**Flight Price Prediction using Machine Learning**

**Introduction:**

This is a blog on the title “Flight ticket price prediction”, a project on machine learning which we will be doing in python. We will go through the entire steps for building a machine learning model that can be used to predict the flight ticket price and we will be understanding them thoroughly. The topics that we are going to cover are:

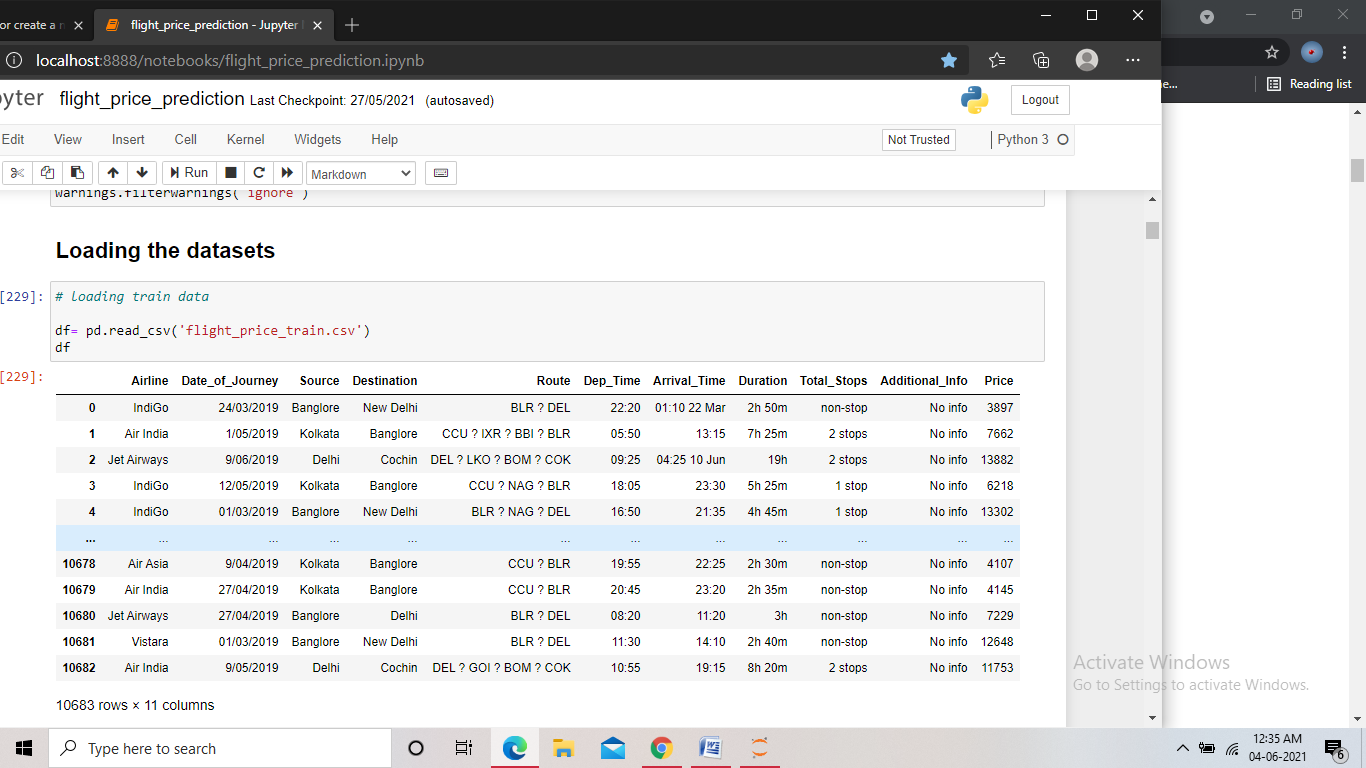
* Problem Definition
* Data Analysis
* EDA Concluding Remarks
* Pre-Processing Pipeline
* Building Machine Learning models
* Concluding Remarks

**Problem Definition:**

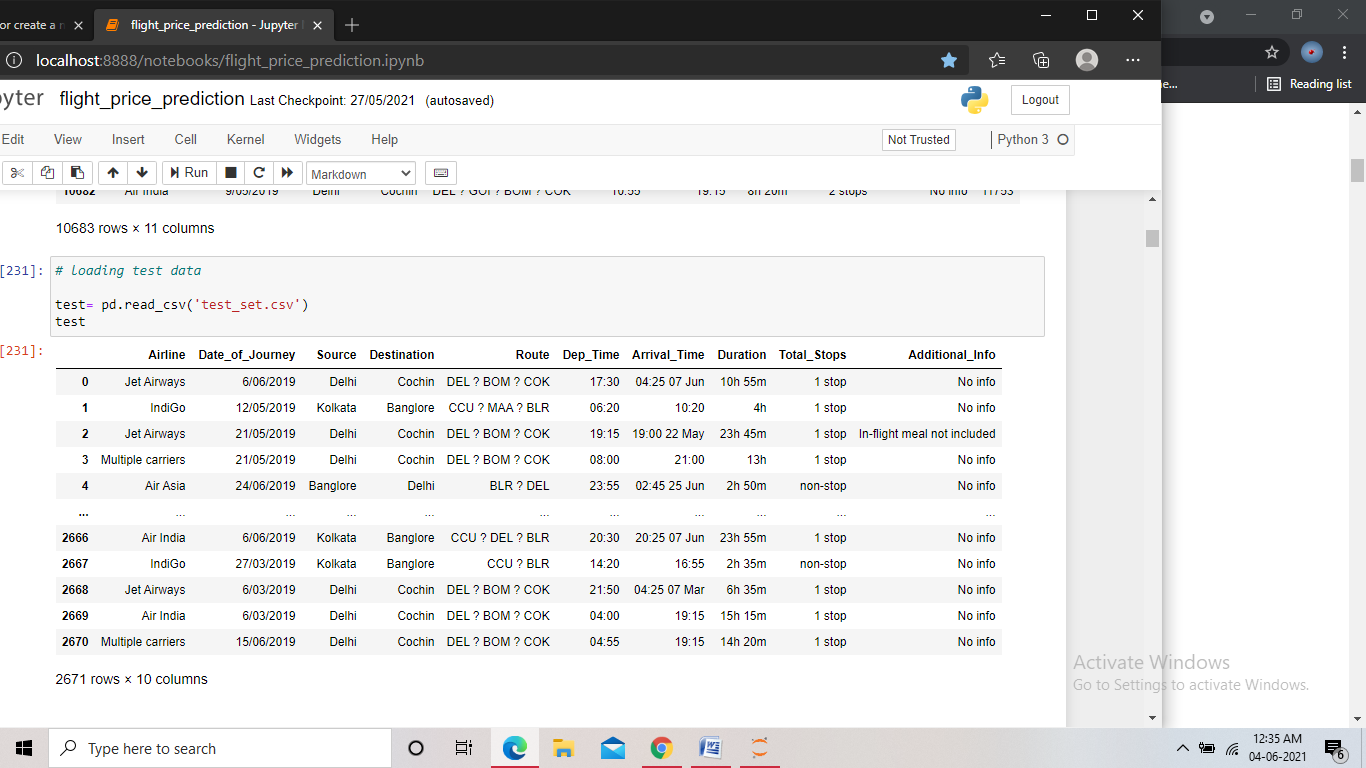
Nowadays more and more people are choosing flight as a travel option, and as the majority of the population in the nation are from middle income household, price of the flights play an important part in choosing a flight for travel. But it may be difficult for a person to know the exact price of the tickets as the price of the flights keeps on fluctuating and is very difficult to predict. Here, machine learning comes into play. By using the price and other data of the previous flights that have operated earlier, we can create a model that can predict the price of the tickets for the upcoming flights.

**Data Analysis:**

In this project, we are provided with prices of flight tickets for various airlines between the months of March and June of 2019 and between various cities. Also we are provided with two datasets- Train data and Test data.



Screenshot of the train dataset



Screenshot of the test dataset

Both the datasets are similar, with the Test data not having the ‘price’ column. Using the Train dataset we have to train and validate our model, and using that model we have to predict the price in the test dataset. The features that are present in the datasets 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.

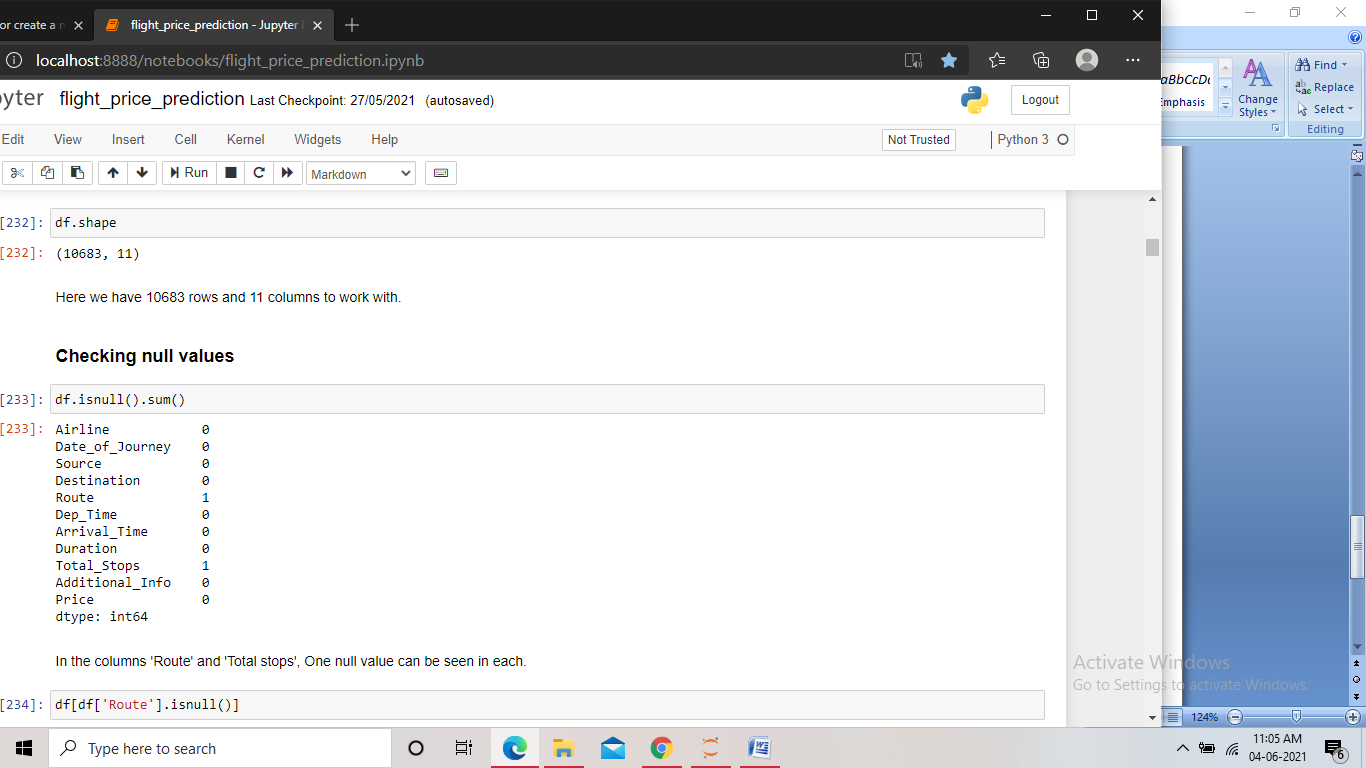
**Additional\_Info**: Additional information about the flight

**Price**: The price of the ticket

The price column is the target column here.

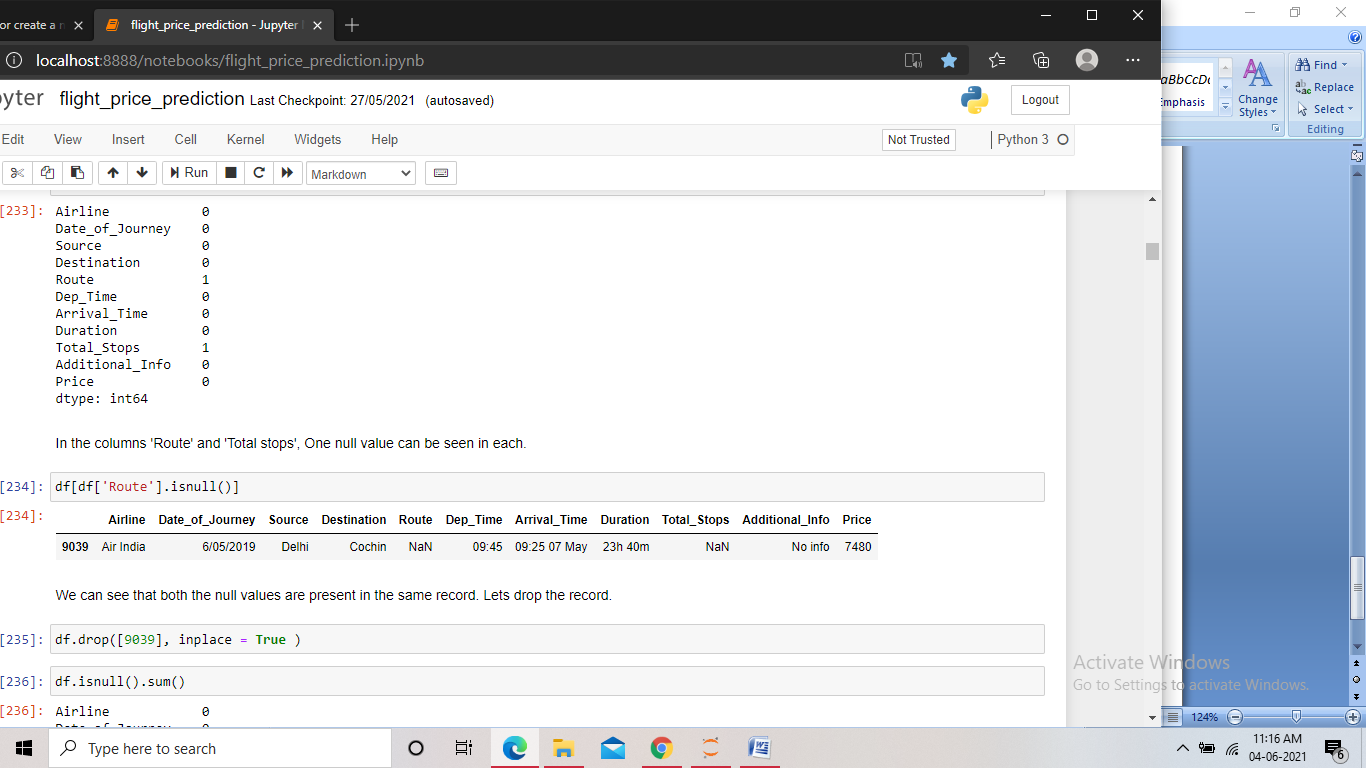
**EDA Concluding Remarks:**

First of all we check the shape of the dataset to get an idea about the size of the data. Then we check for any missing values in the dataset, so that we can treat them.



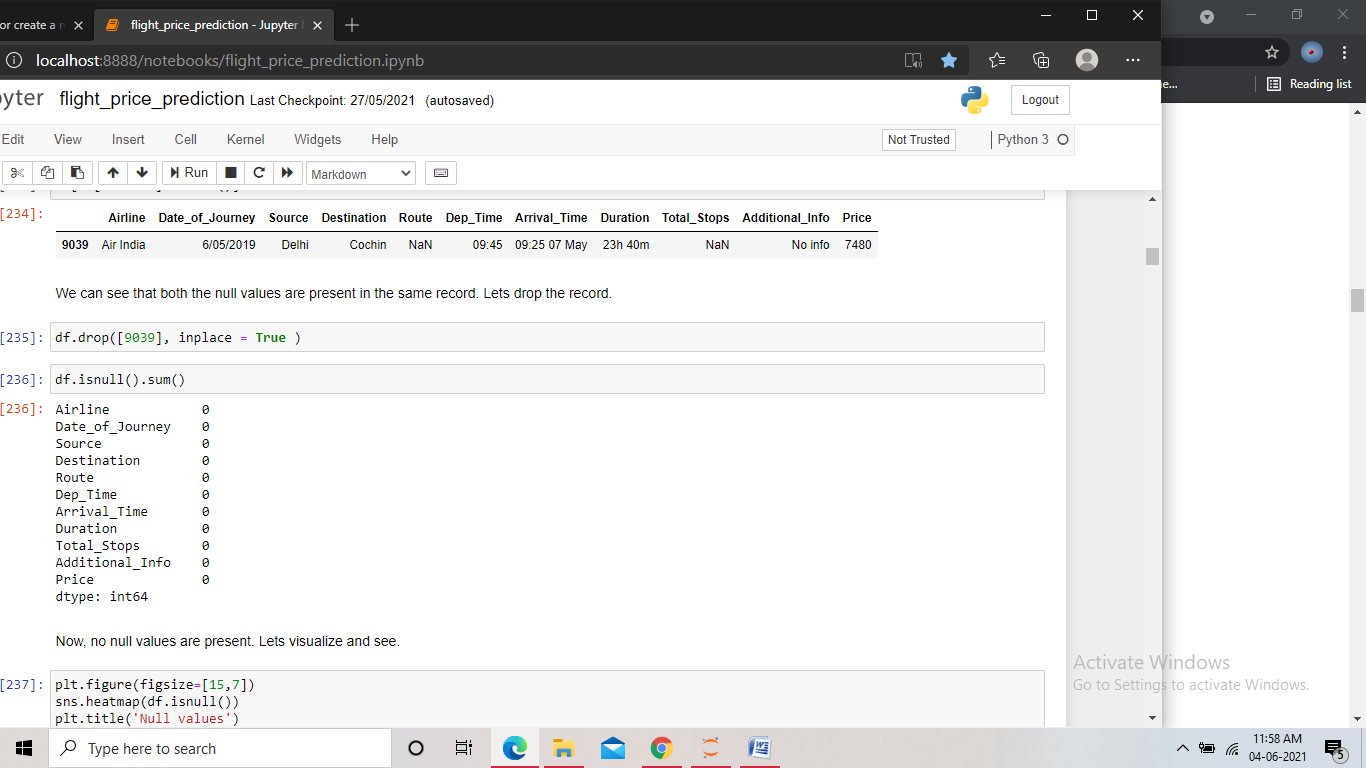
Screenshot of the code

We have found the shape of the train data to be of 11 columns and 10683 rows. Also we have found two missing values in two columns each. After this we check the record that contains the missing values.



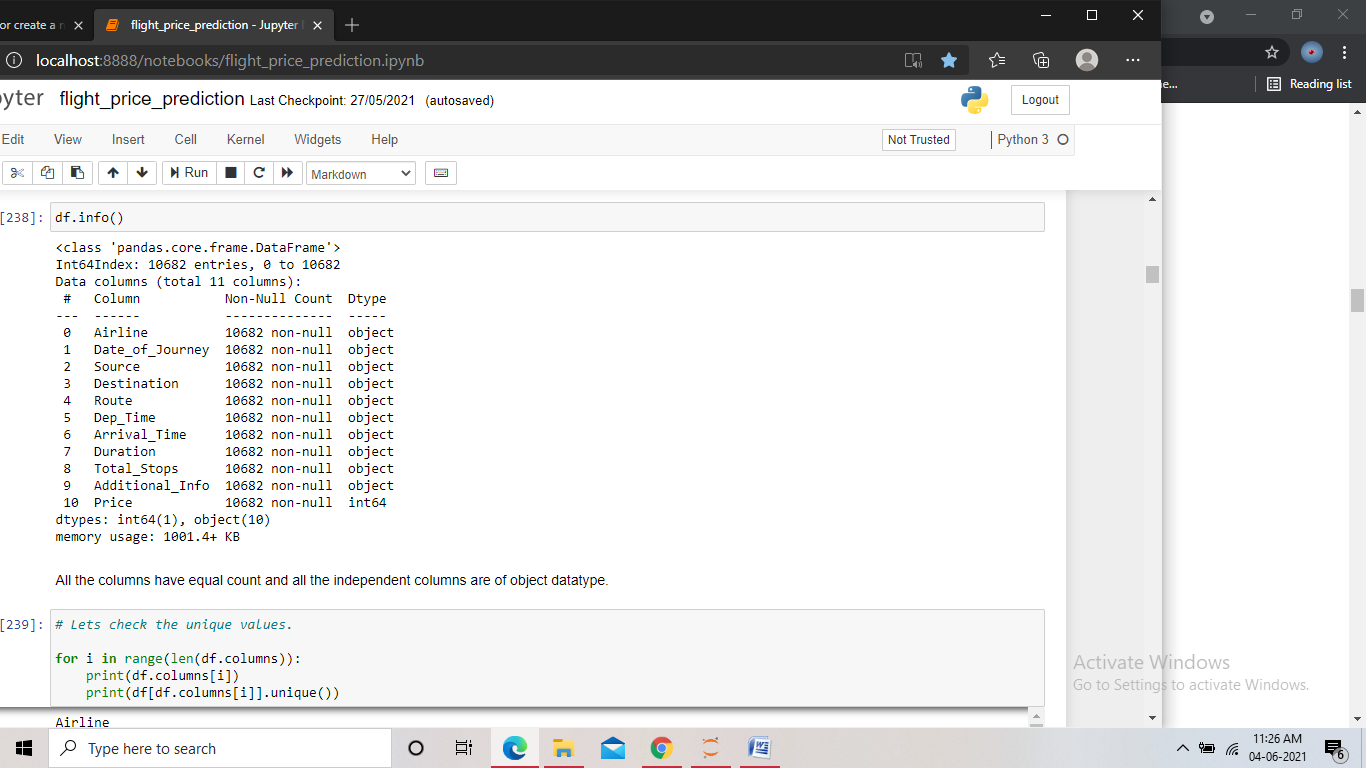
Screenshot of the code

We found that both the missing values belong to the same record. So, we can drop the record to make the dataset free of null values.



Screenshot of the code for dropping the record.

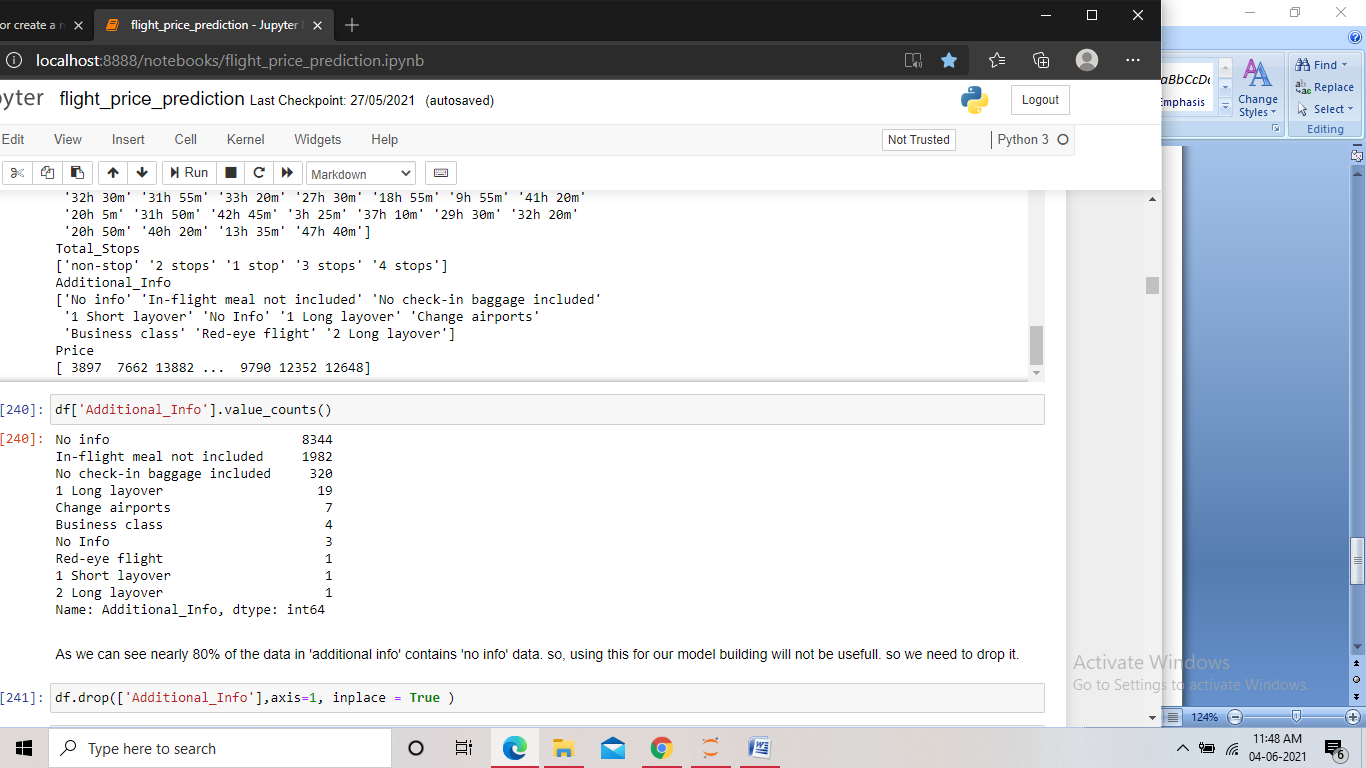
Then we check the info of the data and the unique values present in it.



Screenshot of the code

In the info we check the counts of the data in the columns and the data types; we found all the columns except the “price” column to be of object type. We need to convert them to integer or float for the model building. Using the code to check for the unique values, we check all the unique values of the columns.

Analyzing the unique values we found many columns containing date and time in the records, we need to extract the dates, months and years to separate new columns. We then check the value counts of certain columns.



Screenshot of the code

We found the “Additional info” column containing value “no info” in the nearly 80% of its records. So we can drop the column.