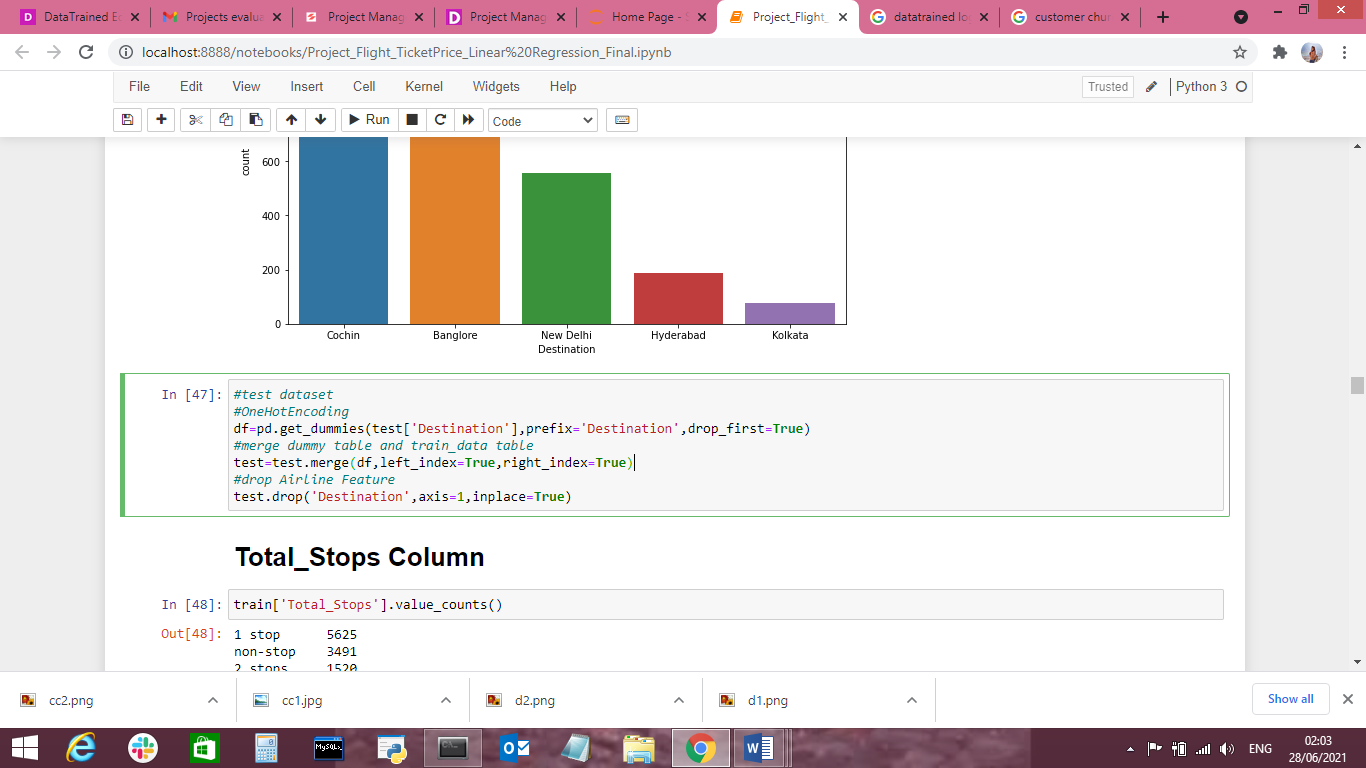


**Destination Column**

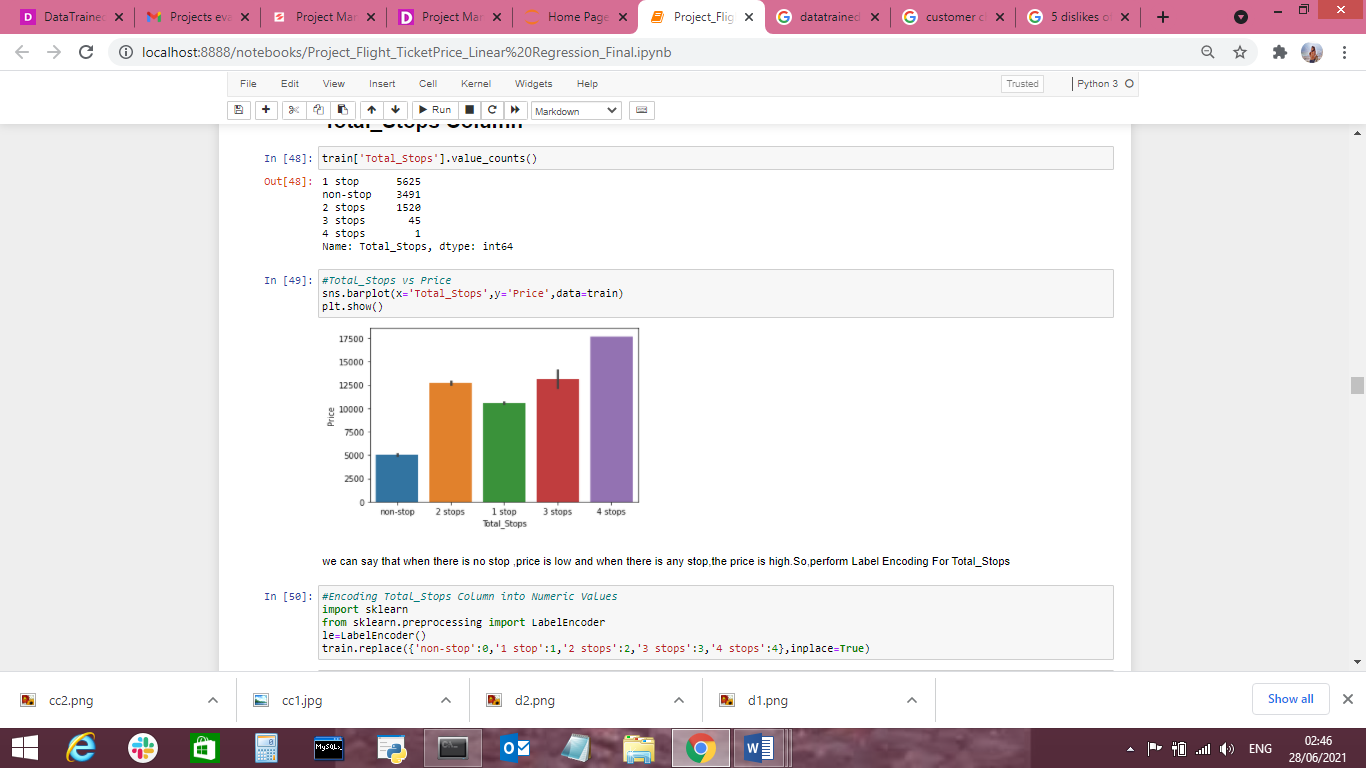


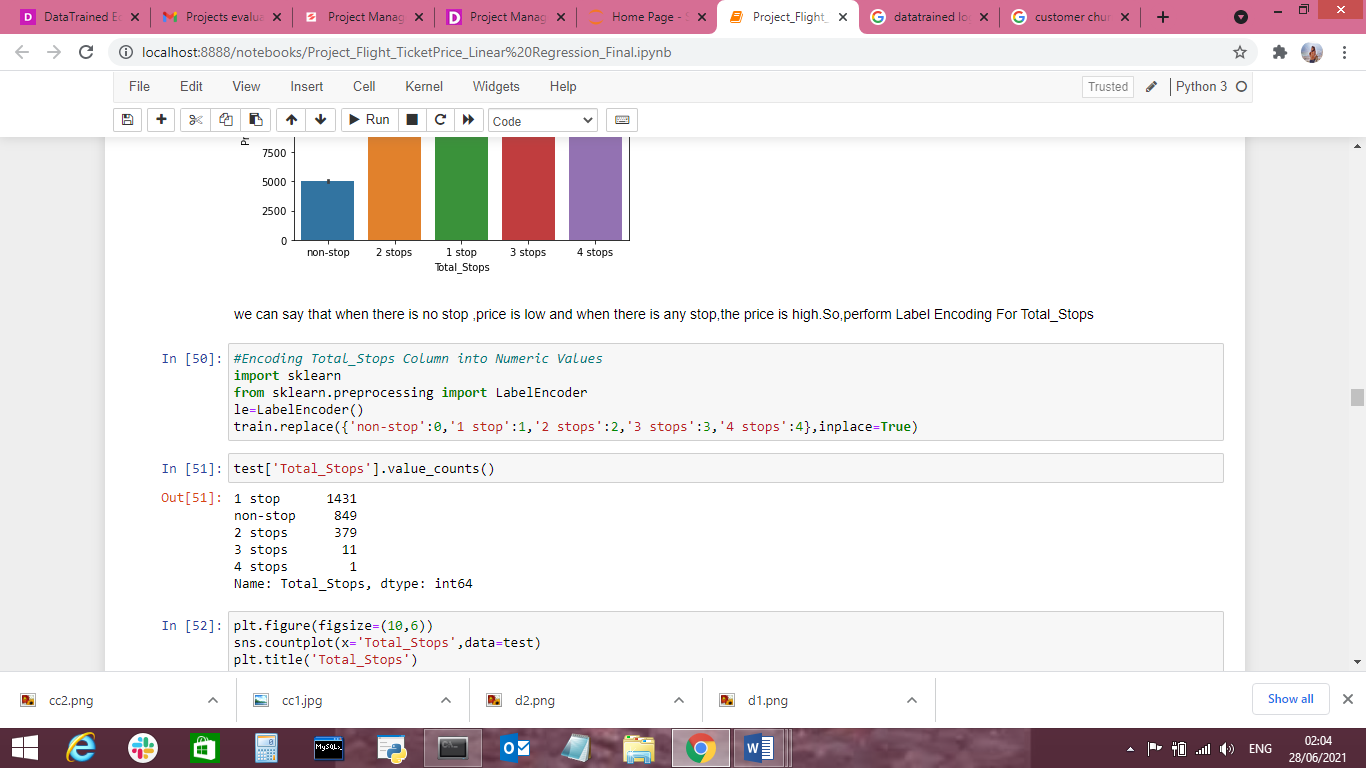


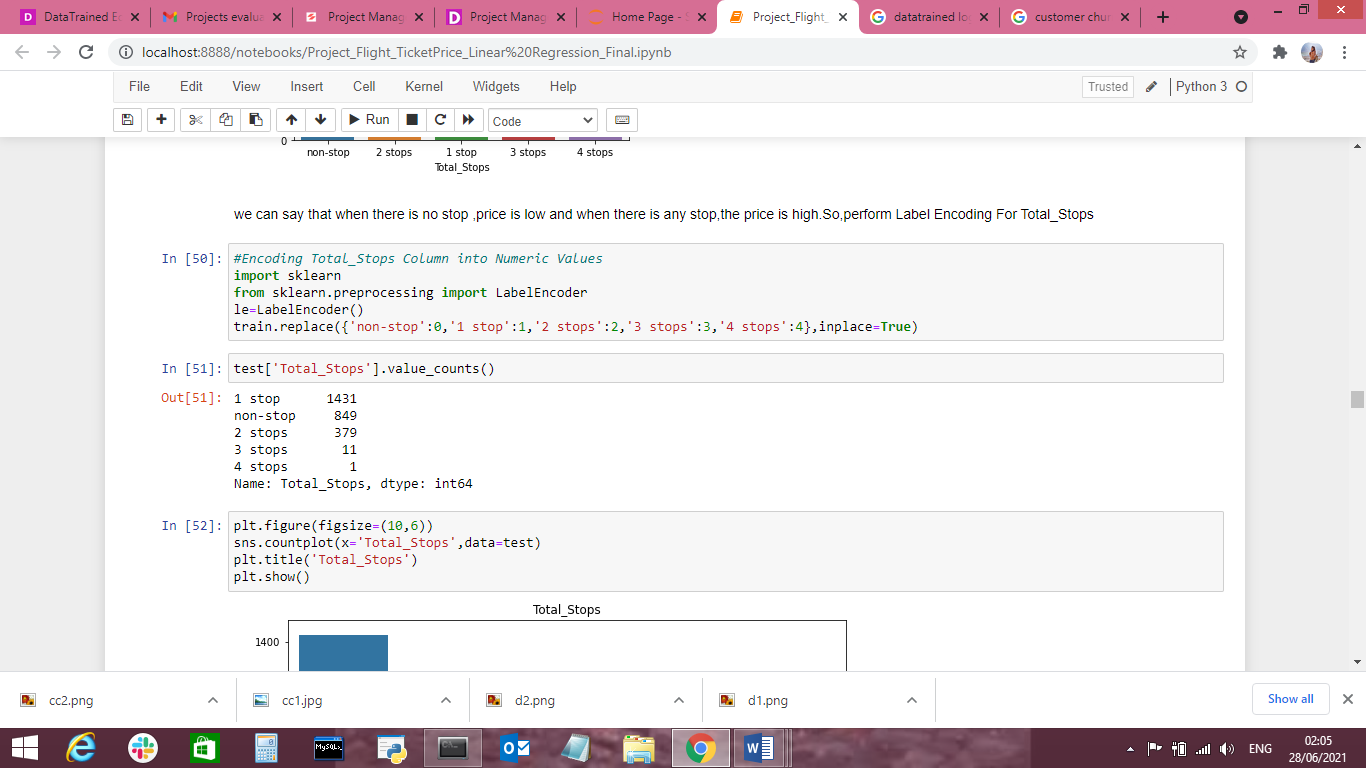
we can say that Destination New Delhi as a slight higher price,where as remaining destinations have the same median prices.So,perfrom oneHotEncoding for Destination feature.



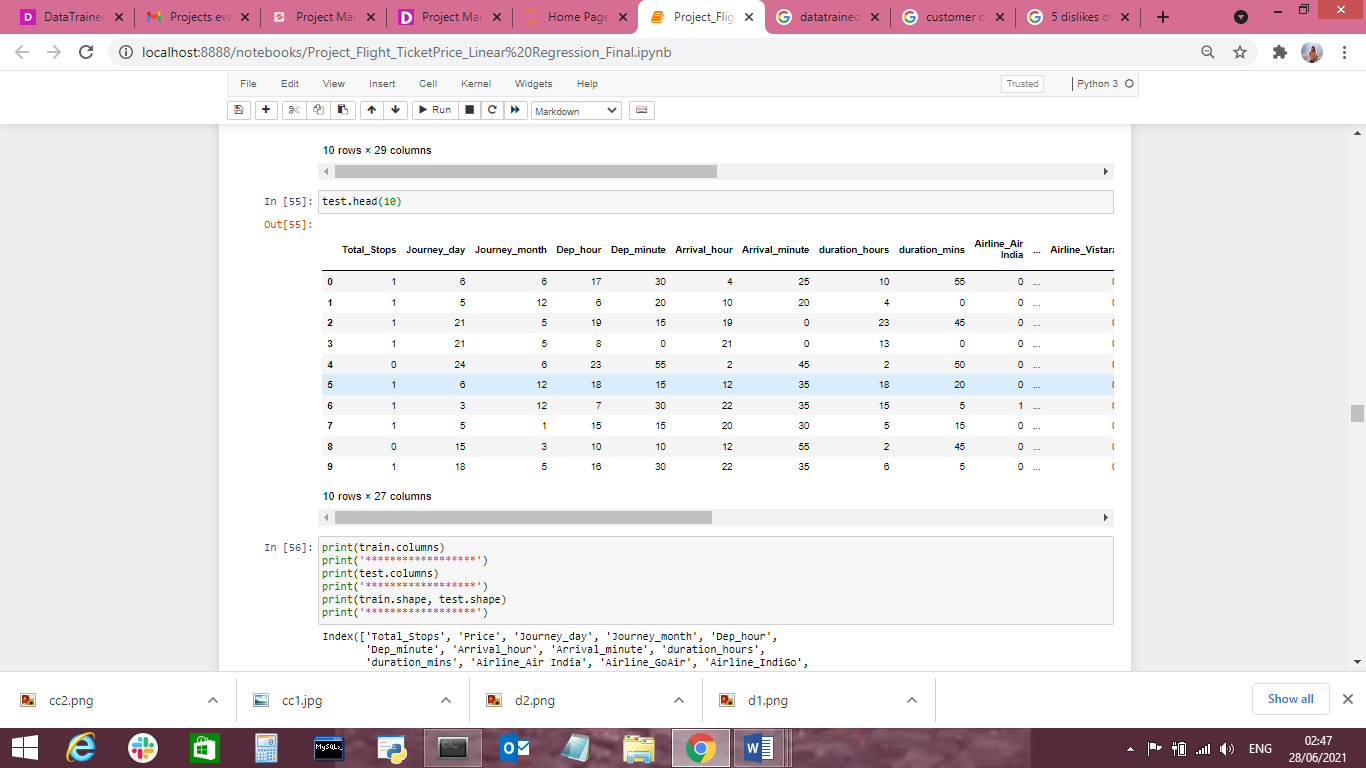
**Total Stop Column**



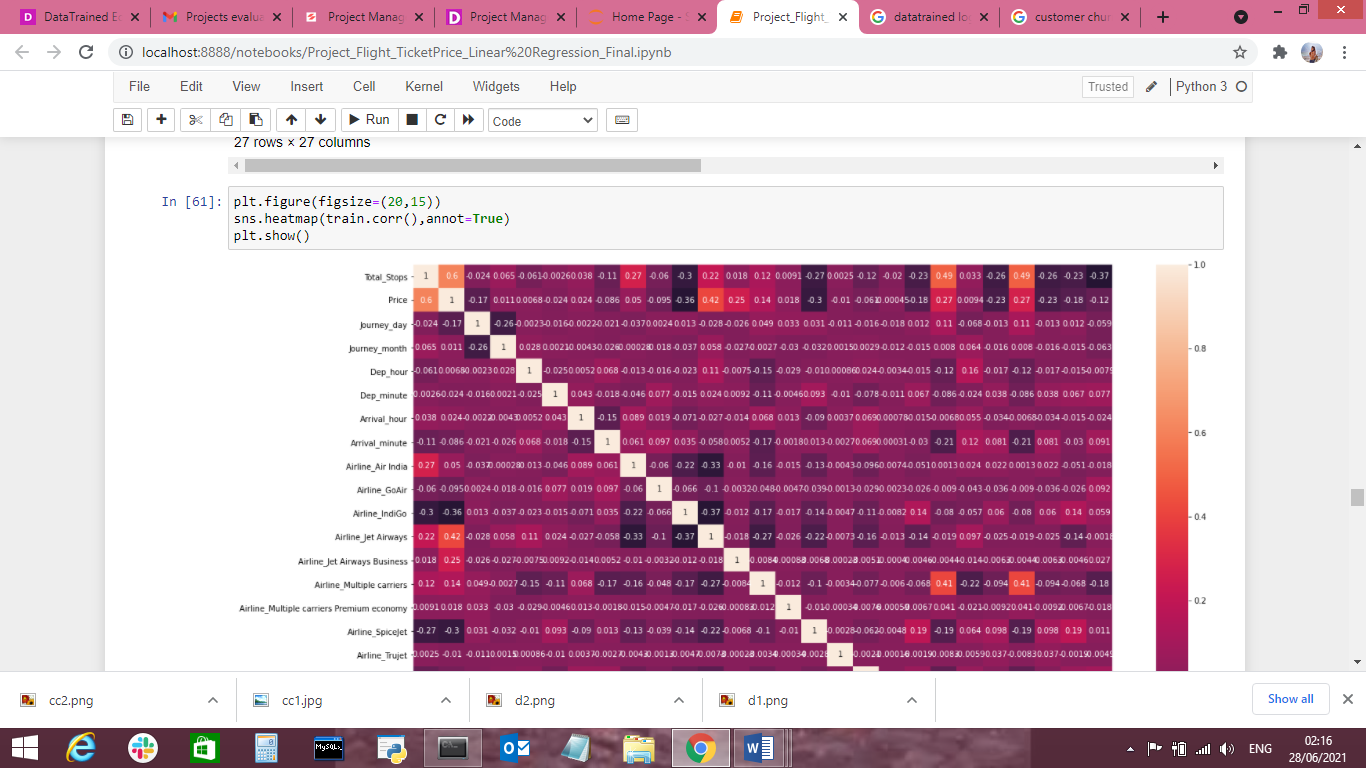


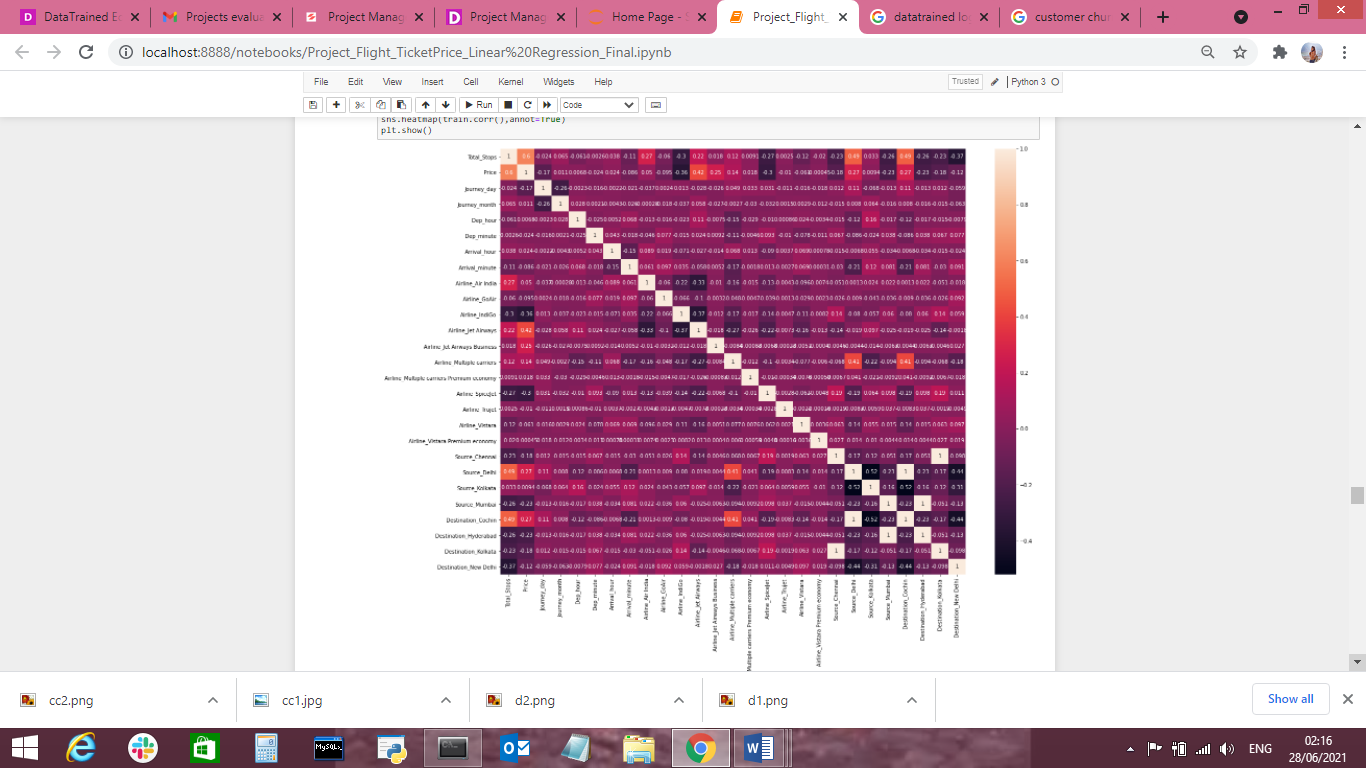


**Semple dataset after worked on categorical data**



**Visualization of Correlation of dependent and independent variables**





**EDA Concluding Remarks**

After performing the transformations, integrations and cleaning of data, we get all the relevant variables and significant information required for building an ML model. We end up having 29 rows and 10,682 records in the dataset. The final dataset consists of important features used for analysis are:

Index(['Total\_Stops', 'Price', 'Journey\_day', 'Journey\_month', 'Dep\_hour',

'Dep\_minute', 'Arrival\_hour', 'Arrival\_minute', 'duration\_hours',

'duration\_mins', 'Airline\_Air India', 'Airline\_GoAir', 'Airline\_IndiGo',

'Airline\_Jet Airways', 'Airline\_Jet Airways Business',

'Airline\_Multiple carriers',

'Airline\_Multiple carriers Premium economy', 'Airline\_SpiceJet',

'Airline\_Trujet', 'Airline\_Vistara', 'Airline\_Vistara Premium economy',

'Source\_Chennai', 'Source\_Delhi', 'Source\_Kolkata', 'Source\_Mumbai',

'Destination\_Cochin', 'Destination\_Hyderabad', 'Destination\_Kolkata',

'Destination\_New Delhi'],

**Concluding Observation Remarks-**

* The standard deviation in the “Route”, “Dep\_Time”, “Arrival\_time”, “Duration” and “Price” column is too high which means that the values in these columns are largely scattered and are not near to the mean value.
* The standard deviation of other columns is not too high which shows us a normal distribution of data and less chances of having skewness.
* The value in the target variable (“Price”) has its minimum price at 1759 and maximum price at 79512. The range is too high.
* The most negatively correlated column is that of the “Total\_Stops” which means more the number of stops less the flight price.
* The most positively correlated variable is “Route”.
* The variables “Route”, “Arrival\_Time”, “Source”, “Month'' and “Dep\_Time” are positively correlated with the Target Variable and variables “Airline”, “Destination”, “Duration”, “Day” and “Total\_Stops” are negatively correlated.

**Pre-processing Pipeline**

**Skewness Correction**

As the range for skewness is threshold +/-0.5, we did not treat skewness in our data. Only variables “Airline”, “Destination”, “Price” and “Month” showed skewness wherein all the input features are of the Object data type and “Price” is the Target Variable. Hence, there is no skewness found in the data.

No need to treat and check outliers and skewness because all column and values are important

**Normalization**

In order to build and train a Machine Learning model, we have to standardize the data and get the values within a particular range for the model to understand data. We have used StandardScaler() technique for normalization because the value ranges are high in the data. Standard Scaler function will get all the values in the dataset within the range of 0 to 1. This will help the ML algorithms to learn data better.

Since the Target Variable is of the continuous type of values, we use Regression Algorithms. In this case, we have used **Linear Regression**.

**Building Machine Learning Models**

Building a model that will help to measure the performance of a better and more refined algorithm is the major goal here. We have used different Regression and Ensemble Techniques to compare and check which algorithm gives better performance and stack them all at the end to see how the model is giving predictions.

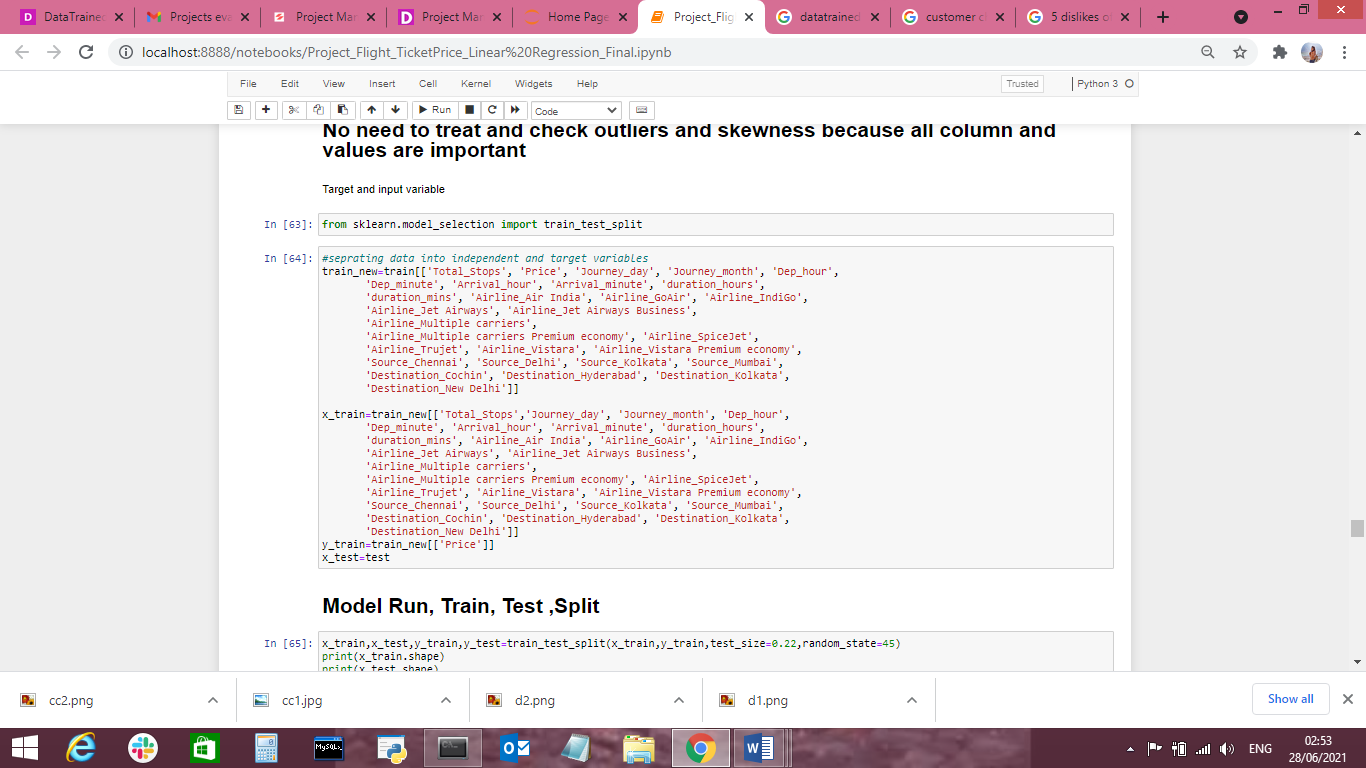
The goal on this step is to increase a benchmark model that serves us a baseline, upon which we are able to degree the overall performance of a better and more tunned algorithim. We're the usage of specific regressin method and comparing them to look which algorithim is giving higher overall performance then other and we can additionally go for a move validation check for due to the fact that the model isn't in overfitting/underfitting .

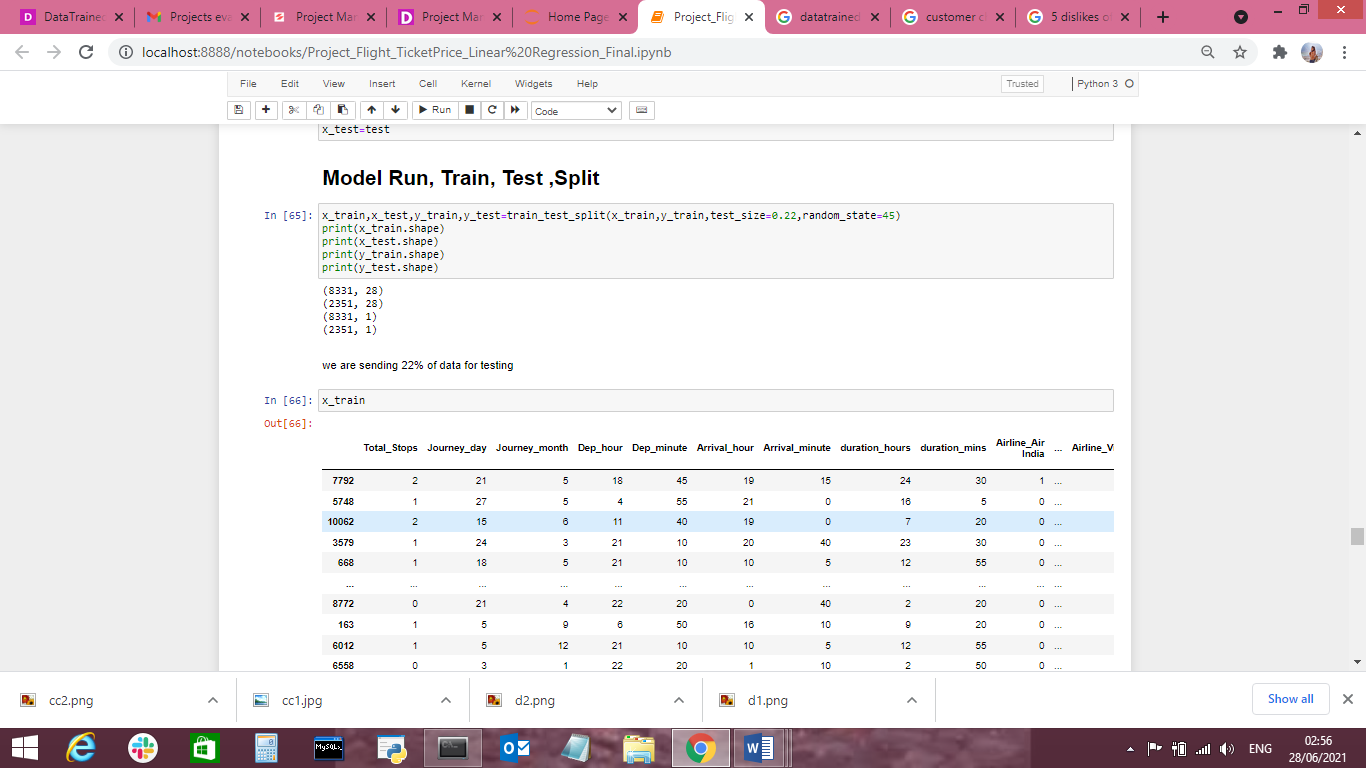
I'm the usage of linear regression from linear model, svr from svm, decidion tree regressor from the tree and also the use of a few ensemble techinques like random forest regressor, gradient boosting regressor and ada improve regressor.

Due to the fact our version is regression type version so we use r2\_score, mean\_squared\_error, and mean\_absolute\_error.

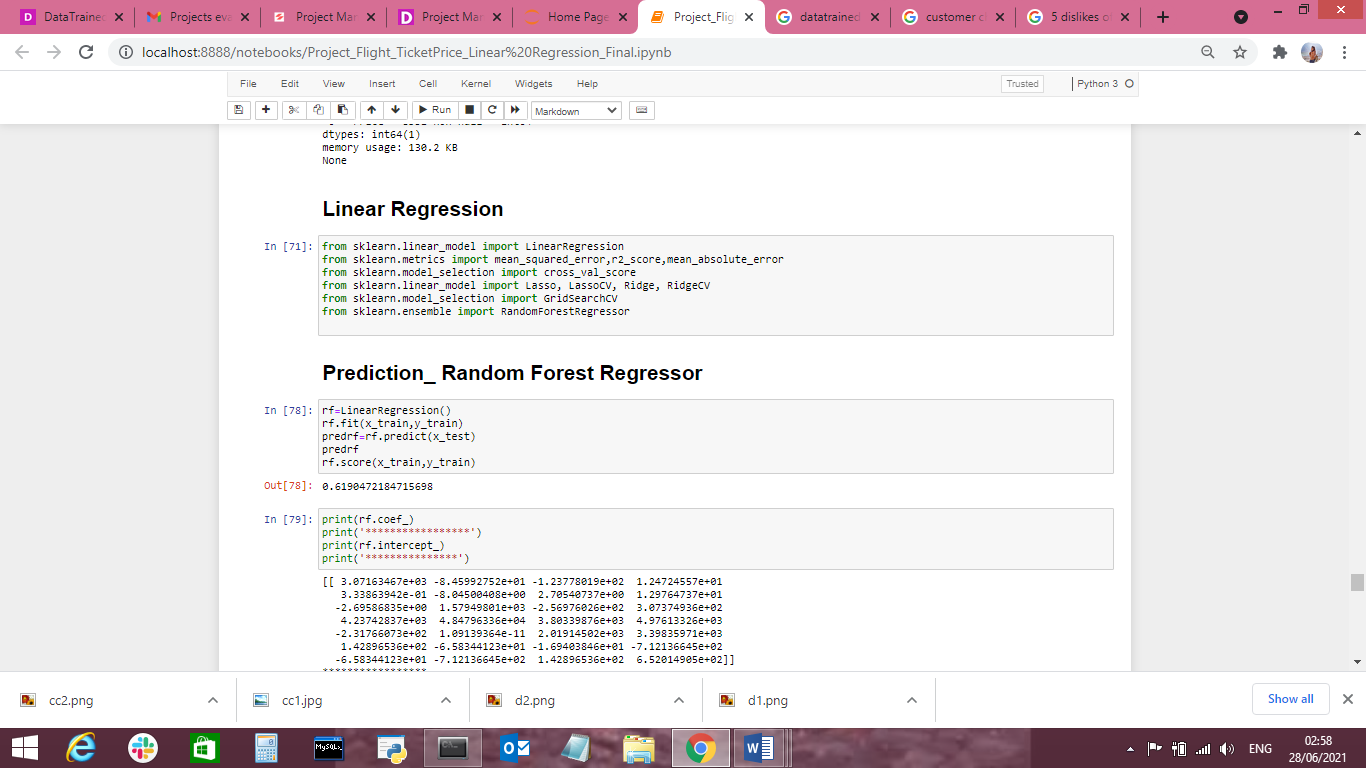
**Linear Regression**

Linear Regression is a Supervised Machine Learning algorithm that performs Regression tasks. Simple Linear Regression analysis is used to identify the correlation between two continuous variables. Prediction error is minimum when we find the best fit line for the given data using Linear Regression Algorithm.

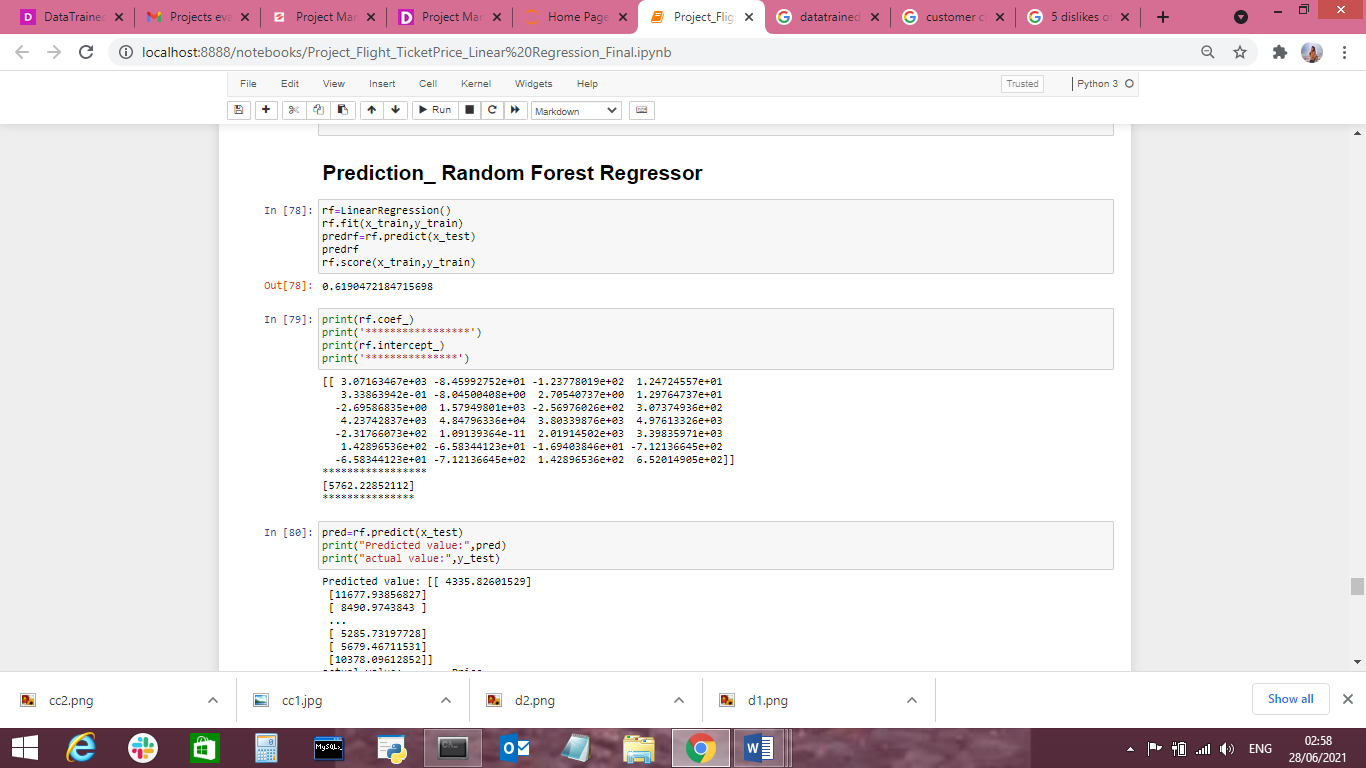


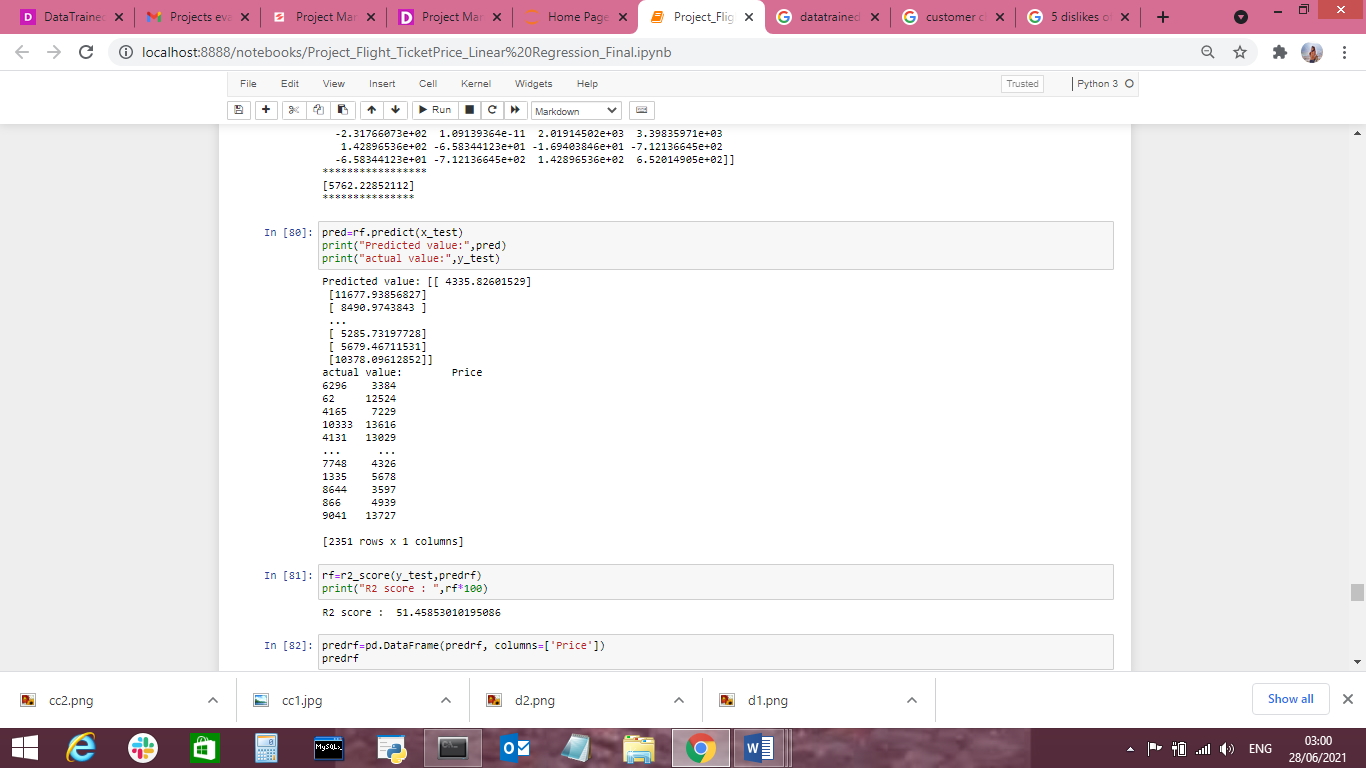


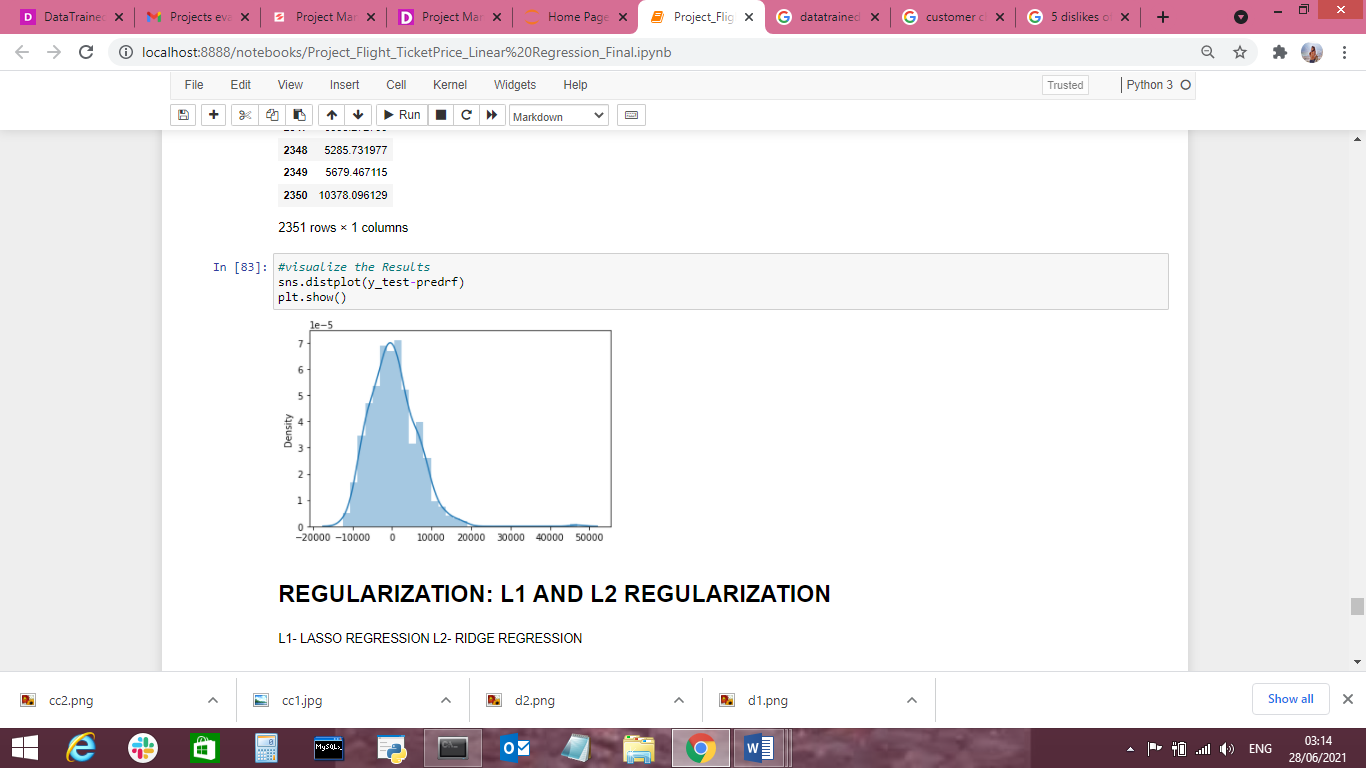
we sent 22% of data for testing



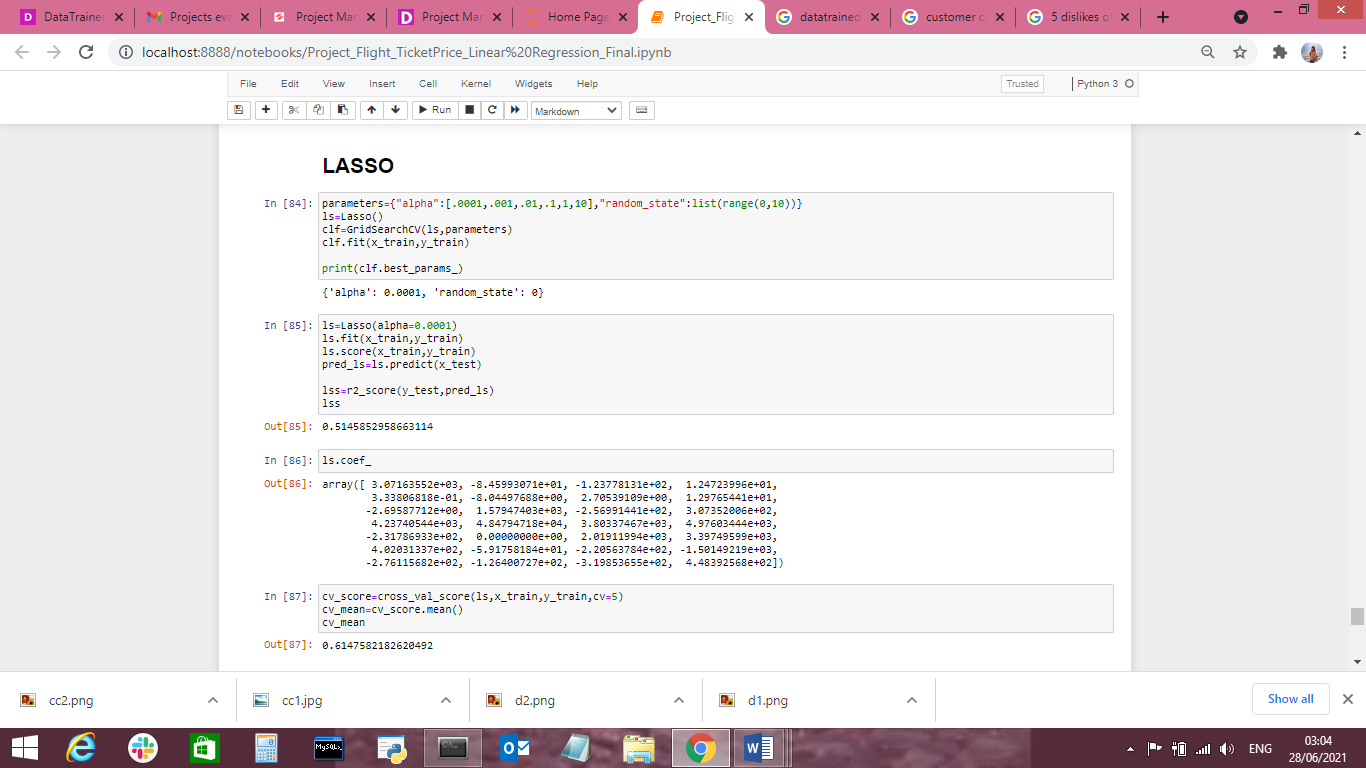
Here we use Random Forest Regressor to check the best R2 score of model run and coefficient value, interception value and prediction of model.

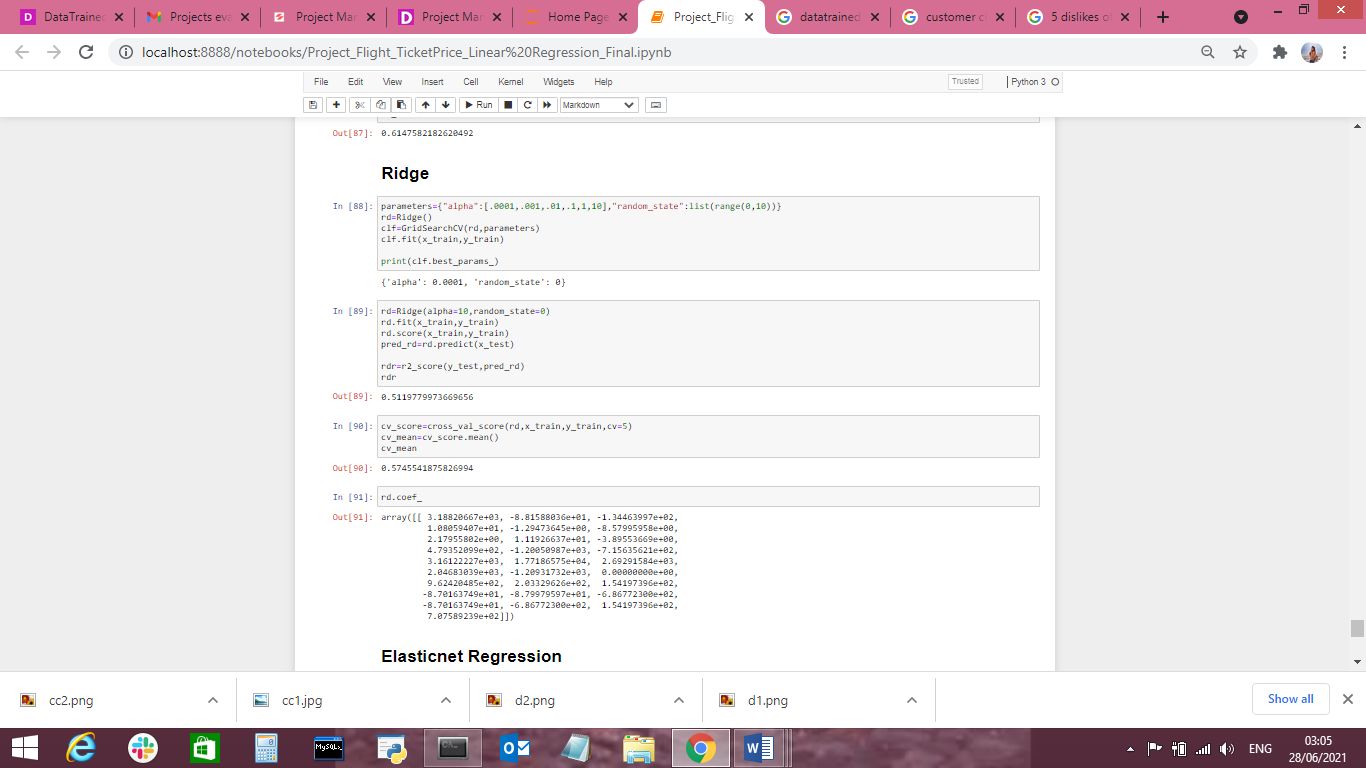






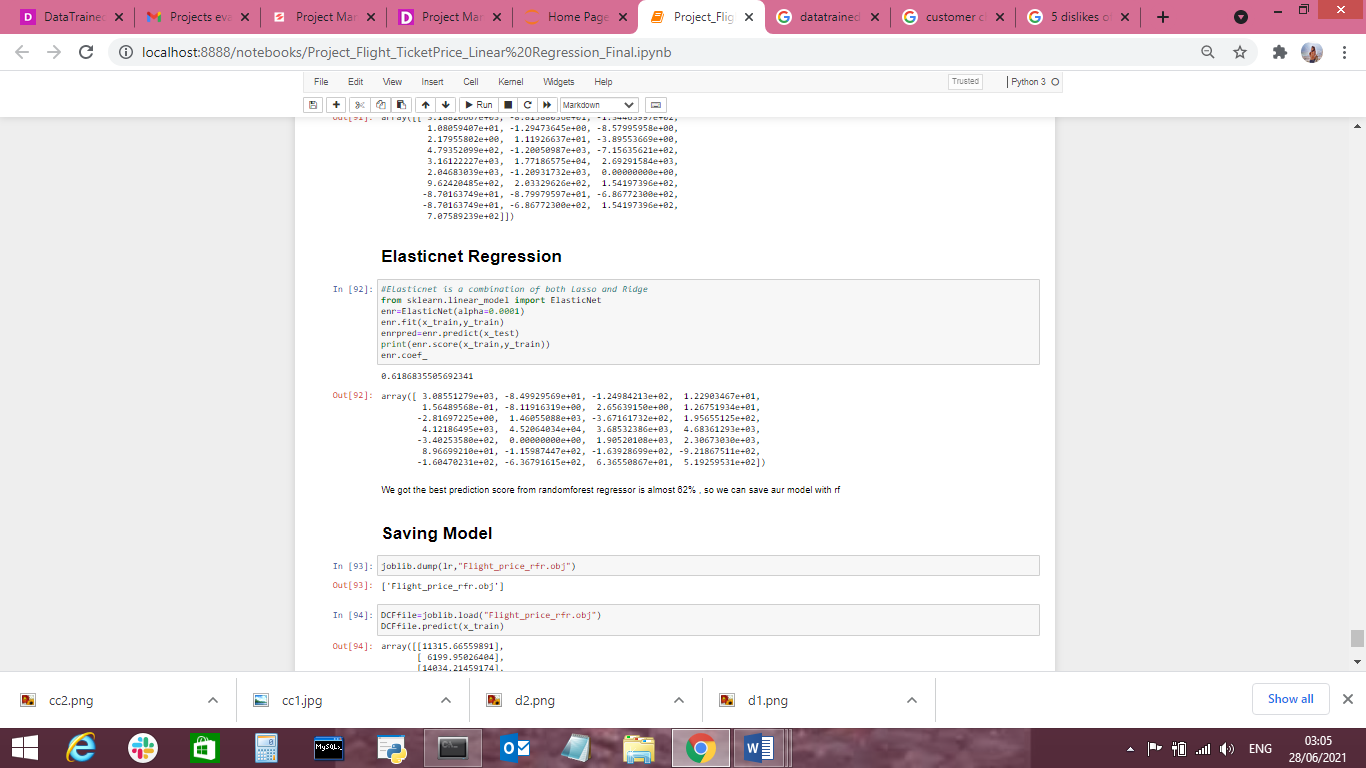
We also used Lasso and ridge regression to check the r2 score, coefficient score and prediction value of model.





A **cross-validation** technique was applied to all the samples and the mean performance of the model is produced.

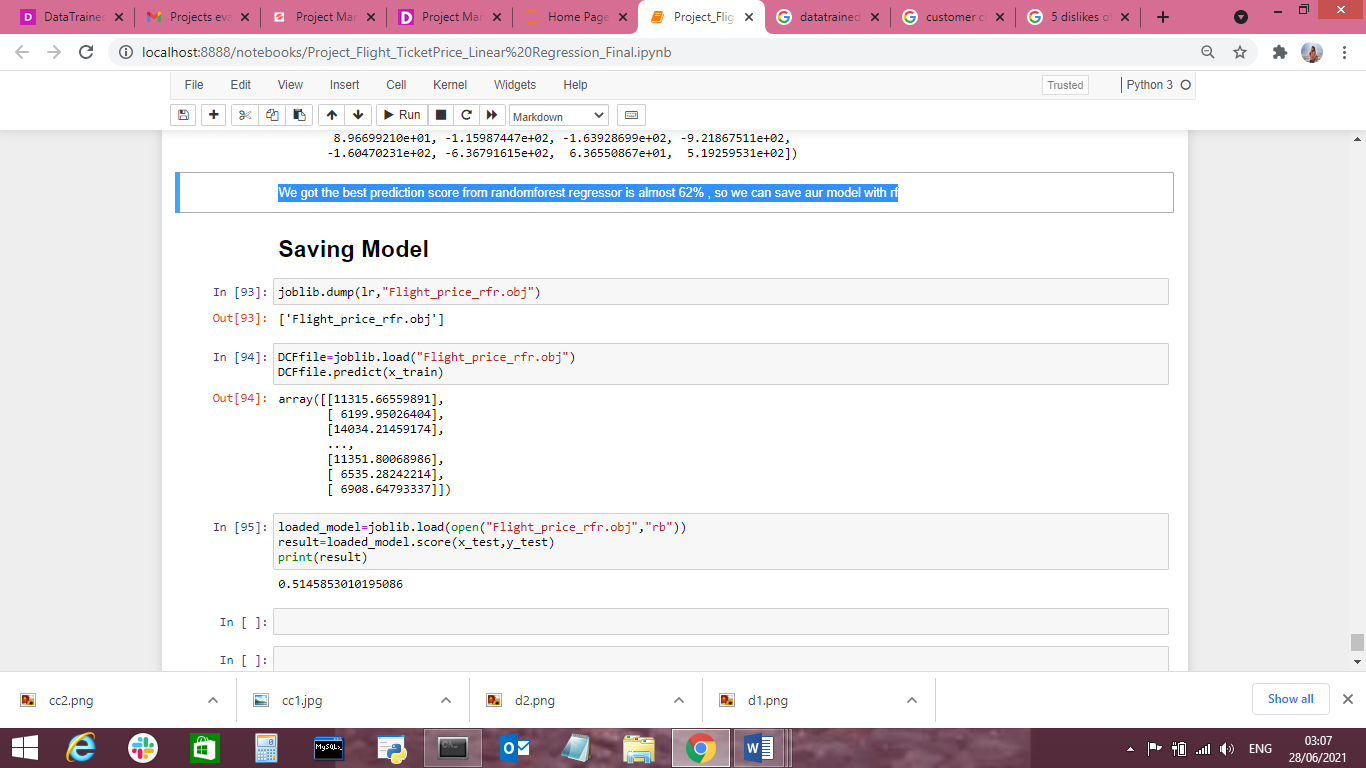
The **best fit line** of the model was seen covering almost all data points which is an illustration of getting the best accuracy and indicates that the model has studied all the points in the data and there are no chances of having overfitting and underfitting issues.



**ElasticNet Regression**

ElasticNet Regression is a regularization regression technique which uses the best of both lasso and ridge regression models by learning from their drawbacks to better the regularization of statistical methods. Hence, we have applied this regularization technique to our regression model to avoid the risk of overfitting. The Regularization technique was put in use with the help of GridSearchCV.

Now as we can see that We got the best prediction score from randomforest regressor is almost 62% , so we can save our model with rf.



**Concluding Remarks**

* There are some instances when an offer is run through an airline due to which the expenses drop all of sudden. These are difficult to comprise in our mathematical models, and for this reason lead to error.
* The Flight Price varies depending at the time of departure, making timeslot used in analysis an critical parameter.
* The r2 score achieved for Linear Regression is 51.45%.
* The accuracy achieved after applying cross validation and ElasticNet Regularization Regression technique comes out to be 61.5%.
* Applying the Ensemble Techniques to these Regression Models in order to achieve stability in the performance of the model, we get r2 score at 51.7% and CV val score at 57%.
* As we save it as the best model.

**Thanks to Visit MY Blog**