Dataset Link: https://www.kaggle.com/datasets/mojtaba142/hotel-booking

Business Problem

In recent years, City Hotel and Resort Hotel have seen high cancellation rates. Each hotel is now dealing with a number of issues as a result, including fewer revenues and less than ideal hotel room use. Consequently, lowering cancellation rates is both hotels' primary goal in order to increase their efficiency in generating revenue, and for us to offer thorough business advice to address this problem.

The analysis of hotel booking cancellations as well as other factors that have no bearing on their business and yearly revenue generation are the main topics of this report.

Assumptions

1. No unusual occurrences between 2015 and 2017 will have a substantial impact on the data used.

2. The information is still current and can be used to analyze a hotel's possible plans in an efficient manner.

3. The biggest factor affecting the effectiveness of earning income is booking cancellations.

4. Cancellations result in vacant room for the booked length of time.

5. Clients make hotel reservations the same year they make cancellations.

Hypothesis

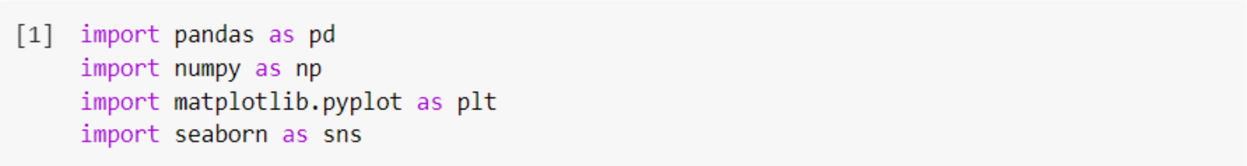
1. More cancellations occur when prices are higher.

2. When there is a longer waiting list, customers tend to cancel more frequently

3. The majority of clients are coming from offline travel agents to make their reservations.

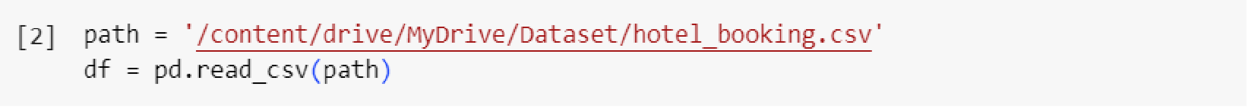
Exploratory Data Analysis

1. Importing Libraries



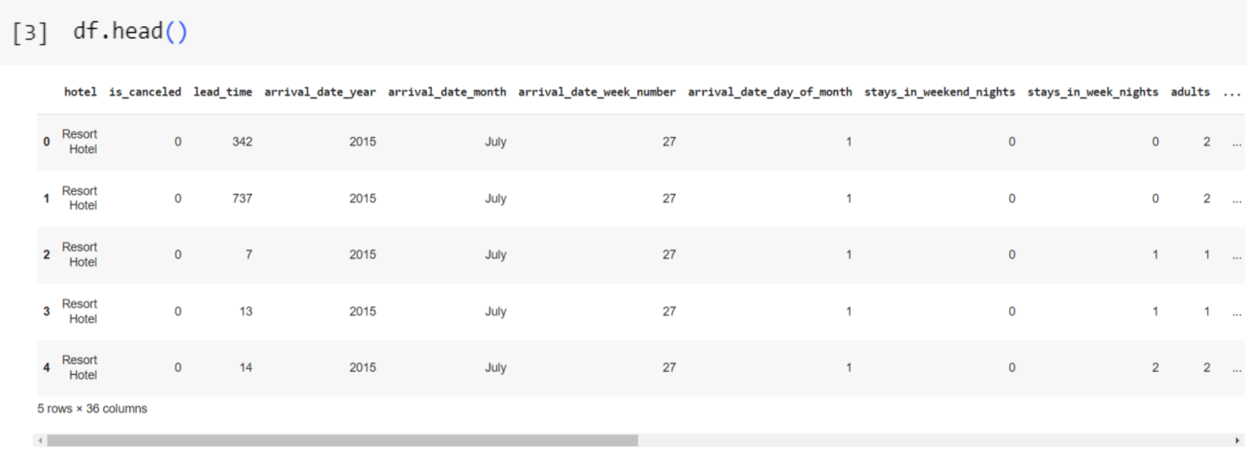
```
[1]  import pandas as pd
     import numpy as np
     import matplotlib.pyplot as plt
     import seaborn as sns
```

2. Loading The Dataset



```
[2]  path = '/content/drive/MyDrive/Dataset/hotel_booking.csv'
     df = pd.read_csv(path)
```

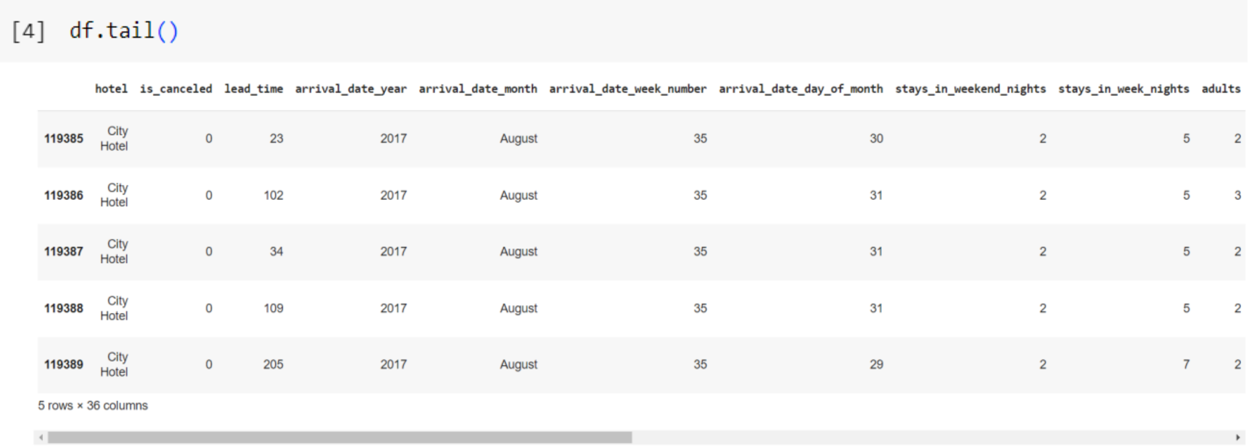
3. Exploring and Cleaning Data



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

	hotel	is_canceled	lead_time	arrival_date_year	arrival_date_month	arrival_date_week_number	arrival_date_day_of_month	stays_in_weekend_nights	stays_in_week_nights	adults	...
0	Resort Hotel	0	342	2015	July	27	1	0	0	2	...
1	Resort Hotel	0	737	2015	July	27	1	0	0	2	...
2	Resort Hotel	0	7	2015	July	27	1	0	1	1	...
3	Resort Hotel	0	13	2015	July	27	1	0	1	1	...
4	Resort Hotel	0	14	2015	July	27	1	0	2	2	...

5 rows × 36 columns

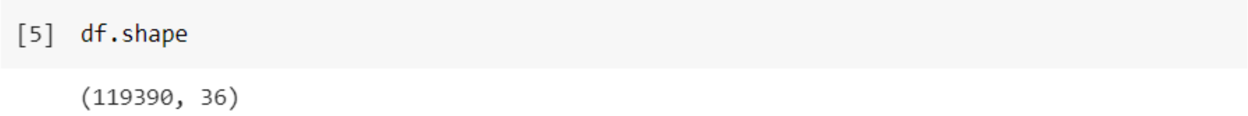


```
[4] df.tail()
```

	hotel	is_canceled	lead_time	arrival_date_year	arrival_date_month	arrival_date_week_number	arrival_date_day_of_month	stays_in_weekend_nights	stays_in_week_nights	adults
119385	City Hotel	0	23	2017	August	35	30	2	5	2
119386	City Hotel	0	102	2017	August	35	31	2	5	3
119387	City Hotel	0	34	2017	August	35	31	2	5	2
119388	City Hotel	0	109	2017	August	35	31	2	5	2
119389	City Hotel	0	205	2017	August	35	29	2	7	2

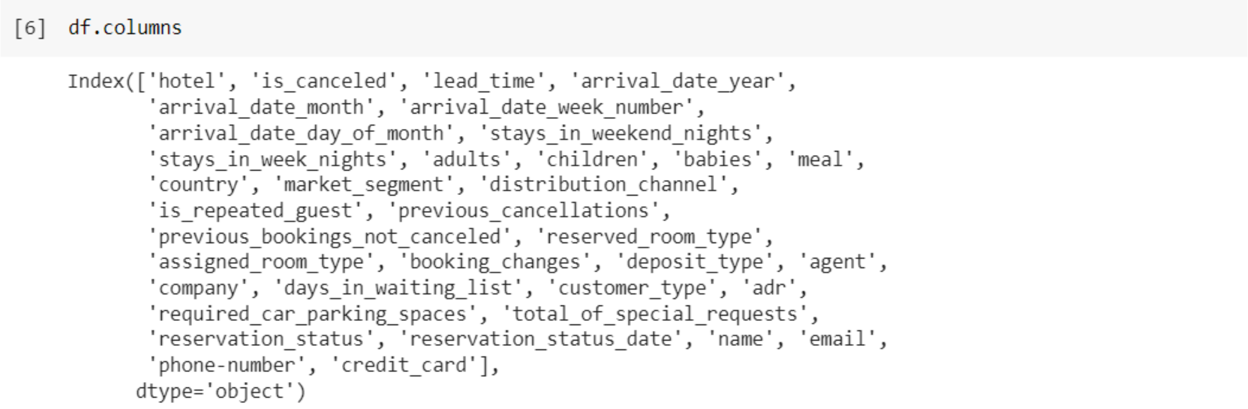
5 rows × 36 columns

Dataset Rows and Columns



```
[5] df.shape

    (119390, 36)
```

Dataset Information



```
[6] df.columns

    Index(['hotel', 'is_canceled', 'lead_time', 'arrival_date_year',
           'arrival_date_month', 'arrival_date_week_number',
           'arrival_date_day_of_month', 'stays_in_weekend_nights',
           'stays_in_week_nights', 'adults', 'children', 'babies', 'meal',
           'country', 'market_segment', 'distribution_channel',
           'is_repeated_guest', 'previous_cancellations',
           'previous_bookings_not_canceled', 'reserved_room_type',
           'assigned_room_type', 'booking_changes', 'deposit_type', 'agent',
           'company', 'days_in_waiting_list', 'customer_type', 'adr',
           'required_car_parking_spaces', 'total_of_special_requests',
           'reservation_status', 'reservation_status_date', 'name', 'email',
           'phone-number', 'credit_card'],
          dtype='object')
```

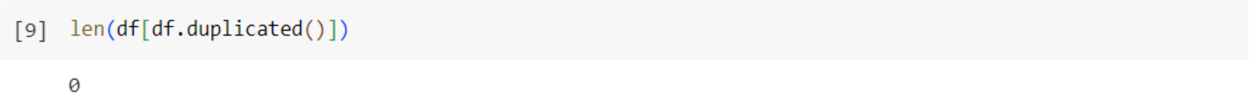
```
[7] df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 119390 entries, 0 to 119389
Data columns (total 36 columns):
 #   Column                          Non-Null Count   Dtype  
---  ------                          --------------   -----  
 0   hotel                           119390 non-null  object 
 1   is_canceled                     119390 non-null  int64  
 2   lead_time                       119390 non-null  int64  
 3   arrival_date_year               119390 non-null  int64  
 4   arrival_date_month              119390 non-null  object 
 5   arrival_date_week_number        119390 non-null  int64  
 6   arrival_date_day_of_month       119390 non-null  int64  
 7   stays_in_weekend_nights         119390 non-null  int64  
 8   stays_in_week_nights            119390 non-null  int64  
 9   adults                          119390 non-null  int64  
 10  children                        119386 non-null  float64
 11  babies                          119390 non-null  int64  
 12  meal                            119390 non-null  object 
 13  country                         118902 non-null  object 
 14  market_segment                  119390 non-null  object 
 15  distribution_channel            119390 non-null  object 
 16  is_repeated_guest               119390 non-null  int64  
 17  previous_cancellations          119390 non-null  int64  
 18  previous_bookings_not_canceled  119390 non-null  int64  
 19  reserved_room_type              119390 non-null  object 
 20  assigned_room_type              119390 non-null  object 
 21  booking_changes                 119390 non-null  int64  
 22  deposit_type                    119390 non-null  object 
 23  agent                           103050 non-null  float64
 24  company                         6797 non-null    float64
 25  days_in_waiting_list            119390 non-null  int64  
 26  customer_type                   119390 non-null  object 
 27  adr                             119390 non-null  float64
 28  required_car_parking_spaces     119390 non-null  int64  
 29  total_of_special_requests       119390 non-null  int64  
 30  reservation_status              119390 non-null  object 
 31  reservation_status_date         119390 non-null  object 
 32  name                            119390 non-null  object 
 33  email                           119390 non-null  object 
 34  phone-number                    119390 non-null  object 
 35  credit_card                     119390 non-null  object 
dtypes: float64(4), int64(16), object(16)
memory usage: 32.8+ MB
```

```
[8] df.describe(include = 'object')
```

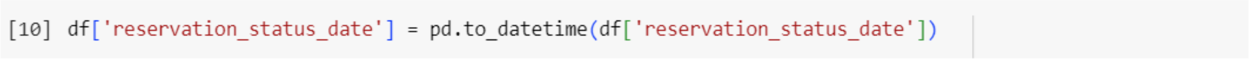
	hotel	arrival_date_month	meal	country	market_segment	distribution_channel	reserved_room_type	assigned_room_type	deposit_type	customer_type	reservation_status	reser
count	119390	119390	119390	118902	119390	119390	119390	119390	119390	119390	119390	
unique	2	12	5	177	8	5	10	12	3	4	3	
top	City Hotel	August	BB	PRT	Online TA	TA/TO	A	A	No Deposit	Transient	Check-Out	
freq	79330	13877	92310	48590	56477	97870	85994	74053	104641	89613	75166	

Checking Duplicates Values

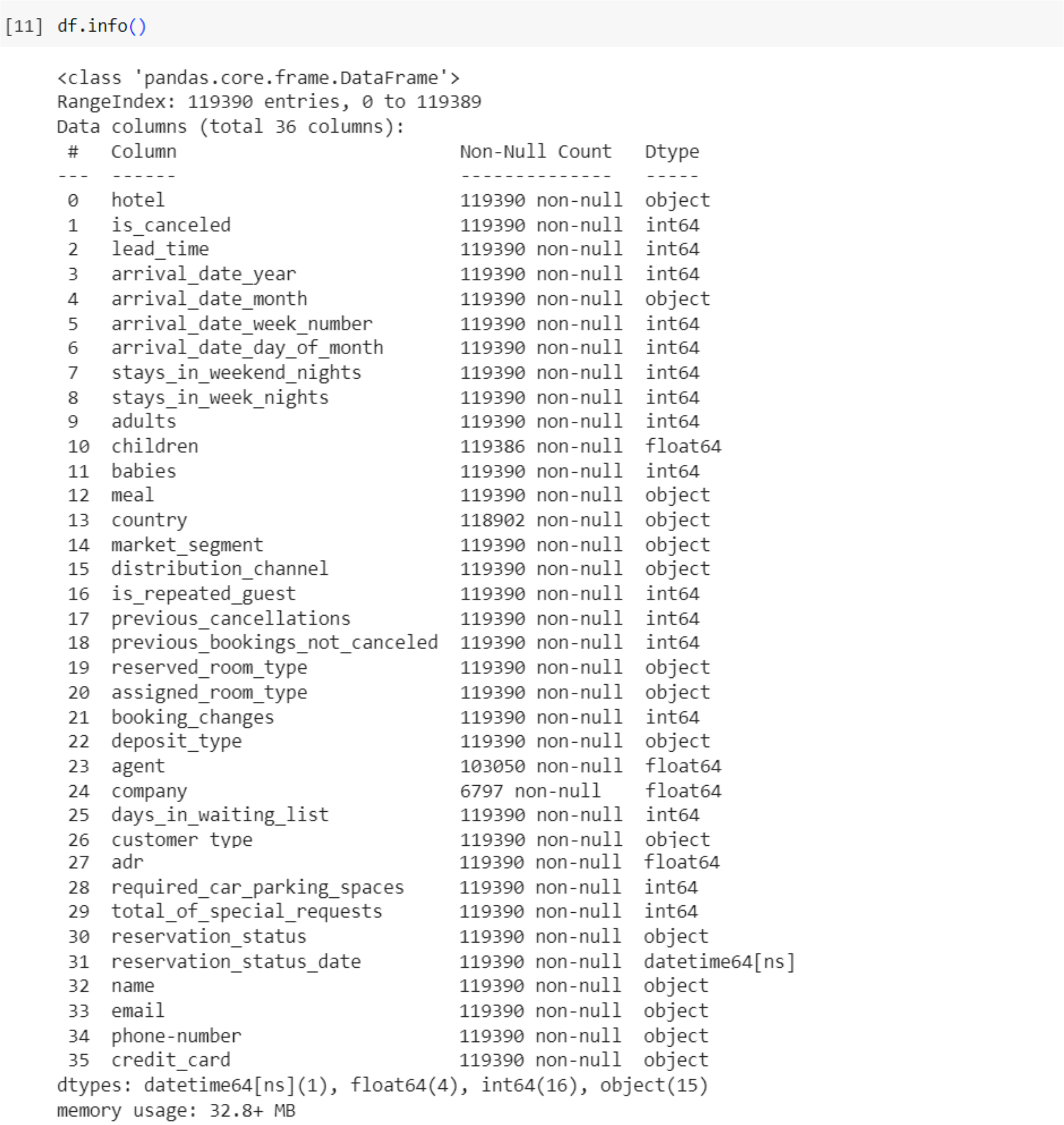


```
[9]  len(df[df.duplicated()])

     0
```

Changing Data Type

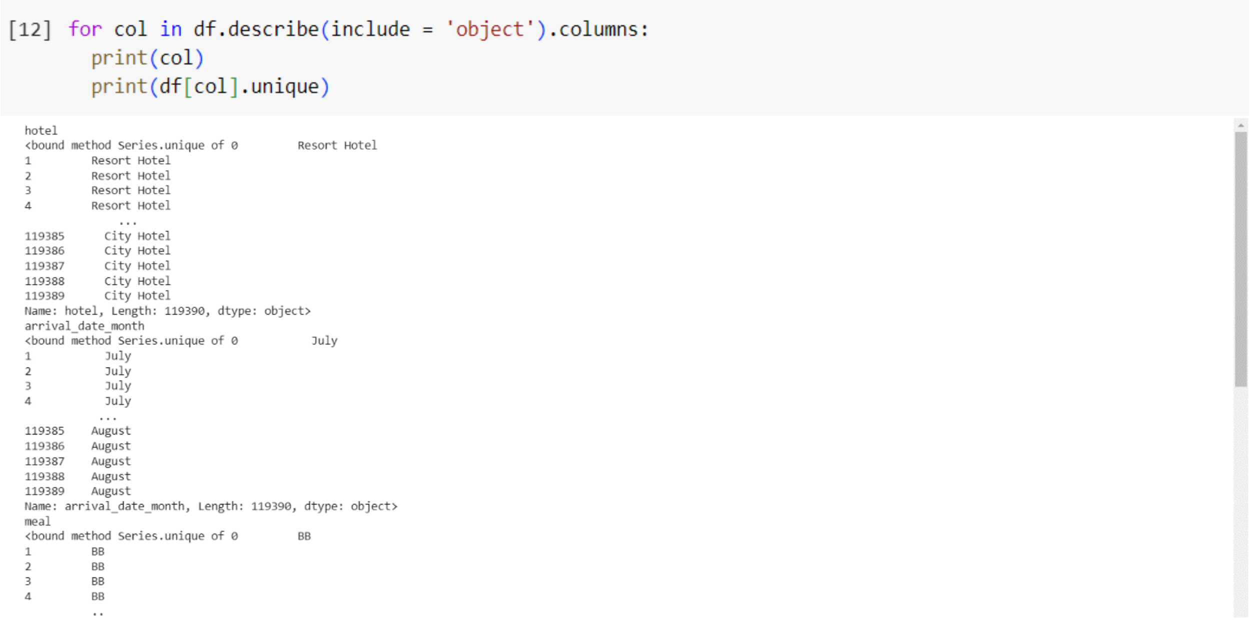


```
[10] df['reservation_status_date'] = pd.to_datetime(df['reservation_status_date'])
```



```
[11] df.info()

     <class 'pandas.core.frame.DataFrame'>
     RangeIndex: 119390 entries, 0 to 119389
     Data columns (total 36 columns):
      #   Column                          Non-Null Count   Dtype
     ---  ------                          --------------   -----
      0   hotel                           119390 non-null  object
      1   is_canceled                     119390 non-null  int64
      2   lead_time                       119390 non-null  int64
      3   arrival_date_year               119390 non-null  int64
      4   arrival_date_month              119390 non-null  object
      5   arrival_date_week_number        119390 non-null  int64
      6   arrival_date_day_of_month       119390 non-null  int64
      7   stays_in_weekend_nights         119390 non-null  int64
      8   stays_in_week_nights            119390 non-null  int64
      9   adults                          119390 non-null  int64
      10  children                        119386 non-null  float64
      11  babies                          119390 non-null  int64
      12  meal                            119390 non-null  object
      13  country                         118902 non-null  object
      14  market_segment                  119390 non-null  object
      15  distribution_channel            119390 non-null  object
      16  is_repeated_guest               119390 non-null  int64
      17  previous_cancellations          119390 non-null  int64
      18  previous_bookings_not_canceled  119390 non-null  int64
      19  reserved_room_type              119390 non-null  object
      20  assigned_room_type              119390 non-null  object
      21  booking_changes                 119390 non-null  int64
      22  deposit_type                    119390 non-null  object
      23  agent                           103050 non-null  float64
      24  company                         6797 non-null    float64
      25  days_in_waiting_list            119390 non-null  int64
      26  customer_type                   119390 non-null  object
      27  adr                             119390 non-null  float64
      28  required_car_parking_spaces     119390 non-null  int64
      29  total_of_special_requests       119390 non-null  int64
      30  reservation_status              119390 non-null  object
      31  reservation_status_date         119390 non-null  datetime64[ns]
      32  name                            119390 non-null  object
      33  email                           119390 non-null  object
      34  phone-number                    119390 non-null  object
      35  credit_card                     119390 non-null  object
     dtypes: datetime64[ns](1), float64(4), int64(16), object(15)
     memory usage: 32.8+ MB
```

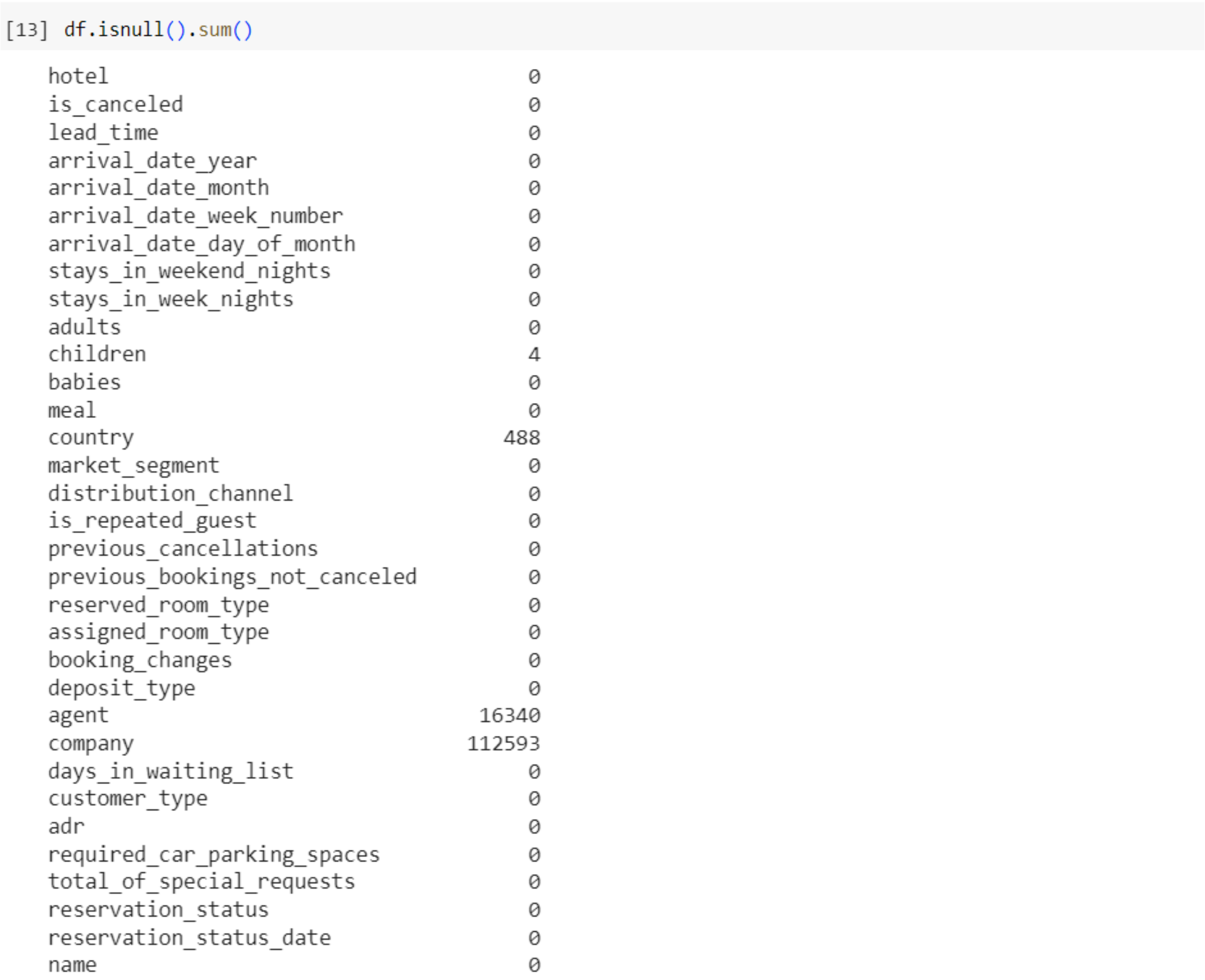
Getting Unique Values



```
[12] for col in df.describe(include = 'object').columns:
       print(col)
       print(df[col].unique)
```

```
hotel
<bound method Series.unique of 0         Resort Hotel
1         Resort Hotel
2         Resort Hotel
3         Resort Hotel
4         Resort Hotel
              ...
119385      City Hotel
119386      City Hotel
119387      City Hotel
119388      City Hotel
119389      City Hotel
Name: hotel, Length: 119390, dtype: object>
arrival_date_month
<bound method Series.unique of 0           July
1           July
2           July
3           July
4           July
           ...
119385    August
119386    August
119387    August
119388    August
119389    August
Name: arrival_date_month, Length: 119390, dtype: object>
meal
<bound method Series.unique of 0         BB
1         BB
2         BB
3         BB
4         BB
          ..
```

Missing Values or Null Values

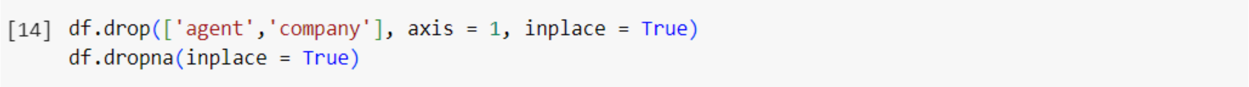


```
[13] df.isnull().sum()
```

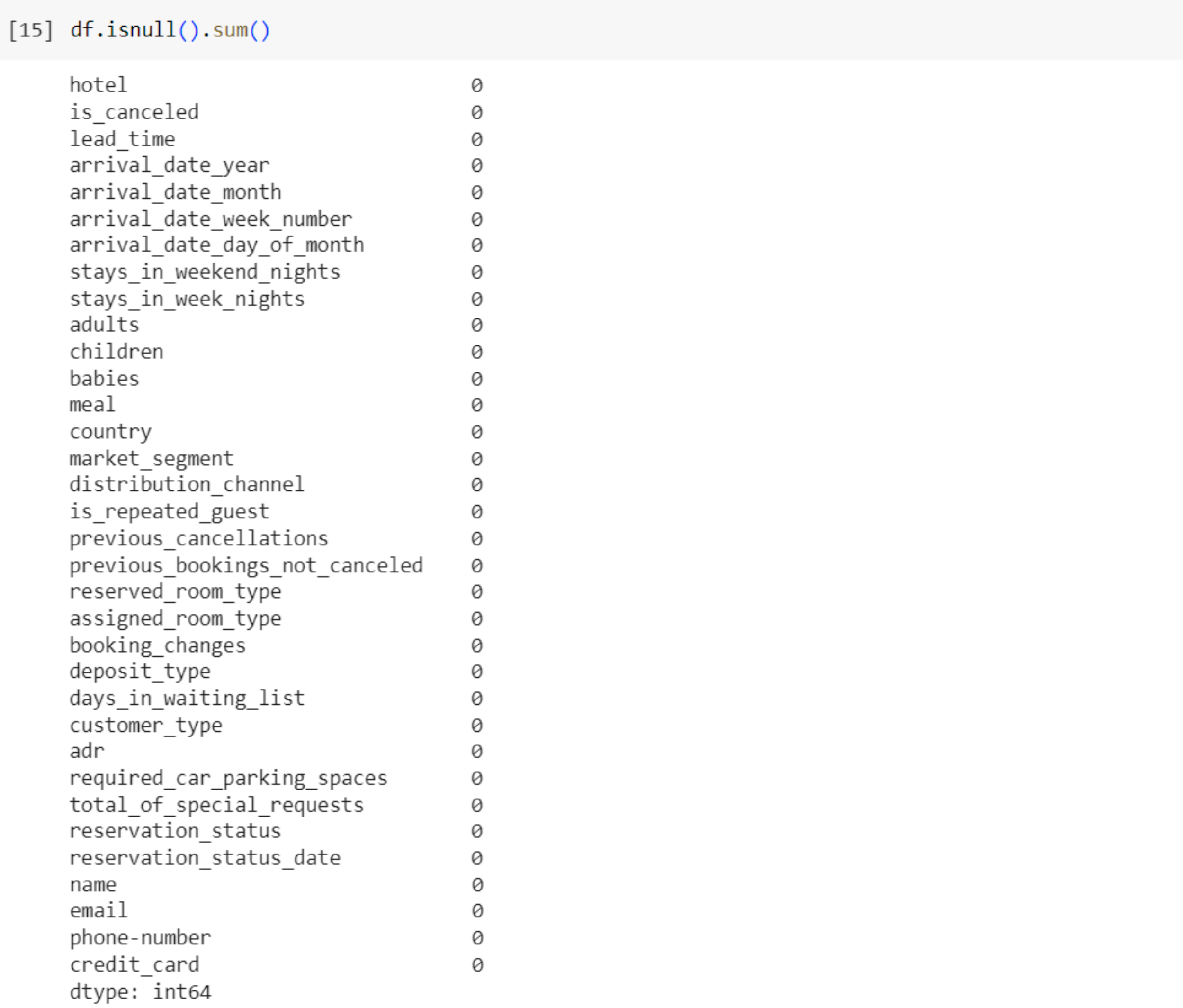
```
hotel                                   0
is_canceled                             0
lead_time                               0
arrival_date_year                       0
arrival_date_month                      0
arrival_date_week_number                0
arrival_date_day_of_month               0
stays_in_weekend_nights                 0
stays_in_week_nights                    0
adults                                  0
children                                4
babies                                  0
meal                                    0
country                               488
market_segment                          0
distribution_channel                    0
is_repeated_guest                       0
previous_cancellations                  0
previous_bookings_not_canceled          0
reserved_room_type                      0
assigned_room_type                      0
booking_changes                         0
deposit_type                            0
agent                               16340
company                            112593
days_in_waiting_list                    0
customer_type                           0
adr                                     0
required_car_parking_spaces             0
total_of_special_requests               0
reservation_status                      0
reservation_status_date                 0
name                                    0
```

```
email                                  0
phone-number                           0
credit_card                            0
dtype: int64
```

Handling Missing Values



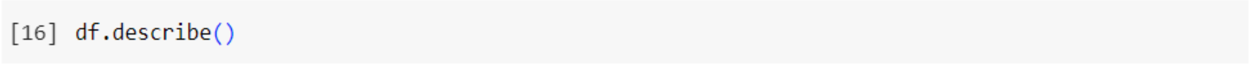
```
[14] df.drop(['agent','company'], axis = 1, inplace = True)
     df.dropna(inplace = True)
```



```
[15] df.isnull().sum()
```

```
hotel                             0
is_canceled                       0
lead_time                         0
arrival_date_year                 0
arrival_date_month                0
arrival_date_week_number          0
arrival_date_day_of_month         0
stays_in_weekend_nights           0
stays_in_week_nights              0
adults                            0
children                          0
babies                            0
meal                              0
country                           0
market_segment                    0
distribution_channel              0
is_repeated_guest                 0
previous_cancellations            0
previous_bookings_not_canceled    0
reserved_room_type                0
assigned_room_type                0
booking_changes                   0
deposit_type                      0
days_in_waiting_list              0
customer_type                     0
adr                               0
required_car_parking_spaces       0
total_of_special_requests         0
reservation_status                0
reservation_status_date           0
name                              0
email                             0
phone-number                      0
credit_card                       0
dtype: int64
```

Identifying and Removing Outliers

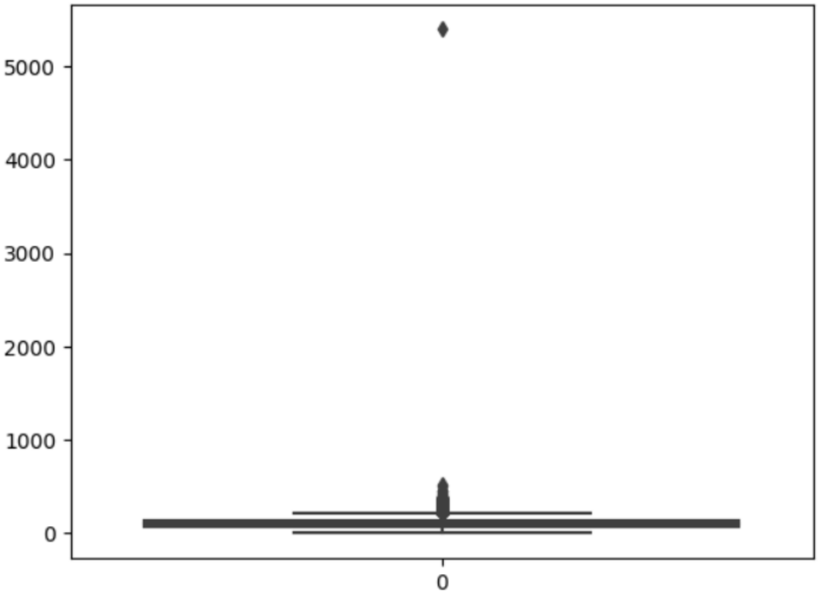


```
[16] df.describe()
```

	is_canceled	lead_time	arrival_date_year	arrival_date_week_number	arrival_date_day_of_month	stays_in_weekend_nights	stays_in_week_nights	adults	children	
count	118898.000000	118898.000000	118898.000000	118898.000000	118898.000000	118898.000000	118898.000000	118898.000000	118898.000000	1
mean	0.371352	104.311435	2016.157656	27.166555	15.800880	0.928897	2.502145	1.858391	0.104207	
std	0.483168	106.903309	0.707459	13.589971	8.780324	0.996216	1.900168	0.578576	0.399172	
min	0.000000	0.000000	2015.000000	1.000000	1.000000	0.000000	0.000000	0.000000	0.000000	
25%	0.000000	18.000000	2016.000000	16.000000	8.000000	0.000000	1.000000	2.000000	0.000000	
50%	0.000000	69.000000	2016.000000	28.000000	16.000000	1.000000	2.000000	2.000000	0.000000	
75%	1.000000	161.000000	2017.000000	38.000000	23.000000	2.000000	3.000000	2.000000	0.000000	
max	1.000000	737.000000	2017.000000	53.000000	31.000000	16.000000	41.000000	55.000000	10.000000	

```
[17] sns.boxplot(df.adr)
```

```
<Axes: >
```



```
[18] Q1 = df['adr'].quantile(0.25)
     Q3 = df['adr'].quantile(0.75)
     IQR = Q3-Q1

     print('Q1:', Q1)
     print('Q3:', Q3)
     print('IQR:', IQR)
```

```
Q1: 70.0
Q3: 126.0
IQR: 56.0
```

```
[19] max_IQR = Q3 + 1.5 * IQR
     min_IQR = Q1 - 1.5 * IQR

     print('min_IQR:', min_IQR)
     print('max_IQR:', max_IQR)
```

```
min_IQR: -14.0
max_IQR: 210.0
```

```
[20] outliers = df.loc[(df['adr'] > max_IQR) | (df['adr'] < min_IQR)]
```

```
[21] new_df = len(df['adr']) - len(outliers)
     print(new_df)
```

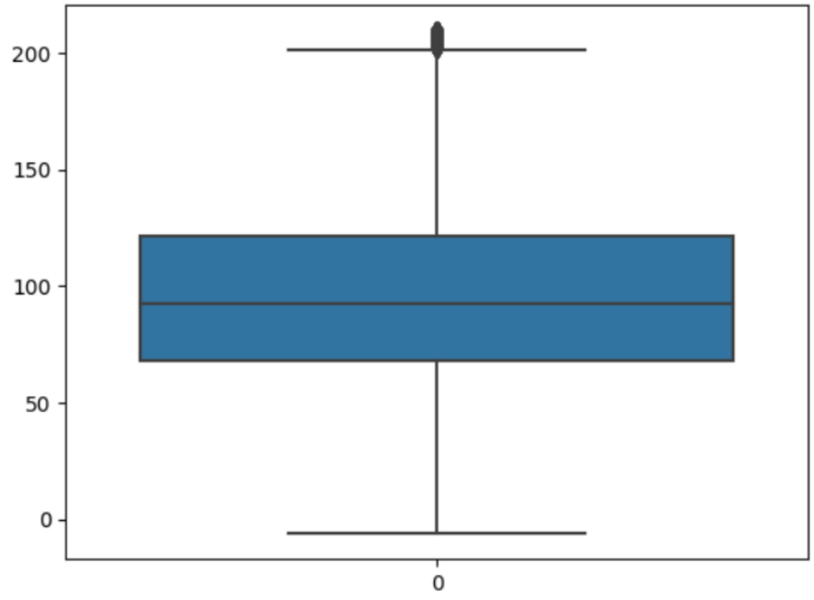
```
115015
```

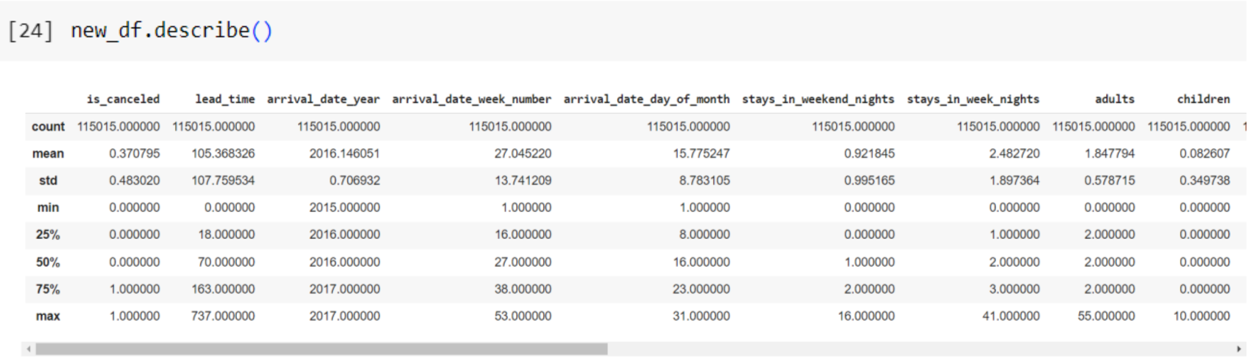
```
[22] new_df = df.loc[(df['adr'] <= max_IQR) & (df['adr'] >= min_IQR)]
     print(new_df)
```

```
               hotel  is_canceled  lead_time  arrival_date_year  \
0       Resort Hotel            0        342               2015   
1       Resort Hotel            0        737               2015   
2       Resort Hotel            0          7               2015   
3       Resort Hotel            0         13               2015   
4       Resort Hotel            0         14               2015   
...              ...          ...        ...                ...   
119384    City Hotel            0         21               2017   
119385    City Hotel            0         23               2017   
119387    City Hotel            0         34               2017   
119388    City Hotel            0        109               2017   
119389    City Hotel            0        205               2017   

       arrival_date_month  arrival_date_week_number  \
0                    July                        27   
1                    July                        27   
2                    July                        27   
3                    July                        27   
4                    July                        27   
...                   ...                       ...   
119384             August                        35   
119385             August                        35   
119387             August                        35   
119388             August                        35   
119389             August                        35   

        arrival_date_day_of_month  stays_in_weekend_nights  \
0                               1                        0   
1                               1                        0   
2                               1                        0   
3                               1                        0   
4                               1                        0   
```

```
[23] sns.boxplot(new_df.adr)
```

```
<Axes: >
```





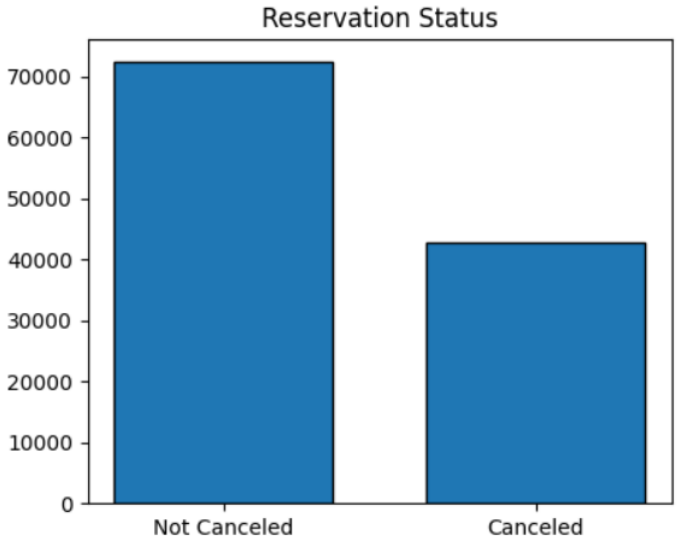
```
[24] new_df.describe()
```

	is_canceled	lead_time	arrival_date_year	arrival_date_week_number	arrival_date_day_of_month	stays_in_weekend_nights	stays_in_week_nights	adults	children
count	115015.000000	115015.000000	115015.000000	115015.000000	115015.000000	115015.000000	115015.000000	115015.000000	115015.000000
mean	0.370795	105.368326	2016.146051	27.045220	15.775247	0.921845	2.482720	1.847794	0.082607
std	0.483020	107.759534	0.706932	13.741209	8.783105	0.995165	1.897364	0.578715	0.349738
min	0.000000	0.000000	2015.000000	1.000000	1.000000	0.000000	0.000000	0.000000	0.000000
25%	0.000000	18.000000	2016.000000	16.000000	8.000000	0.000000	1.000000	2.000000	0.000000
50%	0.000000	70.000000	2016.000000	27.000000	16.000000	1.000000	2.000000	2.000000	0.000000
75%	1.000000	163.000000	2017.000000	38.000000	23.000000	2.000000	3.000000	2.000000	0.000000
max	1.000000	737.000000	2017.000000	53.000000	31.000000	16.000000	41.000000	55.000000	10.000000

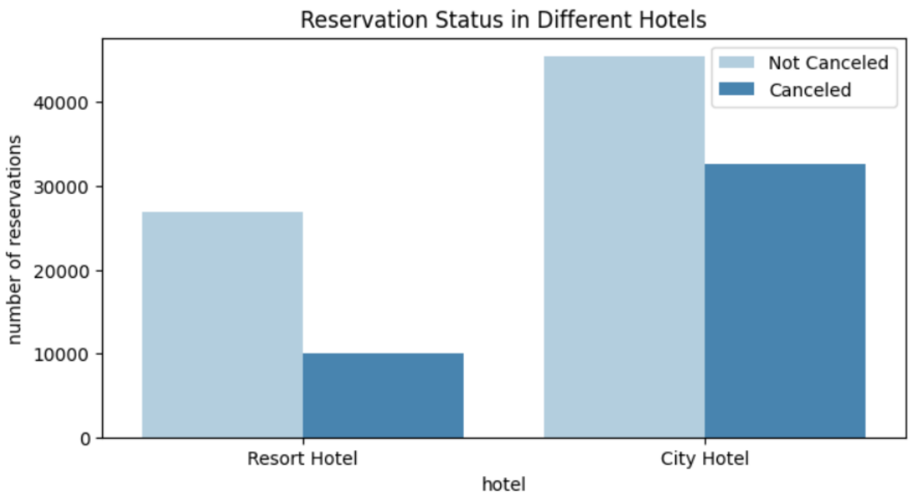
4. Data Analysis and Visualizations

```
[25] cancelled_perc = new_df['is_canceled'].value_counts(normalize = True)
     print(cancelled_perc)

     plt.figure(figsize = (5,4))
     plt.title('Reservation Status')
     plt.bar(['Not Canceled','Canceled'], new_df['is_canceled'].value_counts(), edgecolor = 'k', width = 0.7)
     plt.show()
```

```
0    0.629205
1    0.370795
Name: is_canceled, dtype: float64
```



```
[26] plt.figure(figsize = (8,4))
     ax1=sns.countplot(x = 'hotel', hue = 'is_canceled', data = new_df, palette = 'Blues')
     legend_labels,_ = ax1. get_legend_handles_labels()
     ax1.legend(bbox_to_anchor=(1,1))
     plt.title('Reservation Status in Different Hotels')
     plt.xlabel('hotel')
     plt.ylabel('number of reservations')
     plt.legend(['Not Canceled','Canceled'])
     plt.show()
```



```
[27] resort_hotel = new_df[new_df['hotel'] == 'Resort Hotel']
     resort_hotel['is_canceled'].value_counts(normalize = True)
```

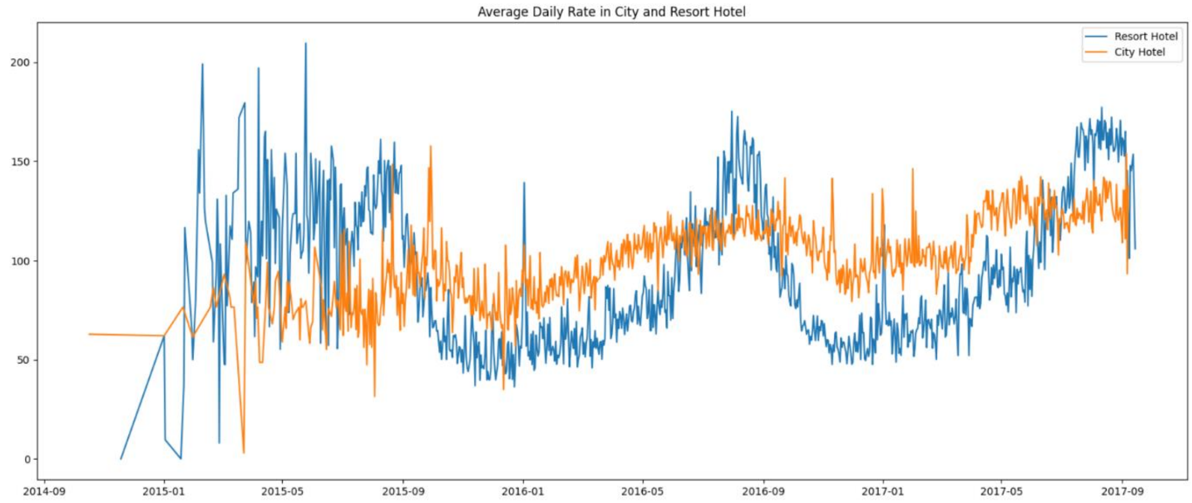
```
0    0.727997
1    0.272003
Name: is_canceled, dtype: float64
```

```
[28] city_hotel = new_df[new_df['hotel'] == 'City Hotel']
     city_hotel['is_canceled'].value_counts(normalize = True)
```

```
0    0.582297
1    0.417703
Name: is_canceled, dtype: float64
```

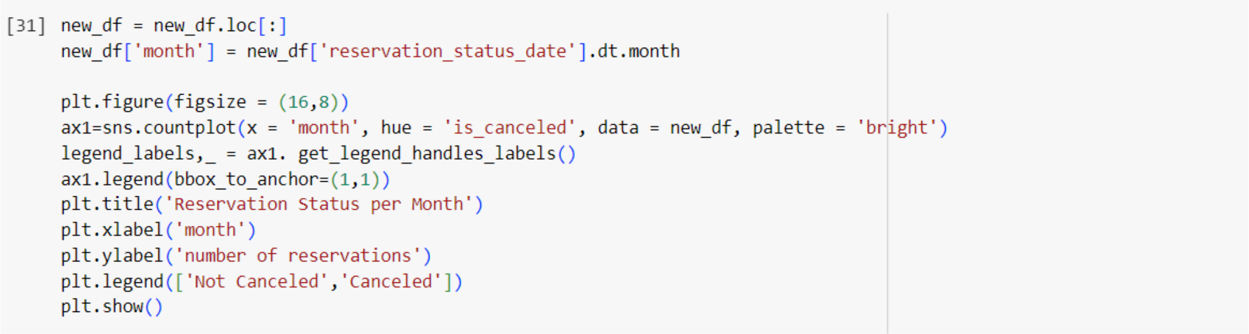
```
[29] resort_hotel = resort_hotel.groupby('reservation_status_date')[['adr']].mean()
     city_hotel = city_hotel.groupby('reservation_status_date')[['adr']].mean()
```

```
[30] plt.figure(figsize = (20,8))
     plt.title('Average Daily Rate in City and Resort Hotel')
     plt.plot(resort_hotel.index, resort_hotel['adr'], label = 'Resort Hotel')
     plt.plot(city_hotel.index, city_hotel['adr'], label = 'City Hotel')
     plt.legend()
     plt.show()
```

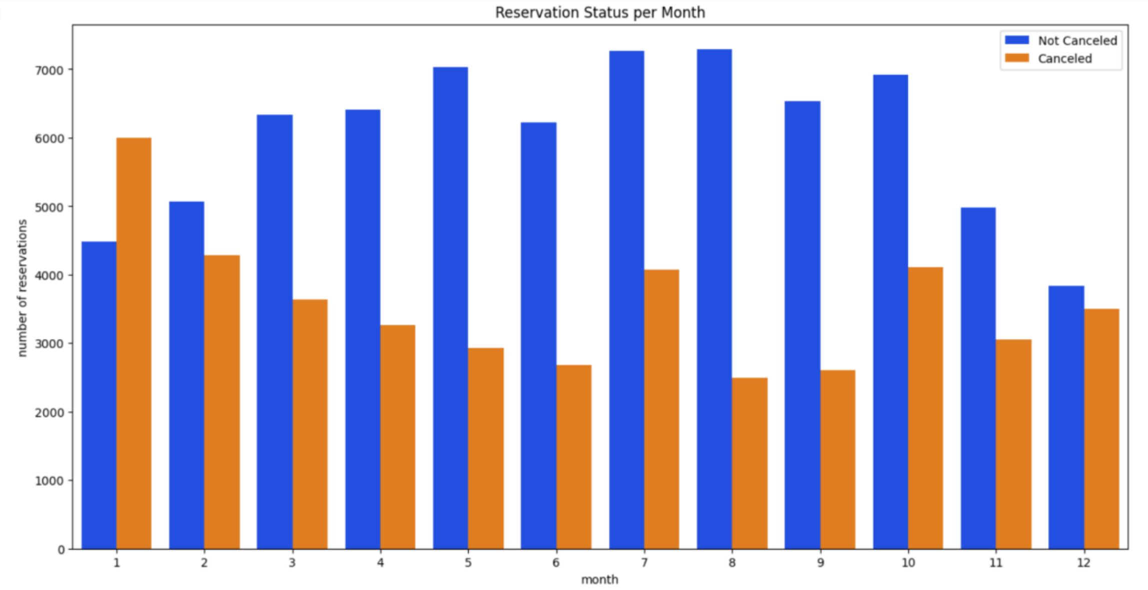


- There are price hikes in dataset. This can be due to price hikes during weekends or on holidays.

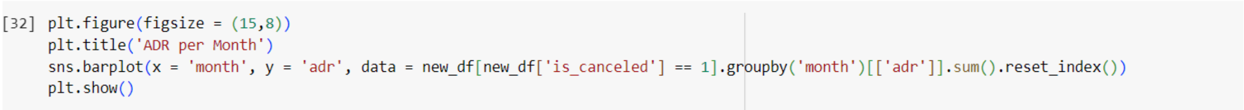
- Prices hikes in resort hotel are much higher than those in city hotel.



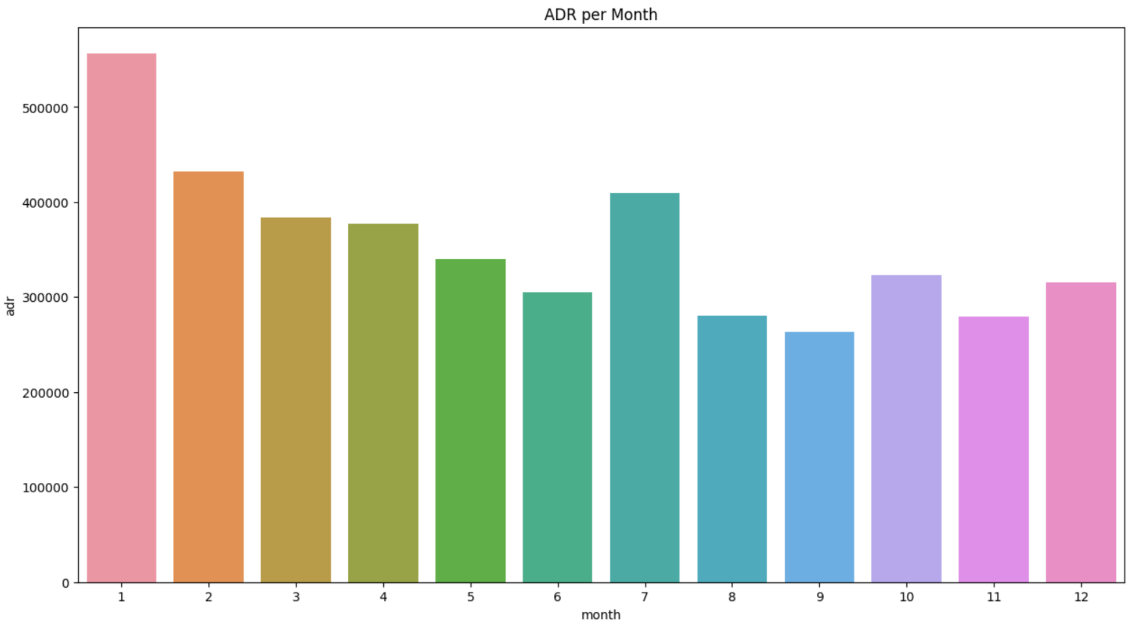
```
[31] new_df = new_df.loc[:]
     new_df['month'] = new_df['reservation_status_date'].dt.month

     plt.figure(figsize = (16,8))
     ax1=sns.countplot(x = 'month', hue = 'is_canceled', data = new_df, palette = 'bright')
     legend_labels,_ = ax1. get_legend_handles_labels()
     ax1.legend(bbox_to_anchor=(1,1))
     plt.title('Reservation Status per Month')
     plt.xlabel('month')
     plt.ylabel('number of reservations')
     plt.legend(['Not Canceled','Canceled'])
     plt.show()
```



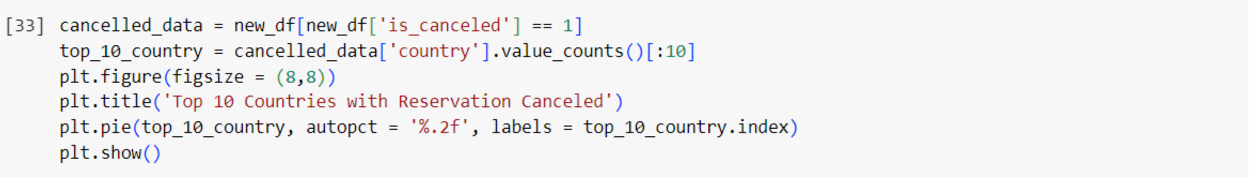
- January has the most cancellations and August has the least cancellations.



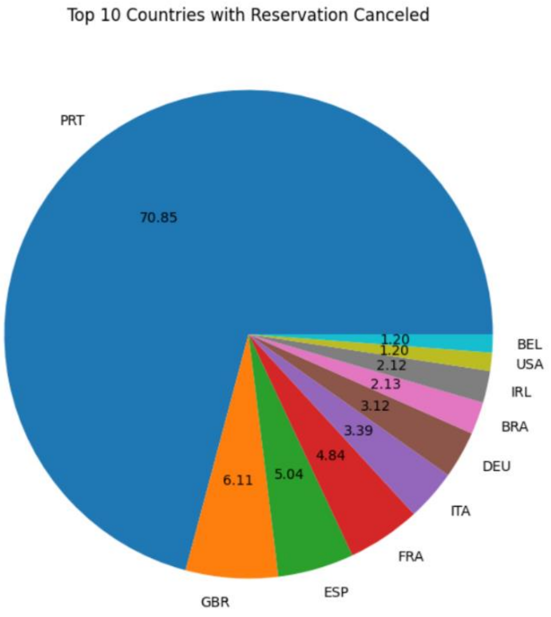
```
[32] plt.figure(figsize = (15,8))
     plt.title('ADR per Month')
     sns.barplot(x = 'month', y = 'adr', data = new_df[new_df['is_canceled'] == 1].groupby('month')[['adr']].sum().reset_index())
     plt.show()
```



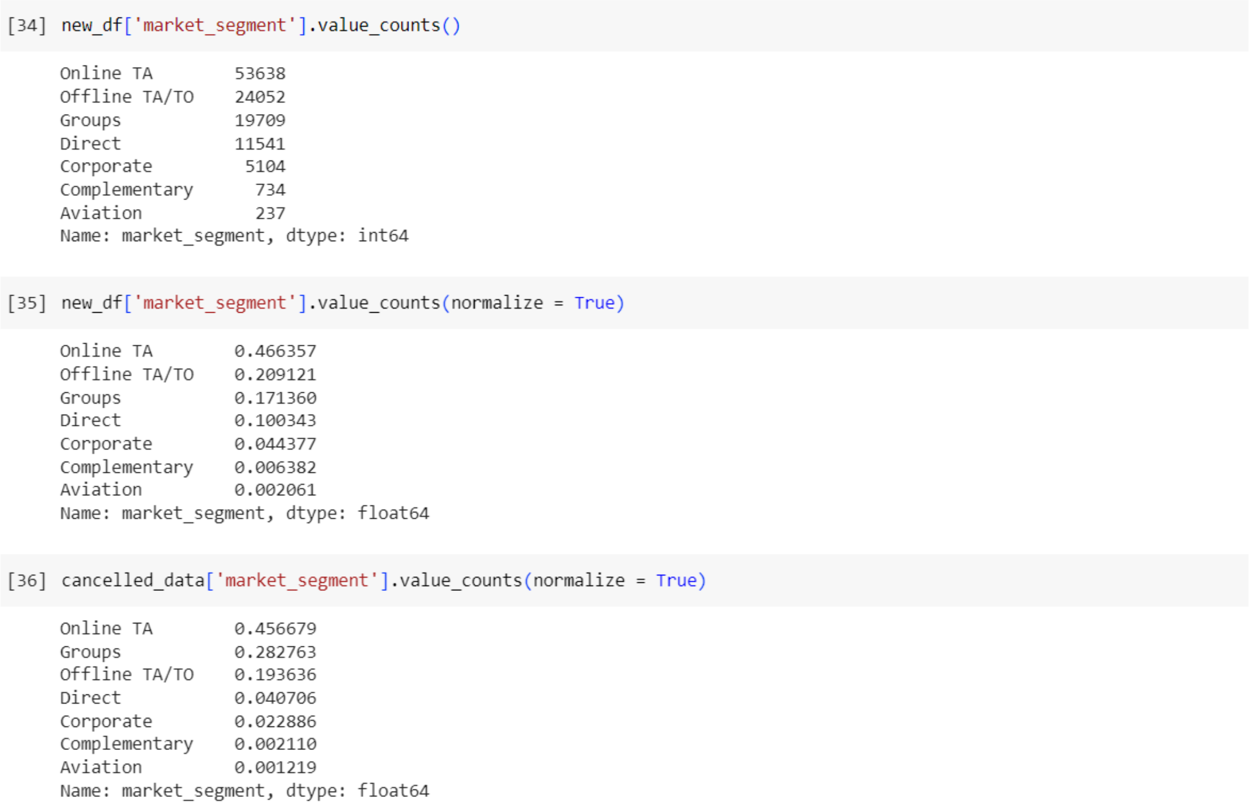
- One of the hypothesis is proved where when price is high, the number of cancellations is more



```
[33] cancelled_data = new_df[new_df['is_canceled'] == 1]
     top_10_country = cancelled_data['country'].value_counts()[:10]
     plt.figure(figsize = (8,8))
     plt.title('Top 10 Countries with Reservation Canceled')
     plt.pie(top_10_country, autopct = '%.2f', labels = top_10_country.index)
     plt.show()
```



- Portugal has the highest cancellations.



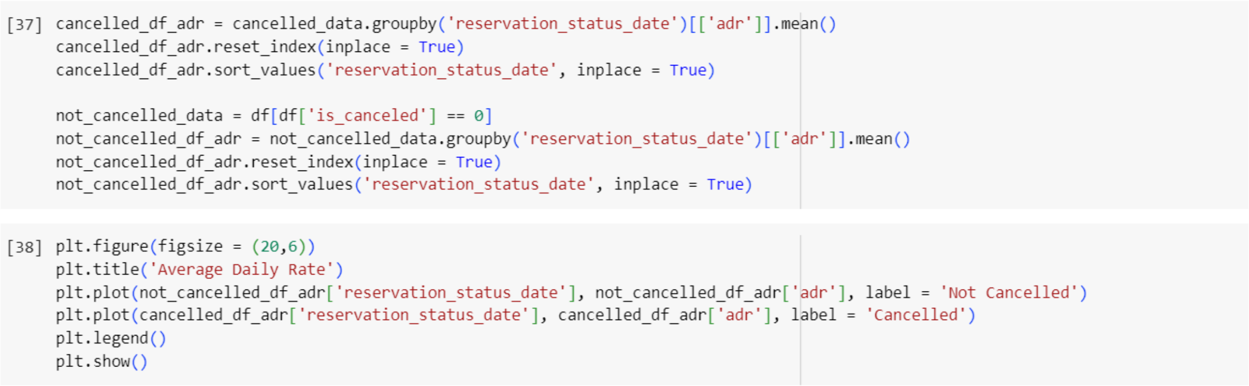
```
[34] new_df['market_segment'].value_counts()

     Online TA        53638
     Offline TA/TO    24052
     Groups           19709
     Direct           11541
     Corporate         5104
     Complementary      734
     Aviation           237
     Name: market_segment, dtype: int64
```

```
[35] new_df['market_segment'].value_counts(normalize = True)

     Online TA        0.466357
     Offline TA/TO    0.209121
     Groups           0.171360
     Direct           0.100343
     Corporate        0.044377
     Complementary    0.006382
     Aviation         0.002061
     Name: market_segment, dtype: float64
```

```
[36] cancelled_data['market_segment'].value_counts(normalize = True)

     Online TA        0.456679
     Groups           0.282763
     Offline TA/TO    0.193636
     Direct           0.040706
     Corporate        0.022886
     Complementary    0.002110
     Aviation         0.001219
     Name: market_segment, dtype: float64
```

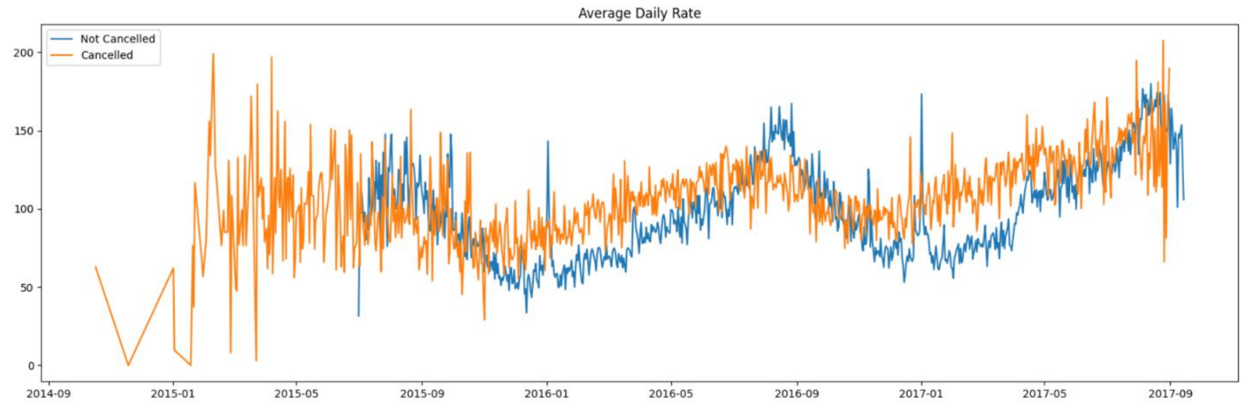
- A whopping 47% of cancellations are from users who booked through online travel agents.

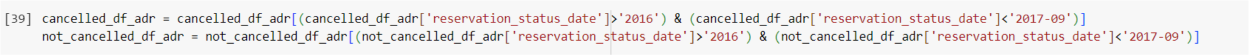
- A possible reason could be that travel agents posted pictures of hotels that don’t the match reality.



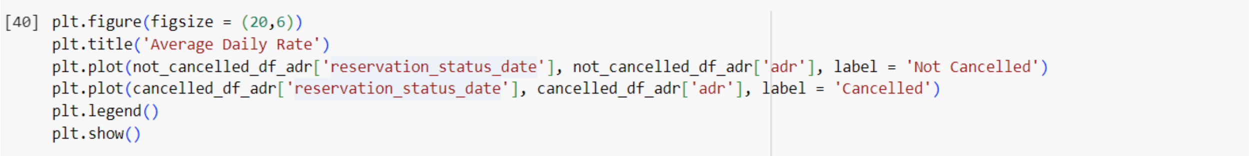
```
[37] cancelled_df_adr = cancelled_data.groupby('reservation_status_date')[['adr']].mean()
     cancelled_df_adr.reset_index(inplace = True)
     cancelled_df_adr.sort_values('reservation_status_date', inplace = True)

     not_cancelled_data = df[df['is_canceled'] == 0]
     not_cancelled_df_adr = not_cancelled_data.groupby('reservation_status_date')[['adr']].mean()
     not_cancelled_df_adr.reset_index(inplace = True)
     not_cancelled_df_adr.sort_values('reservation_status_date', inplace = True)
```

```
[38] plt.figure(figsize = (20,6))
     plt.title('Average Daily Rate')
     plt.plot(not_cancelled_df_adr['reservation_status_date'], not_cancelled_df_adr['adr'], label = 'Not Cancelled')
     plt.plot(cancelled_df_adr['reservation_status_date'], cancelled_df_adr['adr'], label = 'Cancelled')
     plt.legend()
     plt.show()
```

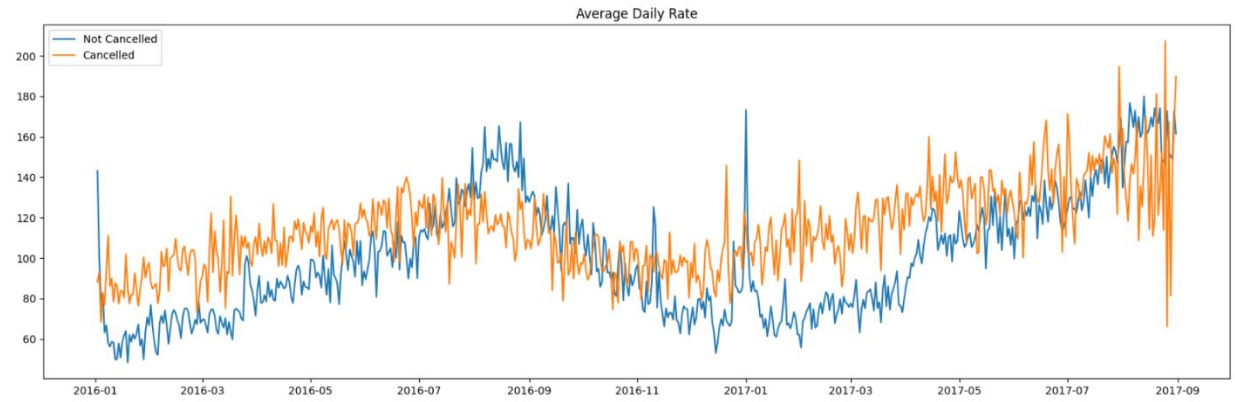




```
[39] cancelled_df_adr = cancelled_df_adr[(cancelled_df_adr['reservation_status_date']>'2016') & (cancelled_df_adr['reservation_status_date']<'2017-09')]
     not_cancelled_df_adr = not_cancelled_df_adr[(not_cancelled_df_adr['reservation_status_date']>'2016') & (not_cancelled_df_adr['reservation_status_date']<'2017-09')]
```



```
[40] plt.figure(figsize = (20,6))
     plt.title('Average Daily Rate')
     plt.plot(not_cancelled_df_adr['reservation_status_date'], not_cancelled_df_adr['adr'], label = 'Not Cancelled')
     plt.plot(cancelled_df_adr['reservation_status_date'], cancelled_df_adr['adr'], label = 'Cancelled')
     plt.legend()
     plt.show()
```



Suggestions

1. Cancellation rates rise as the prices does. In order to prevent cancellations of reservations, hotel could work on their pricing strategies and try to lower the rates for specific hotels based on location. They can also provide some discounts to the customers.

2. As the ratio of the cancelleation and not cancellation of the city hotel is higher than the resort hotel. So the hotels should provide a reasonable discount on the room prices on weekends or on holidays.

3. In the month of January, hotels can start campaigns or marketing with a reasonable amount to increase their revenue as the cancellations is the highest in this month.

4. They can also increase the quality of their hotels and their services mainly in Portugal to reduce the cancellation rates.