HOTEL BOOKING CANCELLATIONS ANALYSIS

JANHI ONG

Tuesday, July 16th, 2024

OVERVIEW OF THE PROJECT

1.Business problem

2.Research question

3.Exploratory data analysis and data cleaning

4.Data analysis and visualizations

5.Analysis and suggestions

1.BUSINESS PROBLEM

- City Hotel and Resort Hotel have seen high cancellation rates
- Issues: reduced revenues and suboptimal hotel room utilization
- Goal: Annalyze hotel booking cancellations and other factors affecting their business and annual revenue generation

2.RESEARCH QUESTIONS

- What variables affect hotel reservation cancellations?
- How can hotel reservation cancellations be reduced?
- How can hotels be assisted in making pricing and promotional decisions?

3.EXPLORATORY DATA ANALYSIS AND DATA CLEANING

#import important libraries and load the dataset

Importing Libraries

In [244…

```
%matplotlib inline
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
from datetime import datetime
warnings.filterwarnings('ignore')
```

Loading the dataset

In [246…

```
hotel_df = pd.read_csv('hotel_booking.csv')
```

3.EXPLORATORY DATA ANALYSIS AND DATA CLEANING

#Understand the structure of the data

In [354…

```
hotel_df.head(10)
```

Out[354…

	hotel	is_canceled	lead_time	arrival_date_year	arrival_date_month	arrival_date_week_number	arrival_date_day_of_month
0	Resort Hotel	0	342	2015	July	27	1
1	Resort Hotel	0	737	2015	July	27	1
2	Resort Hotel	0	7	2015	July	27	1
3	Resort Hotel	0	13	2015	July	27	1
4	Resort Hotel	0	14	2015	July	27	1
5	Resort Hotel	0	14	2015	July	27	1
6	Resort Hotel	0	0	2015	July	27	1
7	Resort Hotel	0	9	2015	July	27	1
8	Resort Hotel	1	85	2015	July	27	1
9	Resort Hotel	1	75	2015	July	27	1

3.EXPLORATORY DATA ANALYSIS AND DATA CLEANING

#Understand the structure of the data

In [346…

```
hotel_df.tail()
```

Out[346…

	hotel	is_canceled	lead_time	arrival_date_year	arrival_date_month	arrival_date_week_number	arrival_date_day_of_mo
119385	City Hotel	0	23	2017	August	35	
119386	City Hotel	0	102	2017	August	35	
119387	City Hotel	0	34	2017	August	35	
119388	City Hotel	0	109	2017	August	35	
119389	City Hotel	0	205	2017	August	35	

5 rows × 31 columns

3.EXPLORATORY DATA ANALYSIS AND DATA CLEANING

#Checking the shape of the dataset

#Understand the variables present in the dataset

In [250…
```
hotel_df.shape
```

Out[250…
```
(119390, 32)
```

In [251…
```
hotel_df.columns
```

Out[251…
```
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'],
      dtype='object')
```

3.EXPLORATORY DATA ANALYSIS AND DATA CLEANING

Displaying detailed information about each column, including data type and non-null counts

In [252…

```
hotel_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 119390 entries, 0 to 119389
Data columns (total 32 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
dtypes: float64(4), int64(16), object(12)
memory usage: 29.1+ MB
```

3.EXPLORATORY DATA ANALYSIS AND DATA CLEANING

Converting the 'reservation_status_date' column to date time format for date-related operations

In [253…
```
hotel_df['reservation_status_date'] = pd.to_datetime(hotel_df['reservation_status_date'], dayfirst=True)
```

In [254…
```
hotel_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 119390 entries, 0 to 119389
Data columns (total 32 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]
dtypes: datetime64[ns](1), float64(4), int64(16), object(11)
memory usage: 29.1+ MB
```

3.EXPLORATORY DATA ANALYSIS AND DATA CLEANING

Displaying updated information to confirm changes in the data types

In [255…

```
hotel_df.describe( include = 'object')
```

Out[255…

	hotel	arrival_date_month	meal	country	market_segment	distribution_channel	reserved_room_type	assigned_roon
count	119390	119390	119390	118902	119390	119390	119390	
unique	2	12	5	177	8	5	10	
top	City Hotel	August	BB	PRT	Online TA	TA/TO	A	
freq	79330	13877	92310	48590	56477	97870	85994	

3.EXPLORATORY DATA ANALYSIS AND DATA CLEANING

Displaying a statistical summary for categorical (object) columns

In [256…

```
for col in hotel_df.describe(include = 'object').columns:
    print(col)
    print(hotel_df[col].unique())
    print('-'*50)
```

```
hotel
['Resort Hotel' 'City Hotel']
--------------------------------------------------
arrival_date_month
['July' 'August' 'September' 'October' 'November' 'December' 'January'
 'February' 'March' 'April' 'May' 'June']
--------------------------------------------------
meal
['BB' 'FB' 'HB' 'SC' 'Undefined']
--------------------------------------------------
country
['PRT' 'GBR' 'USA' 'ESP' 'IRL' 'FRA' nan 'ROU' 'NOR' 'OMN' 'ARG' 'POL'
 'DEU' 'BEL' 'CHE' 'CN' 'GRC' 'ITA' 'NLD' 'DNK' 'RUS' 'SWE' 'AUS' 'EST'
 'CZE' 'BRA' 'FIN' 'MOZ' 'BWA' 'LUX' 'SVN' 'ALB' 'IND' 'CHN' 'MEX' 'MAR'
 'UKR' 'SMR' 'LVA' 'PRI' 'SRB' 'CHL' 'AUT' 'BLR' 'LTU' 'TUR' 'ZAF' 'AGO'
 'ISR' 'CYM' 'ZMB' 'CPV' 'ZWE' 'DZA' 'KOR' 'CRI' 'HUN' 'ARE' 'TUN' 'JAM'
 'HRV' 'HKG' 'IRN' 'GEO' 'AND' 'GIB' 'URY' 'JEY' 'CAF' 'CYP' 'COL' 'GGY'
 'KWT' 'NGA' 'MDV' 'VEN' 'SVK' 'FJI' 'KAZ' 'PAK' 'IDN' 'LBN' 'PHL' 'SEN'
 'SYC' 'AZE' 'BHR' 'NZL' 'THA' 'DOM' 'MKD' 'MYS' 'ARM' 'JPN' 'LKA' 'CUB'
 'CMR' 'BIH' 'MUS' 'COM' 'SUR' 'UGA' 'BGR' 'CIV' 'JOR' 'SYR' 'SGP' 'BDI'
 'SAU' 'VNM' 'PLW' 'QAT' 'EGY' 'PER' 'MLT' 'MWI' 'ECU' 'MDG' 'ISL' 'UZB'
 'NPL' 'BHS' 'MAC' 'TGO' 'TWN' 'DJI' 'STP' 'KNA' 'ETH' 'IRQ' 'HND' 'RWA'
 'KHM' 'MCO' 'BGD' 'IMN' 'TJK' 'NIC' 'BEN' 'VGB' 'TZA' 'GAB' 'GHA' 'TMP'
 'GLP' 'KEN' 'LIE' 'GNB' 'MNE' 'UMI' 'MYT' 'FRO' 'MMR' 'PAN' 'BFA' 'LBY'
 'MLI' 'NAM' 'BOL' 'PRY' 'BRB' 'ABW' 'AIA' 'SLV' 'DMA' 'PYF' 'GUY' 'LCA'
 'ATA' 'GTM' 'ASM' 'MRT' 'NCL' 'KIR' 'SDN' 'ATF' 'SLE' 'LAO']
--------------------------------------------------
market_segment
['Direct' 'Corporate' 'Online TA' 'Offline TA/TO' 'Complementary' 'Groups'
 'Undefined' 'Aviation']
--------------------------------------------------
distribution_channel
['Direct' 'Corporate' 'TA/TO' 'Undefined' 'GDS']
--------------------------------------------------
reserved_room_type
['C' 'A' 'D' 'E' 'G' 'F' 'H' 'L' 'P' 'B']
--------------------------------------------------
assigned_room_type
['C' 'A' 'D' 'E' 'G' 'F' 'I' 'B' 'H' 'P' 'L' 'K']
```

3.EXPLORATORY DATA ANALYSIS AND DATA CLEANING

Checking for missing values in each column of the dataset

In [257…
```
hotel_df.isnull().sum()
```

Out[257…
```
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
dtype: int64
```

In [258…
```
hotel_df.drop(['company','agent'], axis = 1, inplace = True)
```

In [259…
```
hotel_df.dropna(inplace = True)
```

In [260…
```
hotel_df.describe()
```

3.EXPLORATORY DATA ANALYSIS AND DATA CLEANING

Dropping the 'company' and 'agent' columns from the dataset as they have too many missing values
Dropping any remaining rows with missing values to clean the dataset
Displaying a statistical summary of the numerical columns after cleaning the dataset

In [258…
```
hotel_df.drop(['company','agent'], axis = 1, inplace = True)
```

In [259…
```
hotel_df.dropna(inplace = True)
```

In [260…
```
hotel_df.describe()
```

Out[260…

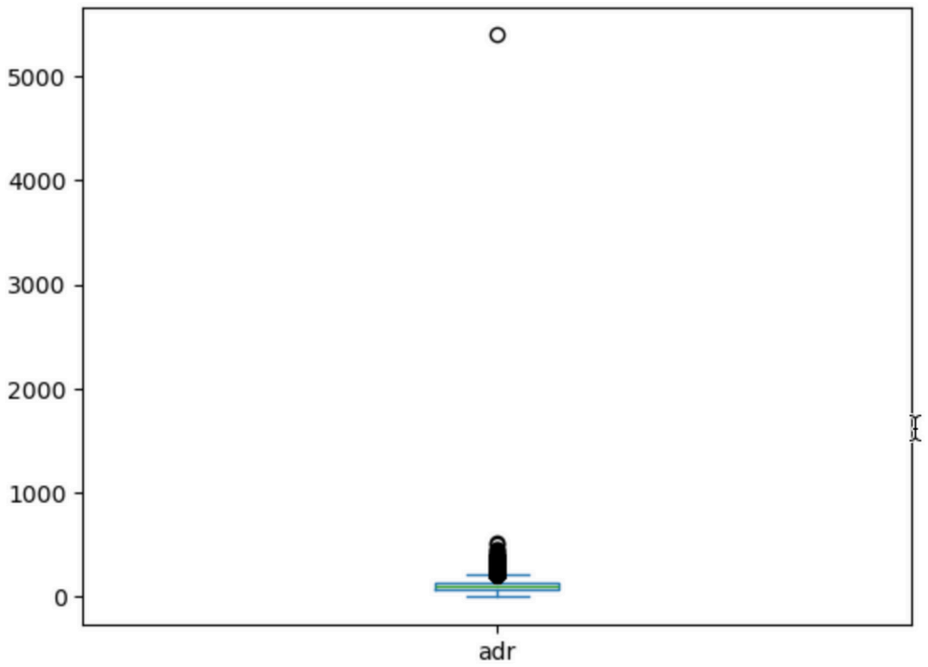
g_changes	days_in_waiting_list	adr	required_car_parking_spaces	total_of_special_requests	reservation_status_date
398.000000	118898.000000	118898.000000	118898.000000	118898.000000	118898
0.221181	2.330754	102.003243	0.061885	0.571683	2016-07-30 07:37:53.336809984
0.000000	0.000000	-6.380000	0.000000	0.000000	2014-10-17 00:00:00
0.000000	0.000000	70.000000	0.000000	0.000000	2016-02-02 00:00:00
0.000000	0.000000	95.000000	0.000000	0.000000	2016-08-08 00:00:00
0.000000	0.000000	126.000000	0.000000	1.000000	2017-02-09 00:00:00
21.000000	391.000000	5400.000000	8.000000	5.000000	2017-09-14 00:00:00
0.652785	17.630452	50.485862	0.244172	0.792678	NaN

3.EXPLORATORY DATA ANALYSIS AND DATA CLEANING

Creating a box plot for the 'adr' column to identify outliers

In [261…
```
hotel_df['adr'].plot(kind = 'box')
```

Out[261… <Axes: >



In [262…
```
hotel_df = hotel_df[hotel_df['adr']<5000]
```

3.EXPLORATORY DATA ANALYSIS AND DATA CLEANING

#Check list for exploratory data analysis and data cleaning

1. Remove duplicated data
2. Handling missing value
3. Correct datatypes
4. Standardize data
5. Remove or Correct Outliners

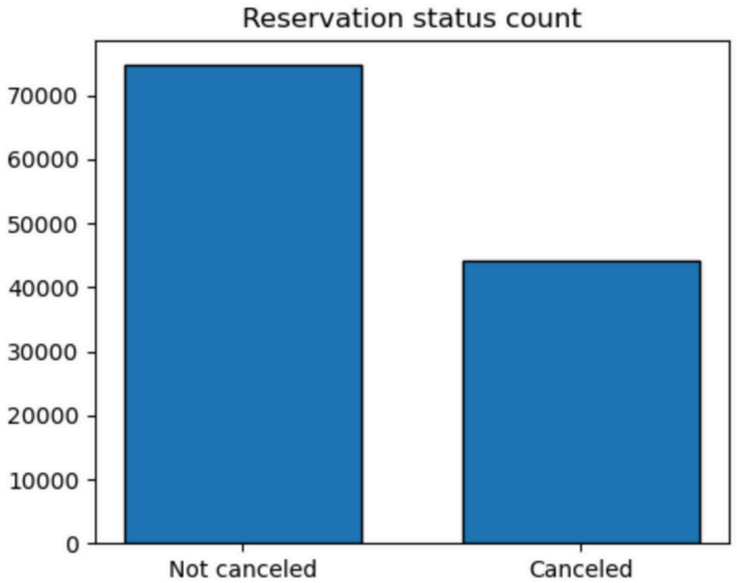
4.DATA ANALYSIS AND VISUALIZATIONS

In [264…

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

plt.figure(figsize = (5,4))
plt.title('Reservation status count')
plt.bar(['Not canceled', 'Canceled'],hotel_df['is_canceled'].value_counts(), edgecolor = 'k', width = 0.7)
```

```
is_canceled
0    0.628653
1    0.371347
Name: proportion, dtype: float64
```

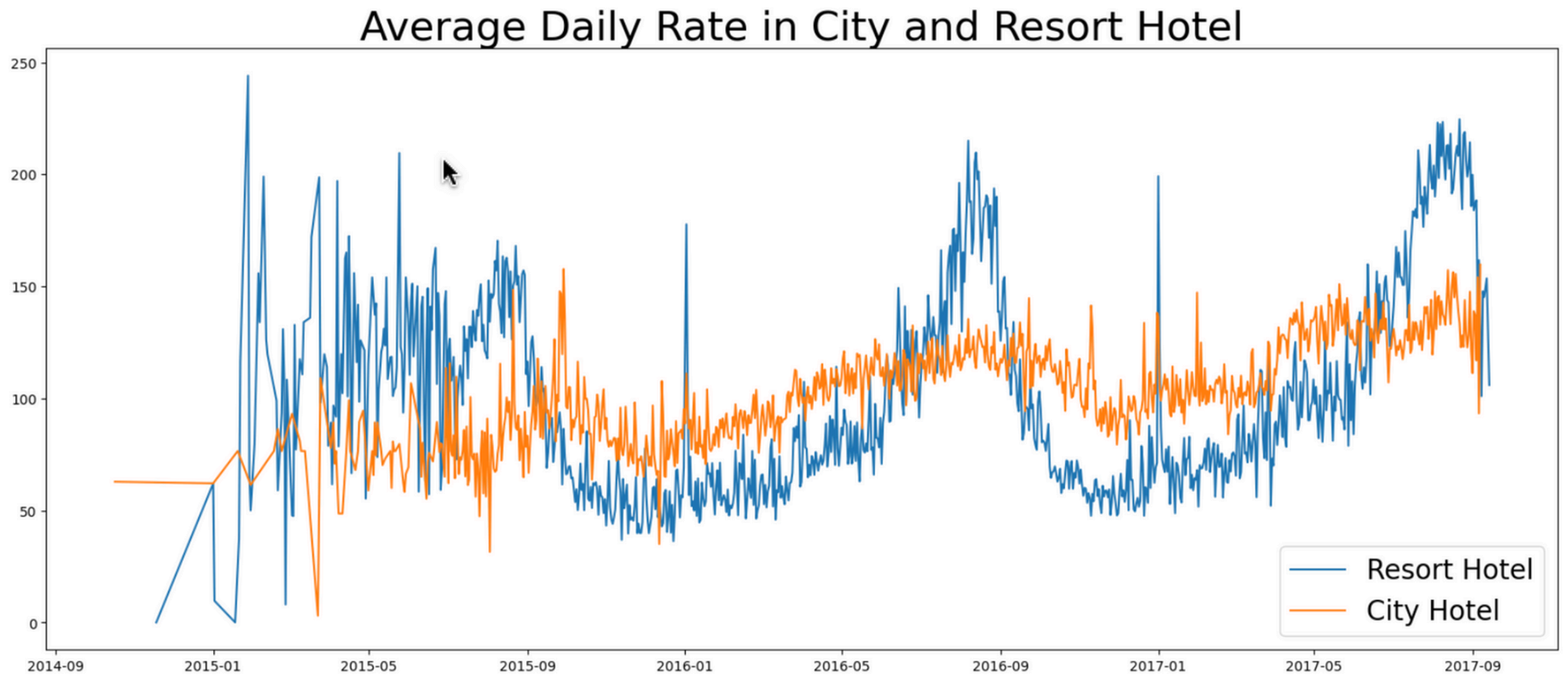
Out[264… <BarContainer object of 2 artists>



4.DATA ANALYSIS AND VISUALIZATIONS

In [270…

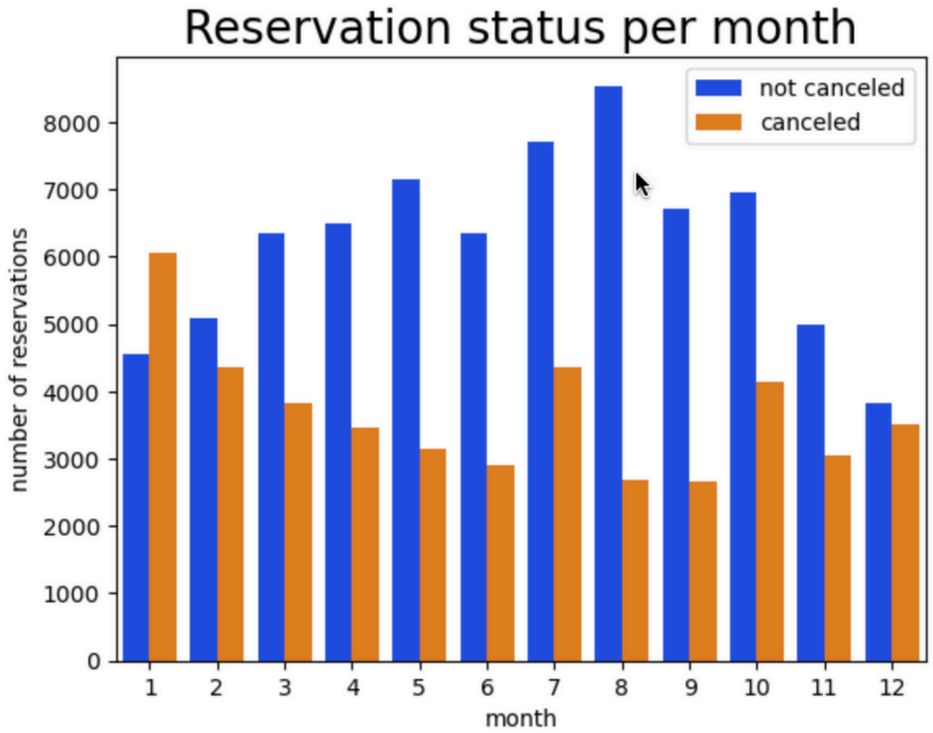
```
plt.figure(figsize = (20,8))
plt.title('Average Daily Rate in City and Resort Hotel', fontsize = 30)
plt.plot(resort_hotel.index, resort_hotel['adr'], label = 'Resort Hotel')
plt.plot(city_hotel.index, city_hotel['adr'], label = 'City Hotel')
plt.legend(fontsize = 20)
plt.show()
```



4.DATA ANALYSIS AND VISUALIZATIONS

In [271…

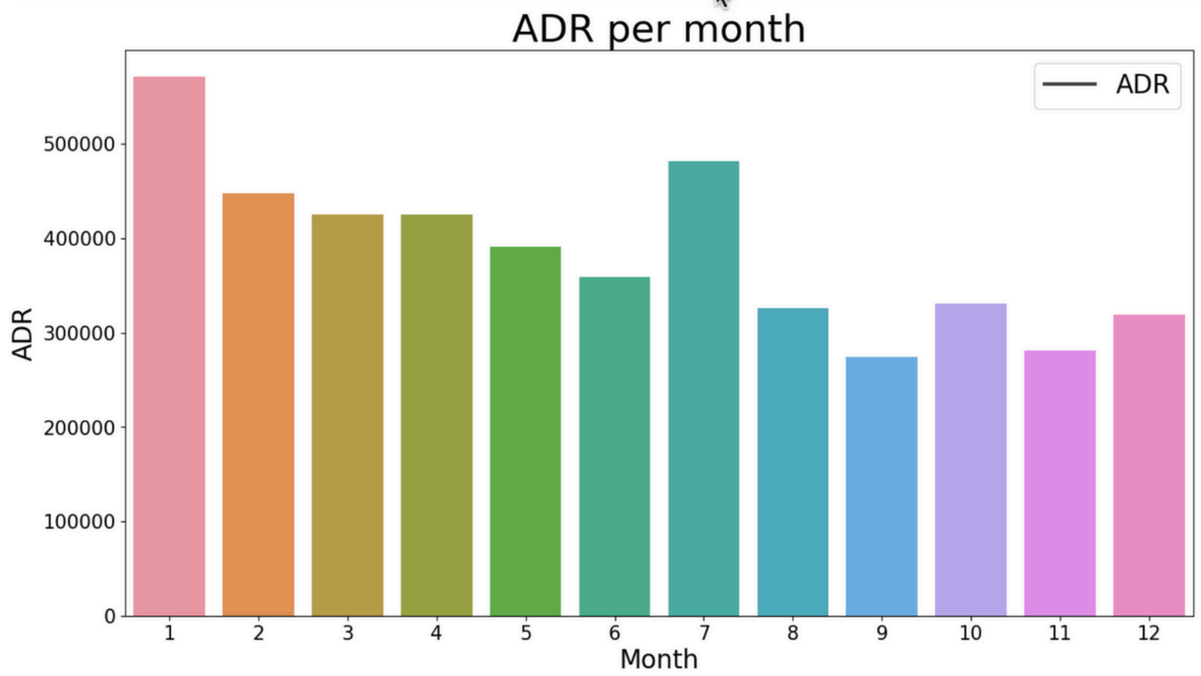
```
hotel_df['month'] = hotel_df['reservation_status_date'].dt.month
plt.figure(figure = (160,20))
ax1 = sns.countplot(x = 'month', hue = 'is_canceled', data = hotel_df, palette = 'bright')
legend_labels,_= ax1. get_legend_handles_labels()
plt.title('Reservation status per month', size = 20)
plt.xlabel('month')
plt.ylabel('number of reservations')
plt.legend(['not canceled', 'canceled'])
plt.show()
```



4.DATA ANALYSIS AND VISUALIZATIONS

In [272…

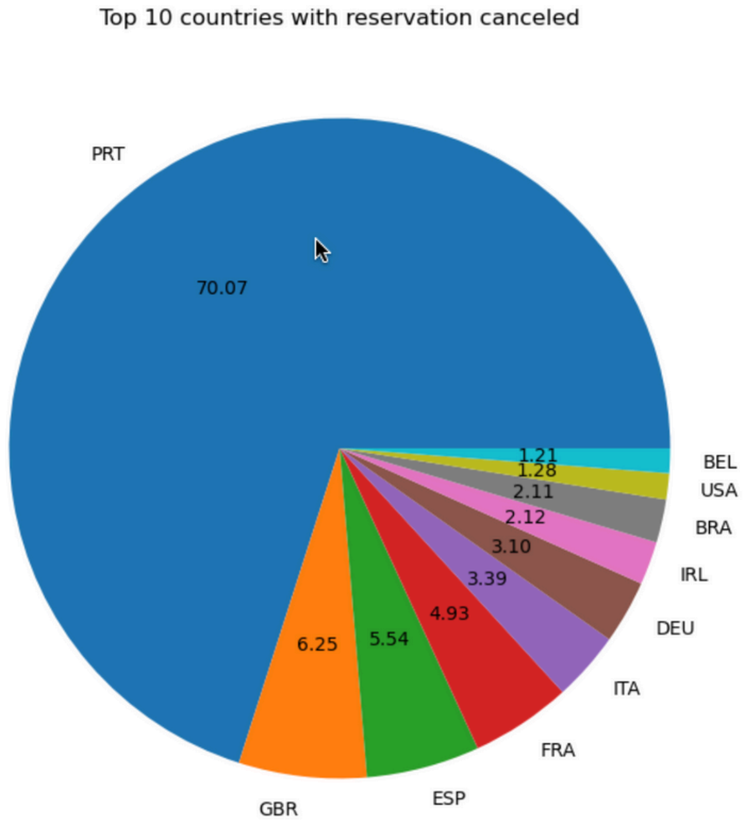
```
plt.figure(figsize = (15,8))
plt.title('ADR per month', fontsize = 30)
sns.barplot( x= 'month', y='adr', data = hotel_df[hotel_df['is_canceled'] == '1'].groupby('month')[['adr']].sum()
grouped_data = hotel_df[hotel_df['is_canceled'] == '1'].groupby('month')[['adr']].sum().reset_index()

# Check if grouped_data is empty before plotting
if not grouped_data.empty:
    sns.barplot(x='month', y='adr', data=grouped_data)
    plt.xlabel('Month', fontsize=20)
    plt.ylabel('ADR', fontsize=20)
    plt.xticks(fontsize=15)
    plt.yticks(fontsize=15)
    plt.legend(['ADR'], fontsize=20)
    plt.show()
else:
    print("No data found for canceled reservations.")
```



4.DATA ANALYSIS AND VISUALIZATIONS

In [374…

```
cancelled_data = hotel_df[hotel_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()
```



4.DATA ANALYSIS AND VISUALIZATIONS

In [422…

```
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_df_adr = not_cancelled_df.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)
```

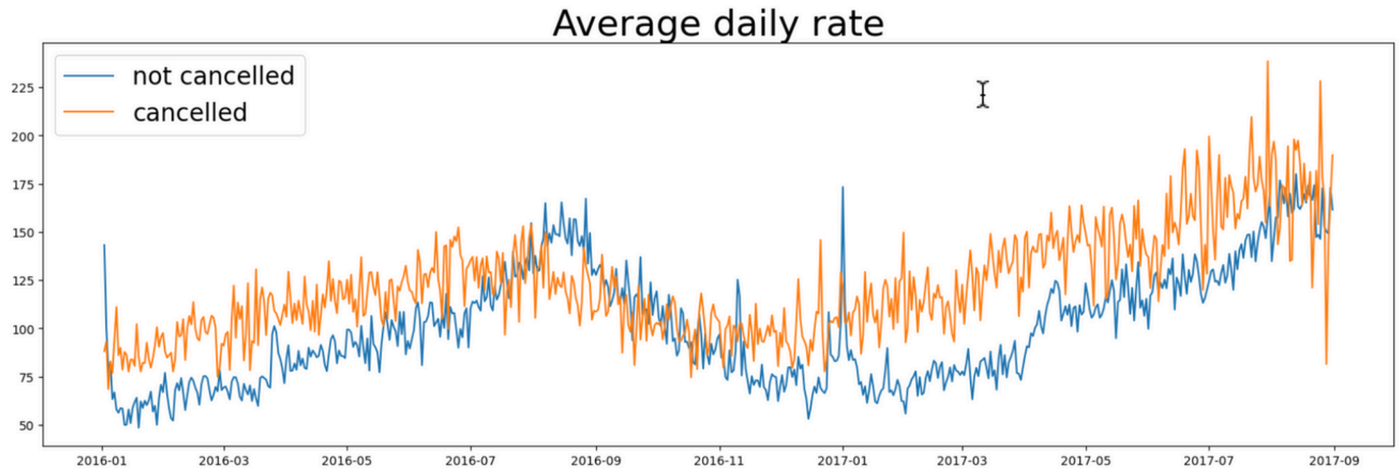
In [424…

```
cancelled_df_adr = cancelled_df_adr[(cancelled_df_adr['reservation_status_date']>'2016') & (cancelled_df_adr['res
not_cancelled_df_adr = not_cancelled_df_adr[(not_cancelled_df_adr['reservation_status_date']>'2016') & (not_cance
```

In [426…

```
plt.figure(figsize = (20,6))
plt.title('Average daily rate', fontsize = 30)
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(fontsize = 20)
```

Out[426… `<matplotlib.legend.Legend at 0x172093d50>`



5.ANALYSIS AND SUGGESTIONS

- **What variables affect hotel reservation cancellations?**
  - Seasonality: High cancellations in August (peak season) and January (post-holiday).
  - Hotel Type: Higher cancellations in resort hotels, likely due to higher rates and leisure travel.
  - ADR Variations: Higher rates during weekends/holidays can deter bookings or lead to cancellations.
  - Booking Channels: Online travel agencies generally have higher cancellation rates.

5.ANALYSIS AND SUGGESTIONS

- **How can hotel reservation cancellations be reduced?**
  - Flexible Policies with Penalties: Encourage early confirmations with structured cancellation fees.
  - Discounts for Non-Cancellable Rates: Attract committed customers by offering lower rates for non-refundable bookings.
  - Predictive Analysis: Use data to identify at-risk reservations and engage guests proactively.
  - Incentivize Direct Bookings: Offer perks for direct bookings to reduce reliance on travel agencies.

5.ANALYSIS AND SUGGESTIONS

- **How can hotels be assisted in making pricing and promotional decisions?**
  - Dynamic Pricing: Adjust rates based on demand (e.g., higher in August, lower in January).
  - Rate Optimization for Peak Times: Use package deals and targeted promotions for weekends and holidays.
  - Segment-Specific Offers: Tailor promotions for different customer groups to boost bookings.
  - Monitor ADR Trends: Regularly review pricing strategies to avoid overpricing.

THANK YOU FOR LISTENING