TASK 1: WEB SCRAPING TO GAIN COMPANY INSIGHTS

BRITISH AIRWAYS

Presenter

Mujahidul Islam

BSc. In Computer Science and Engineering

(Major in Data Science)

(2020-2024)

East West University, Dhaka, Bangladesh

BRITISH AIRWAYS

Introduction

Web scraping is the process of automatically extracting large amounts of data from websites. For this project, we scraped user reviews from British Airways' profile on Airline Quality to analyze customer satisfaction and gather insights on service quality, seat comfort, staff behavior, and other factors. This data provides valuable insights for understanding customer experience and identifying improvement areas for the airline.

BRITISH AIRWAYS

Objective

The main objective of this web scraping project is to:

•Gather authentic customer feedback on British Airways' services.

•Analyze reviews across various parameters such as **Seat Comfort**, **Cabin Staff Service**, **Ground Service**, **Value for Money**, and **Route**.

•Derive insights to understand user satisfaction and pain points, which can be used to inform improvements and enhance the customer experience.

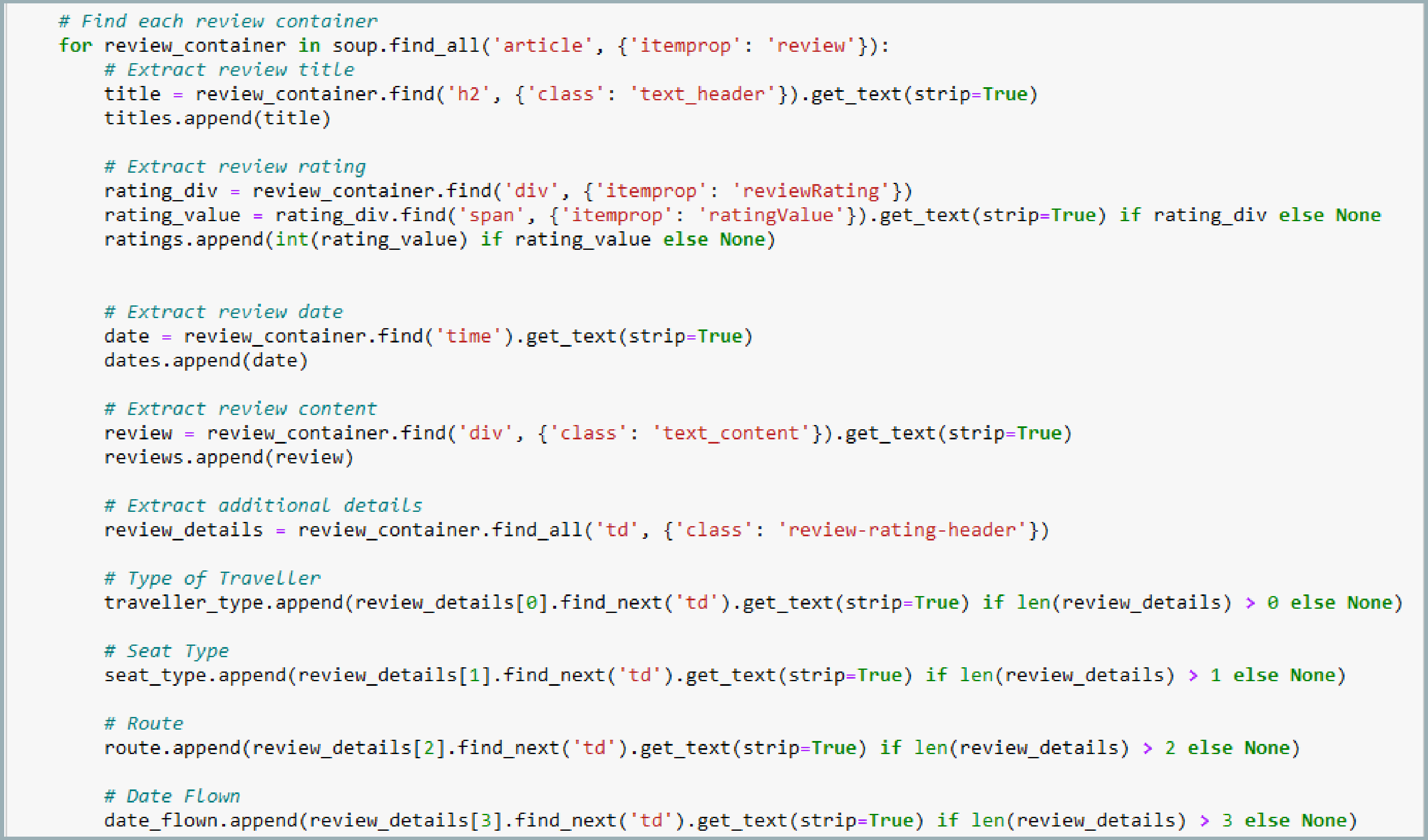
BRITISH AIRWAYS

Key Steps of Web Scraping

- Identify the Data Source
- Set Up the Scraper
- Extract and Clean Data
- Analyze and find insight of Data
- Save Data

BRITISH AIRWAYS

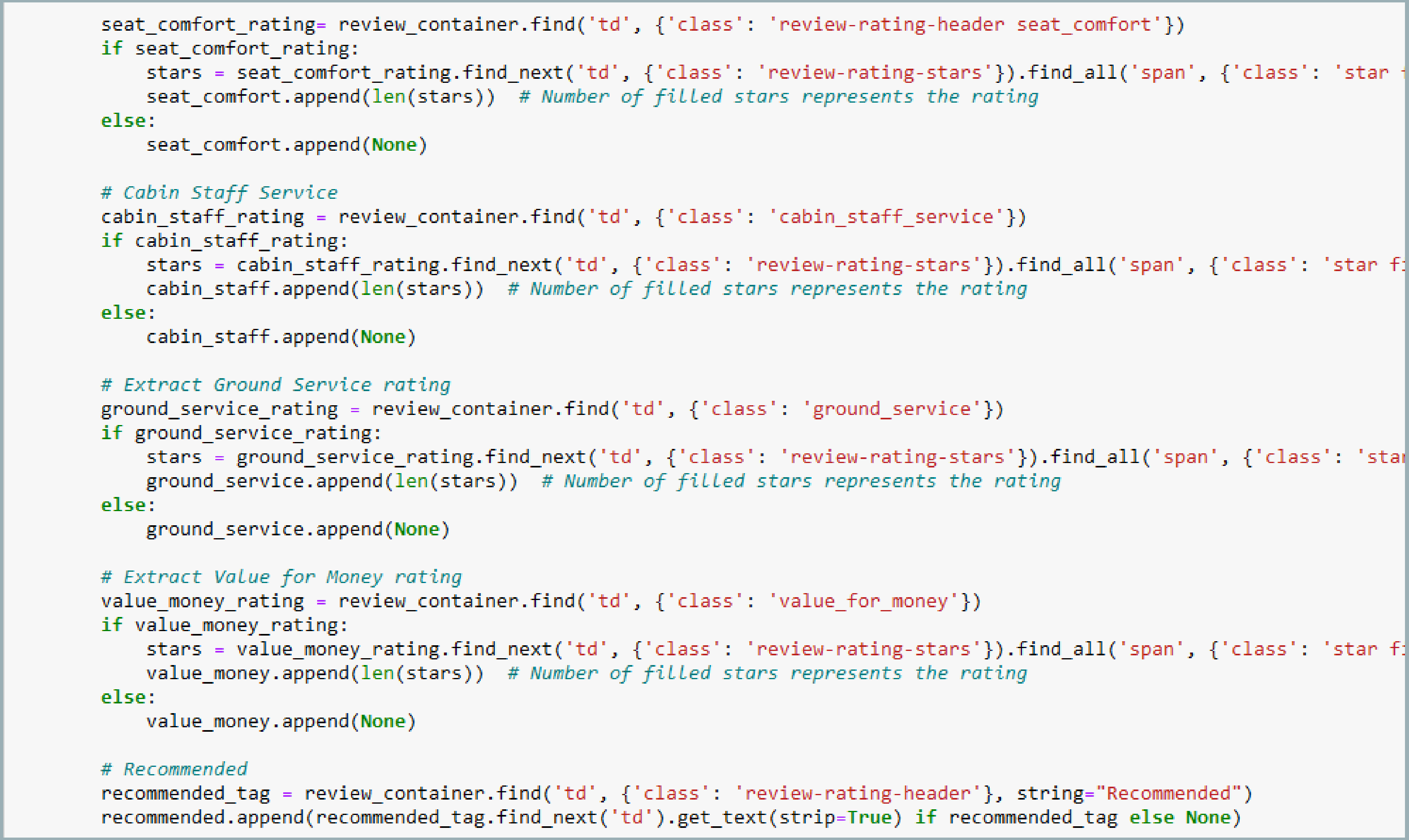
Visuals of Scrapping



```
# Find each review container
for review_container in soup.find_all('article', {'itemprop': 'review'}):
    # Extract review title
    title = review_container.find('h2', {'class': 'text_header'}).get_text(strip=True)
    titles.append(title)

    # Extract review rating
    rating_div = review_container.find('div', {'itemprop': 'reviewRating'})
    rating_value = rating_div.find('span', {'itemprop': 'ratingValue'}).get_text(strip=True) if rating_div else None
    ratings.append(int(rating_value) if rating_value else None)


    # Extract review date
    date = review_container.find('time').get_text(strip=True)
    dates.append(date)

    # Extract review content
    review = review_container.find('div', {'class': 'text_content'}).get_text(strip=True)
    reviews.append(review)

    # Extract additional details
    review_details = review_container.find_all('td', {'class': 'review-rating-header'})

    # Type of Traveller
    traveller_type.append(review_details[0].find_next('td').get_text(strip=True) if len(review_details) > 0 else None)

    # Seat Type
    seat_type.append(review_details[1].find_next('td').get_text(strip=True) if len(review_details) > 1 else None)

    # Route
    route.append(review_details[2].find_next('td').get_text(strip=True) if len(review_details) > 2 else None)

    # Date Flown
    date_flown.append(review_details[3].find_next('td').get_text(strip=True) if len(review_details) > 3 else None)
```

BRITISH AIRWAYS

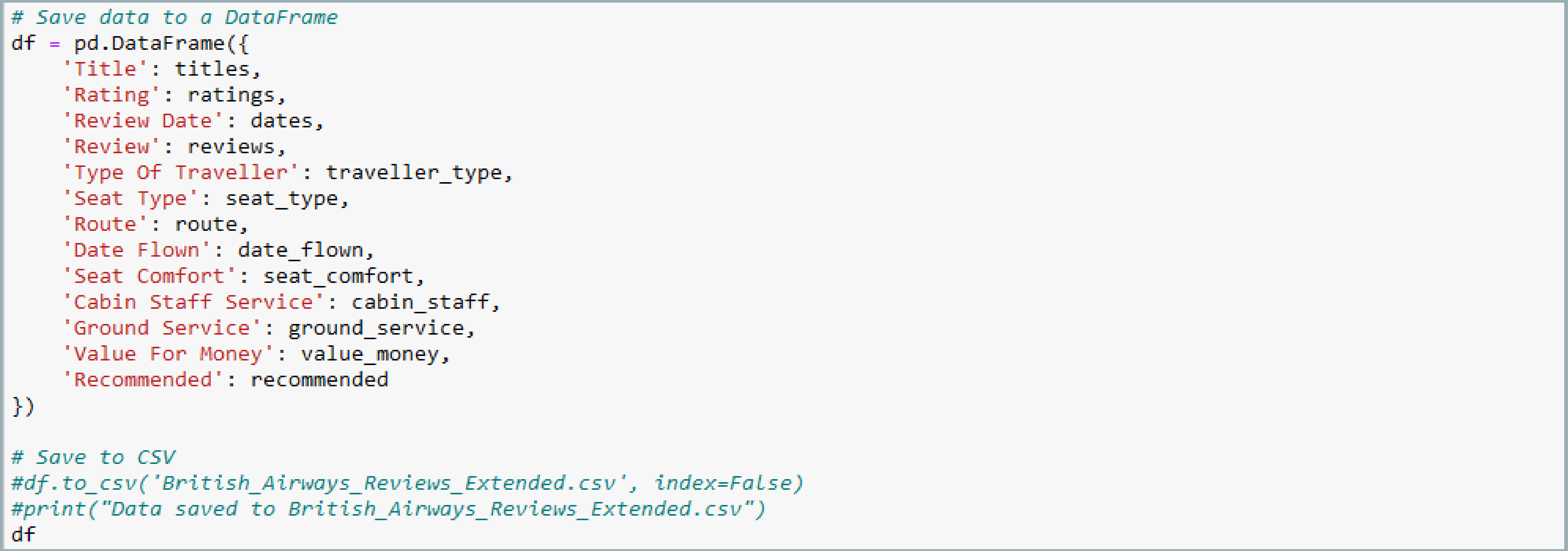
Visuals of Scrapping



```
seat_comfort_rating= review_container.find('td', {'class': 'review-rating-header seat_comfort'})
if seat_comfort_rating:
    stars = seat_comfort_rating.find_next('td', {'class': 'review-rating-stars'}).find_all('span', {'class': 'star f
    seat_comfort.append(len(stars))  # Number of filled stars represents the rating
else:
    seat_comfort.append(None)

# Cabin Staff Service
cabin_staff_rating = review_container.find('td', {'class': 'cabin_staff_service'})
if cabin_staff_rating:
    stars = cabin_staff_rating.find_next('td', {'class': 'review-rating-stars'}).find_all('span', {'class': 'star fi
    cabin_staff.append(len(stars))  # Number of filled stars represents the rating
else:
    cabin_staff.append(None)

# Extract Ground Service rating
ground_service_rating = review_container.find('td', {'class': 'ground_service'})
if ground_service_rating:
    stars = ground_service_rating.find_next('td', {'class': 'review-rating-stars'}).find_all('span', {'class': 'star
    ground_service.append(len(stars))  # Number of filled stars represents the rating
else:
    ground_service.append(None)

# Extract Value for Money rating
value_money_rating = review_container.find('td', {'class': 'value_for_money'})
if value_money_rating:
    stars = value_money_rating.find_next('td', {'class': 'review-rating-stars'}).find_all('span', {'class': 'star fi
    value_money.append(len(stars))  # Number of filled stars represents the rating
else:
    value_money.append(None)

# Recommended
recommended_tag = review_container.find('td', {'class': 'review-rating-header'}, string="Recommended")
recommended.append(recommended_tag.find_next('td').get_text(strip=True) if recommended_tag else None)
```

BRITISH AIRWAYS

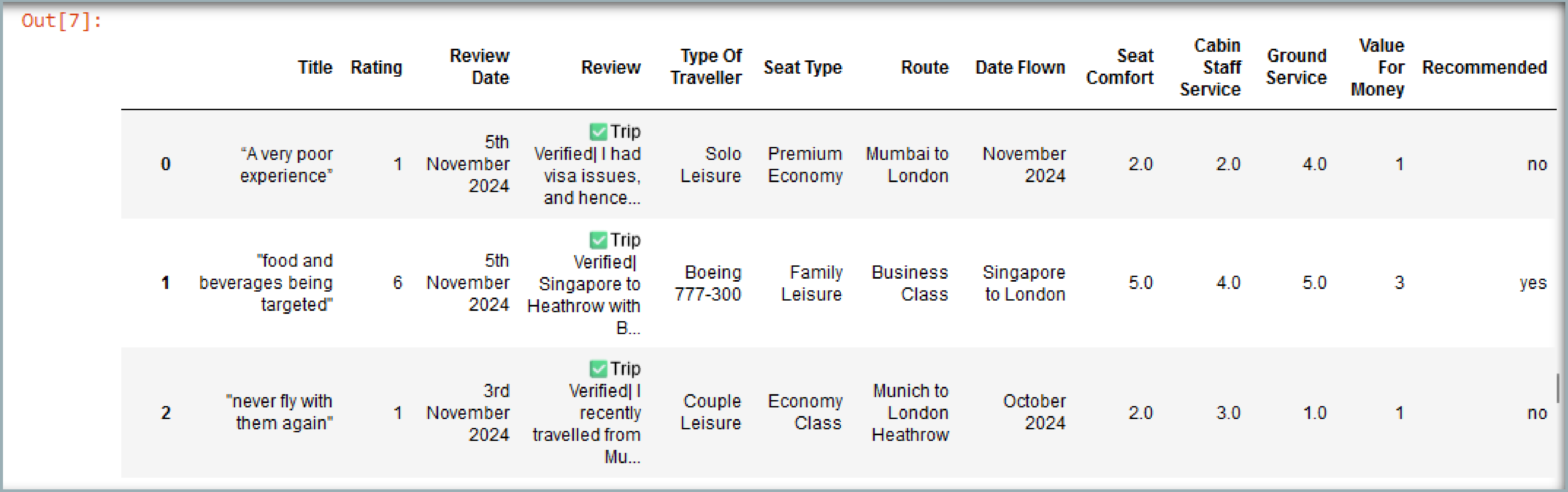
Visuals of Scrapping



```
# Save data to a DataFrame
df = pd.DataFrame({
    'Title': titles,
    'Rating': ratings,
    'Review Date': dates,
    'Review': reviews,
    'Type Of Traveller': traveller_type,
    'Seat Type': seat_type,
    'Route': route,
    'Date Flown': date_flown,
    'Seat Comfort': seat_comfort,
    'Cabin Staff Service': cabin_staff,
    'Ground Service': ground_service,
    'Value For Money': value_money,
    'Recommended': recommended
})

# Save to CSV
#df.to_csv('British_Airways_Reviews_Extended.csv', index=False)
#print("Data saved to British_Airways_Reviews_Extended.csv")
df
```

BRITISH AIRWAYS

Dataset



Out[7]:

	Title	Rating	Review Date	Review	Type Of Traveller	Seat Type	Route	Date Flown	Seat Comfort	Cabin Staff Service	Ground Service	Value For Money	Recommended
0	“A very poor experience”	1	5th November 2024	✅Trip Verified| I had visa issues, and hence...	Solo Leisure	Premium Economy	Mumbai to London	November 2024	2.0	2.0	4.0	1	no
1	"food and beverages being targeted"	6	5th November 2024	✅Trip Verified| Singapore to Heathrow with B...	Boeing 777-300	Family Leisure	Business Class	Singapore to London	5.0	4.0	5.0	3	yes
2	"never fly with them again"	1	3rd November 2024	✅Trip Verified| I recently travelled from Mu...	Couple Leisure	Economy Class	Munich to London Heathrow	October 2024	2.0	3.0	1.0	1	no

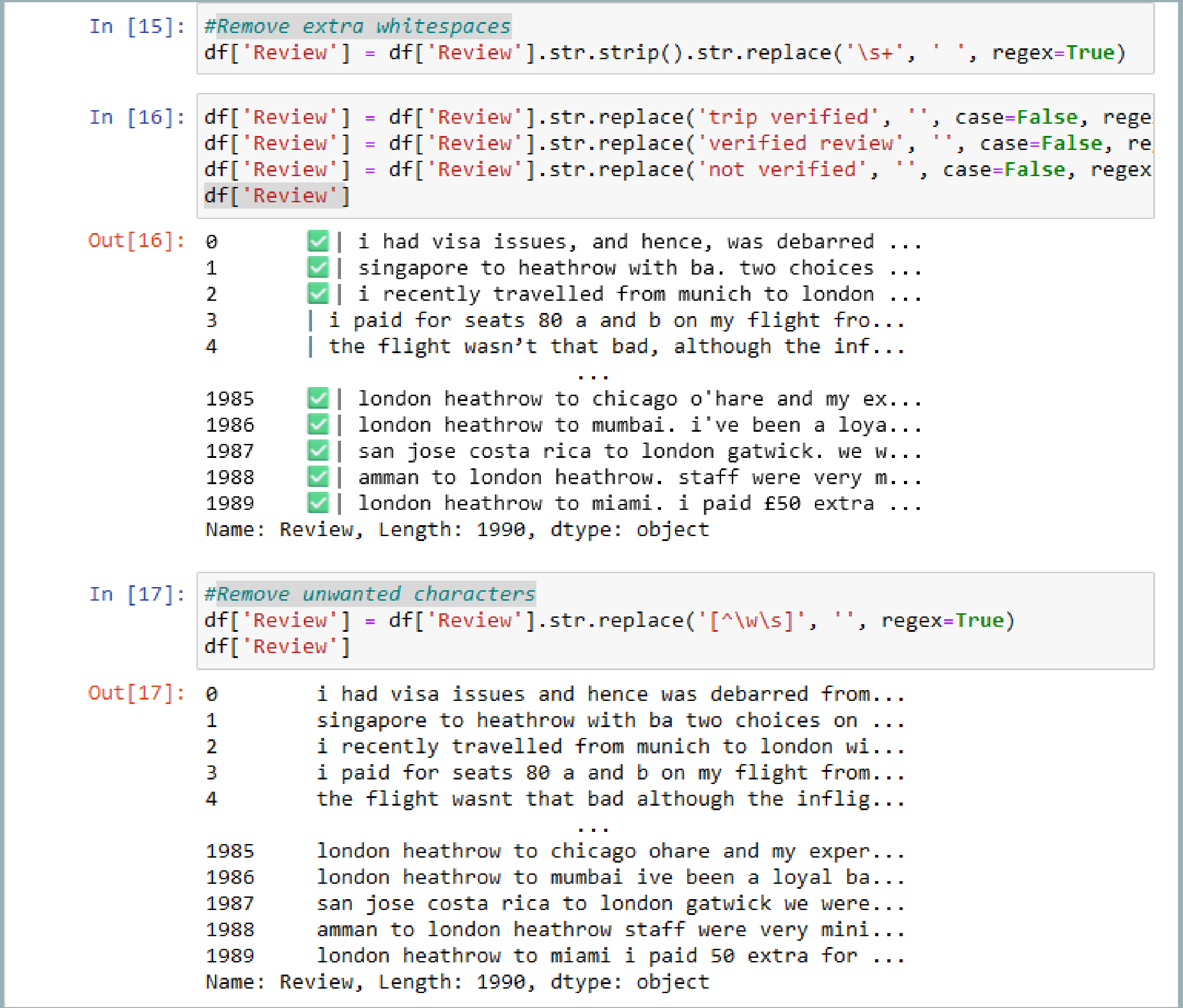
BRITISH AIRWAYS

Cleaning and Processing

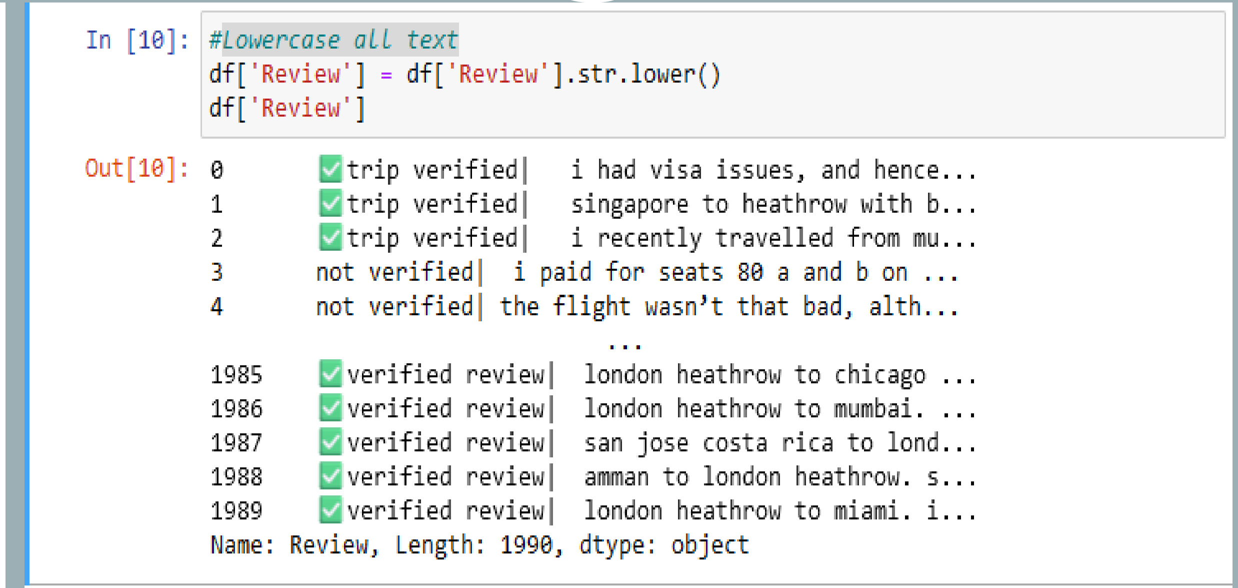
- Lowercase all text
- Remove extra whitespaces
- Remove unwanted characters
- Spelling Correction
- Perform Sentiment Analysis

BRITISH AIRWAYS

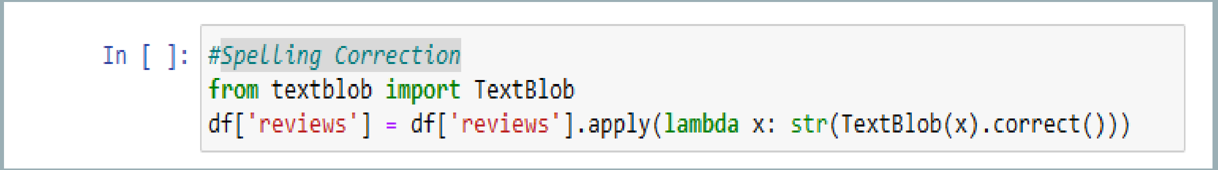
Visuals of Cleaning



```
In [15]: #Remove extra whitespaces
         df['Review'] = df['Review'].str.strip().str.replace('\s+', ' ', regex=True)

In [16]: df['Review'] = df['Review'].str.replace('trip verified', '', case=False, rege
         df['Review'] = df['Review'].str.replace('verified review', '', case=False, re
         df['Review'] = df['Review'].str.replace('not verified', '', case=False, regex
         df['Review']

Out[16]: 0       ✅ | i had visa issues, and hence, was debarred ...
         1       ✅ | singapore to heathrow with ba. two choices ...
         2       ✅ | i recently travelled from munich to london ...
         3       | i paid for seats 80 a and b on my flight fro...
         4       | the flight wasn’t that bad, although the inf...
                                       ...
         1985    ✅ | london heathrow to chicago o'hare and my ex...
         1986    ✅ | london heathrow to mumbai. i've been a loya...
         1987    ✅ | san jose costa rica to london gatwick. we w...
         1988    ✅ | amman to london heathrow. staff were very m...
         1989    ✅ | london heathrow to miami. i paid £50 extra ...
         Name: Review, Length: 1990, dtype: object

In [17]: #Remove unwanted characters
         df['Review'] = df['Review'].str.replace('[^\w\s]', '', regex=True)
         df['Review']

Out[17]: 0         i had visa issues and hence was debarred from...
         1         singapore to heathrow with ba two choices on ...
         2         i recently travelled from munich to london wi...
         3         i paid for seats 80 a and b on my flight from...
         4         the flight wasnt that bad although the inflig...
                                       ...
         1985      london heathrow to chicago ohare and my exper...
         1986      london heathrow to mumbai ive been a loyal ba...
         1987      san jose costa rica to london gatwick we were...
         1988      amman to london heathrow staff were very mini...
         1989      london heathrow to miami i paid 50 extra for ...
         Name: Review, Length: 1990, dtype: object
```



```
In [10]: #Lowercase all text
         df['Review'] = df['Review'].str.lower()
         df['Review']

Out[10]: 0       ✅trip verified|   i had visa issues, and hence...
         1       ✅trip verified|   singapore to heathrow with b...
         2       ✅trip verified|   i recently travelled from mu...
         3       not verified|  i paid for seats 80 a and b on ...
         4       not verified| the flight wasn’t that bad, alth...
                                    ...
         1985    ✅verified review|  london heathrow to chicago ...
         1986    ✅verified review|  london heathrow to mumbai. ...
         1987    ✅verified review|  san jose costa rica to lond...
         1988    ✅verified review|  amman to london heathrow. s...
         1989    ✅verified review|  london heathrow to miami. i...
         Name: Review, Length: 1990, dtype: object
```



```
In [ ]: #Spelling Correction
        from textblob import TextBlob
        df['reviews'] = df['reviews'].apply(lambda x: str(TextBlob(x).correct()))
```

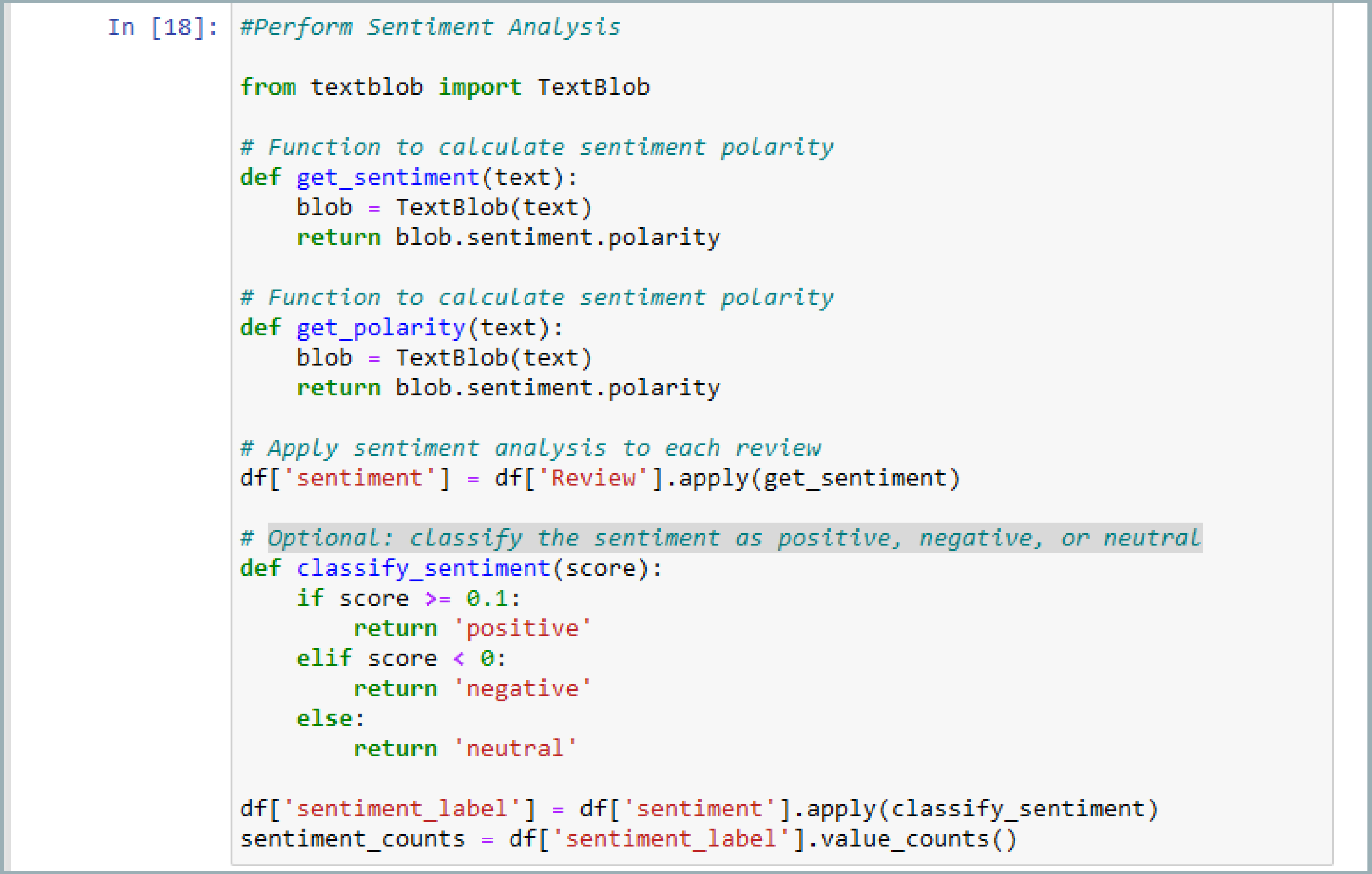
BRITISH AIRWAYS

Analysis (Based on Review)

- Perform Sentiment Analysis
- Apply sentiment analysis to each review
- Optional: classify the sentiment as positive, negative, or neutral
- Apply the polarity function to create a new 'Polarity' column
- Apply the classification function to create a new 'Sentiment' column

BRITISH AIRWAYS

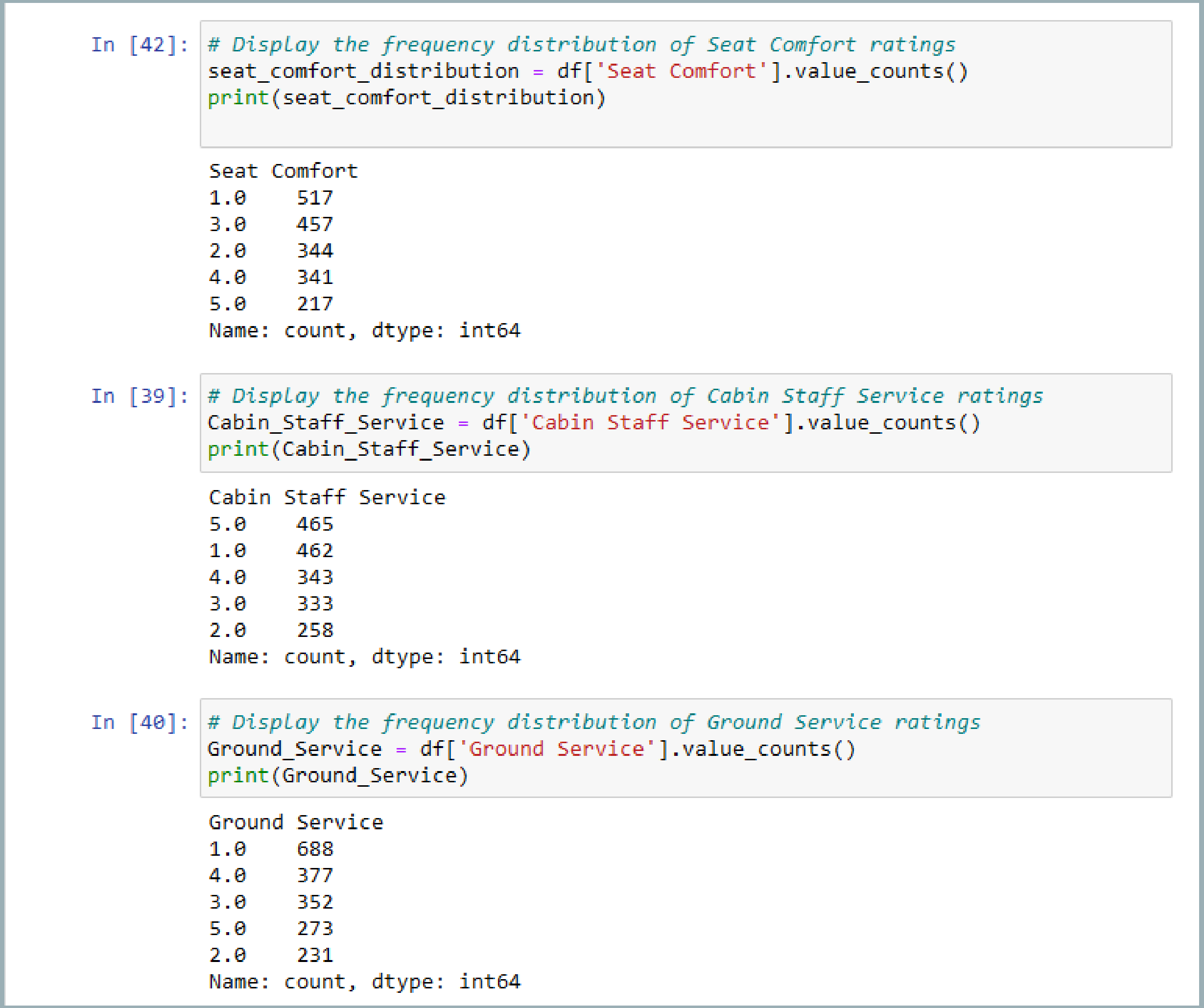
Visuals of Analysis



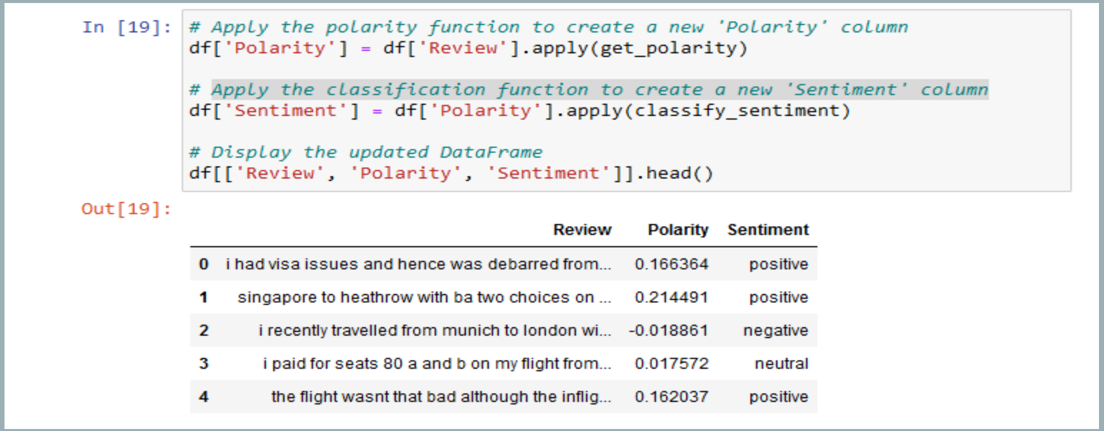
```
In [18]: #Perform Sentiment Analysis

         from textblob import TextBlob

         # Function to calculate sentiment polarity
         def get_sentiment(text):
             blob = TextBlob(text)
             return blob.sentiment.polarity

         # Function to calculate sentiment polarity
         def get_polarity(text):
             blob = TextBlob(text)
             return blob.sentiment.polarity

         # Apply sentiment analysis to each review
         df['sentiment'] = df['Review'].apply(get_sentiment)

         # Optional: classify the sentiment as positive, negative, or neutral
         def classify_sentiment(score):
             if score >= 0.1:
                 return 'positive'
             elif score < 0:
                 return 'negative'
             else:
                 return 'neutral'

         df['sentiment_label'] = df['sentiment'].apply(classify_sentiment)
         sentiment_counts = df['sentiment_label'].value_counts()
```

BRITISH AIRWAYS

Visuals of Analysis

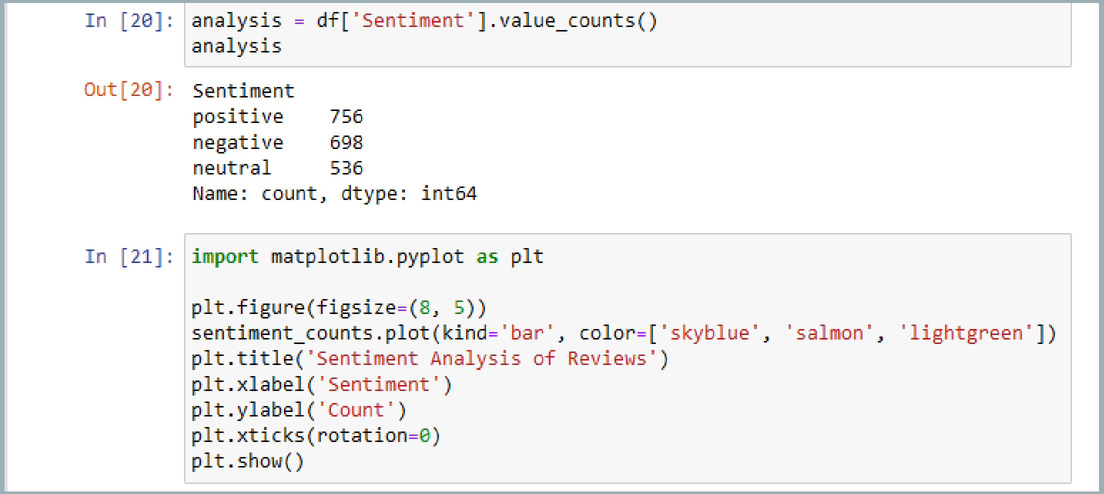


```
In [42]: # Display the frequency distribution of Seat Comfort ratings
         seat_comfort_distribution = df['Seat Comfort'].value_counts()
         print(seat_comfort_distribution)

         Seat Comfort
         1.0    517
         3.0    457
         2.0    344
         4.0    341
         5.0    217
         Name: count, dtype: int64

In [39]: # Display the frequency distribution of Cabin Staff Service ratings
         Cabin_Staff_Service = df['Cabin Staff Service'].value_counts()
         print(Cabin_Staff_Service)

         Cabin Staff Service
         5.0    465
         1.0    462
         4.0    343
         3.0    333
         2.0    258
         Name: count, dtype: int64

In [40]: # Display the frequency distribution of Ground Service ratings
         Ground_Service = df['Ground Service'].value_counts()
         print(Ground_Service)

         Ground Service
         1.0    688
         4.0    377
         3.0    352
         5.0    273
         2.0    231
         Name: count, dtype: int64
```



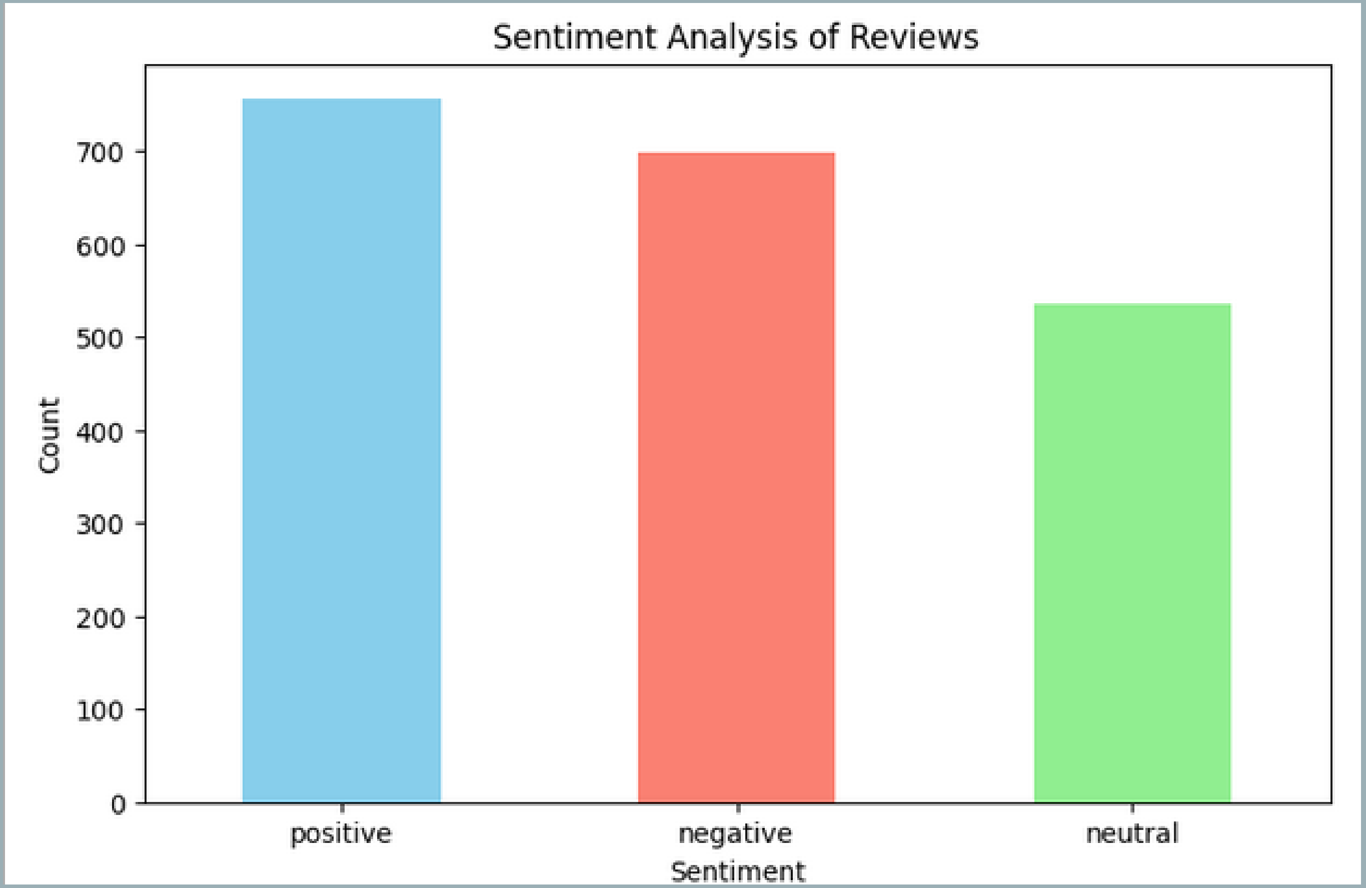
```
In [19]: # Apply the polarity function to create a new 'Polarity' column
         df['Polarity'] = df['Review'].apply(get_polarity)

         # Apply the classification function to create a new 'Sentiment' column
         df['Sentiment'] = df['Polarity'].apply(classify_sentiment)

         # Display the updated DataFrame
         df[['Review', 'Polarity', 'Sentiment']].head()

Out[19]:
```

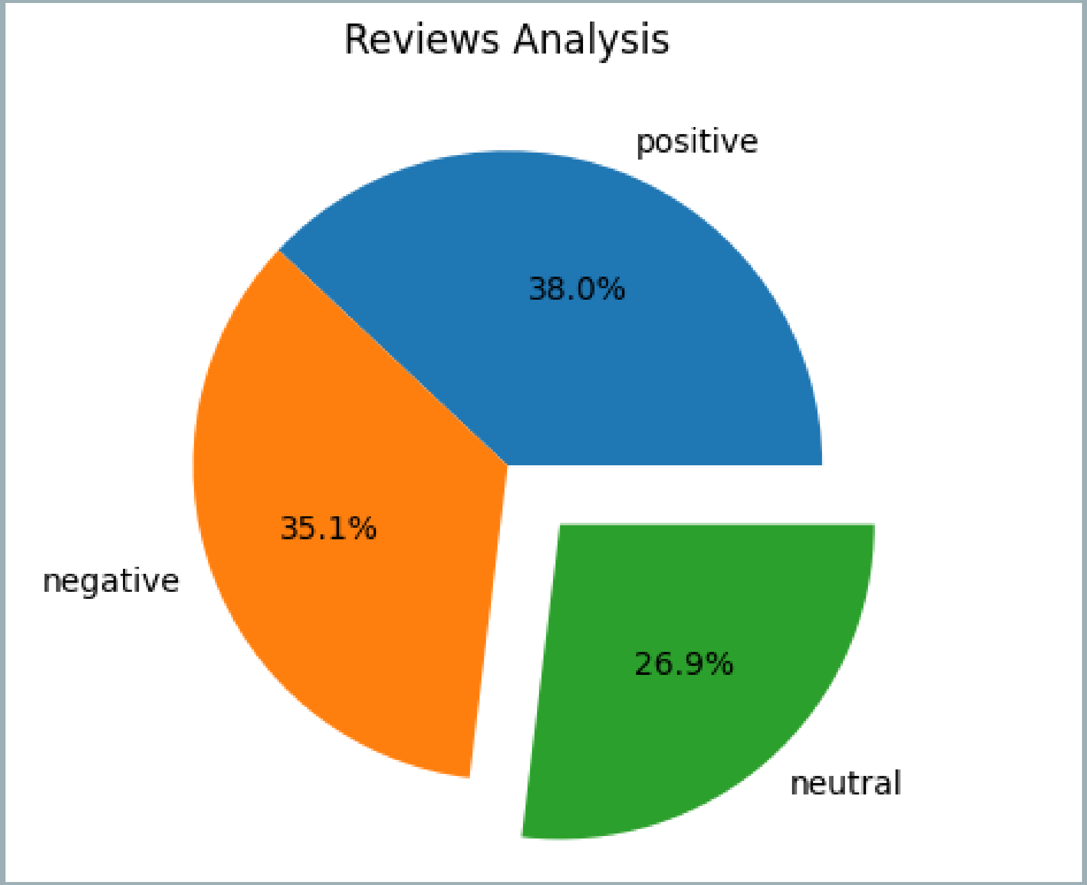
	Review	Polarity	Sentiment
0	i had visa issues and hence was debarred from...	0.166364	positive
1	singapore to heathrow with ba two choices on ...	0.214491	positive
2	i recently travelled from munich to london wi...	-0.018861	negative
3	i paid for seats 80 a and b on my flight from...	0.017572	neutral
4	the flight wasnt that bad although the inflig...	0.162037	positive



```
In [20]: analysis = df['Sentiment'].value_counts()
         analysis

Out[20]: Sentiment
         positive    756
         negative    698
         neutral     536
         Name: count, dtype: int64

In [21]: import matplotlib.pyplot as plt

         plt.figure(figsize=(8, 5))
         sentiment_counts.plot(kind='bar', color=['skyblue', 'salmon', 'lightgreen'])
         plt.title('Sentiment Analysis of Reviews')
         plt.xlabel('Sentiment')
         plt.ylabel('Count')
         plt.xticks(rotation=0)
         plt.show()
```

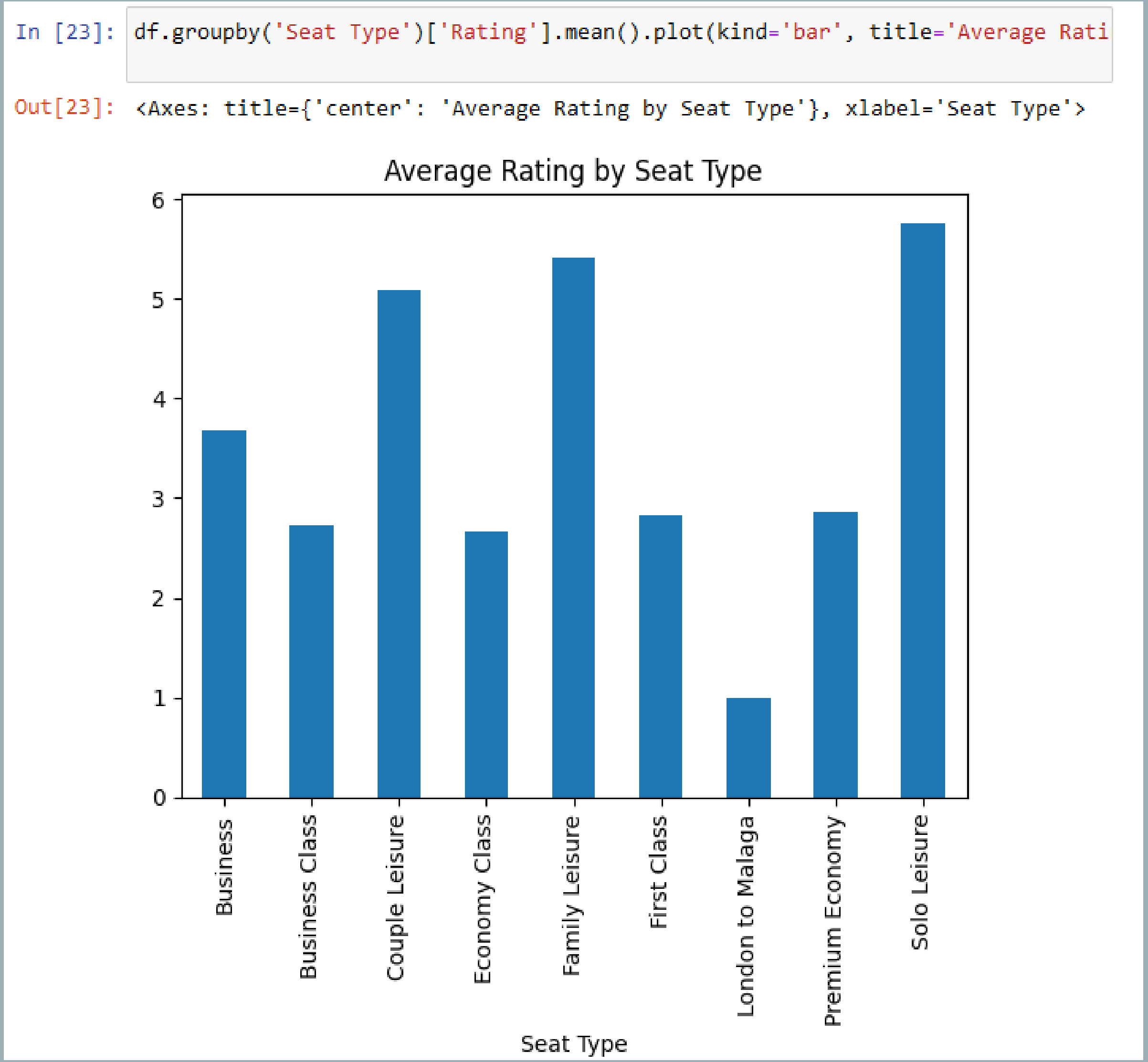
BRITISH AIRWAYS



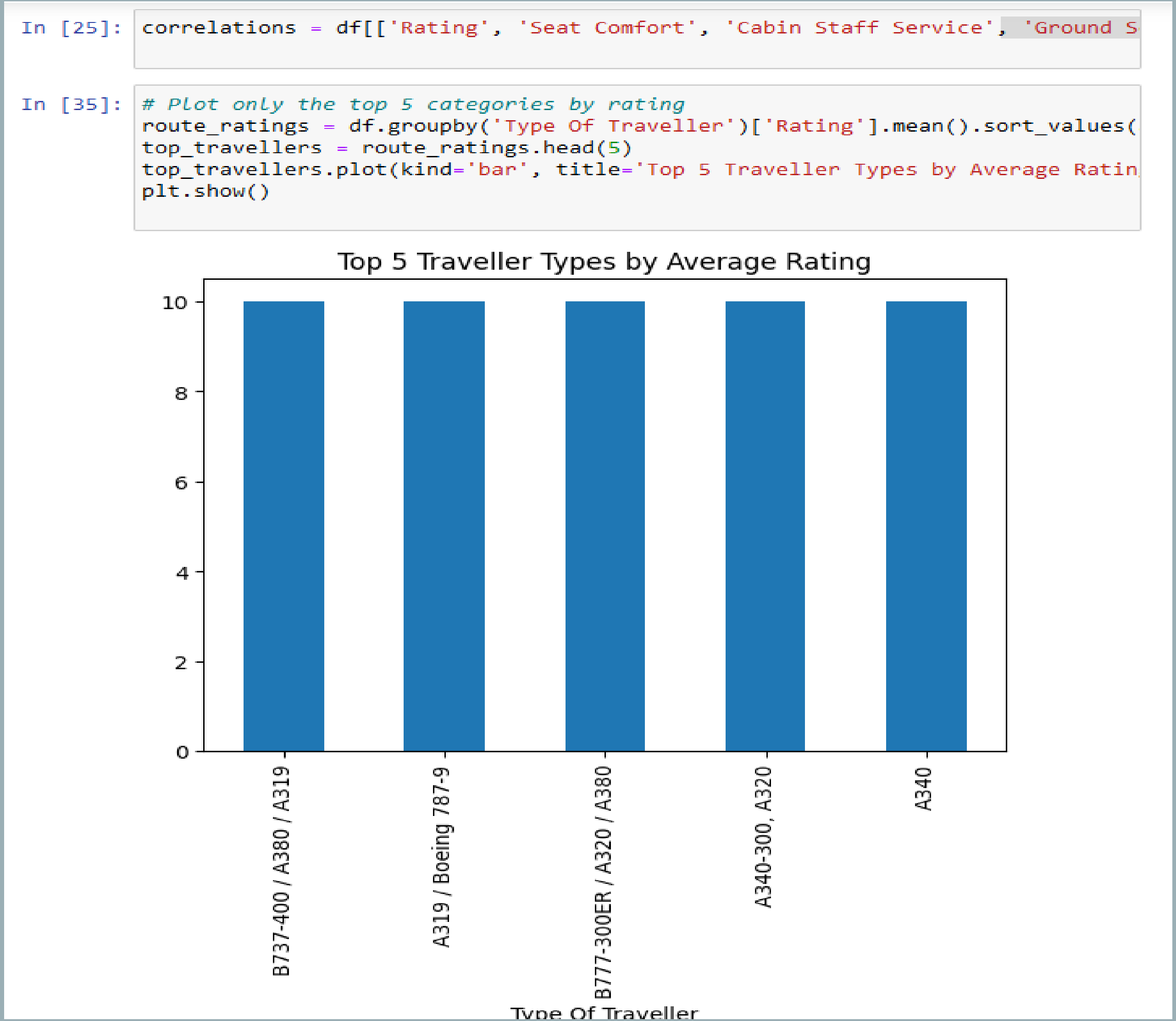


BRITISH AIRWAYS

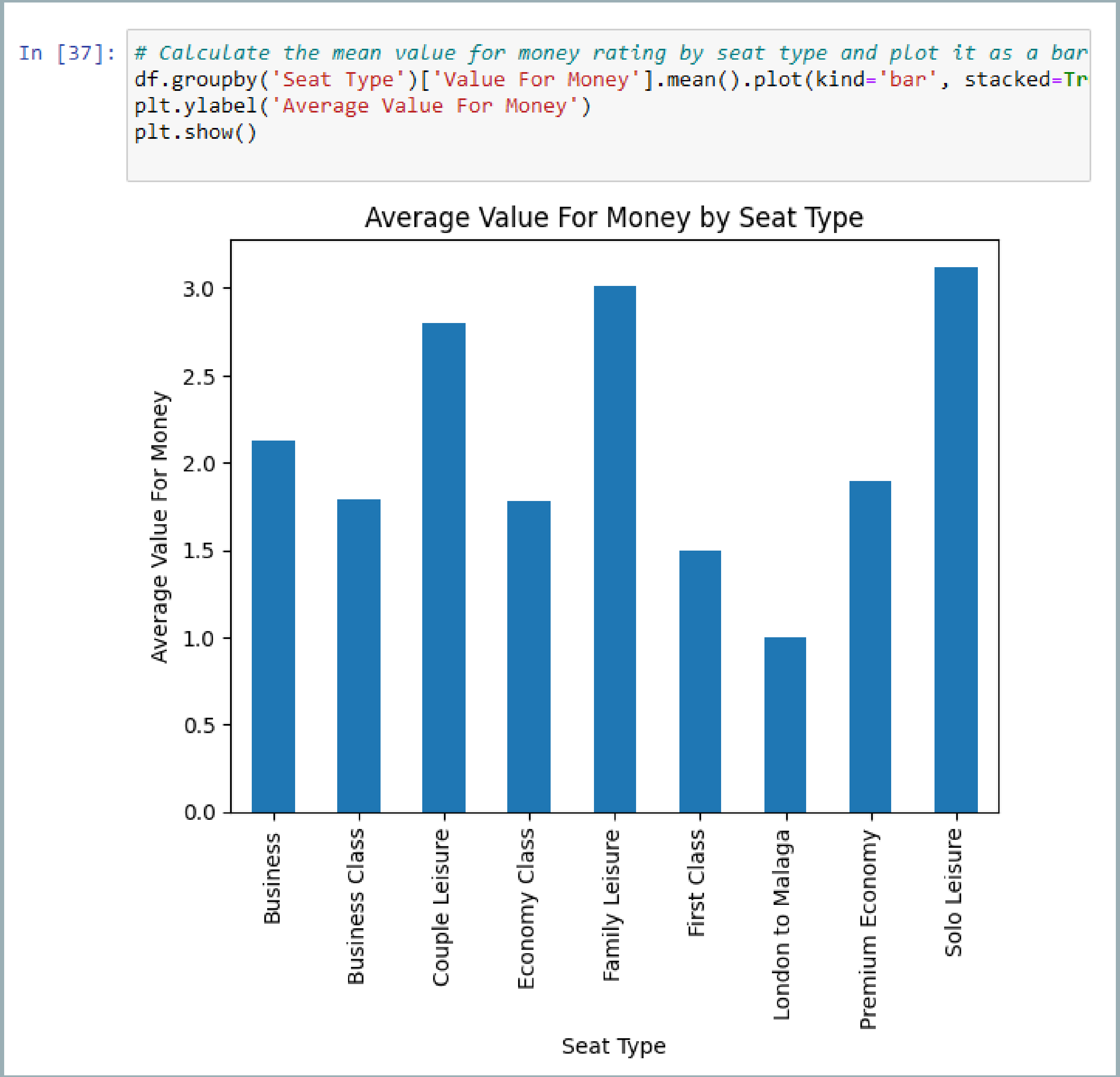
Analysis (Based on other columns)



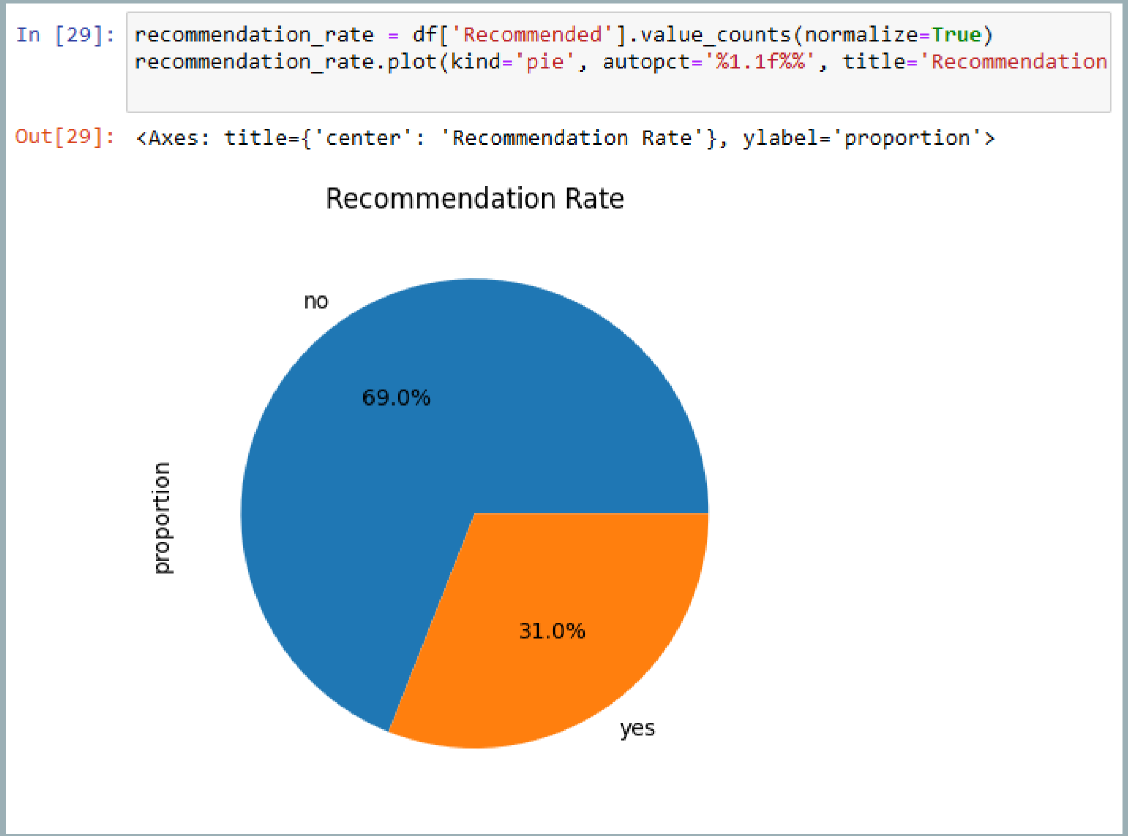
```
In [23]: df.groupby('Seat Type')['Rating'].mean().plot(kind='bar', title='Average Rati

Out[23]: <Axes: title={'center': 'Average Rating by Seat Type'}, xlabel='Seat Type'>
```



```
In [25]: correlations = df[['Rating', 'Seat Comfort', 'Cabin Staff Service', 'Ground S

In [35]: # Plot only the top 5 categories by rating
         route_ratings = df.groupby('Type Of Traveller')['Rating'].mean().sort_values(
         top_travellers = route_ratings.head(5)
         top_travellers.plot(kind='bar', title='Top 5 Traveller Types by Average Ratin
         plt.show()
```



```
In [37]: # Calculate the mean value for money rating by seat type and plot it as a bar
         df.groupby('Seat Type')['Value For Money'].mean().plot(kind='bar', stacked=Tr
         plt.ylabel('Average Value For Money')
         plt.show()
```



```
In [29]: recommendation_rate = df['Recommended'].value_counts(normalize=True)
         recommendation_rate.plot(kind='pie', autopct='%1.1f%%', title='Recommendation

Out[29]: <Axes: title={'center': 'Recommendation Rate'}, ylabel='proportion'>
```

THANK YOU

BRITISH AIRWAYS