Project: **Hotel Booking Analysis**

Perspective: Market Insights and Revenue Growth

Auther Name : **Muhammad Irfan**

Email : mi641033@gmail.com
Github : https://github.com/irfan5006\ LinkedIn :https://www.linkedin.com/in/muhammad-irfan-a15359247

> **`About Dataset`**

Context: This dataset contains 119390 observations for a City Hotel and a Resort Hotel. Each observation represents a hotel booking between the 1st of July 2015 and 31st of August 2017, including booking that effectively arrived and booking that were canceled.

Content: Since this is hotel real data, all data elements pertaining hotel or costumer identification were deleted. Four Columns, 'name', 'email', 'phone number' and 'credit_card' have been artificially created and added to the dataset.

Acknowledgements: The data is originally from the article Hotel Booking Demand Datasets, written by Nuno Antonio, Ana Almeida, and Luis Nunes for Data in Brief, Volume 22, February 2019.

> **Hotel Booking Analysis**

- High cancellation rates impact revenue.
- Efficiency enhancement the goal.
- Analysis covers booking cancellations.
- Focus on revenue-related factors.
- Provide actionable business insights.

Assumptions:

- ⚙ Unforeseen events or circumstances between 2015 and 2017 will not significantly impact the data.
- The information remains up-to-date and can be effectively used to analyze a hotel's potential strategies.
- ✖ There are no unexpected drawbacks to the hotel implementing any recommended methods.
- The suggested solutions are not currently in use by the hotels.
- The primary factor influencing income generation is the occurrence of booking cancellations.
- Cancellations lead to unoccupied rooms for the duration of the original booking.
- Clients make hotel reservations in the same year as their cancellations.

Research Questions:

1- What variables influence hotel reservation cancellations?
2- How can we optimize hotel reservation cancellations?
3- How can hotels receive assistance in pricing and promotional decisions?

Hypotheses:

1- An increase in prices leads to a higher rate of cancellations.
2- Longer waiting lists correlate with a higher cancellation rate among customers.
3- The majority of clients make reservations through offline travel agents.

Import Libraries

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

> **`Import The Hotel Booking Analysis Dataset`**

In [ ]:
```
df = pd.read_csv('E:\Data_Sacience\Projects_EDA\hotel_booking.csv')
```

Exploratory Data Analysis (EDA)

In [ ]:
```
df.head(4).T
```

Out[ ]:

	0	1	2	
hotel	Resort Hotel	Resort Hotel	Resort Hotel	Resort
is_canceled	0	0	0	
lead_time	342	737	7	
arrival_date_year	2015	2015	2015	
arrival_date_month	July	July	July	
arrival_date_week_number	27	27	27	
arrival_date_day_of_month	1	1	1	
stays_in_weekend_nights	0	0	0	
stays_in_week_nights	0	0	1	
adults	2	2	1	
children	0.0	0.0	0.0	
babies	0	0	0	
meal	BB	BB	BB	
country	PRT	PRT	GBR	
market_segment	Direct	Direct	Direct	Corp
distribution_channel	Direct	Direct	Direct	Corp
is_repeated_guest	0	0	0	
previous_cancellations	0	0	0	
previous_bookings_not_canceled	0	0	0	
reserved_room_type	C	C	A	
assigned_room_type	C	C	C	
booking_changes	3	4	0	
deposit_type	No Deposit	No Deposit	No Deposit	No De
agent	NaN	NaN	NaN	
company	NaN	NaN	NaN	
days_in_waiting_list	0	0	0	
customer_type	Transient	Transient	Transient	Tran
adr	0.0	0.0	75.0	
required_car_parking_spaces	0	0	0	
total_of_special_requests	0	0	0	
reservation_status	Check-Out	Check-Out	Check-Out	Chec
reservation_status_date	2015-07-01	2015-07-01	2015-07-02	2015-
name	Ernest Barnes	Andrea Baker	Rebecca Parker	Laura M
email	Ernest.Barnes31@outlook.com	Andrea_Baker94@aol.com	Rebecca_Parker@comcast.net	Laura_M@gmai
phone-number	669-792-1661	858-637-6955	652-885-2745	364-656
credit_card	**********4322	**********9157	**********3734	**********

In [ ]:
```
df.shape
```

Out[ ]:
```
(119390, 36)
```

In [ ]:
```
pd.set_option('display.max_rows', None)
```

In [ ]:
```
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
```

> **"Four Columns, 'name', 'email', 'phone number' and 'credit_card' have been artificially created and added to the dataset."**

In [ ]:
```
# Now Remove this Columns
df.drop(['name', 'email','phone-number','credit_card' ], axis = 1 , inplace = True )
```

In [ ]:
```
df.columns
```

Out[ ]:
```
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')
```

> **Handle_Missing Values**

In [ ]:
```
df.isnull().sum()
```

Out[ ]:
```
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 [ ]:
```
# Check the Percentage of Missing Values
df.isnull().sum() / len(df) * 100
```

Out[ ]:
```
hotel                            0.000000
is_canceled                      0.000000
lead_time                        0.000000
arrival_date_year                0.000000
arrival_date_month               0.000000
arrival_date_week_number         0.000000
arrival_date_day_of_month        0.000000
stays_in_weekend_nights          0.000000
stays_in_week_nights             0.000000
adults                           0.000000
children                         0.003350
babies                           0.000000
meal                             0.000000
country                          0.408744
market_segment                   0.000000
distribution_channel             0.000000
is_repeated_guest                0.000000
previous_cancellations           0.000000
previous_bookings_not_canceled   0.000000
reserved_room_type               0.000000
assigned_room_type               0.000000
booking_changes                  0.000000
deposit_type                     0.000000
agent                           13.686238
company                         94.306893
days_in_waiting_list             0.000000
customer_type                    0.000000
adr                              0.000000
required_car_parking_spaces      0.000000
total_of_special_requests        0.000000
reservation_status               0.000000
reservation_status_date          0.000000
dtype: float64
```

In [ ]:
```
# According to percentage iF above 70 % we remove the Column
df.drop(['company'], axis = 1,  inplace = True)
```

In [ ]:
```
agent_median = df['agent'].median()
agent_median
```

Out[ ]:
```
14.0
```

In [ ]:
```
# Filling the missing Values by Median
df['agent'].fillna(agent_median, inplace = True)
```

In [ ]:
```

```

In [ ]:
```
df['agent'].head(5)
```

Out[ ]:
```
0     14.0
1     14.0
2     14.0
3    304.0
4    240.0
Name: agent, dtype: float64
```

In [ ]:
```
# Check Missing Values.
df['country'].isnull().sum()
#
```

Out[ ]:
```
488
```

In [ ]:
```
# Now Fill The Missing Values by mode
country_mode = df['country'].mode()[0]
df['country'].fillna(country_mode, inplace = True)
country_mode
```

Out[ ]:
```
'PRT'
```

In [ ]:
```
# Check missing Values.
df['country'].isnull().sum()
```

Out[ ]:
```
0
```

In [ ]:
```
df_child = df['children'].mean()
df['children'].fillna(df_child, inplace = True)
```

In [ ]:
```
# Check again missing Values.
df['country'].isnull().sum()
```

Out[ ]:
```
0
```

In [ ]:
```
# Now 'reservation_status_date' dtype is object now convert into datetime
df['reservation_status_date'] = pd.to_datetime(df['reservation_status_date'])
```

In [ ]:
```
# Now check
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 119390 entries, 0 to 119389
Data columns (total 31 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                        119390 non-null  float64
 11  babies                          119390 non-null  int64
 12  meal                            119390 non-null  object
 13  country                         119390 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                           119390 non-null  float64
 24  days_in_waiting_list            119390 non-null  int64
 25  customer_type                   119390 non-null  object
 26  adr                             119390 non-null  float64
 27  required_car_parking_spaces     119390 non-null  int64
 28  total_of_special_requests       119390 non-null  int64
 29  reservation_status              119390 non-null  object
 30  reservation_status_date         119390 non-null  datetime64[ns]
dtypes: datetime64[ns](1), float64(3), int64(16), object(11)
memory usage: 28.2+ MB
```

In [ ]:
```
# Statical Summery
df.describe().T
```

Out[ ]:

	count	mean	min	25%	50%	75%	max	std
is_canceled	119390.0	0.370416	0.0	0.0	0.0	1.0	1.0	0.482918
lead_time	119390.0	104.011416	0.0	18.0	69.0	160.0	737.0	106.863097
arrival_date_year	119390.0	2016.156554	2015.0	2016.0	2016.0	2017.0	2017.0	0.707476
arrival_date_week_number	119390.0	27.165173	1.0	16.0	28.0	38.0	53.0	13.605138
arrival_date_day_of_month	119390.0	15.798241	1.0	8.0	16.0	23.0	31.0	8.780829
stays_in_weekend_nights	119390.0	0.927599	0.0	0.0	1.0	2.0	19.0	0.998613
stays_in_week_nights	119390.0	2.500302	0.0	1.0	2.0	3.0	50.0	1.908286
adults	119390.0	1.856403	0.0	2.0	2.0	2.0	55.0	0.579261
children	119390.0	0.10389	0.0	0.0	0.0	0.0	10.0	0.398555
babies	119390.0	0.007949	0.0	0.0	0.0	0.0	10.0	0.097436
is_repeated_guest	119390.0	0.031912	0.0	0.0	0.0	0.0	1.0	0.175767
previous_cancellations	119390.0	0.087118	0.0	0.0	0.0	0.0	26.0	0.844336
previous_bookings_not_canceled	119390.0	0.137097	0.0	0.0	0.0	0.0	72.0	1.497437
booking_changes	119390.0	0.221124	0.0	0.0	0.0	0.0	21.0	0.652306
agent	119390.0	76.744392	1.0	9.0	14.0	152.0	535.0	105.904658
days_in_waiting_list	119390.0	2.321149	0.0	0.0	0.0	0.0	391.0	17.594721
adr	119390.0	101.831122	-6.38	69.29	94.575	126.0	5400.0	50.53579
required_car_parking_spaces	119390.0	0.062518	0.0	0.0	0.0	0.0	8.0	0.245291
total_of_special_requests	119390.0	0.571363	0.0	0.0	0.0	1.0	5.0	0.792798
reservation_status_date	119390	2016-07-30 00:24:47.883407104	2014-10-17 00:00:00	2016-02-01 00:00:00	2016-08-07 00:00:00	2017-02-08 00:00:00	2017-09-14 00:00:00	NaN

In [ ]:
```
df.describe(include= 'object').T
```

Out[ ]:

	count	unique	top	freq
hotel	119390	2	City Hotel	79330
arrival_date_month	119390	12	August	13877
meal	119390	5	BB	92310
country	119390	177	PRT	49078
market_segment	119390	8	Online TA	56477
distribution_channel	119390	5	TA/TO	97870
reserved_room_type	119390	10	A	85994
assigned_room_type	119390	12	A	74053
deposit_type	119390	3	No Deposit	104641
customer_type	119390	4	Transient	89613
reservation_status	119390	3	Check-Out	75166

In [ ]:
```
df.describe(exclude= 'object').T
```

Out[ ]:

	count	mean	min	25%	50%	75%	max	std
is_canceled	119390.0	0.370416	0.0	0.0	0.0	1.0	1.0	0.482918
lead_time	119390.0	104.011416	0.0	18.0	69.0	160.0	737.0	106.863097
arrival_date_year	119390.0	2016.156554	2015.0	2016.0	2016.0	2017.0	2017.0	0.707476
arrival_date_week_number	119390.0	27.165173	1.0	16.0	28.0	38.0	53.0	13.605138
arrival_date_day_of_month	119390.0	15.798241	1.0	8.0	16.0	23.0	31.0	8.780829
stays_in_weekend_nights	119390.0	0.927599	0.0	0.0	1.0	2.0	19.0	0.998613
stays_in_week_nights	119390.0	2.500302	0.0	1.0	2.0	3.0	50.0	1.908286
adults	119390.0	1.856403	0.0	2.0	2.0	2.0	55.0	0.579261
children	119390.0	0.10389	0.0	0.0	0.0	0.0	10.0	0.398555
babies	119390.0	0.007949	0.0	0.0	0.0	0.0	10.0	0.097436
is_repeated_guest	119390.0	0.031912	0.0	0.0	0.0	0.0	1.0	0.175767
previous_cancellations	119390.0	0.087118	0.0	0.0	0.0	0.0	26.0	0.844336
previous_bookings_not_canceled	119390.0	0.137097	0.0	0.0	0.0	0.0	72.0	1.497437
booking_changes	119390.0	0.221124	0.0	0.0	0.0	0.0	21.0	0.652306
agent	119390.0	76.744392	1.0	9.0	14.0	152.0	535.0	105.904658
days_in_waiting_list	119390.0	2.321149	0.0	0.0	0.0	0.0	391.0	17.594721
adr	119390.0	101.831122	-6.38	69.29	94.575	126.0	5400.0	50.53579
required_car_parking_spaces	119390.0	0.062518	0.0	0.0	0.0	0.0	8.0	0.245291
total_of_special_requests	119390.0	0.571363	0.0	0.0	0.0	1.0	5.0	0.792798
reservation_status_date	119390	2016-07-30 00:24:47.883407104	2014-10-17 00:00:00	2016-02-01 00:00:00	2016-08-07 00:00:00	2017-02-08 00:00:00	2017-09-14 00:00:00	NaN

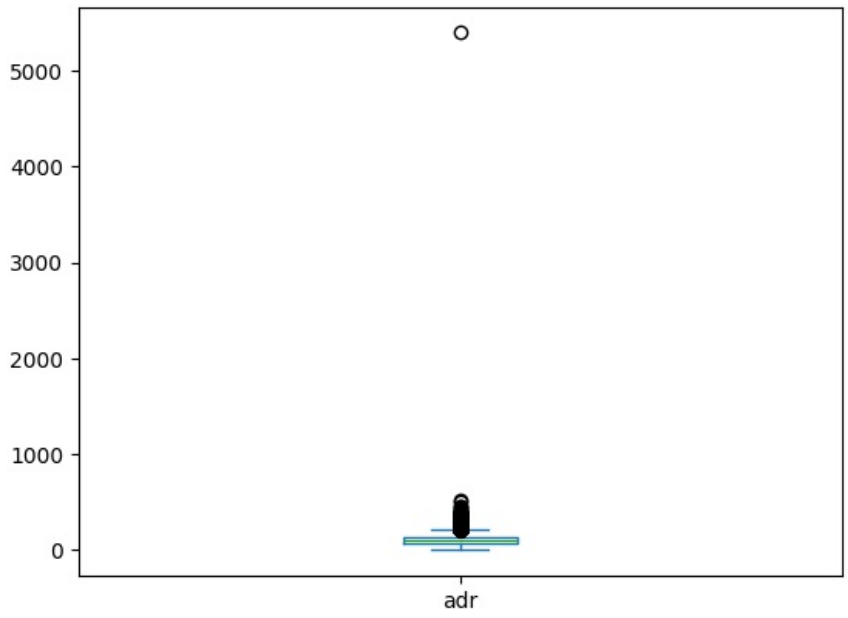
In [ ]:

```
for col in df.describe(include='object').columns:
    print('COL_NAME ->', col)
    print('UNIQUE_VALUES ->', df[col].unique())
    print('------------------------------------------------------------------------')
```

```
COL_NAME -> hotel
UNIQUE_VALUES -> ['Resort Hotel' 'City Hotel']
----------------------------------------------------------------------
COL_NAME -> arrival_date_month
UNIQUE_VALUES -> ['July' 'August' 'September' 'October' 'November' 'December' 'January'
 'February' 'March' 'April' 'May' 'June']
----------------------------------------------------------------------
COL_NAME -> meal
UNIQUE_VALUES -> ['BB' 'FB' 'HB' 'SC' 'Undefined']
----------------------------------------------------------------------
COL_NAME -> country
UNIQUE_VALUES -> ['PRT' 'GBR' 'USA' 'ESP' 'IRL' 'FRA' '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']
----------------------------------------------------------------------
COL_NAME -> market_segment
UNIQUE_VALUES -> ['Direct' 'Corporate' 'Online TA' 'Offline TA/TO' 'Complementary' 'Groups'
 'Undefined' 'Aviation']
----------------------------------------------------------------------
COL_NAME -> distribution_channel
UNIQUE_VALUES -> ['Direct' 'Corporate' 'TA/TO' 'Undefined' 'GDS']
----------------------------------------------------------------------
COL_NAME -> reserved_room_type
UNIQUE_VALUES -> ['C' 'A' 'D' 'E' 'G' 'F' 'H' 'L' 'P' 'B']
----------------------------------------------------------------------
COL_NAME -> assigned_room_type
UNIQUE_VALUES -> ['C' 'A' 'D' 'E' 'G' 'F' 'I' 'B' 'H' 'P' 'L' 'K']
----------------------------------------------------------------------
COL_NAME -> deposit_type
UNIQUE_VALUES -> ['No Deposit' 'Refundable' 'Non Refund']
----------------------------------------------------------------------
COL_NAME -> customer_type
UNIQUE_VALUES -> ['Transient' 'Contract' 'Transient-Party' 'Group']
----------------------------------------------------------------------
COL_NAME -> reservation_status
UNIQUE_VALUES -> ['Check-Out' 'Canceled' 'No-Show']
----------------------------------------------------------------------
```

In [ ]:
```
df['adr'].plot(kind='box') # To check the outlier
```

Out[ ]: <Axes: >



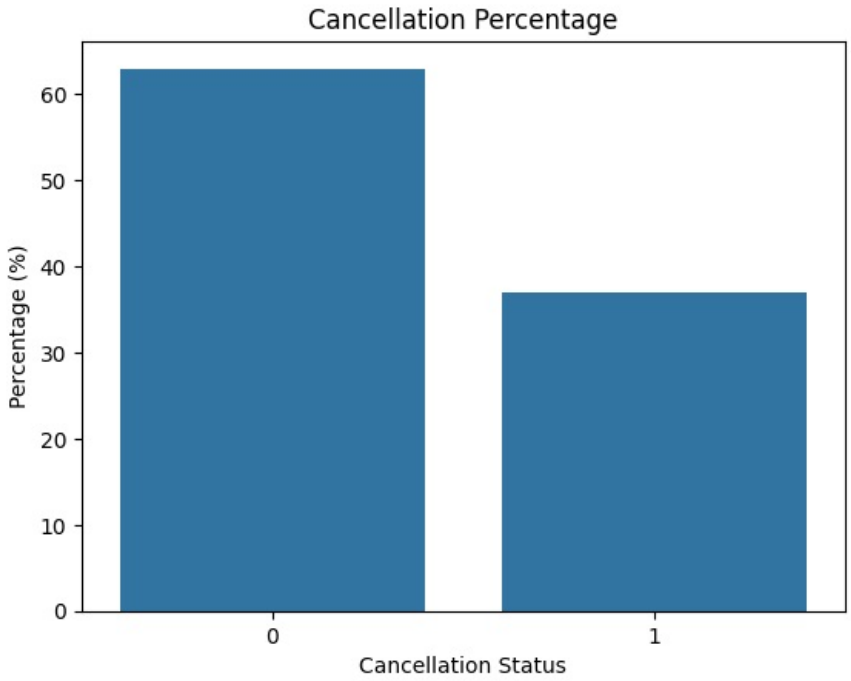
'Data Vizualizations'

In [ ]:
```
# Check The cancellation and reservation Percetage .
```

```
cancel_percentage = df['is_canceled'].value_counts(normalize=True) * 100
print(f'The Total Reserved_order percentage is: {cancel_percentage[0]:.2f}%.')
print(f'The Total cancel_percentage is: {cancel_percentage[1]:.2f}%.')
```

```
The Total Reserved_order percentage is: 62.96%.
The Total cancel_percentage is: 37.04%.
```

In [ ]:
```
# Plot the Barplot .
sns.barplot(x=cancel_percentage.index, y=cancel_percentage.values)
plt.title('Cancellation Percentage')
plt.xlabel('Cancellation Status')
plt.ylabel('Percentage (%)')
plt.show()
```



> - 0 Mean here the not_cancelled Reservations.
> - 1 Mean here the cancelled Reservations.

In [ ]:
```
# Calculate the cancellation percentage for each hotel
cancel_percentage_hotel = df.groupby('hotel')['is_canceled'].mean() * 100

# Print the cancellation percentages for each hotel type
print(f'The Total Reserved_order percentage for City Hotel is: {cancel_percentage_hotel["City Hotel"]:.2f}%.')
print(f'The Total Reserved_order percentage for Resort Hotel is: {cancel_percentage_hotel["Resort Hotel"]:.2f}%
```

```
The Total Reserved_order percentage for City Hotel is: 41.73%.
The Total Reserved_order percentage for Resort Hotel is: 27.76%.
```

In [ ]:
```
# Create a countplot
ax = sns.countplot(x='hotel', hue='is_canceled', data=df)
# Customize the plot
ax.set_title('Hotel Booking Cancellations by Hotel Type')
ax.set_xlabel('Hotel Type')
ax.set_ylabel('Count')

# Show the plot
plt.show()
```

Location:

- Resort hotels are typically located in scenic and tranquil destinations, - offering a getaway from urban life.
- City hotels, on the other hand, are situated in bustling metropolitan areas, making them convenient for business and city exploration.

Booking Volume:

- Resort hotels often have fewer bookings due to their focus on exclusivity and relaxation.
- City hotels tend to have a higher volume of bookings, catering to business travelers and tourists.

Price Point:

- Resort hotels are associated with a relatively higher price point , making them a choice for those seeking a luxurious vacation.
- City hotels may offer a range of price options, making them accessible to a broader spectrum of travelers.
- These bullet points succinctly describe the key differences between resort hotels and city hotels.

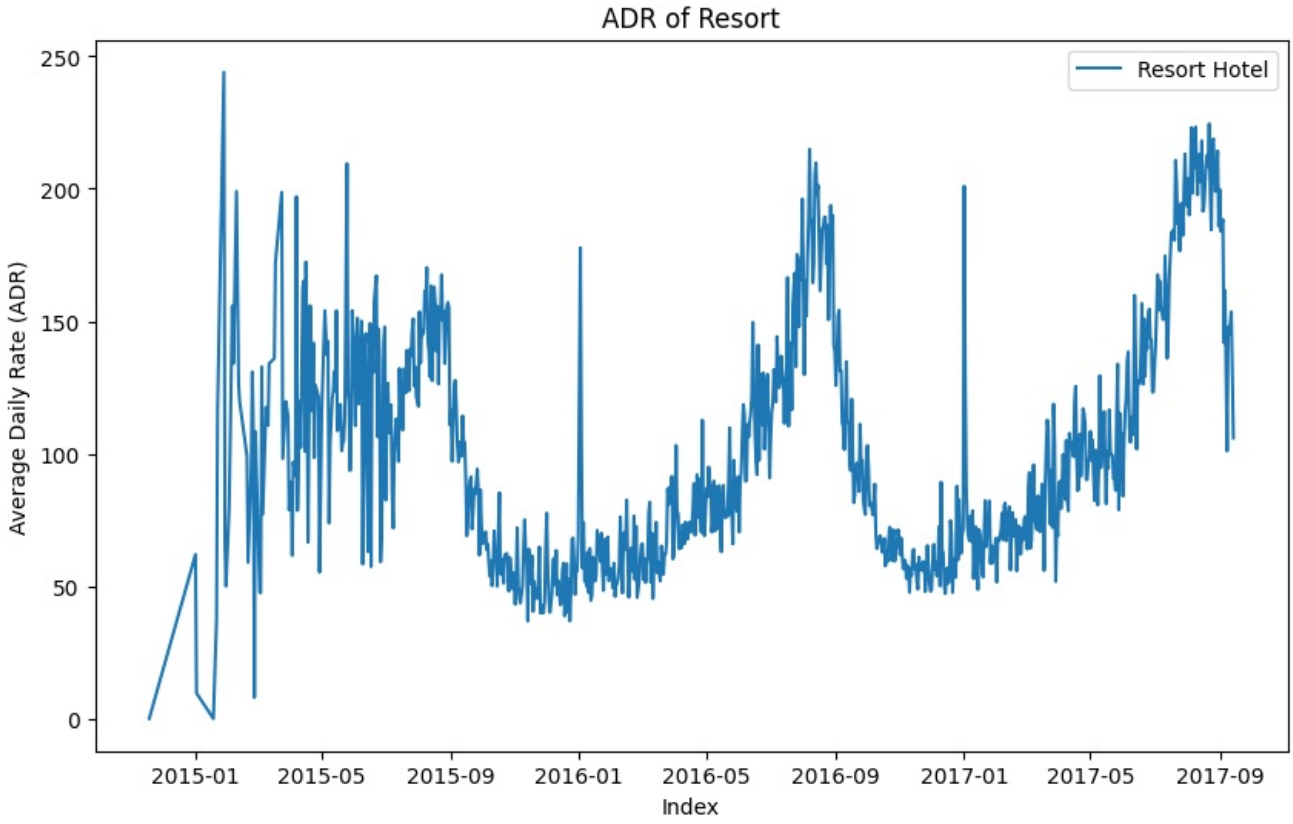
In [ ]:
```
# Filter the DataFrame for Resort Hotel
resort_hotel = df[df['hotel'] == 'Resort Hotel']

# Calculate the cancellation percentages for Resort Hotel
cancel_percentage_resort = resort_hotel['is_canceled'].value_counts(normalize=True) * 100

# Print the cancellation percentages for Resort Hotel
print('Cancellation percentages for Resort Hotel:')
print(cancel_percentage_resort)

# Filter the DataFrame for City Hotel
city_hotel = df[df['hotel'] == 'City Hotel']

# Calculate the cancellation percentages for City Hotel
cancel_percentage_city = city_hotel['is_canceled'].value_counts(normalize=True) * 100

# Print the cancellation percentages for City Hotel
print('Cancellation percentages for City Hotel:')
print(cancel_percentage_city)
```

```
Cancellation percentages for Resort Hotel:
is_canceled
0    72.236645
1    27.763355
Name: proportion, dtype: float64
Cancellation percentages for City Hotel:
is_canceled
0    58.273037
1    41.726963
Name: proportion, dtype: float64
```

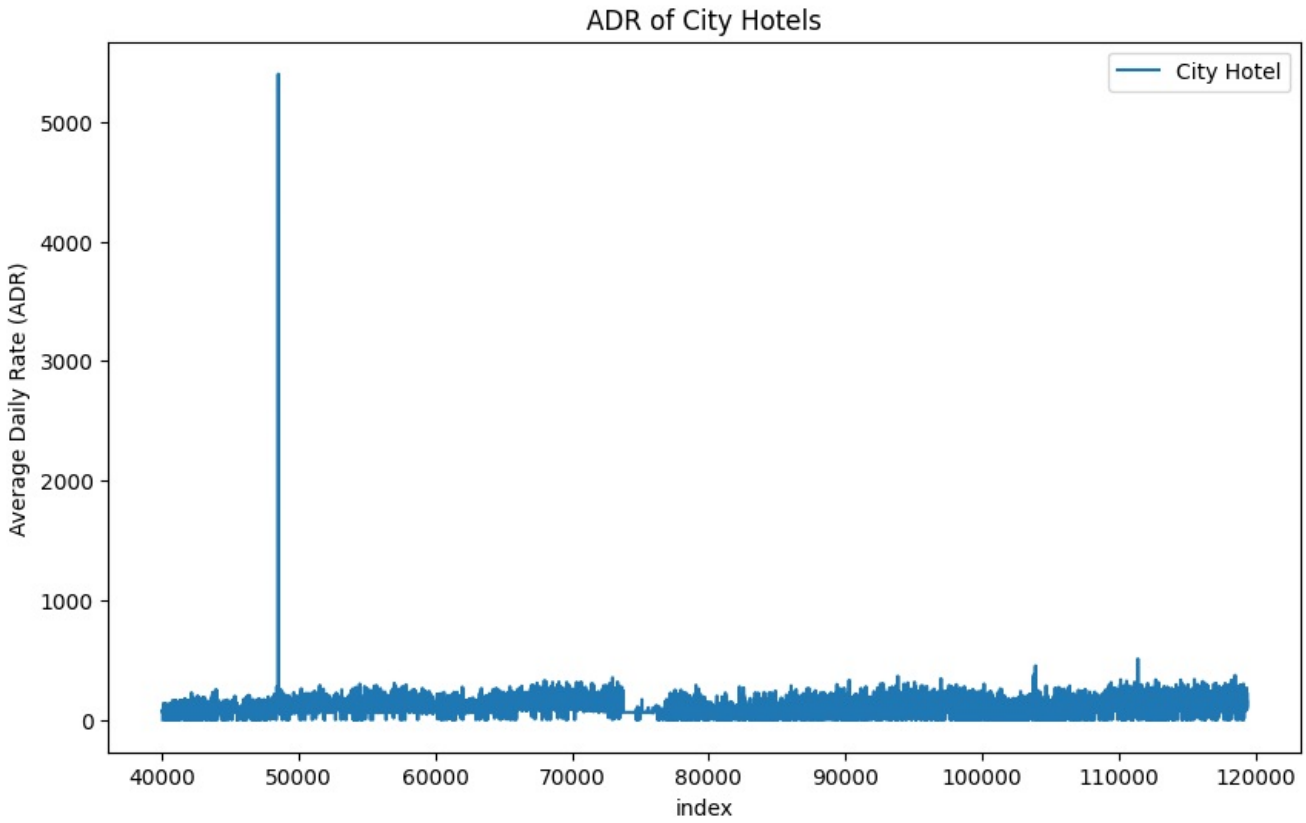
In [ ]:
```
resort_hotel = resort_hotel.groupby('reservation_status_date')[['adr']].mean()
city_hotel_hotel = city_hotel.groupby('reservation_status_date')[['adr']].mean()
```

In [ ]:
```
# Set a larger figure size to create more space
plt.figure(figsize=(10, 6))
```

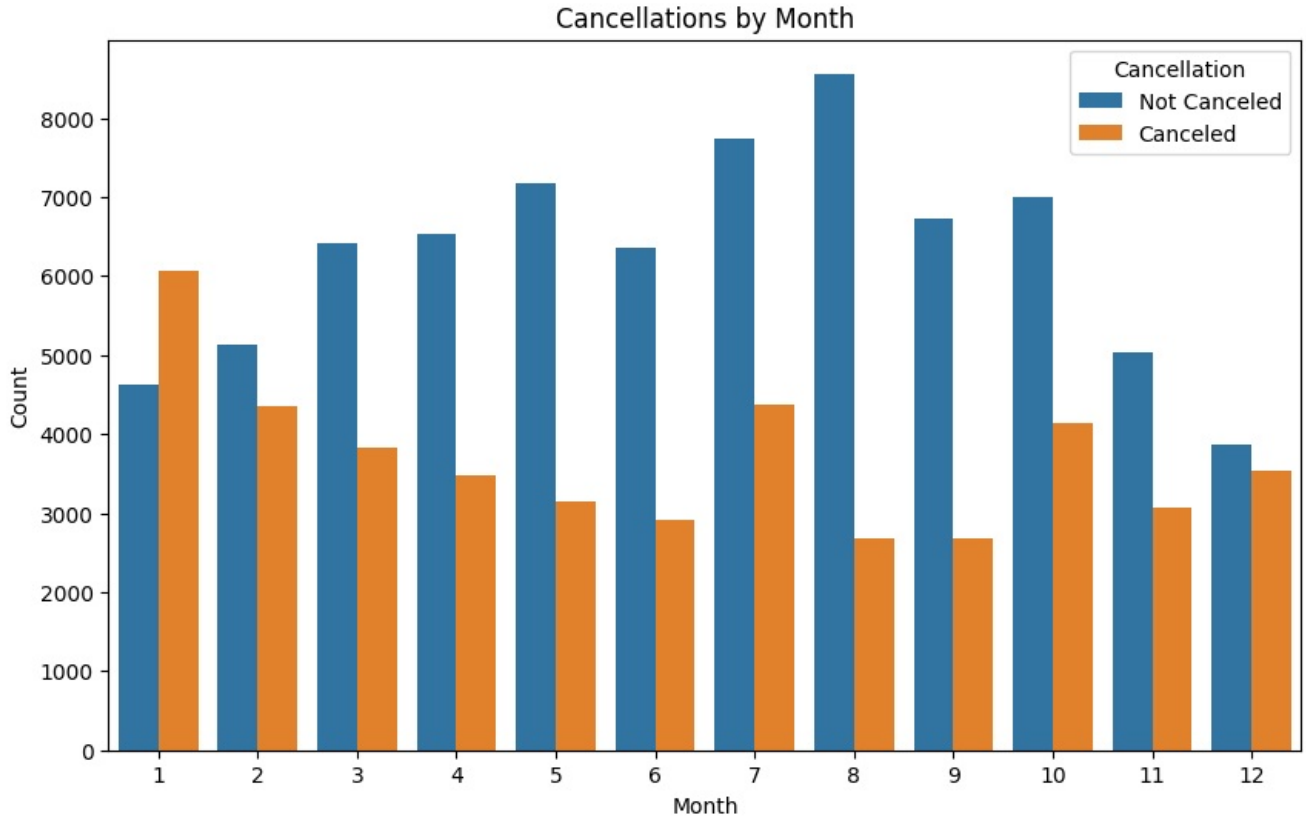
```
# Assuming you have the 'resort_hotel' and 'city_hotel' DataFrames
sns.lineplot(data=resort_hotel, x=resort_hotel.index, y='adr', label='Resort Hotel')
# now make a bins of reservation_status_date
# sns.lineplot(data=city_hotel, x=city_hotel.index, y='adr', label='City Hotel')

plt.xlabel('Index')
plt.ylabel('Average Daily Rate (ADR)')
plt.title('ADR of Resort')
plt.legend()
plt.show()
```



In [ ]:
```
# Set a larger figure size to create more space
plt.figure(figsize=(10, 6))

# Assuming you have the 'resort_hotel' and 'city_hotel' DataFrames
# sns.lineplot(data=resort_hotel, x=resort_hotel.index, y='adr', label='Resort Hotel')
# now make a bins of reservation_status_date
sns.lineplot(data=city_hotel, x=city_hotel.index, y='adr', label='City Hotel')

plt.xlabel('index')
plt.ylabel('Average Daily Rate (ADR)')
plt.title('ADR of City Hotels')
plt.legend()
plt.show()
```



In [ ]:
```
df['month'] = df['reservation_status_date'].dt.month

plt.figure(figsize=(10, 6))

ax = sns.countplot(x='month', hue='is_canceled', data=df)

plt.xlabel('Month')
plt.ylabel('Count')
plt.title('Cancellations by Month')
plt.legend(title='Cancellation', labels=['Not Canceled', 'Canceled'])

plt.show()
```



> - **Month with the highest number of confirmed reservations:**

Label the bar for August as "Confirmed Reservations."

- **Month with the highest number of canceled reservations:**

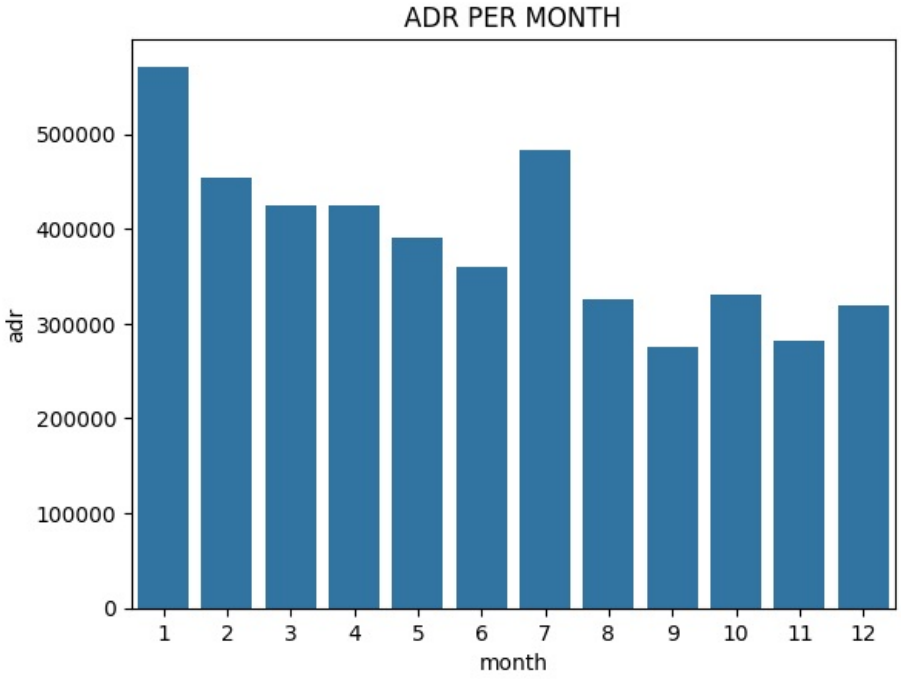
Label the bar for August as "Canceled Reservations."

- **Month with the most canceled reservations:**

Label the bar for January as "Most Canceled Reservations."

ADR Per Month Wise canceled Reservations:

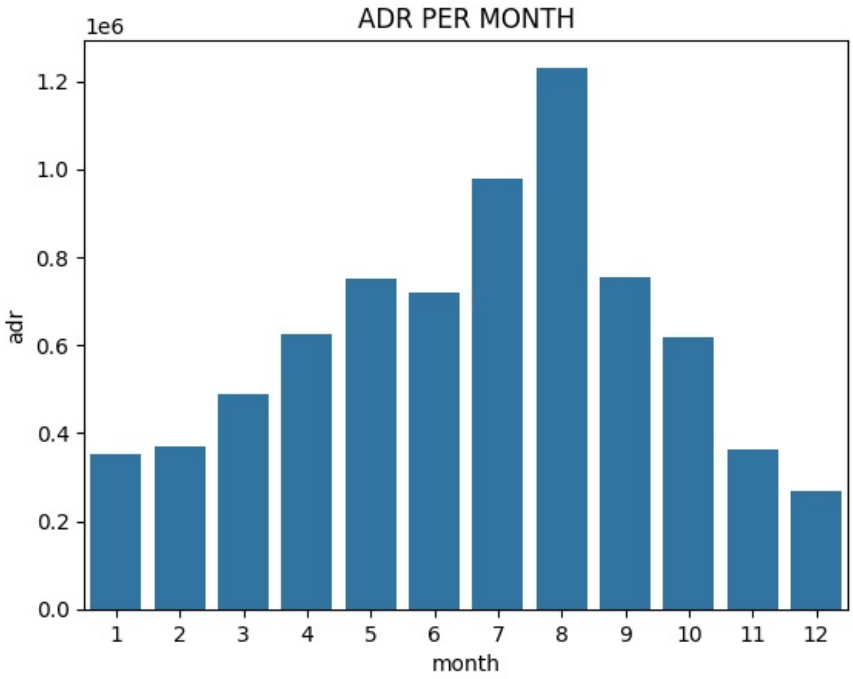
In [ ]:
```
df_wise_month = df[df['is_canceled']==1].groupby('month')[['adr']].sum().reset_index()
sns.barplot(x = 'month', y = 'adr', data= df_wise_month)
plt.title("ADR PER MONTH")
plt.figure(figsize=(8,6))
plt.show()
```




```
<Figure size 800x600 with 0 Axes>
```

ADR Per Month Wise Not-canceled Reservations:

In [ ]:
```
df_wise_month = df[df['is_canceled']==0].groupby('month')[['adr']].sum().reset_index()
sns.barplot(x = 'month', y = 'adr', data= df_wise_month)
plt.title("ADR PER MONTH")
plt.figure(figsize=(8,6))
plt.show()
```




```
<Figure size 800x600 with 0 Axes>
```

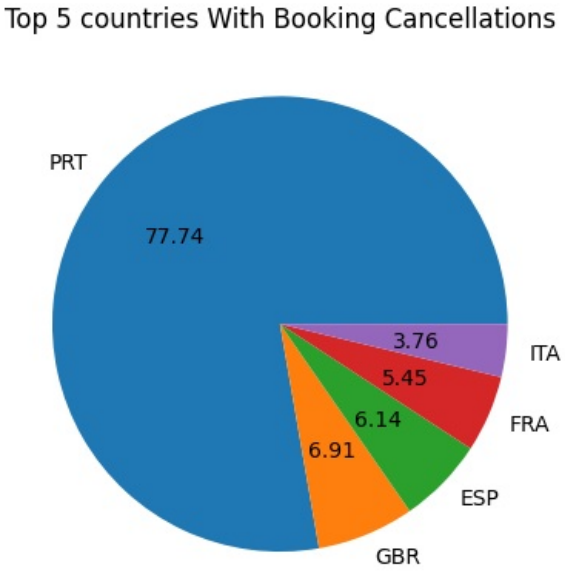
> - **When the prices higher then Cancellation Higher.**

In [ ]:
```
cancel_df = df[df['is_canceled'] == 1]
top_5_countries = cancel_df['country'].value_counts().head(5)

print('Top 5 countries with the most cancellations:')
for country, count in top_5_countries.items():
    print(f"The country {country} has {count} cancellations.")
```

```
Top 5 countries with the most cancellations:
The country PRT has 27586 cancellations.
The country GBR has 2453 cancellations.
The country ESP has 2177 cancellations.
The country FRA has 1934 cancellations.
The country ITA has 1333 cancellations.
```

In [ ]:
```
plt.pie(top_5_country, autopct='%.2f', labels=top_5_country.index)
plt.title('Top 5 countries With Booking Cancellations')
```

Out[ ]:
```
Text(0.5, 1.0, 'Top 5 countries With Booking Cancellations')
```



`Cancellations in Top 5 Countries:`

> - **Portugal (PRT)** stands out with the highest number of booking cancellations, totaling 27,586, showcasing its considerable impact on hotel occupancy.

> - **The United Kingdom** (GBR) and Spain (ESP) closely follow in the list of countries with notable booking fluctuations, emphasizing their significance in the hotel industry.

- **France (FRA)** and Italy (ITA) contribute with 1,934 and 1,333 cancellations, respectively, underlining their influence on hotel reservations.

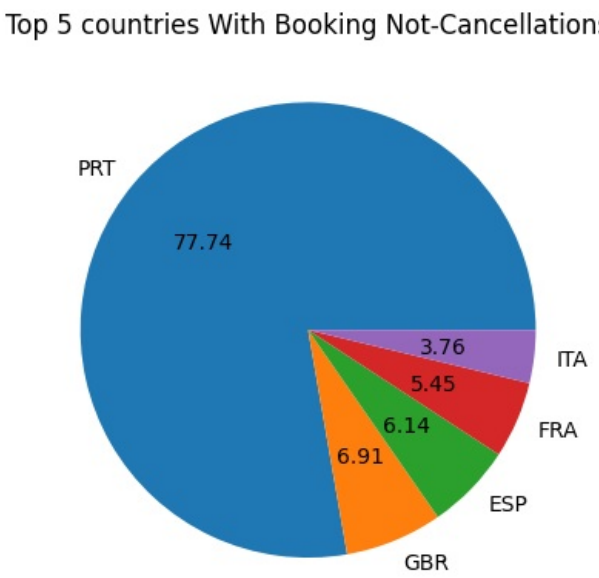
- Monitoring and understanding the booking behavior of these top 5 countries is vital for optimizing hotel operations and addressing industry challenges effectively.

In [ ]:
```
not_cancel_df = df[df['is_canceled'] == 0]
top_5_countries = not_cancel_df['country'].value_counts().head(5)

print('Top 5 countries with the most successful reservations:')
for country, count in top_5_countries.items():
    print(f"The country {country} has {count} successful reservations.")
```

```
Top 5 countries with the most successful reservations:
The country PRT has 21492 successful reservations.
The country GBR has 9676 successful reservations.
The country FRA has 8481 successful reservations.
The country ESP has 6391 successful reservations.
The country DEU has 6069 successful reservations.
```

In [ ]:
```
plt.pie(top_5_country, autopct='%.2f', labels=top_5_country.index)
plt.title('Top 5 countries With Booking Not-Cancellations')
```

Out[ ]:
```
Text(0.5, 1.0, 'Top 5 countries With Booking Not-Cancellations')
```



`Not-Cancellations in Top 5 Countries:`

- □□ **Portugal (PRT):** Leads with 21,492 not-cancellations, indicating strong commitment to reservations.

- □□ **United Kingdom (GBR):** Follows with 9,676 not-cancellations, showing a significant number of confirmed bookings.

- □□ **France (FRA):** Records 8,481 not-cancellations, reflecting consistent bookings in the French market.

- □□ **Spain (ESP):** Presents 6,391 not-cancellations, highlighting a stable reservation pattern in Spain.

- □□ **Germany (DEU):** Contributes with 6,069 not-cancellations, indicating a strong presence in the hotel market.

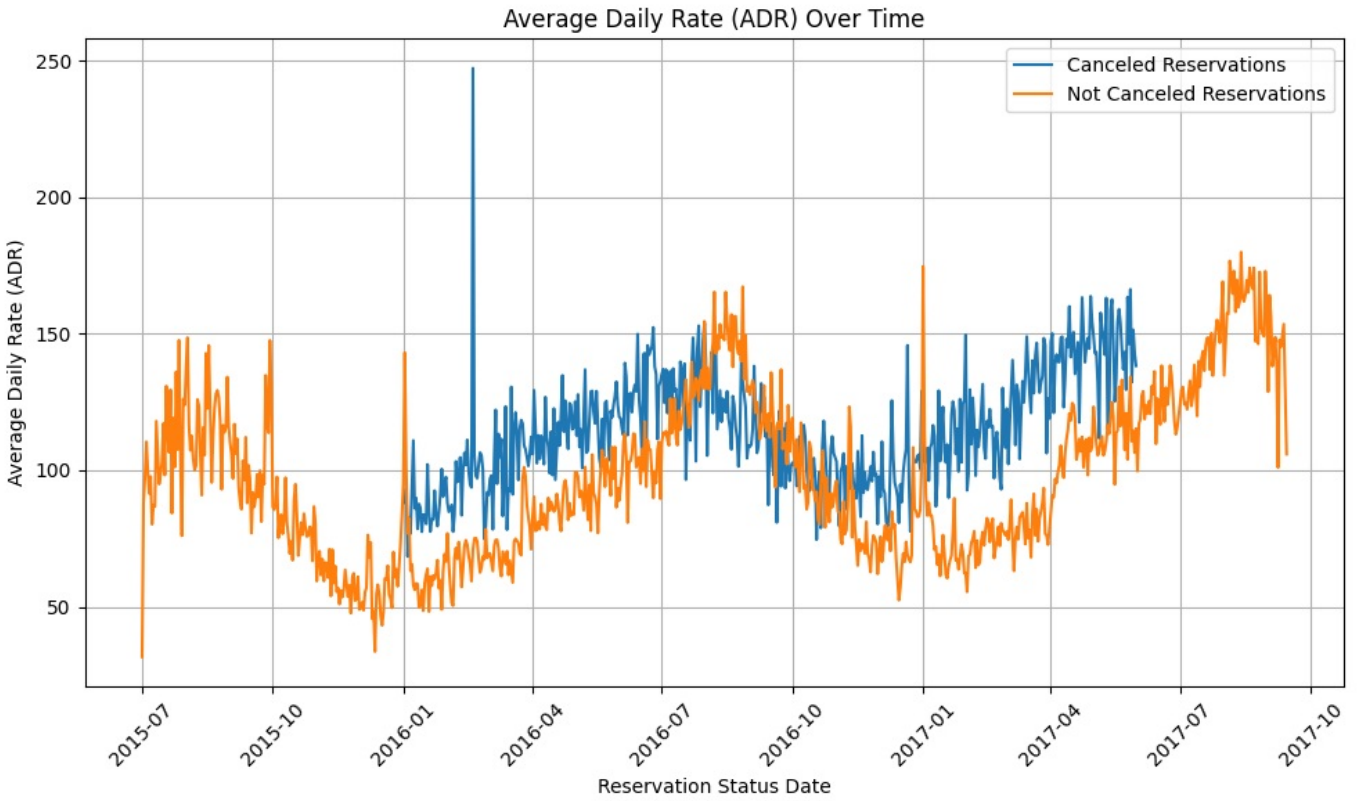
Let's Check the Customers are Coming Offine:

In [ ]:
```
df['market_segment'].value_counts(normalize=True)
```

Out[ ]:
```
market_segment
Online TA        0.473046
Offline TA/TO    0.202856
Groups           0.165935
Direct           0.105587
Corporate        0.044350
Complementary    0.006223
Aviation         0.001985
Undefined        0.000017
Name: proportion, dtype: float64
```

In [ ]:
```
# Reset and sort the data
```

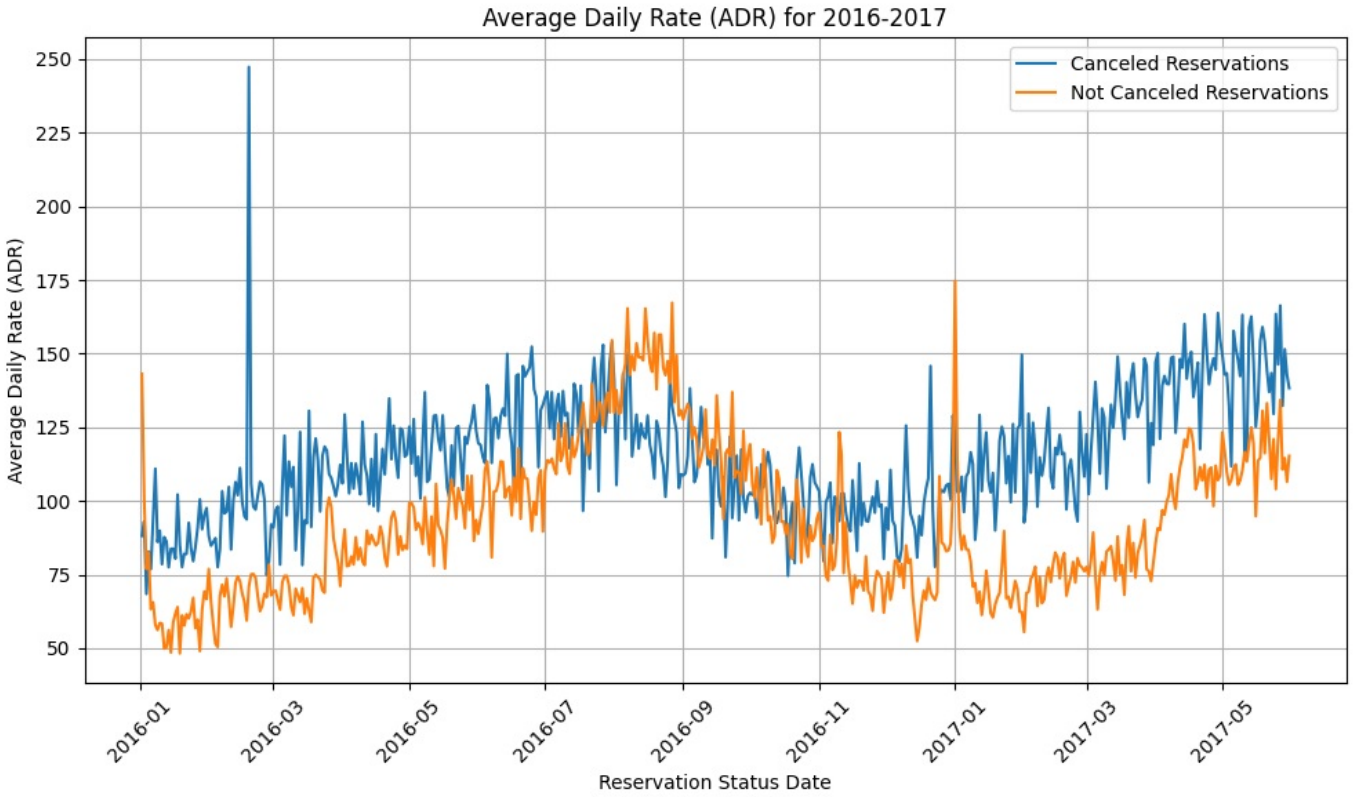
```
cancel_df_adr.reset_index(inplace=True)
cancel_df_adr.sort_values('reservation_status_date', inplace=True)

# Calculate the average daily rate (ADR) for not canceled reservations
not_cancel_df = df[df['is_canceled'] == 0]
not_cancel_df_adr = not_cancel_df.groupby('reservation_status_date')[['adr']].mean()
not_cancel_df_adr.reset_index(inplace=True)
not_cancel_df_adr.sort_values('reservation_status_date', inplace=True)

# Create a line plot using Matplotlib to visualize ADR by Reservation Status Date
plt.figure(figsize=(10, 6))
plt.plot(cancel_df_adr['reservation_status_date'], cancel_df_adr['adr'], label='Canceled Reservations')
plt.plot(not_cancel_df_adr['reservation_status_date'], not_cancel_df_adr['adr'], label='Not Canceled Reservatio

# Customize the plot
plt.title('Average Daily Rate (ADR) Over Time')
plt.xlabel('Reservation Status Date')
plt.ylabel('Average Daily Rate (ADR)')
plt.legend()
plt.grid(True)

# Display the plot
plt.xticks(rotation=45)  # Rotate x-axis labels for better readability
plt.tight_layout()
plt.show()
```



In [ ]:
```
market_segmant_base_df = market_segmant_base.to_frame().reset_index()
```

In [ ]:
```
# Filter the data for a specific date range (from '2016' to '2017-06')
cancel_df_adr = cancel_df_adr[(cancel_df_adr['reservation_status_date'] > '2016') & (cancel_df_adr['reservation_
not_cancel_df_adr = not_cancel_df_adr[(not_cancel_df_adr['reservation_status_date'] > '2016') & (not_cancel_df_a

# Extract x and y data
x_cancel = cancel_df_adr['reservation_status_date']
y_cancel = cancel_df_adr['adr']
x_not_cancel = not_cancel_df_adr['reservation_status_date']
y_not_cancel = not_cancel_df_adr['adr']

# Create a line plot to visualize ADR for the filtered date range using Matplotlib
plt.figure(figsize=(10, 6))
plt.plot(x_cancel, y_cancel, label='Canceled Reservations', marker='')
plt.plot(x_not_cancel, y_not_cancel, label='Not Canceled Reservations', marker='')

# Customize the plot
plt.title('Average Daily Rate (ADR) for 2016-2017')
plt.xlabel('Reservation Status Date')
plt.ylabel('Average Daily Rate (ADR)')
plt.legend()
plt.grid(True)

# Display the plot
```

```
plt.xticks(rotation=45)  # Rotate x-axis labels for better readability
plt.tight_layout()
plt.show()
```



- **ADR (Average Daily Rate) varies over time.**
- **Canceled reservations have different ADR trends compared to non-canceled ones.**
- **Analysis covers the period from 2016 to mid-2017.**
- **Potential pricing strategies and demand fluctuations in the hotel industry.**

Regards: Muhammad Irfan EDA_Expert.