THIRUVALLUVAR UNIVERSITY

PERIYAR ARTS COLLEGE

CUDDALORE – 607001.

DEPARTMENT OF COMPUTER APPLICATIONS

MACHINE LEARNING WITH PYTHON

Project Title : Optimizing Flight Booking Decisions through Machine Learning Price Predictions

Team Id : NM2023TMID35485

Team Leader : VIGNESH V(tvu10520CA48)

Team member: SURIYAPRAKASH S(tvu10520CA45)

Team member: ARUN P(tvu10520CA09)

Team member: SILAMBARASAN A(tvu10520CA41)

Introduction

This report contains the information about project of Optimizing Flight Booking Decisions through Machine Learning Price Predictions.

Overview:

Flight booking decisions can be difficult, with travelers often struggling to balance their budget constraints with their preferred travel dates and airline preferences. Machine learning (ML) can help travelers make more informed decisions by predicting flight prices based on historical data and current market trends.

ML algorithms can analyze large amounts of data from various sources, such as airline websites, travel agencies, and social media platforms, to identify patterns and trends that influence flight prices. By using these patterns, algorithms can make accurate predictions on how prices will change in the future, enabling travelers to make more informed decisions on when and where to book their flights.

With the help of ML-powered price prediction tools, travelers can optimize their flight booking decisions and save money by booking at the right time and choosing the best airline and travel dates for their needs. Additionally, airlines and travel agencies can benefit from these tools by improving their pricing strategies and increasing customer satisfaction.

Overall, the use of ML in flight booking can greatly improve the travel experience for both travelers and businesses, making it easier and more efficient to book flights and travel to new destinations.

Purpose:

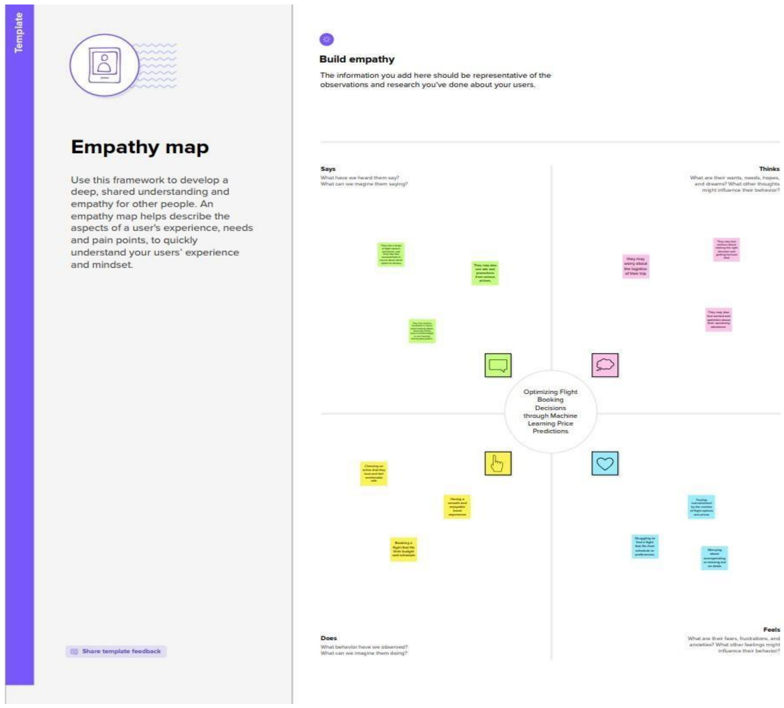
The purpose of using machine learning for optimizing flight booking decisions through price predictions is to help travelers make more informed decisions about their flights, leading to cost savings and a better travel experience. The project aims to provide accurate predictions of flight prices based on historical data and current market trends, which can help travelers determine the best time to book their flights and choose the most cost-effective travel dates and airlines.

The use of machine learning in this project also benefits airlines and travel agencies by enabling them to optimize their pricing strategies, increase customer satisfaction, and gain a competitive edge in the industry.

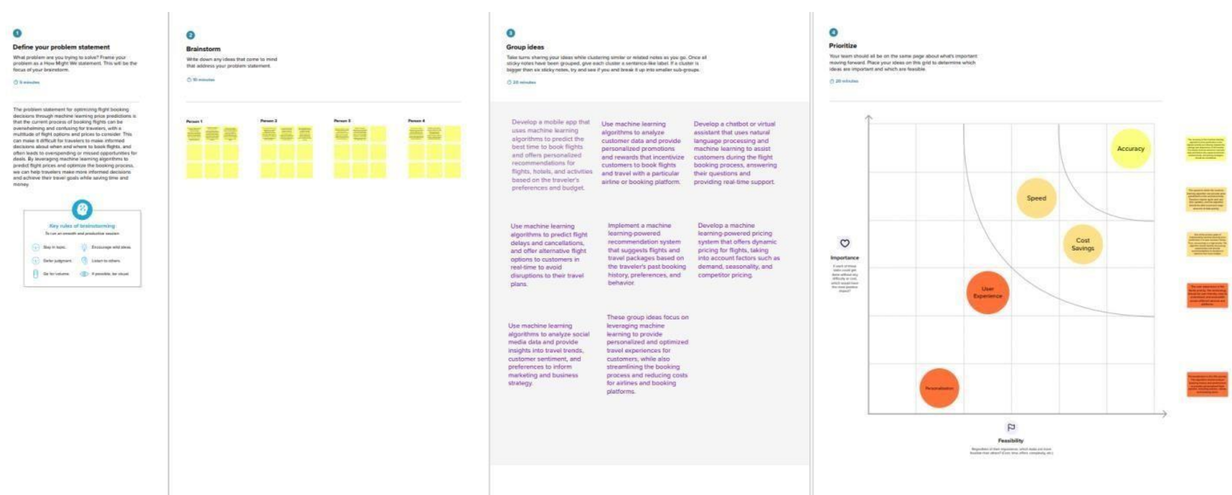
Overall, the purpose of this project is to leverage the power of machine learning to improve the travel experience for both travelers and businesses, making it easier and more efficient to book flights and travel to new destinations.

Problem Definition and Design Thinking:

Empathy map:



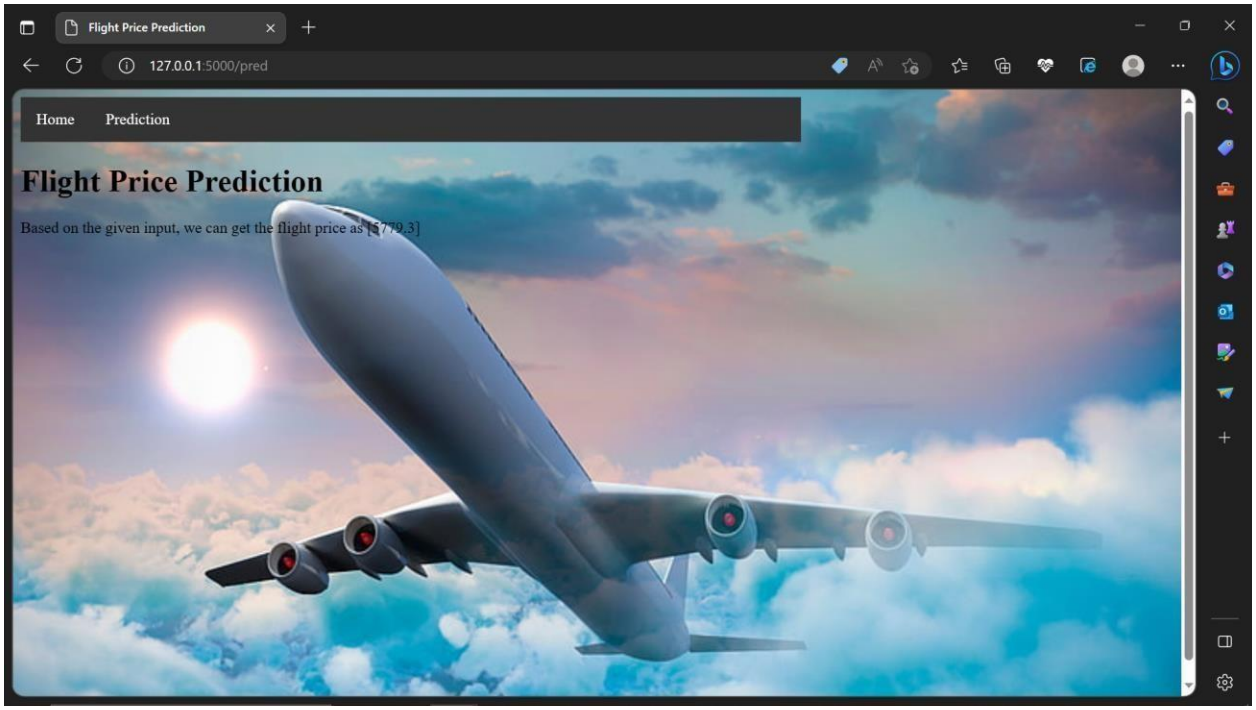
Ideation and Brainstorming :



Result:

The use of machine learning for optimizing flight booking decisions through price predictions has the potential to revolutionize the travel industry. By analyzing large amounts of data from various sources, machine learning algorithms can make accurate price predictions and help travelers make more informed decisions about their flights.

Output: (final findings)



Advantages:

- **Accuracy:** Machine learning algorithms can analyze vast amounts of data and identify patterns that may not be obvious to human analysts. This can lead to more accurate price predictions, helping travelers make more informed decisions.
- **Time-saving:** Machine learning algorithms can quickly analyze data from various sources and provide realtime price predictions, saving travelers time and effort.
- **Cost-saving:** Accurate price predictions can help travelers save money by booking flights at the right time, reducing the cost of air travel.
- **Customization:** Machine learning algorithms can consider individual preferences and travel habits, providing customized recommendations to travelers.

Disadvantages:

- **Data accuracy:** Machine learning algorithms rely on accurate and up-to-date data to provide accurate predictions. If the data used for training the algorithm is incomplete or inaccurate, the predictions may be unreliable.
- **Technical expertise:** Developing and implementing machine learning algorithms requires technical expertise and resources, which may be a challenge for smaller companies or individuals.
- **Unforeseen events:** Machine learning algorithms cannot predict unforeseen events, such as natural disasters or political unrest, which can affect flight prices.
- **Privacy concerns:** The use of personal data for machine learning algorithms may raise privacy concerns, and companies must ensure they comply with relevant regulations and protect customer data.

Applications

The application of this project is in the travel industry, particularly in the areas of online travel agencies, airline websites, and travel search engines. By implementing machine learning algorithms for flight booking decisions, companies can provide their customers with more accurate and personalized recommendations for flights, resulting in cost savings and an enhanced travel experience.

This also extends to airlines and travel agencies, as they can use machine learning algorithms to optimize their pricing strategies and improve their revenue management. By analyzing data on customer behavior and market trends, companies can adjust their prices in real-time to attract more customers and increase profitability.

By using machine learning algorithms to analyze data from multiple sources, companies can provide their customers with more comprehensive travel recommendations and an overall better travel experience.

In travel industry, where machine learning algorithms can be used to optimize flight booking decisions, improve revenue management, and enhance the travel experience for both businesses and customers.

Conclusion

By analyzing large amounts of data from various sources, machine learning algorithms can make accurate price predictions and help travelers make more informed decisions about their flights.

The benefits of using machine learning for flight booking decisions include increased accuracy, time and cost savings, customization, and improved pricing strategies for airlines and travel agencies. However, there are potential drawbacks, such as the need for accurate and up-to-date data, technical expertise, unforeseen events, and privacy concerns.

Despite these challenges, the application of machine learning for flight booking decisions is a promising development for the travel industry. Companies that adopt these technologies are likely to gain a competitive advantage by providing better pricing strategies and personalized recommendations to their customers, resulting in increased customer satisfaction and loyalty.

Future scope:

The future scope of this project is vast and promising, with the potential to further improve the travel experience for both travelers and businesses.

Integration with augmented reality: The integration of machine learning algorithms with augmented reality could help travelers to visualize and interact with their travel plans in real-time, providing a more immersive and personalized travel experience.

Expansion to other modes of transportation: The application of machine learning for optimizing travel decisions can be extended beyond flight bookings to include other modes of transportation such as trains, buses, and taxis. This would allow travelers to make more informed decisions about their travel plans and choose the most costeffective and efficient modes of transportation.

Integration with smart cities: The integration of machine learning algorithms with smart city infrastructure could help travelers to navigate their destinations more efficiently, providing real-time traffic information, parking recommendations, and other helpful information.

Personalized travel recommendations: As machine learning algorithms become more advanced, they will be able to provide more personalized travel recommendations based on a traveler's preferences and past travel history. This would allow companies to offer customized travel packages and improve customer loyalty.

Increased use of natural language processing: The use of natural language processing in conjunction with machine learning algorithms could enable travelers to interact with virtual travel assistants using natural language, providing a more intuitive and user-friendly experience.

Appendix:

Project (2).ipynb

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import imblearn
import sklearn
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier, GradientBoostingClassifier
from sklearn.tree import DecisionTreeClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import f1_score
from sklearn.metrics import confusion_matrix, f1_score, accuracy_score
import warnings
import pickle
from scipy import stats
warnings.filterwarnings('ignore')
plt.style.use("fivethirtyeight")
```

```
[2]  data=pd.read_csv("/content/Data_Train.csv")
```

```
[3]  data.head()
```

	Airline	Date_of_Journey	Source	Destination	Route	Dep_Time	Arrival_Time	Duration	Total_Stops	Additional_Info	Price
0	IndiGo	24/03/2019	Banglore	New Delhi	BLR ? DEL	22:20	01:10 22 Mar	2h 50m	non-stop	No info	3897
1	Air India	1/05/2019	Kolkata	Banglore	CCU ? IXR ? BBI ? BLR	05:50	13:15	7h 25m	2 stops	No info	7662
2	Jet Airways	9/06/2019	Delhi	Cochin	DEL ? LKO ? BOM ? COK	09:25	04:25 10 Jun	19h	2 stops	No info	13882
3	IndiGo	12/05/2019	Kolkata	Banglore	CCU ? NAG ? BLR	18:05	23:30	5h 25m	1 stop	No info	6218
4	IndiGo	01/03/2019	Banglore	New Delhi	BLR ? NAG ? DEL	16:50	21:35	4h 45m	1 stop	No info	13302

```
for i in data:
  print(i,data[i].unique())
```

```
Airline ['IndiGo' 'Air India' 'Jet Airways' 'SpiceJet' 'Multiple carriers' 'GoAir'
 'Vistara' 'Air Asia' 'Vistara Premium economy' 'Jet Airways Business'
 'Multiple carriers Premium economy' 'Trujet']
Date_of_Journey ['24/03/2019' '1/05/2019' '9/06/2019' '12/05/2019' '01/03/2019'
 '24/06/2019' '12/03/2019' '27/05/2019' '1/06/2019' '18/04/2019'
 '9/05/2019' '24/04/2019' '3/03/2019' '15/04/2019' '12/06/2019'
 '6/03/2019' '21/03/2019' '3/04/2019' '6/05/2019' '15/05/2019'
 '18/06/2019' '15/06/2019' '6/04/2019' '18/05/2019' '27/06/2019'
 '21/05/2019' '06/03/2019' '3/06/2019' '15/03/2019' '3/05/2019'
 '9/03/2019' '6/06/2019' '24/05/2019' '09/03/2019' '1/04/2019'
 '21/04/2019' '21/06/2019' '27/03/2019' '18/03/2019' '12/04/2019'
 '9/04/2019' '1/03/2019' '03/03/2019' '27/04/2019']
Source ['Banglore' 'Kolkata' 'Delhi' 'Chennai' 'Mumbai']
Destination ['New Delhi' 'Banglore' 'Cochin' 'Kolkata' 'Delhi' 'Hyderabad']
Route ['BLR ? DEL' 'CCU ? IXR ? BBI ? BLR' 'DEL ? LKO ? BOM ? COK'
 'CCU ? NAG ? BLR' 'BLR ? NAG ? DEL' 'CCU ? BLR' 'BLR ? BOM ? DEL'
 'DEL ? BOM ? COK' 'DEL ? BLR ? COK' 'MAA ? CCU' 'CCU ? BOM ? BLR'
 'DEL ? AMD ? BOM ? COK' 'DEL ? PNQ ? COK' 'DEL ? CCU ? BOM ? COK'
 'BLR ? COK ? DEL' 'DEL ? IDR ? BOM ? COK' 'DEL ? LKO ? COK'
 'CCU ? GAU ? DEL ? BLR' 'DEL ? NAG ? BOM ? COK' 'CCU ? MAA ? BLR'
 'DEL ? HYD ? COK' 'CCU ? HYD ? BLR' 'DEL ? COK' 'CCU ? DEL ? BLR'
 'BLR ? BOM ? AMD ? DEL' 'BOM ? DEL ? HYD' 'DEL ? MAA ? COK' 'BOM ? HYD'
 'DEL ? BHO ? BOM ? COK' 'DEL ? JAI ? BOM ? COK' 'DEL ? ATQ ? BOM ? COK'
 'DEL ? JDH ? BOM ? COK' 'CCU ? BBI ? BOM ? BLR' 'BLR ? MAA ? DEL'
 'DEL ? GOI ? BOM ? COK' 'DEL ? BDQ ? BOM ? COK' 'CCU ? JAI ? BOM ? BLR'
```

```
data.Date_of_Journey
```

```
0         24/03/2019
1          1/05/2019
2          9/06/2019
3         12/05/2019
4         01/03/2019
             ...
10678      9/04/2019
10679     27/04/2019
10680     27/04/2019
10681     01/03/2019
10682      9/05/2019
Name: Date_of_Journey, Length: 10683, dtype: object
```

```
data.Dep_Time=data.Dep_Time.str.split(':')
```

```
data['Travel_Hours']=data.Duration.str[0]
data['Travel_Hours']=data['Travel_Hours'].str.split('h')
data['Travel_Hours']=data['Travel_Hours'].str[0]
data.Travel_Hours=data.Travel_Hours
data['Travel_Mins']=data.Duration.str[1]
data.Travel_Mins=data.Travel_Mins.str.split('m')
data.Travel_Mins=data.Travel_Mins.str[0]
```

```
data.Total_Stops.replace('non_stop',0,inplace=True)
data.Total_Stops=data.Total_Stops.str.split(' ')
data.Total_Stops=data.Total_Stops.str[0]
```

```
data.Additional_Info.unique()
```

```
array(['No info', 'In-flight meal not included',
       'No check-in baggage included', '1 Short layover', 'No Info',
       '1 Long layover', 'Change airports', 'Business class',
       'Red-eye flight', '2 Long layover'], dtype=object)
```

```
data.Additional_Info.replace('No Info','No Info',inplace=True)
```

```
try:
    as_int = int(64)
except ValueError:

 data.Total_Stops=data.Total_Stops.astype('int64')
 data.Date=data.Date.astype('int64')
 data.Month=data.Month.astype('int64')
 data.Year=data.Year.astype('int64')
 data.Dep_Time_Hour=data.Dep_Time_Hour.astype('int64')
 data.Dep_Time_Mins=data.Dep_Time_Mins.astype('int64')
 data.Arrival_date=data.Arrival_date.astype('int64')
 data.Arrival_Time_Hour=data.Arrival_Time_Hour.astype('int64')
 data.Arrival_Time_Mins=data.Arrival_Time_Mins.astype('int64')
 data.Travel_Mins=data.Travel_Mins.astype('int64')
```

```
data[data['Travel_Hours']=='5m']
```

	Airline	Source	Destination	Total_Stops	Additional_Info	Price	Date	Month	Year	City1	City2	City3	Dep_Time_Hour	Dep_Time_Mins	Arrival_c
6474	Air India	Mumbai	Hyderabad	2	No info	17327	6	03	2019	BOM ? GOI ? PNQ ? HYD	NaN	None	16	50	

```
[31] data.drop(index=6474,inplace=True,axis=0)
```

```
[32] data.Travel_Hours=data.Travel_Hours.astype('int64')
```

```
[33] categorical=['Airline','Source','Destination','Additional_Info','City']
     numerical=['Total_Stops','Date','Month','Year','Dep_Time_Hour','Dep_Time_Mins','Arrival_date','Arrival_Time_Hour','Arrival_Time_Mins','Travel_Hours',
```

```
[34] from sklearn.preprocessing import LabelEncoder
     le=LabelEncoder()
```

```
[35] data.Airline=le.fit_transform(data.Airline)
     data.Source=le.fit_transform(data.Source)
     data.Destination=le.fit_transform(data.Destination)
     data.Total_Stops=le.fit_transform(data.Total_Stops)
     data.City1=le.fit_transform(data.City1)
     data.City2=le.fit_transform(data.City2)
     data.City3=le.fit_transform(data.City3)
     data.Additional_Info=le.fit_transform(data.Additional_Info)
     data.head()
```

	Airline	Source	Destination	Total_Stops	Additional_Info	Price	Date	Month	Year	City1	City2	City3	Dep_Time_Hour	Dep_Time_Mins	Arrival_date
0	3	0	5	4	8	3897	24	03	2019	18	0	0	22	20	22
1	1	3	0	1	8	7662	1	05	2019	84	0	0	05	50	1
2	4	2	1	1	8	13882	9	06	2019	118	0	0	09	25	10
3	3	3	0	0	8	6218	12	05	2019	91	0	0	18	05	12
4	3	0	5	0	8	13302	01	03	2019	29	0	0	16	50	01

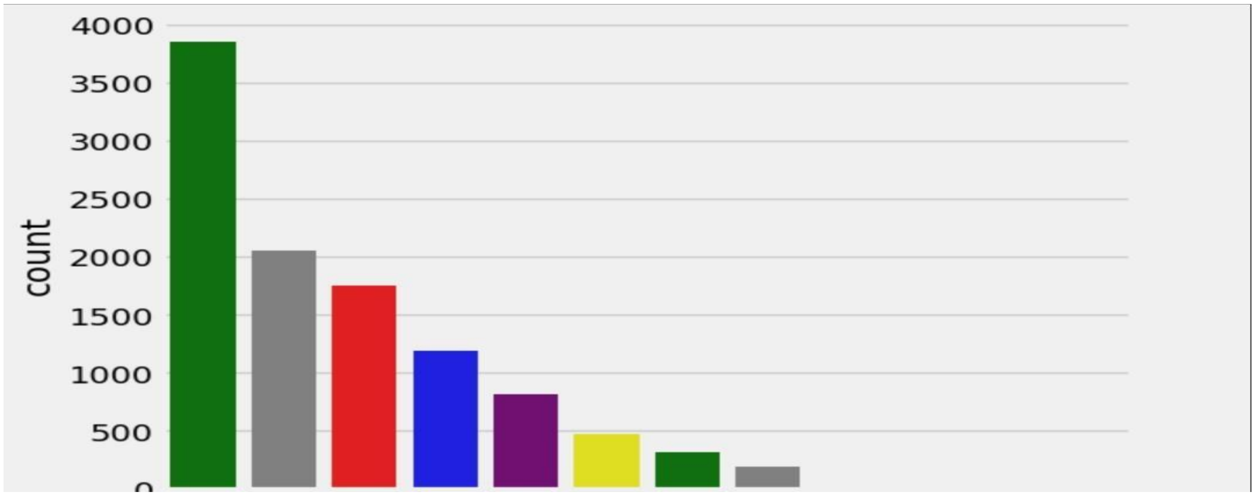
```
data= data[['Airline','Source','Destination','Date','Month','Year','Dep_Time_Hour','Dep_Time_Mins','Arrival_date','Arrival_Time_Hour','Arrival_Time_M:
```

```
data.head()
```

	Airline	Source	Destination	Date	Month	Year	Dep_Time_Hour	Dep_Time_Mins	Arrival_date	Arrival_Time_Hour	Arrival_Time_Mins	Price
0	3	0	5	24	03	2019	22	20	22	01	10	3897
1	1	3	0	1	05	2019	05	50	1	13	15	7662
2	4	2	1	9	06	2019	09	25	10	04	25	13882
3	3	3	0	12	05	2019	18	05	12	23	30	6218
4	3	0	5	01	03	2019	16	50	01	21	35	13302

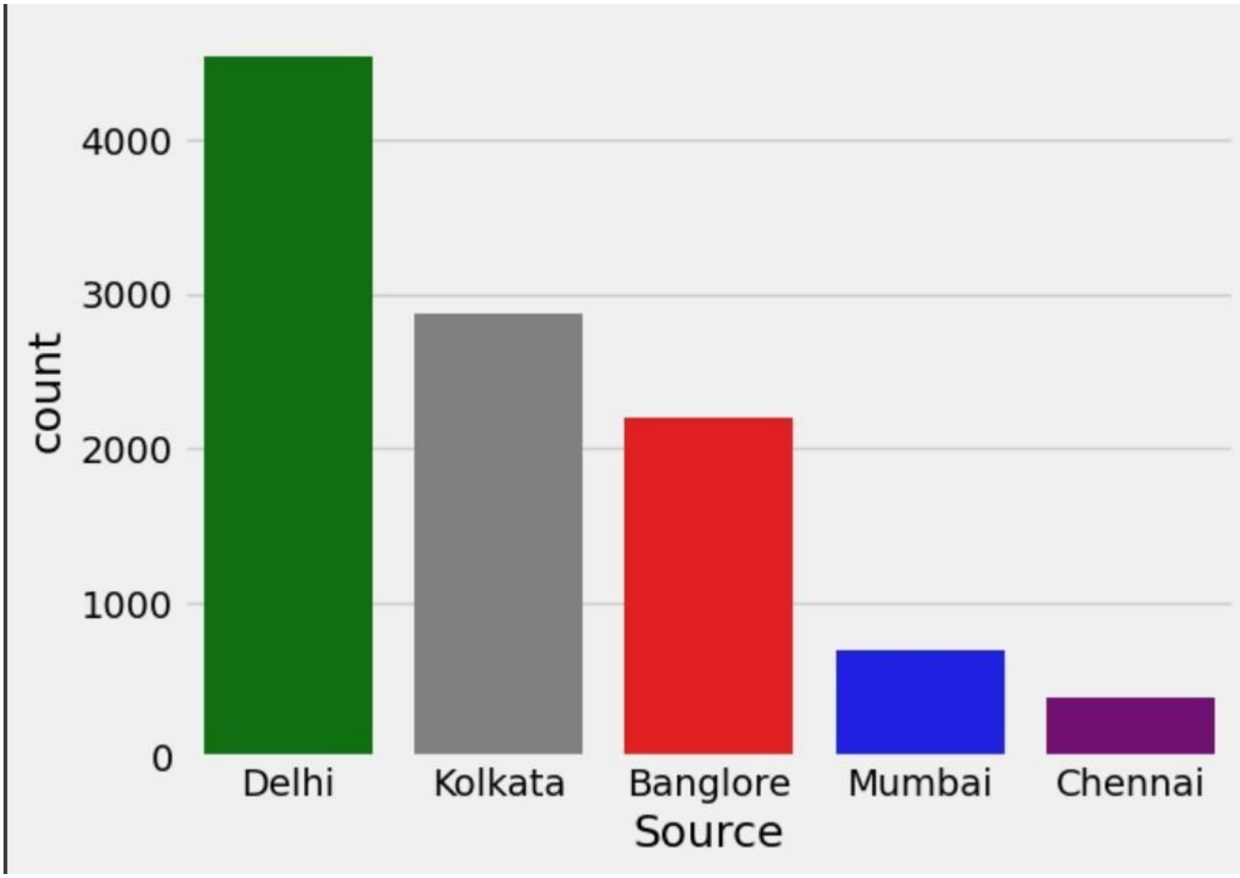
```
freqs = df["Airline"].value_counts()
order = freqs.index.tolist()
cols = ["green","grey","red","blue","purple","yellow"]
ax = sns.countplot(x=df["Airline"], order=order, palette=cols)
print(freqs)
```

```
Jet Airways                          3849
IndiGo                               2053
Air India                            1752
Multiple carriers                    1196
SpiceJet                              818
Vistara                               479
Air Asia                              319
GoAir                                 194
Multiple carriers Premium economy      13
Jet Airways Business                    6
Vistara Premium economy                 3
Trujet                                  1
Name: Airline, dtype: int64
```



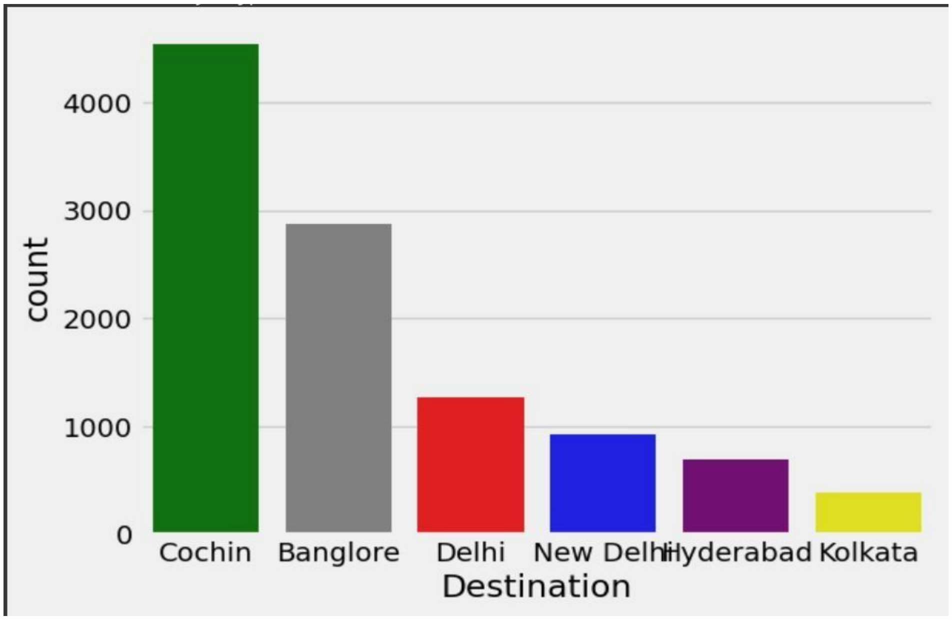

```
freqs2 = df["Source"].value_counts()
order = freqs2.index.tolist()
cols = ["green","grey","red","blue","purple","yellow"]
ax = sns.countplot(x=df["Source"], order=order, palette=cols)
print(freqs2)
```

```
Delhi       4537
Kolkata     2871
Banglore    2197
Mumbai       697
Chennai      381
Name: Source, dtype: int64
```

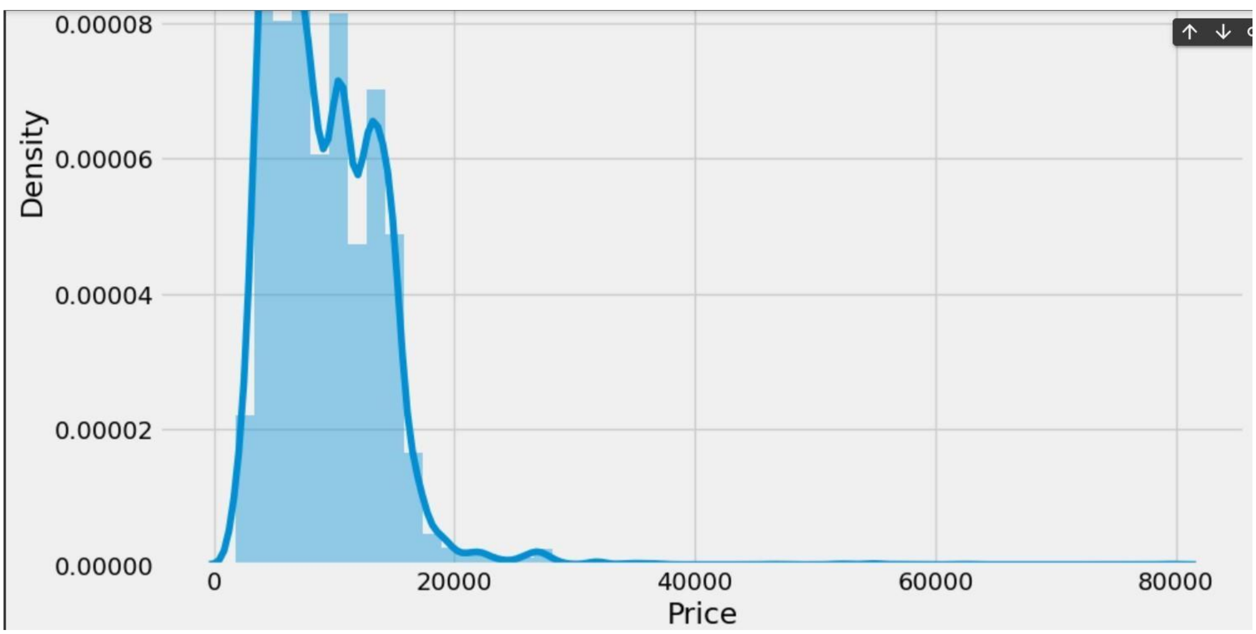



```
freqs3 = df["Destination"].value_counts()


order = freqs3.index.tolist()


cols = ["green","grey","red","blue","purple","yellow"]

ax = sns.countplot(x=df["Destination"], order=order, palette=cols)
print(freqs3)
```

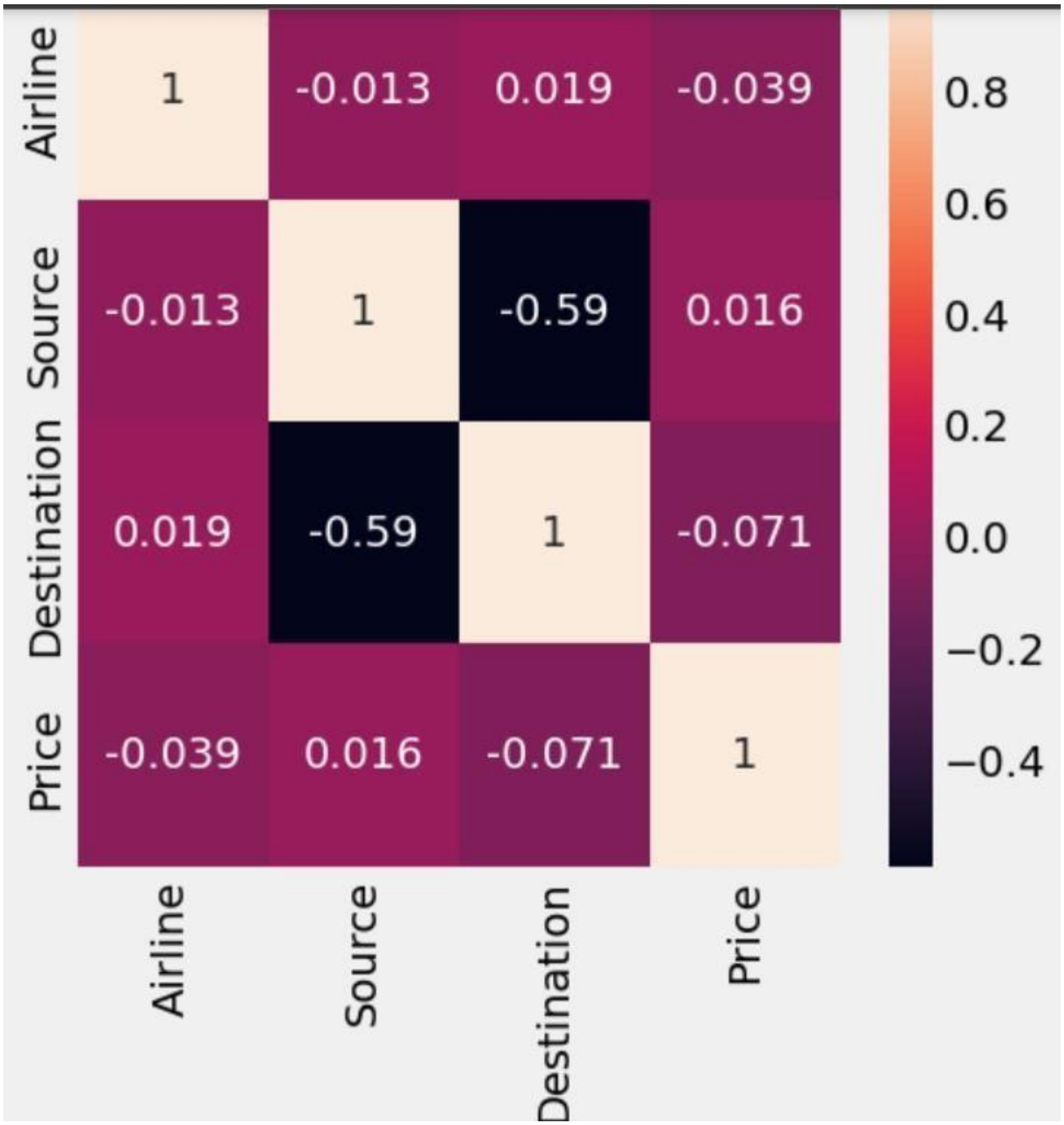
```
Cochin       4537
Banglore     2871
Delhi        1265
New Delhi     932
Hyderabad     697
Kolkata       381
Name: Destination, dtype: int64
```




```
plt.figure(figsize=(15,8))
sns.distplot(data.Price)
```

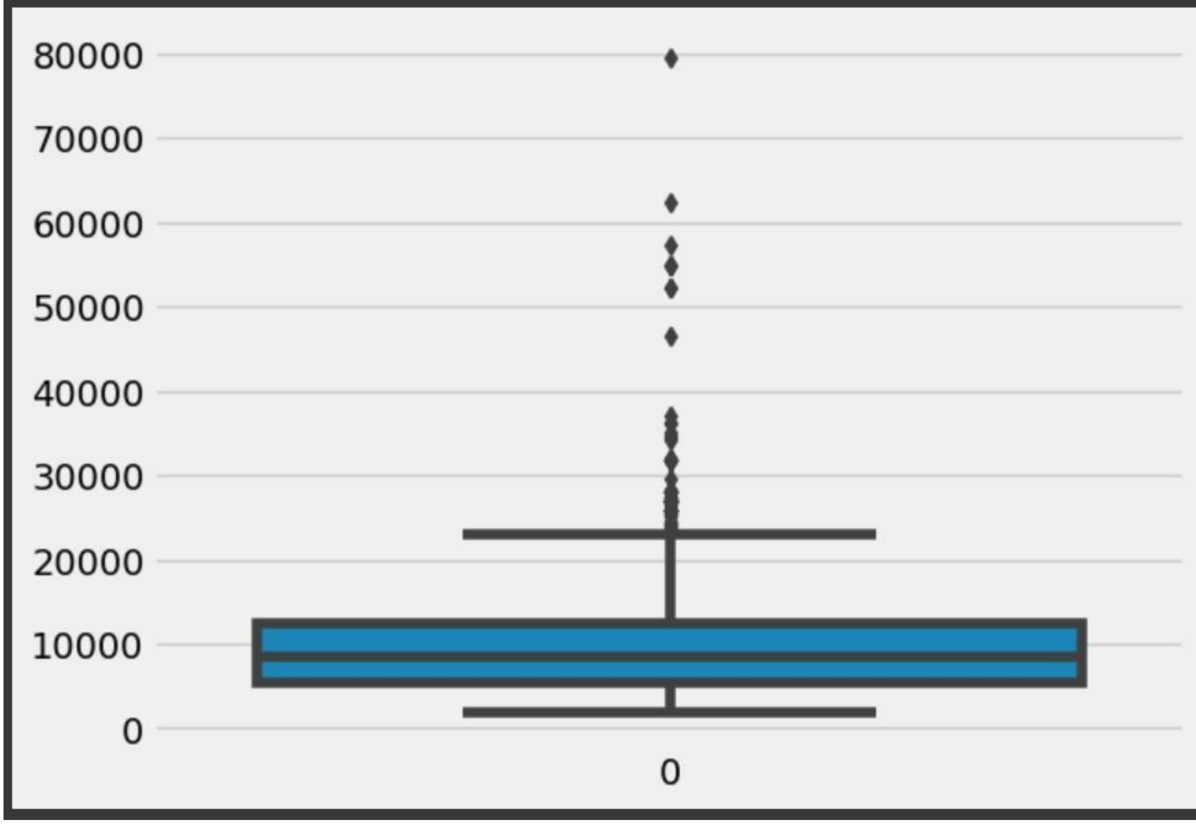



```
plt.figure(figsize=(10, 10))
sns.heatmap(data.corr(),annot=True)
```



Airline
Source
Destination
Price
1
-0.013
0.019
-0.039
-0.013
1
-0.59
0.016
0.019
-0.59
1
-0.071
-0.039
0.016
-0.071
1
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4

```python
import seaborn as sns
sns.boxplot(data.Price);
```




```python
y=data['Price']
x=data.drop(columns=['Price'],axis=1)
```

```python
from sklearn.preprocessing import StandardScaler
ss=StandardScaler()
```

```python
x_scaled = ss.fit_transform(x)
```

```python
x_scaled = pd.DataFrame(x_scaled,columns=x.columns)
x_scaled.head()
```

	Airline	Source	Destination	Date	Month	Year	Dep_Time_Hour	Dep_Time_Mins	Arrival_date	Arrival_Time_Hour	Arrival_Time_Mins
0	-0.410805	-1.658435	2.416778	1.237288	-1.467707	0.0	1.654268	-0.234932	0.955750	-1.800319	-0.889984
1	-1.261152	0.890299	-0.973732	-1.475307	0.250153	0.0	-1.303000	1.363674	-1.524648	-0.050813	-0.587017
2	0.014369	0.040721	-0.295630	-0.531796	1.109082	0.0	-0.607172	0.031502	-0.461621	-1.362943	0.018918
3	-0.410805	0.890299	-0.973732	-0.177979	0.250153	0.0	0.958440	-1.034235	-0.225392	1.407109	0.321885
4	-0.410805	-1.658435	2.416778	-1.475307	-1.467707	0.0	0.610527	1.363674	-1.524648	1.115525	0.624852

```
[52] from sklearn.model_selection import train_test_split
     x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2,random_state=42)
```

```
[53] x_train.head()
```

	Airline	Source	Destination	Date	Month	Year	Dep_Time_Hour	Dep_Time_Mins	Arrival_date	Arrival_Time_Hour	Arrival_Time_Mins
10005	6	2	1	27	05	2019	08	30	27	19	15
3684	4	2	1	9	05	2019	11	30	10	12	35
1034	8	2	1	24	04	2019	15	45	24	22	05
3909	6	2	1	21	03	2019	12	50	22	01	35
3088	1	2	1	24	06	2019	17	15	25	19	15

```
y_train.head()
```

```
10005     9149
3684     12373
1034      5583
3909      7695
3088     11972
Name: Price, dtype: int64
```

```
gb=GradientBoostingRegressor()
gb_res=RandomizedSearchCV(estimator=gb,param_distributions=param_grid,cv=3,verbose=2,n_jobs=-1)
gb_res.fit(x_train,y_train)
```

```
Fitting 3 folds for each of 10 candidates, totalling 30 fits
▸ RandomizedSearchCV
▸ estimator: GradientBoostingRegressor
  ▸ GradientBoostingRegressor
```

```
rfr=RandomForestRegressor(n_estimators=10,max_features='sqrt',max_depth=None)
rfr.fit(x_train,y_train)
y_train_pred=rfr.predict(x_train)
y_test_pred=rfr.predict(x_test)
print("train accuracy",r2_score(y_train_pred,y_train))
print("test accuracy",r2_score(y_test_pred,y_test))
```

```
train accuracy 0.9201086463734791
test accuracy 0.7633404855809773
```

```python
from sklearn.neighbors import KNeighborsRegressor
from sklearn.metrics import confusion_matrix, f1_score, accuracy_score
knn=KNeighborsRegressor(n_neighbors=2,algorithm='auto',metric_params=None,n_jobs=-1)
knn.fit(x_train,y_train)
y_train_pred=knn.predict(x_train)
y_test_pred=knn.predict(x_test)
print("train accuracy",r2_score(y_train_pred,y_train))
print("test accuracy",r2_score(y_test_pred,y_test))
```

```
train accuracy 0.8282052673851559
test accuracy 0.4817675302068355
```

```python
rfr=RandomForestRegressor(n_estimators=10,max_features='sqrt',max_depth=None)
rfr.fit(x_train,y_train)
y_train_pred=rfr.predict(x_train)
y_test_pred=rfr.predict(x_test)
print("train accuracy",r2_score(y_train_pred,y_train))
print("test accuracy",r2_score(y_test_pred,y_test))
```

```
train accuracy 0.9279564761911743
test accuracy 0.7704790605144969
```

```python
import pickle
pickle.dump(rfr,open('model.pkl','wb'))
```

Home.html

*home.html - Notepad

File Edit View

```html
<!DOCTYPE html>
<html>
<title>Flight Price Prediction</title>
<head>
    <style>
        ul {
          list-style-type: none;
          margin: 0;
          padding: 0;
          overflow: hidden;
          background-color: #333;
        }

        li {
          float: left;
        }

        li a {
          display: block;
          color: white;
          text-align: center;
          padding: 14px 16px;
          text-decoration: none;
        }

        li a:hover {
          background-color: #111;
        }
        </style>
</head>

<body style="background-image: url('/static/flight2.jpg'); background-repeat: no-repeat;background-size: cover;background-attachment:
fixed;  min-height: 100vh; max-width: 800px;">
    <ul>
        <li><a class="active" href="/">Home</a></li>
        <li><a href="/predict">Prediction</a></li>
      </ul>
    <h1>Flight Price Prediction</h1>
<div>This project aims to predict ticket prices for upcoming flights to help customers in selecting the optimum time for travel and the
cheapest flight to the desired destination. A random forest regression model is applied to forecast the flight prices based on data scraped from Kayak.</div>
</body>
</html>
```

Ln 36, Col 42 110% Windows (CRLF) UTF-8

Predict.html

*predict.html - Notepad

File Edit View

```
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Flight Price Prediction</title>


    <!-- BootStrap -->
    <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.0/css/bootstrap.min.css"
        integrity="sha384-9aIt2nRpC12Uk9gS9baDl411NQApFmC26EwAOH8WgZl5MYYxFfc+NcPb1dKGj7Sk" crossorigin="anonymous">


    <!-- css -->
    <link rel="stylesheet" href="static/css/styles.css">

    <style>
        ul {
          list-style-type: none;
          margin: 0;
          padding: 0;
          overflow: hidden;
          background-color: #333;
        }

        li {
          float: left;
        }

        li a {
          display: block;
          color: white;
          text-align: center;
          padding: 14px 16px;
          text-decoration: none;
        }

        li a:hover {
          background-color: #111;
        }
        </style>
</head>

<body style="background-image: url('/static/flight2.jpg'); background-repeat: no-repeat;background-size: cover;background-attachment: fixed;
 min-height: 100vh; max-width: 800px;">
    <ul>
        <li><a class="active" href="/">Home</a></li>
        <li><a href="/predict">Prediction</a></li>
      </ul>
    <h1>Flight Price Prediction</h1>

    <!-- As a heading -->
    <!--<nav class="navbar navbar-inverse navbar-fixed-top">
        <div class="container-fluid">
            <div class="navbar-header">
```

Ln 45, Col 1 100% Windows (CRLF) UTF-8

*predict.html - Notepad

File Edit View

```
      </ul>
    <h1>Flight Price Prediction</h1>

    <!-- As a heading -->
    <!--<nav class="navbar navbar-inverse navbar-fixed-top">
        <div class="container-fluid">
            <div class="navbar-header">
                <a class="navbar-brand" href="/">FLIGHT PRICE PREDICTION</a>
            </div>
        </div>
    </nav> -->

    <br><br><br>


    <div class="container">


        <form action="\pred" method="post">


            <div class="row">
                <div class="col-sm-6">
                    <div class="card">
                        <div class="card-body">
                            <h5 class="card-title">Departure Date</h5>
                            <!-- Departure -->
                            <input type="text" name="Date" id="Date" required="required">
                        </div>
                    </div>
                </div>
                <div class="col-sm-6">
                    <div class="card">
                        <div class="card-body">
                            <h5 class="card-title">Departure Month</h5>
                            <!-- Departure -->
                            <input type="text" name="Month" id="Month" required="required">
                        </div>
                    </div>

                </div>

            </div>
            <br>
            <div class="row">
                <div class="col-sm-6">
                    <div class="card">
                        <div class="card-body">
                            <h5 class="card-title">Departure Year</h5>
                            <!-- Departure -->
                            <input type="text" name="Year" id="Year" required="required">
                        </div>
                    </div>
                </div>
                <div class="col-sm-6">
```

Ln 45, Col 1 100% Windows (CRLF) UTF-8

*predict.html - Notepad

File Edit View

```html
                    <!-- Departure -->
                    <input type="text" name="Dep_Time_Hour" id="Dep_Time_Hour" required="required">
                </div>
            </div>
        </div>
    </div>

        <br>
    <div class="row">
        <div class="col-sm-6">
            <div class="card">
                <div class="card-body">
                    <h5 class="card-title">Departure Time Mins</h5>
                    <!-- Arrival -->
                    <input type="text" name="Dep_Time_Mins" id="Dep_Time_Mins" required="required">
                </div>
            </div>
        </div>
        <div class="col-sm-6">
            <div class="card">
                <div class="card-body">
                    <h5 class="card-title">Arrival Time</h5>
                    <!-- Arrival -->
                    <input type="text" name="Arrival_date" id="Arrival_date" required="required">
                </div>
            </div>
        </div>
    </div>
    <br>
    <div class="row">
        <div class="col-sm-6">
            <div class="card">
                <div class="card-body">
                    <h5 class="card-title">Arrival Time Hours</h5>
                    <!-- Arrival -->
                    <input type="text" name="Arrival_Time_Hours" id="Arrival_Time_Hours" required="required">
                </div>
            </div>
        </div>
        <div class="col-sm-6">
            <div class="card">
                <div class="card-body">
                    <h5 class="card-title">Arrival Time Mins</h5>
                    <!-- Arrival -->
                    <input type="text" name="Arrival_Time_Mins" id="Arrival_Time_Mins" required="required">
                </div>
            </div>
        </div>
    </div>
    <br>
    <div class="row">
        <div class="col-sm-6">
            <div class="card">
                <div class="card-body">
                    <!-- Source -->
                    <h5 class="card-title">Source</h5>
```

Ln 45, Col 1 100% Windows (CRLF) UTF-8

*predict.html - Notepad

File Edit View

```html
                        <option value="1">Kolkata</option>
                        <option value="2">Mumbai</option>
                        <option value="3">Chennai</option>
                    </select>
                </div>
            </div>
        </div>
        <div class="col-sm-6">
            <div class="card">
                <div class="card-body">
                    <h5 class="card-title">Destination</h5>
                    <!-- Destination -->
                    <select name="Destination" id="Destination" required="required">
                        <option value="0">Cochin</option>
                        <option value="1">Delhi</option>
                        <option value="2">New Delhi</option>
                        <option value="3">Hyderabad</option>
                        <option value="4">Kolkata</option>
                    </select>
                </div>
            </div>
        </div>
    </div>
    <br>
    <div class="row">
        <div class="col-sm-6">
            <div class="card">
                <div class="card-body">
                    <h5 class="card-title">Which Airline you want to travel?</h5>
                    <!-- Airline -->
                    <select name="airline" id="airline" required="required">
                        <option value="0">Jet Airways</option>
                        <option value="1">IndiGo</option>
                        <option value="2">Air India</option>
                        <option value="3">Multiple carriers</option>
                        <option value="4">SpiceJet</option>
                        <option value="5">Vistara</option>
                        <option value="6">Air Asia</option>
                        <option value="7">GoAir</option>
                        <option value="8">Multiple carriers Premium economy
                        </option>
                        <option value="9">Jet Airways Business</option>
                        <option value="10">Vistara Premium economy</option>
                        <option value="11">Trujet</option>
                    </select>
                </div>
            </div>
        </div>
    </div>



    <br>
    <br>
    <br>
    <!-- Submit -->
```

Ln 45, Col 1 100% Windows (CRLF) UTF-8

*predict.html - Notepad

File Edit View

```
                              <select name="airline" id="airline" required="required">
                                  <option value="0">Jet Airways</option>
                                  <option value="1">IndiGo</option>
                                  <option value="2">Air India</option>
                                  <option value="3">Multiple carriers</option>
                                  <option value="4">SpiceJet</option>
                                  <option value="5">Vistara</option>
                                  <option value="6">Air Asia</option>
                                  <option value="7">GoAir</option>
                                  <option value="8">Multiple carriers Premium economy
                                  </option>
                                  <option value="9">Jet Airways Business</option>
                                  <option value="10">Vistara Premium economy</option>
                                  <option value="11">Trujet</option>
                              </select>
                          </div>
                      </div>
                  </div>
              </div>



              <br>
              <br>
              <br>
              <!-- Submit -->
              <input type="submit" value="Submit" class="btn btn-secondary">
          </form>

          <br>
          <br>
          <h3>{{ prediction_text }}</h3>

          <br>
          <br>
          <p></p>

      </div>



      <!-- JavaScript -->
      <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"
          integrity="sha384-DfXdz2htPH0lsSSs5nCTpuj/zy4C+OGpamoFVy38MVBnE+IbbVYUew+OrCXaRkfj"
          crossorigin="anonymous"></script>
      <script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/popper.min.js"
          integrity="sha384-Q6E9RHvbIyZFJoft+2mJbHaEWldlvI9IOYy5n3zV9zzTtmI3UksdQRVvoxMfooAo"
          crossorigin="anonymous"></script>
      <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.0/js/bootstrap.min.js"
          integrity="sha384-OgVRvuATP1z7JjHLkuOU7Xw704+h835Lr+6QL9UvYjZE3Ipu6Tp75j7Bh/kR0JKI"
          crossorigin="anonymous"></script>

</body>

</html>
```

Ln 45, Col 1 | 100% | Windows (CRLF) | UTF-8

Submit.html

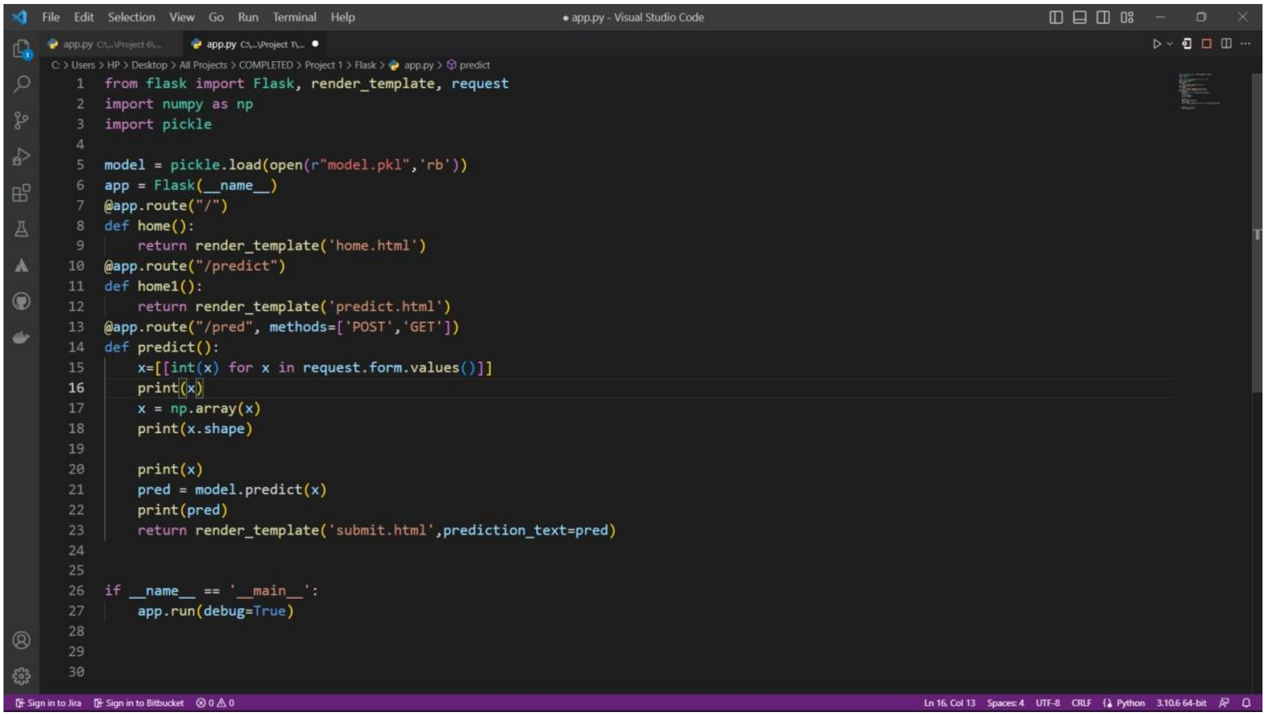
*submit.html - Notepad

File Edit View

```
<!DOCTYPE html>
<html>
<title>Flight Price Prediction</title>
<head>
    <style>
        ul {
          list-style-type: none;
          margin: 0;
          padding: 0;
          overflow: hidden;
          background-color: #333;
        }

        li {
          float: left;
        }

        li a {
          display: block;
          color: white;
          text-align: center;
          padding: 14px 16px;
          text-decoration: none;
        }

        li a:hover {
          background-color: #111;
        }
        </style>
</head>

<body style="background-image: url('/static/flight2.jpg'); background-repeat: no-repeat;background-size: cover;background-attachment: fixed;
min-height: 100vh; max-width: 800px;">
    <ul>
        <li><a class="active" href="/">Home</a></li>
        <li><a href="/predict">Prediction</a></li>
      </ul>
    <h1>Flight Price Prediction</h1>
<div>Based on the given input, we can get the flight price as {{prediction_text }}</div>
</body>
</html>
```

Ln 33, Col 1 | 100% | Windows (CRLF) | UTF-8

App.py


File Edit Selection View Go Run Terminal Help
● app.py - Visual Studio Code
app.py C:\...\Project 6\...
app.py C:\...\Project 1\...
C: > Users > HP > Desktop > All Projects > COMPLETED > Project 1 > Flask > app.py > predict
from flask import Flask, render_template, request
import numpy as np
import pickle

model = pickle.load(open(r"model.pkl",'rb'))
app = Flask(__name__)
@app.route("/")
def home():
    return render_template('home.html')
@app.route("/predict")
def home1():
    return render_template('predict.html')
@app.route("/pred", methods=['POST','GET'])
def predict():
    x=[[int(x) for x in request.form.values()]]
    print(x)
    x = np.array(x)
    print(x.shape)

    print(x)
    pred = model.predict(x)
    print(pred)
    return render_template('submit.html',prediction_text=pred)

if __name__ == '__main__':
    app.run(debug=True)
Sign in to Jira
Sign in to Bitbucket
0 0
Ln 16, Col 13 Spaces: 4 UTF-8 CRLF Python 3.10.6 64-bit