

Soaring Through Data: Unveiling Insights in Flight Analytics



Name : Ashmin Bhaumik (12201289)
Email : ashmin.bhaumik03@gmail.com

"Just as the sky is vast and boundless, so too are the insights hidden within our flight data, waiting to be discovered and explored."





EXECUTIVE SUMMARY

- Introduction
- Methodology
- Results
- Visualization – Charts
- Conclusion



INTRODUCTION

Today's dynamic world of air travel, understanding flight data is crucial for **optimizing journeys** and **enhancing** passenger **experiences**. Our analysis **delves** into a **comprehensive dataset** encompassing **airlines**, **flight identifiers**, **source** and **destination** cities, **departure** and **arrival times**, and various other essential metrics. By **examining flight duration**, stopovers, class of service, remaining days to departure, and **pricing details**, we aim to **uncover patterns** and **insights** that can **streamline** travel **planning** and **inform strategic decisions**. This **exploration** not only **reveals** operational efficiencies but also **enhances** the **overall passenger experience** through **data-driven insights**. Join us as we navigate through this intricate tapestry of flight information.



- ❑ Data Collection Sources:
 - From SIR
- ❑ Data Exploration
- ❑ Data Cleaning
- ❑ Data Visualization
- ❑ Storytelling
- ❑ Presentation



merican





Data Cleaning

Importing all the Libraries needed

```
[1]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sb
```

Importing Data

```
[5]: data = pd.read_csv(r"D:\FIFTHFORCE\Flight_Data.csv", encoding = "unicode_escape")
data.head(10)
```

```
C:\Users\ashmi\AppData\Local\Temp\ipykernel_1700\447626745.py:1: DtypeWarning: Columns (11,12) have mixed types. Specify dtype option on import or set low_memory=False.
data = pd.read_csv(r"D:\FIFTHFORCE\Flight_Data.csv", encoding = "unicode_escape")
```

```
[5]:
```

	Unnamed: 0	airline	flight	source_city	departure_time	stops	arrival_time	destination_city	class	duration	days_left	price	date
0	0	SpiceJet	SG-8709	Delhi	Evening	zero	Night	Mumbai	Economy	2.17	1	5953	01-01-2018
1	1	SpiceJet	SG-8157	Delhi	Early_Morning	zero	Morning	Mumbai	Economy	2.33	1	5953	02-01-2018
2	2	AirAsia	I5-764	Delhi	Early_Morning	zero	Early_Morning	Mumbai	Economy	2.17	1	5956	03-01-2018
3	3	Vistara	UK-995	Delhi	Morning	zero	Afternoon	Mumbai	Economy	2.25	1	5955	04-01-2018
4	4	Vistara	UK-963	Delhi	Morning	zero	Morning	Mumbai	Economy	2.33	1	5955	05-01-2018
5	5	Vistara	UK-945	Delhi	Morning	zero	Afternoon	Mumbai	Economy	2.33	1	5955	06-01-2018
6	6	Vistara	UK-927	Delhi	Morning	zero	Morning	Mumbai	Economy	2.08	1	6060	07-01-2018
7	7	Vistara	UK-951	Delhi	Afternoon	zero	Evening	Mumbai	Economy	2.17	1	6060	08-01-2018
8	8	GO_FIRST	G8-334	Delhi	Early_Morning	zero	Morning	Mumbai	Economy	2.17	1	5954	09-01-2018
9	9	GO_FIRST	G8-336	Delhi	Afternoon	zero	Evening	Mumbai	Economy	2.25	1	5954	10-01-2018

```
[71]: dataCopy.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 301206 entries, 0 to 301205
Data columns (total 13 columns):
#   Column              Non-Null Count  Dtype  
---  --
0   Unnamed: 0           301206 non-null  int64  
1   airline              300754 non-null  object  
2   flight               301206 non-null  object  
3   source_city          301206 non-null  object  
4   departure_time       301206 non-null  object  
5   stops                301206 non-null  object  
6   arrival_time         301206 non-null  object  
7   destination_city     301206 non-null  object  
8   class                301006 non-null  object  
9   duration              300956 non-null  float64 
10  days_left            301206 non-null  int64  
11  price                301106 non-null  object  
12  date                  358 non-null     object  
dtypes: float64(1), int64(2), object(10)
memory usage: 29.9+ MB
```

```
•[70]: dataCopy.isnull().sum()
# dataCopy["date"].isnull().sum()
```

```
[70]: Unnamed: 0           0
airline              452
flight                0
source_city           0
departure_time        0
stops                 0
arrival_time          0
destination_city       0
class                 200
duration              250
days_left            0
price                 100
date                 300848
dtype: int64
```



Data Cleaning

Data Profiling

Unnamed columns should be dropped.

Airline Column

* 'nan' values should be replaced with mod values.

Flight Column

* No need to change anything.

Source_city Column

* No need to change anything.

Departure_time Column

* No need to change anything.

Stops Column

* No need to change anything.

Arrival_time Column

* No need to change anything.

Destination_city Column

* No need to change anything.

Class Column

* 'nan' values should be replaced with mod values.

Duration Column

* 'nan' values should be replaced with mod values.

Days_left Column

* No need to change anything.

Price Column

* 'nan' values should be replaced with mod values.

Date Column

* Date column should be removed. More than 70% data is not Present.

```
[82]: # Dropping unnamed Column...
dataCopy.drop(dataCopy.columns[dataCopy.columns.str.contains('unnamed', case=False)], axis=1, inplace=True)
dataCopy
```

```
[82]:
```

	airline	flight	source_city	departure_time	stops	arrival_time	destination_city	class	duration	days_left	price	date
0	SpiceJet	SG-8709	Delhi	Evening	zero	Night	Mumbai	Economy	2.17	1	5953	01-01-2018
1	SpiceJet	SG-8157	Delhi	Early_Morning	zero	Morning	Mumbai	Economy	2.33	1	5953	02-01-2018
2	AirAsia	I5-764	Delhi	Early_Morning	zero	Early_Morning	Mumbai	Economy	2.17	1	5956	03-01-2018
3	Vistara	UK-995	Delhi	Morning	zero	Afternoon	Mumbai	Economy	2.25	1	5955	04-01-2018
4	Vistara	UK-963	Delhi	Morning	zero	Morning	Mumbai	Economy	2.33	1	5955	05-01-2018
...
301201	Vistara	UK-747	Delhi	Early_Morning	one	Night	Mumbai	Economy	113.83	1	14985	15-03-2018
301202	Air_India	AI-512	Delhi	Afternoon	one	Night	Mumbai	Economy	97.58	1	14670	16-03-2018
301203	Air_India	AI-537	Delhi	Evening	one	Morning	Mumbai	Economy	115.83	1	15195	17-03-2018
301204	Vistara	UK-977	Delhi	Evening	zero	Night	Mumbai	Economy	622.25	1	15720	18-03-2018
301205	Vistara	UK-809	Delhi	Evening	one	Night	Mumbai	Economy	124.42	1	15983	19-03-2018

301206 rows x 12 columns

```
[92]: # Nan Values are replaced with mod values...
Airline_mode = dataCopy['airline'].mode()[0] # [0] is the index value we need...
Airline_mode

dataCopy["airline"] = dataCopy['airline'].fillna(Airline_mode)
dataCopy["airline"]
```

```
[92]: 0      SpiceJet
1      SpiceJet
2      AirAsia
3      Vistara
4      Vistara
```




Data Cleaning

```
[106]: # Date column dropped...
dataCopy.drop(columns=['date'])
dataCopy
```

```
[106]:
```

	airline	flight	source_city	departure_time	stops	arrival_time	destination_city	class	duration	days_left	price
0	SpiceJet	SG-8709	Delhi	Evening	zero	Night	Mumbai	Economy	2.17	1	5953
1	SpiceJet	SG-8157	Delhi	Early_Morning	zero	Morning	Mumbai	Economy	2.33	1	5953
2	AirAsia	I5-764	Delhi	Early_Morning	zero	Early_Morning	Mumbai	Economy	2.17	1	5956
3	Vistara	UK-995	Delhi	Morning	zero	Afternoon	Mumbai	Economy	2.25	1	5955
4	Vistara	UK-963	Delhi	Morning	zero	Morning	Mumbai	Economy	2.33	1	5955
...
301201	Vistara	UK-747	Delhi	Early_Morning	one	Night	Mumbai	Economy	113.83	1	14985
301202	Air_India	AI-512	Delhi	Afternoon	one	Night	Mumbai	Economy	97.58	1	14670
301203	Air_India	AI-537	Delhi	Evening	one	Morning	Mumbai	Economy	115.83	1	15195
301204	Vistara	UK-977	Delhi	Evening	zero	Night	Mumbai	Economy	622.25	1	15720
301205	Vistara	UK-809	Delhi	Evening	one	Night	Mumbai	Economy	124.42	1	15983

301206 rows × 11 columns

```
[107]: # Null values are totally removed...
dataCopy.isnull().sum()
```

```
[107]: airline      0
flight        0
source_city   0
departure_time 0
stops         0
arrival_time  0
destination_city 0
class         0
duration      0
days_left    0
price         0
```




Data Cleaning

[53]:

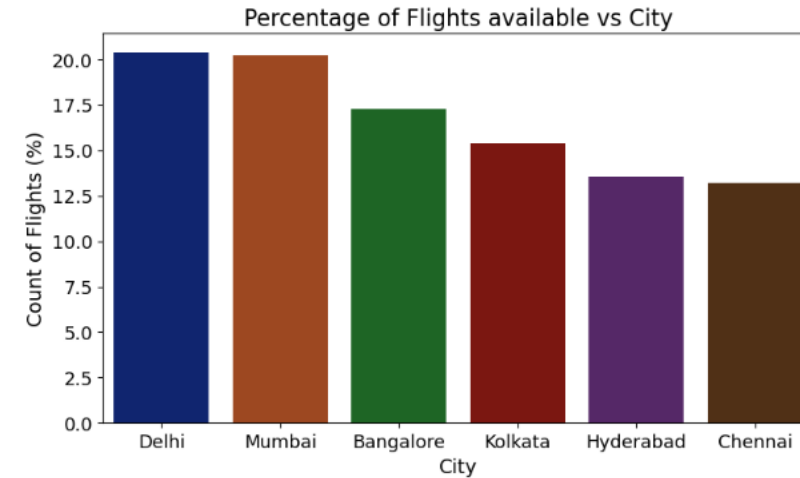
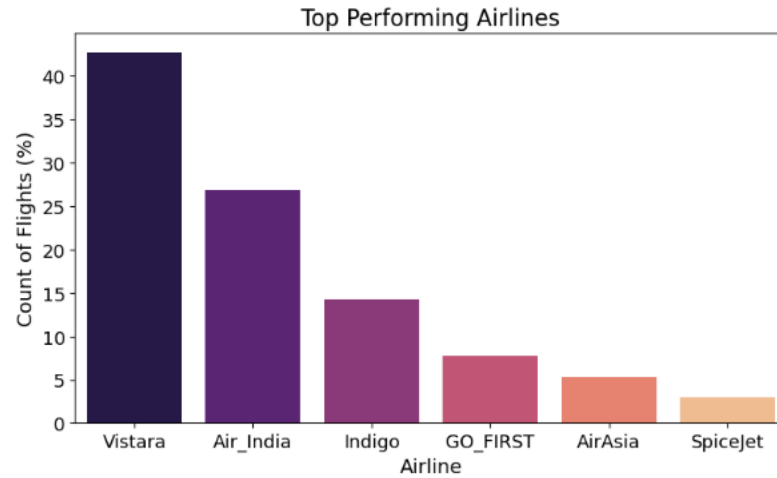
	airline	flight	source_city	departure_time	arrival_time	destination_city	class	duration	days_left	price	Time_duration	Total_stops
0	SpiceJet	SG-8709	Delhi	Evening	Night	Mumbai	Economy	2.17	1	5953	0 days 02:16:00	0
1	SpiceJet	SG-8157	Delhi	Morning	Morning	Mumbai	Economy	2.33	1	5953	0 days 02:33:00	0
2	AirAsia	I5-764	Delhi	Morning	Morning	Mumbai	Economy	2.17	1	5956	0 days 02:16:00	0
3	Vistara	UK-995	Delhi	Morning	Afternoon	Mumbai	Economy	2.25	1	5955	0 days 02:25:00	0
4	Vistara	UK-963	Delhi	Morning	Morning	Mumbai	Economy	2.33	1	5955	0 days 02:33:00	0
5	Vistara	UK-945	Delhi	Morning	Afternoon	Mumbai	Economy	2.33	1	5955	0 days 02:33:00	0
6	Vistara	UK-927	Delhi	Morning	Morning	Mumbai	Economy	2.08	1	6060	0 days 02:08:00	0
7	Vistara	UK-951	Delhi	Afternoon	Evening	Mumbai	Economy	2.17	1	6060	0 days 02:16:00	0
8	GO_FIRST	G8-334	Delhi	Morning	Morning	Mumbai	Economy	2.17	1	5954	0 days 02:16:00	0
9	GO_FIRST	G8-336	Delhi	Afternoon	Evening	Mumbai	Economy	2.25	1	5954	0 days 02:25:00	0

[54]: `data.describe()`

	duration	days_left	price	Total_stops
count	301206.000000	301206.000000	3.012060e+05	301206.000000
mean	12.202056	26.042277	2.111804e+04	0.924550
std	7.135976	13.564973	2.544767e+04	0.397471
min	0.830000	1.000000	1.105000e+03	0.000000
25%	6.830000	15.000000	4.788000e+03	1.000000
50%	11.250000	26.000000	7.455000e+03	1.000000
75%	16.170000	38.000000	4.252100e+04	1.000000
max	30.180000	49.000000	2.512780e+06	2.000000



Data Analysis and Plotting



Conclusion

- Vistara Flights are more in number than other Flights.
- Availability of Flights of Delhi and Mumbai are more in number.

```
[64]: dataCopy.head(10)
```

	airline	flight	source_city	departure_time	stops	arrival_time	destination_city	class	duration	days_left	price
0	SpiceJet	SG-8709	Delhi	Evening	zero	Night	Mumbai	Economy	2.17	1	5953
1	SpiceJet	SG-8157	Delhi	Early_Morning	zero	Morning	Mumbai	Economy	2.33	1	5953
2	AirAsia	I5-764	Delhi	Early_Morning	zero	Early_Morning	Mumbai	Economy	2.17	1	5956
3	Vistara	UK-995	Delhi	Morning	zero	Afternoon	Mumbai	Economy	2.25	1	5955
4	Vistara	UK-963	Delhi	Morning	zero	Morning	Mumbai	Economy	2.33	1	5955
5	Vistara	UK-945	Delhi	Morning	zero	Afternoon	Mumbai	Economy	2.33	1	5955



Data Analysis and Plotting

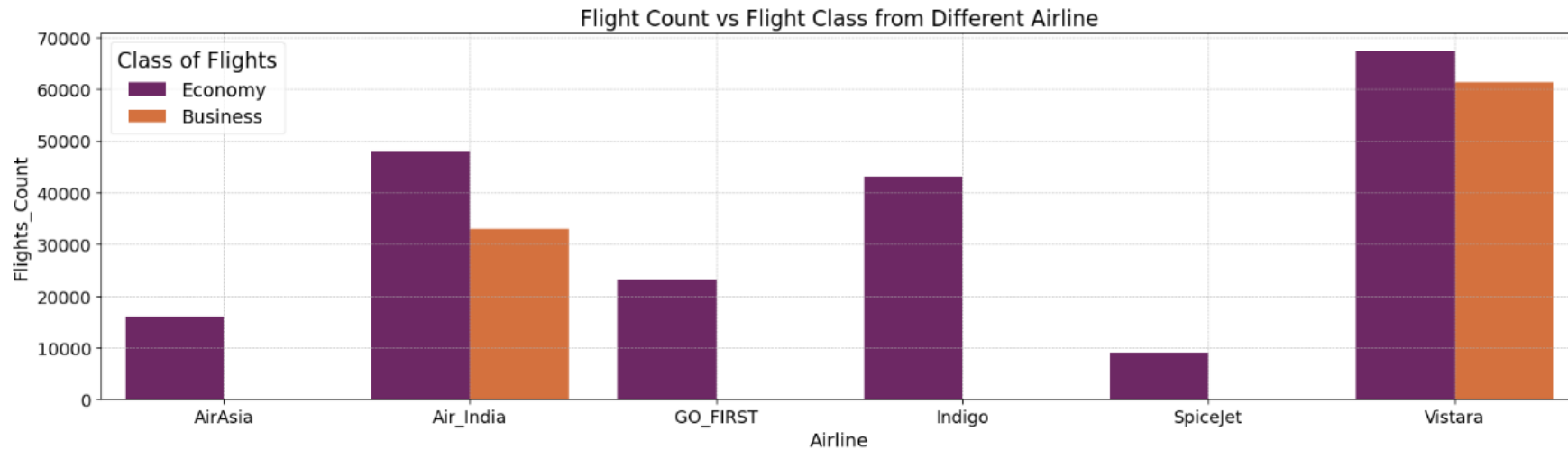


Conclusion

- People prefer to take Morning Flight from there respective Source_City.



Data Analysis and Plotting



Conclusion

- Passengers prefer to choose Business Class exclusively from Air India and Vistara.
- Among these, Vistara is the most preferred airline for Business Class
- Similarly, Vistara also holds the highest preference for Economy Class among travelers.

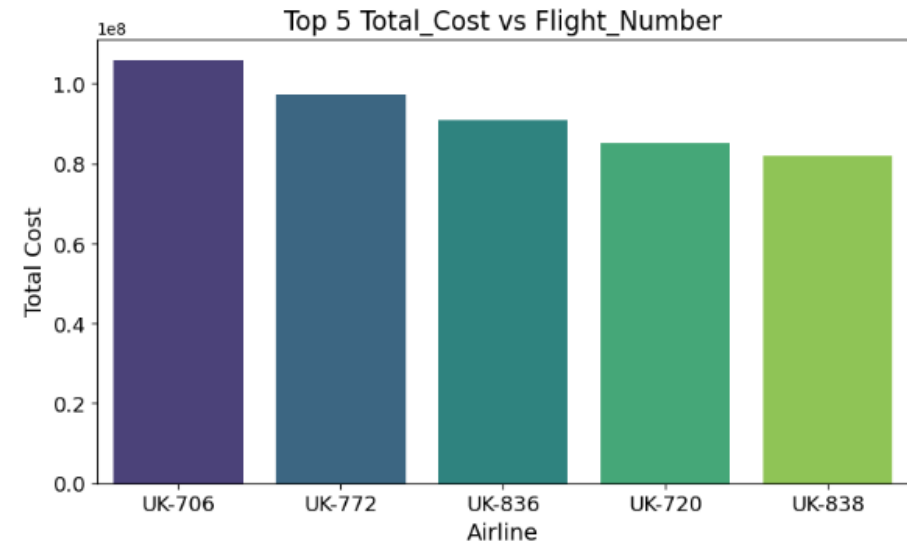
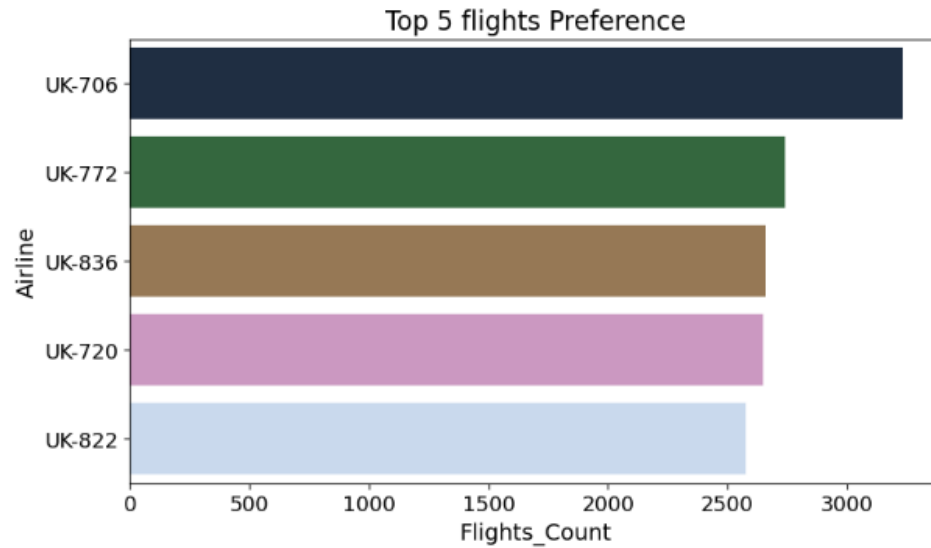
```
[242]: dataCopy.head(10)
```

```
[242]:
```

	airline	flight	source_city	departure_time	stops	arrival_time	destination_city	class	duration	days_left	price
0	SpiceJet	SG-8709	Delhi	Evening	zero	Night	Mumbai	Economy	2.17	1	5953.0
1	SpiceJet	SG-8157	Delhi	Early_Morning	zero	Morning	Mumbai	Economy	2.33	1	5953.0
2	AirAsia	I5-764	Delhi	Early_Morning	zero	Early_Morning	Mumbai	Economy	2.17	1	5956.0
3	Vistara	UK-995	Delhi	Morning	zero	Afternoon	Mumbai	Economy	2.25	1	5955.0
4	Vistara	UK-963	Delhi	Morning	zero	Morning	Mumbai	Economy	2.33	1	5955.0



Data Analysis and Plotting



Conclusion

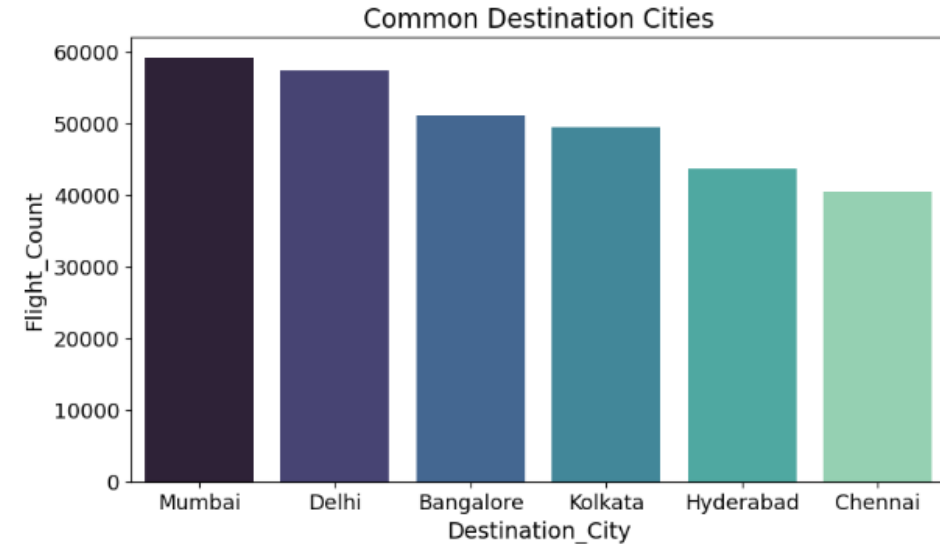
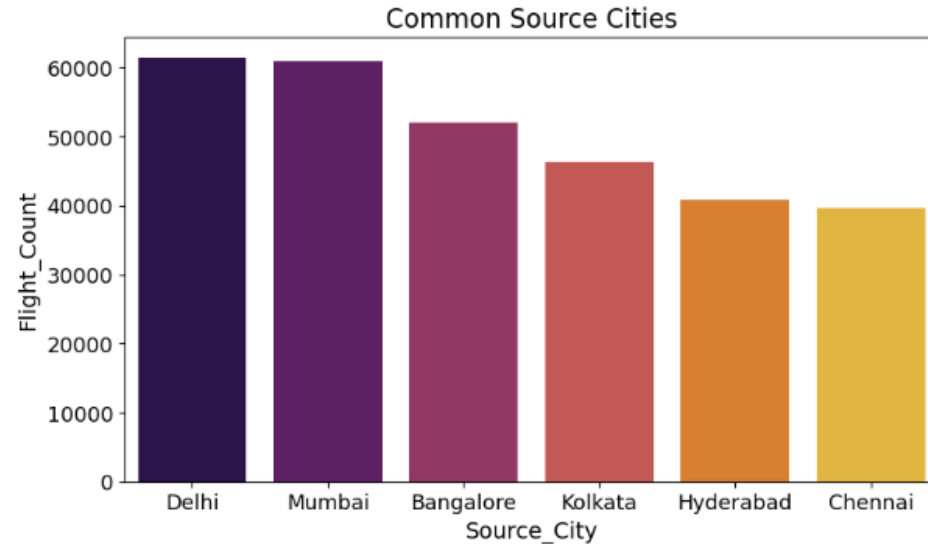
- The airline UK-706 operates a greater number of flights.
- UK-822 has a higher number of flights compared to UK-838, yet UK-838 generates more revenue than UK-822.
- UK-706 and UK-772 exhibit a dramatic variation in the number of flights. However, the total cost remains relatively stable between the two.

```
[177]: dataCopy.size
```

```
[177]: 3313266
```



Data Analysis and Plotting



Conclusion

- The most frequent source city is Delhi.
- The most common destination city is Mumbai.

```
[312]: dataCopy.head(10)
```

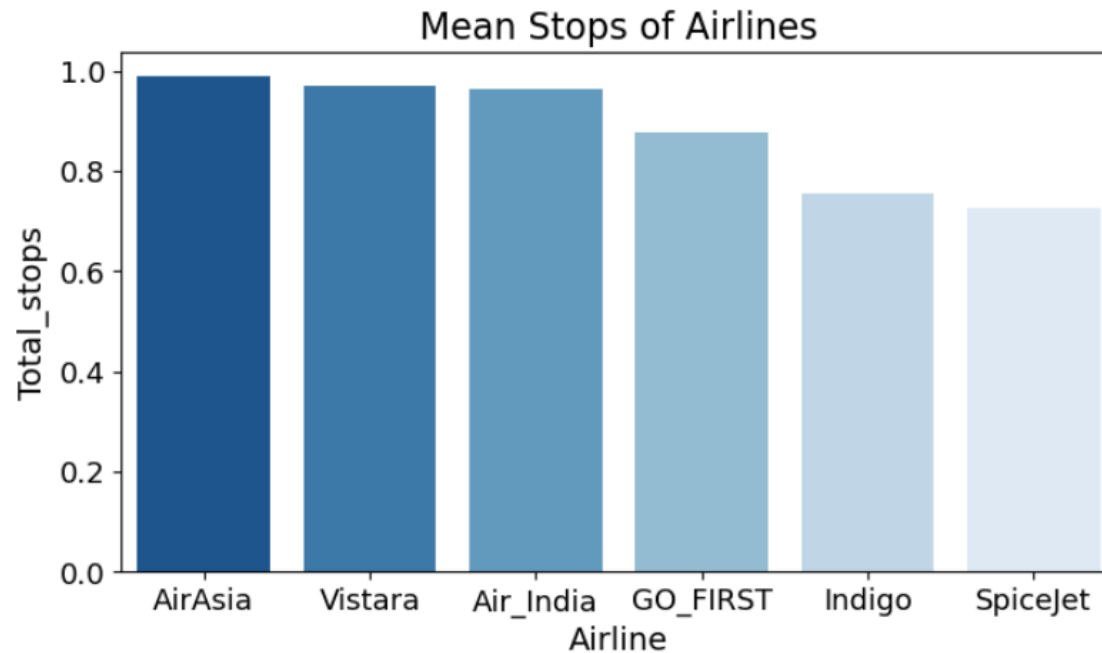
```
[312]:
```

	airline	flight	source_city	departure_time	stops	arrival_time	destination_city	class	duration	days_left	price
0	SpiceJet	SG-8709	Delhi	Evening	zero	Night	Mumbai	Economy	2.17	1	5953.0
1	SpiceJet	SG-8157	Delhi	Early_Morning	zero	Morning	Mumbai	Economy	2.33	1	5953.0
2	AirAsia	I5-764	Delhi	Early_Morning	zero	Early_Morning	Mumbai	Economy	2.17	1	5956.0



Data Analysis and Plotting

```
original_palette = sns.color_palette('Blues')  
# Reverse the palette  
reversed_palette = original_palette[::-1]  
sns.barplot(x = "airline", y = "Total_stops", data = stops, ax = axis,  
            palette = reversed_palette)  
axis.set_xlabel('Airline', fontsize=14)  
axis.set_ylabel('Total_stops', fontsize=14)  
axis.tick_params(axis='both', labelsize=13)  
axis.set_title('Mean Stops of Airlines', fontsize = 16)  
plt.show()
```



Conclusion

- AirAsia has more stops per flight compared to the other airlines.



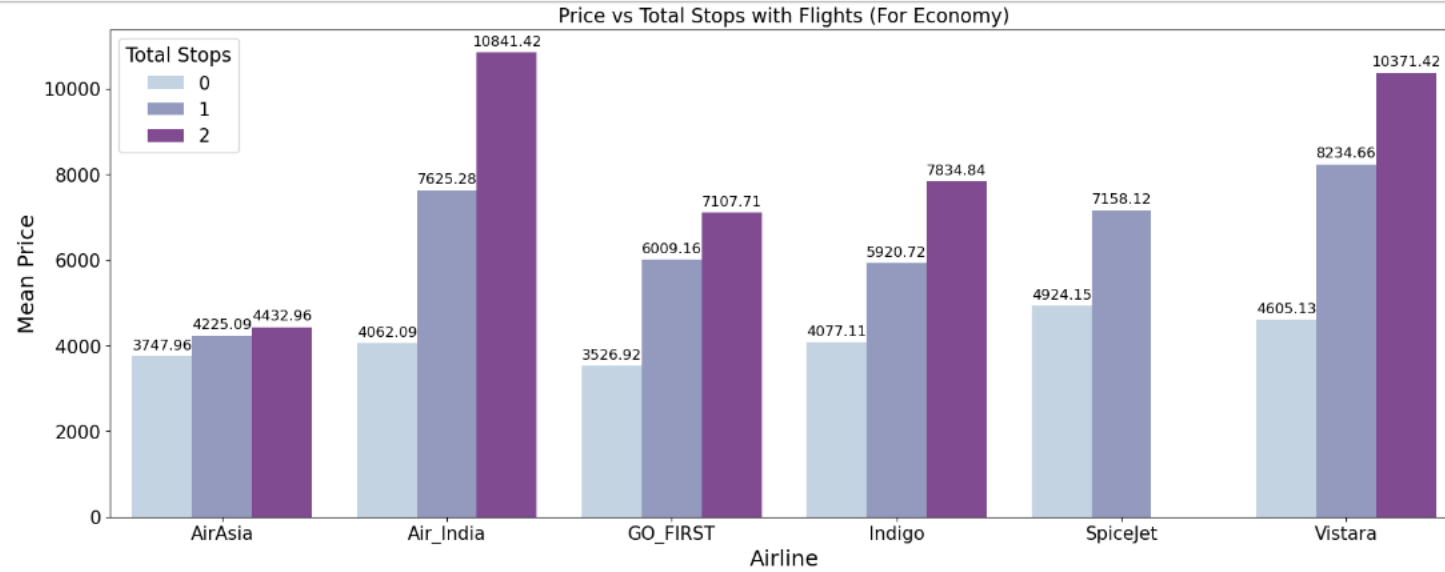
Data Analysis and Plotting

```
plt.show()
```



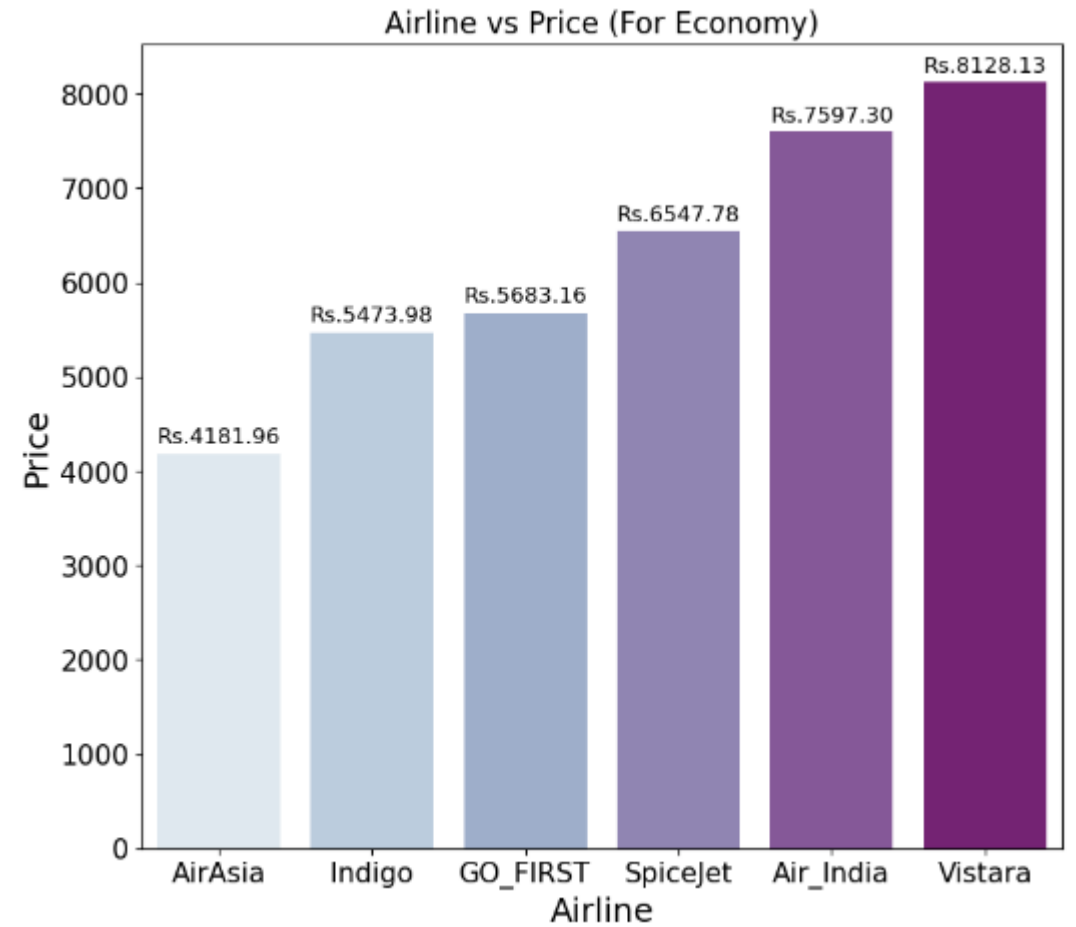
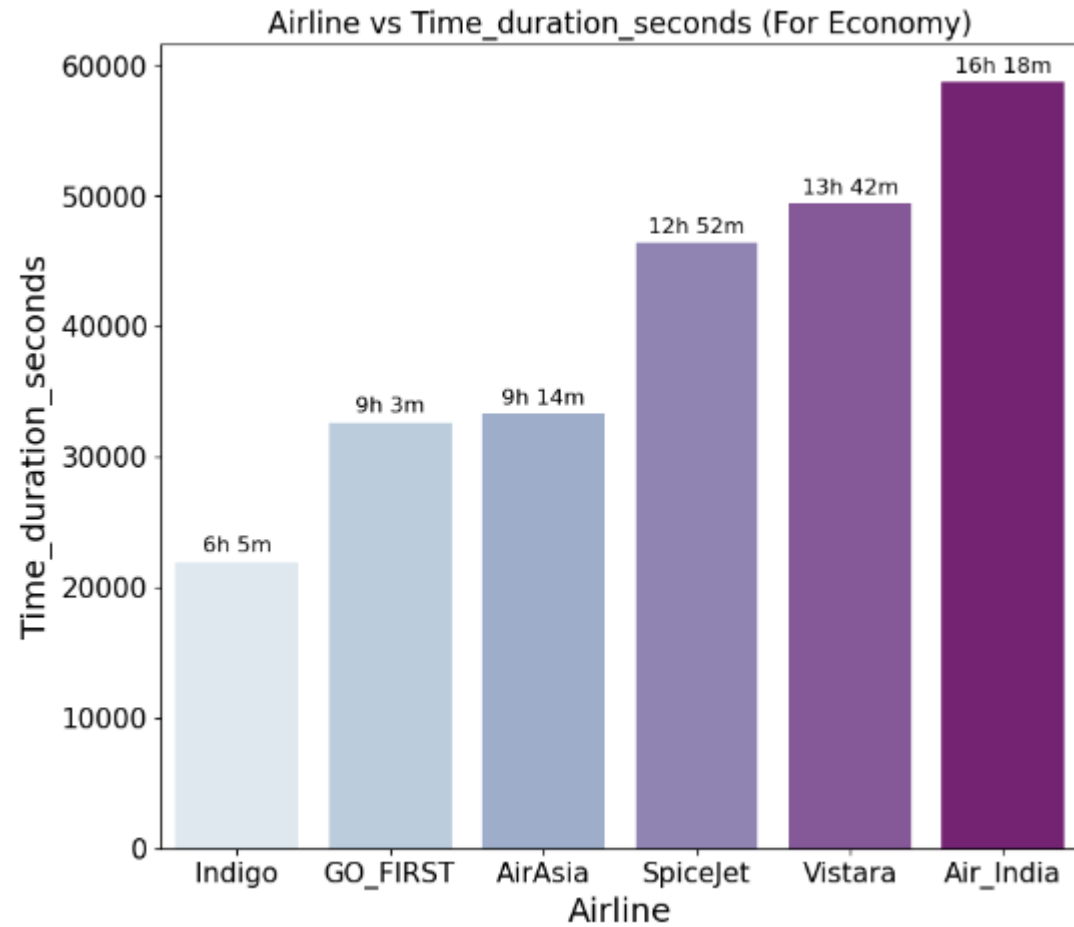


Data Analysis and Plotting



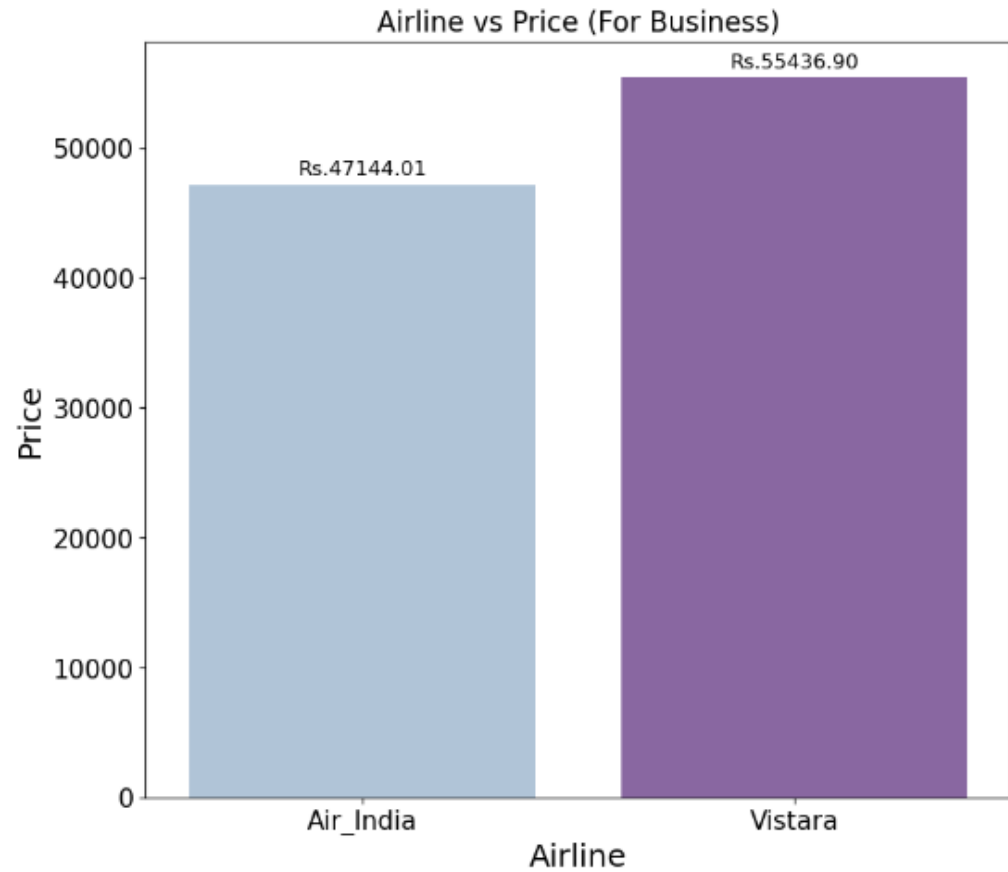
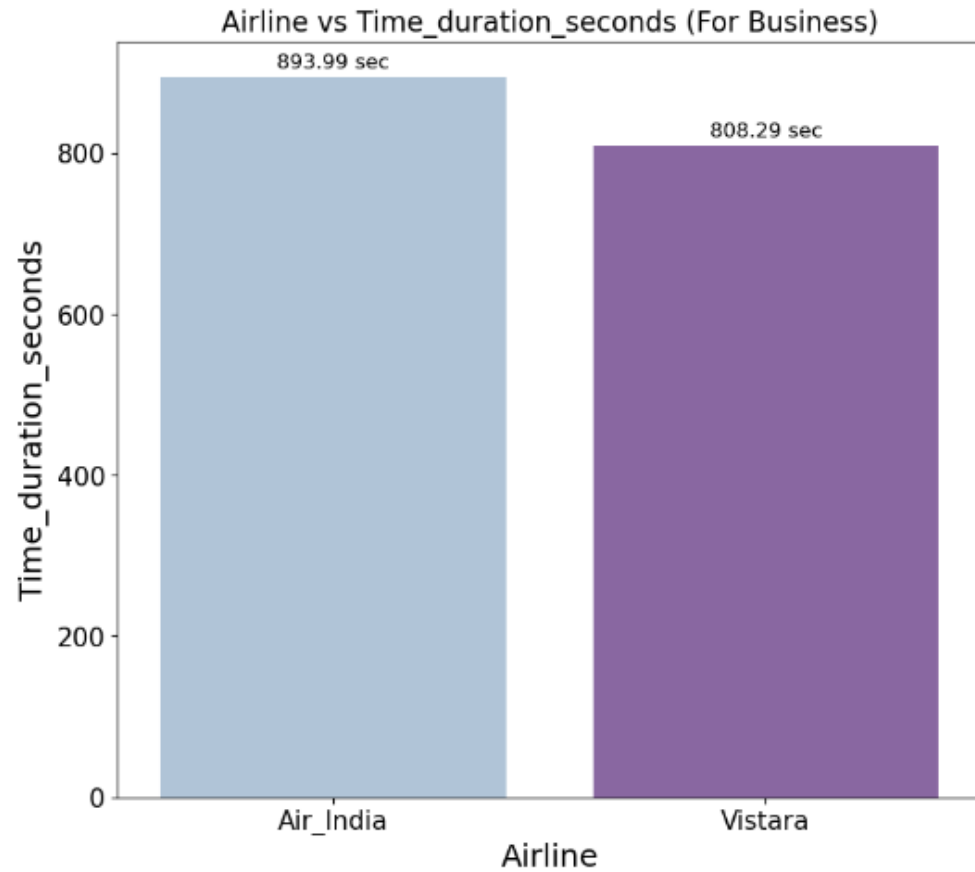


Data Analysis and Plotting



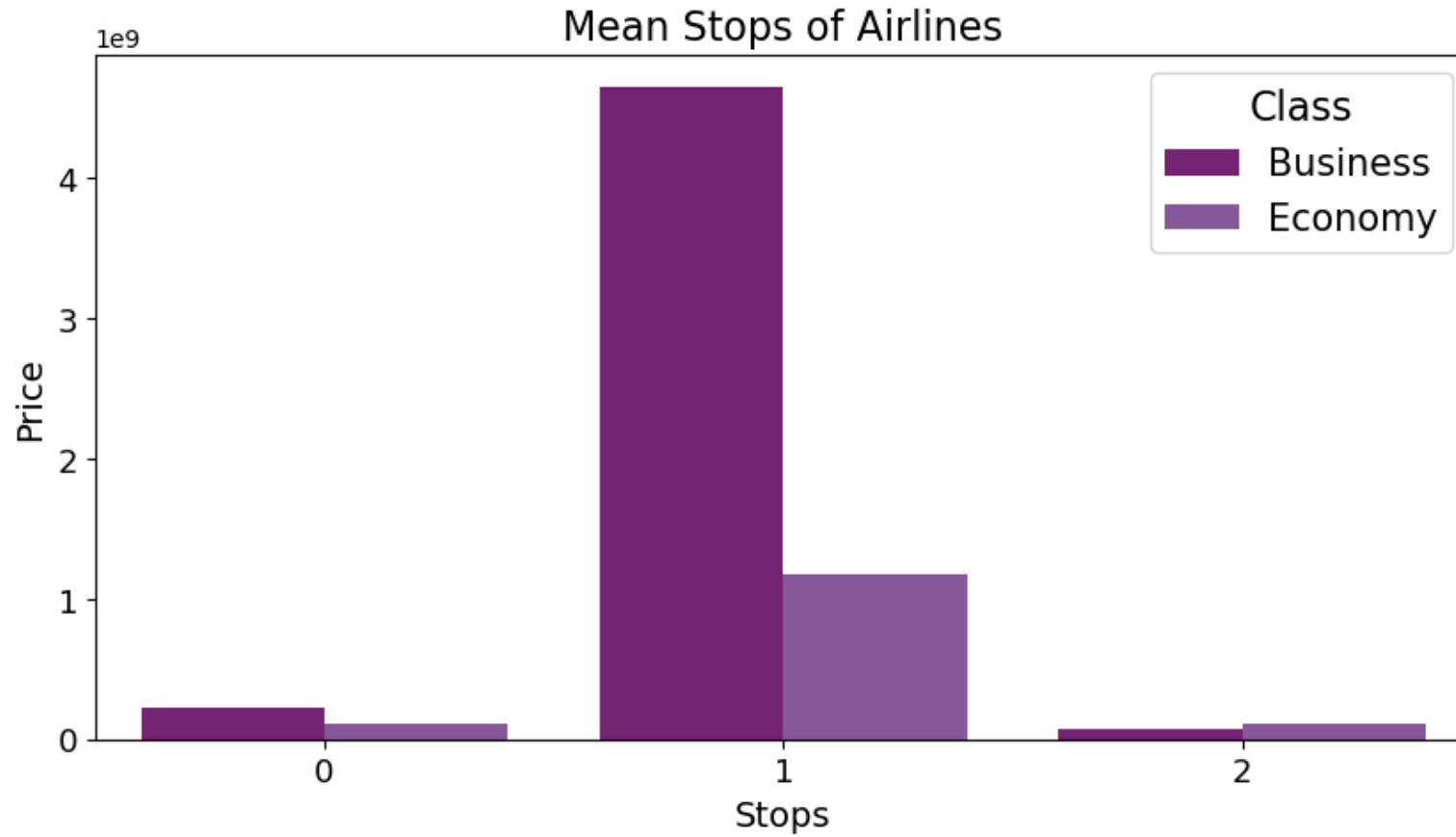


Data Analysis and Plotting





Data Analysis and Plotting





Vistara emerges as the **predominant airline**, with a **higher frequency of flights** and a significant **preference** for both **Business and Economy classes**. Flights from **Delhi and Mumbai** are the most **abundant**, with **morning departures** being favored. Among specific flights, **UK-706** stands out for its **volume**, while **UK-838**, despite **fewer flights** compared to **UK-822**, generates **greater revenue**. The total **cost** remains consistent **between UK-706 and UK-772**, despite **notable differences** in **flight numbers**. **Delhi and Mumbai** dominate as the **leading source and destination** cities. For the **fastest flights**, **SpiceJet** excels in **Bangalore**, **Vistara** in **Chennai**, **AirAsia** in **Mumbai**, and **Air India** in **Hyderabad**. **AirAsia**, however, **tends** to have **more stops per flight**. **Business Flights** with **1 stoppage** is more **costly** than **other price rates**.



THANK
YOU