Project Name - Hotel Bookings Analysis

Project Summary -

- **The purpose of the analysis:** This project aims to analyze the high cancellation rates at City Hotel and Resort Hotel.
- The goal is to understand the underlying reasons for these cancellations and their impact on the hotels' revenue and room utilization.
- The project will also explore other factors unrelated to their business and yearly revenue generation.

Problem Statements -

- **High Cancellation Rates:** Both City Hotel and Resort Hotel have been experiencing high cancellation rates in recent years. This has led to a decrease in revenue and suboptimal room utilization.

Importing the necessary libraries

In [1]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings('ignore')
```

Load Hotel Booking Dataset

In [2]:
```
Hotel_Booking = pd.read_csv('hotel_bookings 2.csv')
Hotel_Booking
```

Out[2]:

	hotel	is_canceled	lead_time	arrival_date_year	arrival_date_month	arrival_date_wee
0	Resort Hotel	0	342	2015	July	
1	Resort Hotel	0	737	2015	July	
2	Resort Hotel	0	7	2015	July	
3	Resort Hotel	0	13	2015	July	
4	Resort Hotel	0	14	2015	July	
...	...	...	...	...	...	
119385	City Hotel	0	23	2017	August	
119386	City Hotel	0	102	2017	August	
119387	City Hotel	0	34	2017	August	
119388	City Hotel	0	109	2017	August	
119389	City Hotel	0	205	2017	August	

119390 rows × 32 columns

About the Dataset – Airbnb Bookings

- This Hotel Booking dataset contains nearly 119390 observations , with 32 columns of data.

##UNDERSTAND THE GIVEN VARIABLES

hotel:- Type of hotel (Resort Hotel or City Hotel).

is_canceled:- If the booking was canceled (1) or not (0).

lead_time:- Number of days between the booking date and arrival date

arrival_date_year:- Year of arrival

arrival_date_month:- Month of arrival

arrival_date_week_number:- Week number of the year for arrival date

arrival_date_day_of_month:- Day of the month of arrival

stays_in_weekend_nights:- Number of weekend nights (Saturday or Sunday) the guest stayed or booked to stay at the hotel

stays_in_week_nights:- Number of week nights (Monday to Friday) the guest stayed or booked to stay at the hotel

adults:- Number of adults

children:- Number of children

babies:- Number of babies

meal:- Type of meal booked

country:- Country of origin of the guest

market_segment:- Market segment designation

distribution_channel:- Booking distribution channel

is_repeated_guest:- If the guest was a repeated guest (1) or not (0)

previous_cancellations:- Number of previous bookings that were cancelled by the customer prior to the current booking

previous_bookings_not_canceled:- Number of previous bookings not cancelled by the customer prior to the current booking

reserved_room_type:- Code of room type reserved

assigned_room_type:- Code for the type of room assigned to the booking

booking_changes:- Number of changes/amendments made to the booking from the moment the booking was entered on the PMS until the moment of check-in or cancellation

deposit_type:- Indication on if the customer made a deposit to guarantee the booking

agent:- ID of the travel agency that made the booking

company:- ID of the company/entity that made the booking or responsible for paying the booking

days_in_waiting_list:- Number of days the booking was in the waiting list before it was confirmed to the customer

customer_type:- Type of booking

adr:- Average Daily Rate as defined by dividing the sum of all lodging transactions by the total number of staying nights

required_car_parking_spaces:- Number of car parking spaces required by the customer

total_of_special_requests:- Number of special requests made by the customer

reservation_status:- Last reservation status, assuming one of three categories: Canceled, Check-Out, No-Show

reservation_status_date:- Date at which the last status was set

Data Exploration and Data Cleaning

In [3]:
```
Hotel_Booking.head()
```

Out[3]:

	hotel	is_canceled	lead_time	arrival_date_year	arrival_date_month	arrival_date_week_nu
0	Resort Hotel	0	342	2015	July	
1	Resort Hotel	0	737	2015	July	
2	Resort Hotel	0	7	2015	July	
3	Resort Hotel	0	13	2015	July	
4	Resort Hotel	0	14	2015	July	

5 rows × 32 columns

In [4]:
```
#checking what are the variables here:
Hotel_Booking.columns
```

Out[4]:
```
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')
```

In [5]:
```
Hotel_Booking.head().T
```

Out[5]:

	0	1	2	3	4
hotel	Resort Hotel	Resort Hotel	Resort Hotel	Resort Hotel	Resort Hotel
is_canceled	0	0	0	0	0
lead_time	342	737	7	13	14
arrival_date_year	2015	2015	2015	2015	2015
arrival_date_month	July	July	July	July	July
arrival_date_week_number	27	27	27	27	27
arrival_date_day_of_month	1	1	1	1	1
stays_in_weekend_nights	0	0	0	0	0
stays_in_week_nights	0	0	1	1	2
adults	2	2	1	1	2
children	0.0	0.0	0.0	0.0	0.0
babies	0	0	0	0	0
meal	BB	BB	BB	BB	BB
country	PRT	PRT	GBR	GBR	GBR
market_segment	Direct	Direct	Direct	Corporate	Online TA
distribution_channel	Direct	Direct	Direct	Corporate	TA/TO
is_repeated_guest	0	0	0	0	0
previous_cancellations	0	0	0	0	0
previous_bookings_not_canceled	0	0	0	0	0
reserved_room_type	C	C	A	A	A
assigned_room_type	C	C	C	A	A
booking_changes	3	4	0	0	0
deposit_type	No Deposit	No Deposit	No Deposit	No Deposit	No Deposit
agent	NaN	NaN	NaN	304.0	240.0
company	NaN	NaN	NaN	NaN	NaN
days_in_waiting_list	0	0	0	0	0
customer_type	Transient	Transient	Transient	Transient	Transient
adr	0.0	0.0	75.0	75.0	98.0
required_car_parking_spaces	0	0	0	0	0
total_of_special_requests	0	0	0	0	1
reservation_status	Check-Out	Check-Out	Check-Out	Check-Out	Check-Out
reservation_status_date	1/7/2015	1/7/2015	2/7/2015	2/7/2015	3/7/2015

In [6]:
```
#checking shape of Hotel Booking dataset
Hotel_Booking.shape
```

Out[6]: (119390, 32)

In [7]:
```
#basic information about the dataset
Hotel_Booking.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
```

In [8]:
```
Hotel_Booking['reservation_status_date'] = pd.to_datetime(Hotel_Booking['re
```

In [9]:
```
# checking null values of each columns
Hotel_Booking.isnull().sum()
```

Out[9]:
```
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
```

Country are not that much of null values, so first we are good to fill those with some substitutes.

In [10]:
```
Hotel_Booking['country'].fillna('unknown',inplace=True)
```

now, the columns **agent** and **company** have total null values agent is 12193 and company 82137.

agent and **company** column is not required for our analysis . We're good to drop this column.

In [11]:
```
Hotel_Booking.drop(['company', 'agent'], axis = 1, inplace = True)
Hotel_Booking.dropna(inplace = True)                                    #r
```

In [12]:
```
# checking null values of each columns
Hotel_Booking.isnull().sum()                    #no null values present in
```

Out[12]:
```
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
dtype: int64
```

In [13]:
```
for col in Hotel_Booking.describe(include = 'object').columns:
    print(col)
    print(Hotel_Booking[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' 'unknown' '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'
 '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']
--------------------------------------------------
deposit_type
['No Deposit' 'Refundable' 'Non Refund']
--------------------------------------------------
customer_type
['Transient' 'Contract' 'Transient-Party' 'Group']
--------------------------------------------------
reservation_status
['Check-Out' 'Canceled' 'No-Show']
--------------------------------------------------
```

Describe the Dataset and removing outliers

In [14]:
```
# describe the DataFrame
Hotel_Booking.describe()
```

Out[14]:

	is_canceled	lead_time	arrival_date_year	arrival_date_week_number	arrival_date
count	119386.000000	119386.000000	119386.000000	119386.000000	
mean	0.370395	104.014801	2016.156593	27.165003	
min	0.000000	0.000000	2015.000000	1.000000	
25%	0.000000	18.000000	2016.000000	16.000000	
50%	0.000000	69.000000	2016.000000	28.000000	
75%	1.000000	160.000000	2017.000000	38.000000	
max	1.000000	737.000000	2017.000000	53.000000	
std	0.482913	106.863286	0.707456	13.605334	

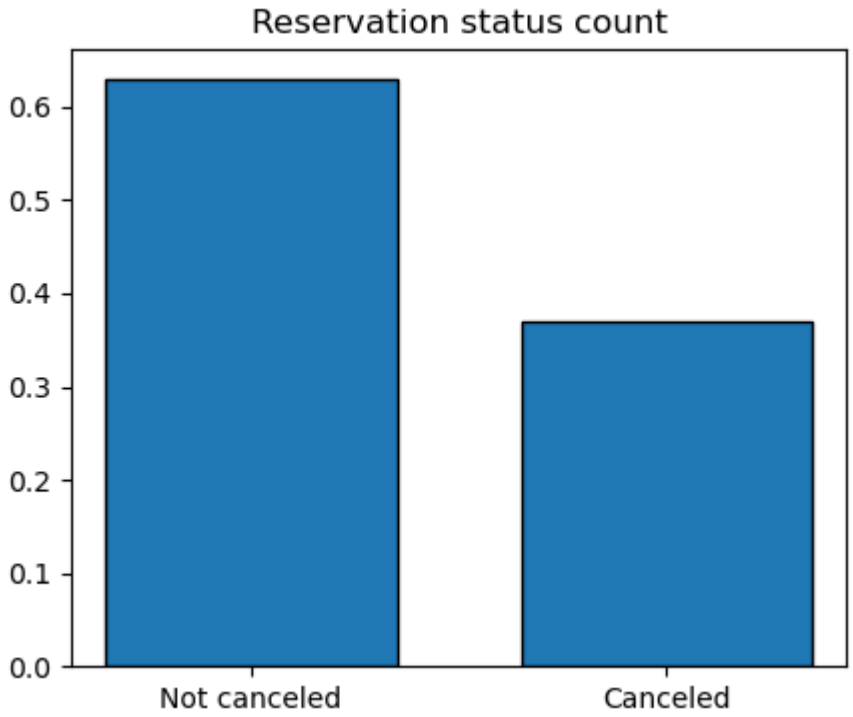
Note - adr (Average Daily Rate) column is very important so we have to find big outliers in important columns first.

In [15]:
```
Hotel_Booking = Hotel_Booking[Hotel_Booking['adr']<5000]
```

Data Analysis and Visualizations

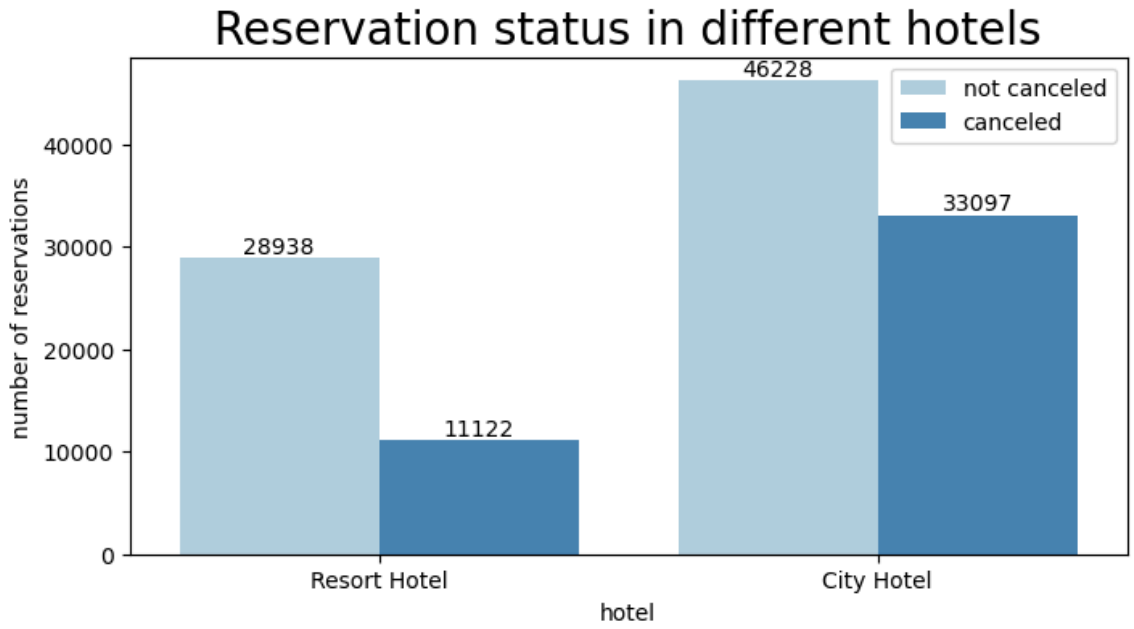
In [16]:
```
# Calculate cancellation percentage
cancelled_perc =  Hotel_Booking['is_canceled'].value_counts(normalize=True)
print(cancelled_perc)

# Create a bar chart to visualize reservation status counts
plt.figure(figsize=(5, 4))
plt.title('Reservation status count')
plt.bar(['Not canceled', 'Canceled'], cancelled_perc, edgecolor='k', width=
plt.show()
```

```
is_canceled
0    0.62961
1    0.37039
Name: proportion, dtype: float64
```



The graph shows that 72.84% of hotel bookings were not canceled, while 27.15% were canceled

In [17]:
```
plt.figure(figsize = (8,4))
ax1= sns.countplot(x = 'hotel', hue = 'is_canceled', data = Hotel_Booking, 
legend_labels,_ = ax1. get_legend_handles_labels()
ax1.legend(bbox_to_anchor=(1,1))
for bars in ax1.containers:
    ax1.bar_label(bars)
plt.title('Reservation status in different hotels', size = 20)
plt.xlabel('hotel')
plt.ylabel('number of reservations')
plt.legend(['not canceled', 'canceled'])
plt.show()
```



- "Based on the data, it appears 'Resort Hotel' might have more reservations than 'City Hotel'."
- The data shows 'City Hotel' has 36880 not canceled reservations and 15765 canceled reservations, while 'Resort Hotel' has 24956 not canceled reservations and 7287 canceled reservations.
- In comparison to resort hotels, city hotels have more bookings. It's possible that resorthotels are more expensive than those in cities.

In [18]:
```
resort_hotel = Hotel_Booking[Hotel_Booking['hotel'] == 'Resort Hotel']
resort_hotel['is_canceled'].value_counts(normalize = True)
```

Out[18]:
```
is_canceled
0    0.722366
1    0.277634
Name: proportion, dtype: float64
```

- Index: 0 (represents not canceled) and 1 (represents canceled)
- Resort Hotel 77.39% were not canceled and 22.60% were canceled.

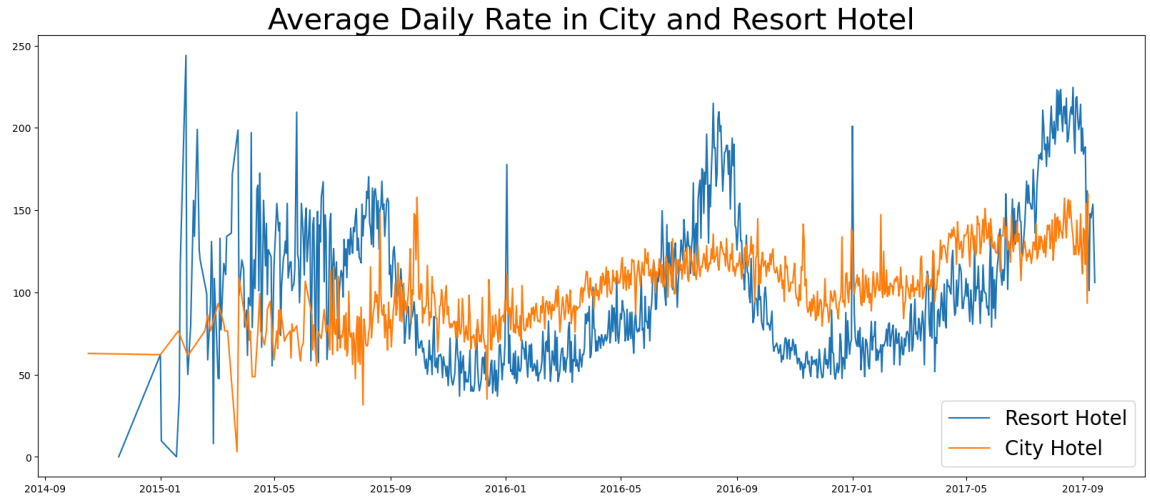
In [19]:
```
city_hotel = Hotel_Booking[Hotel_Booking['hotel'] == 'City Hotel']
city_hotel['is_canceled'].value_counts(normalize = True)
```

Out[19]:
```
is_canceled
0    0.582767
1    0.417233
Name: proportion, dtype: float64
```

- Index: 0 (represents not canceled) and 1 (represents canceled)
- "City Hotel" bookings 70.05% were not canceled and 29.94% were canceled.
- By comparing the cancellation rates for "Resort Hotel" (previously calculated) and "City Hotel", you can see that "City Hotel" has a slightly higher cancellation rate.

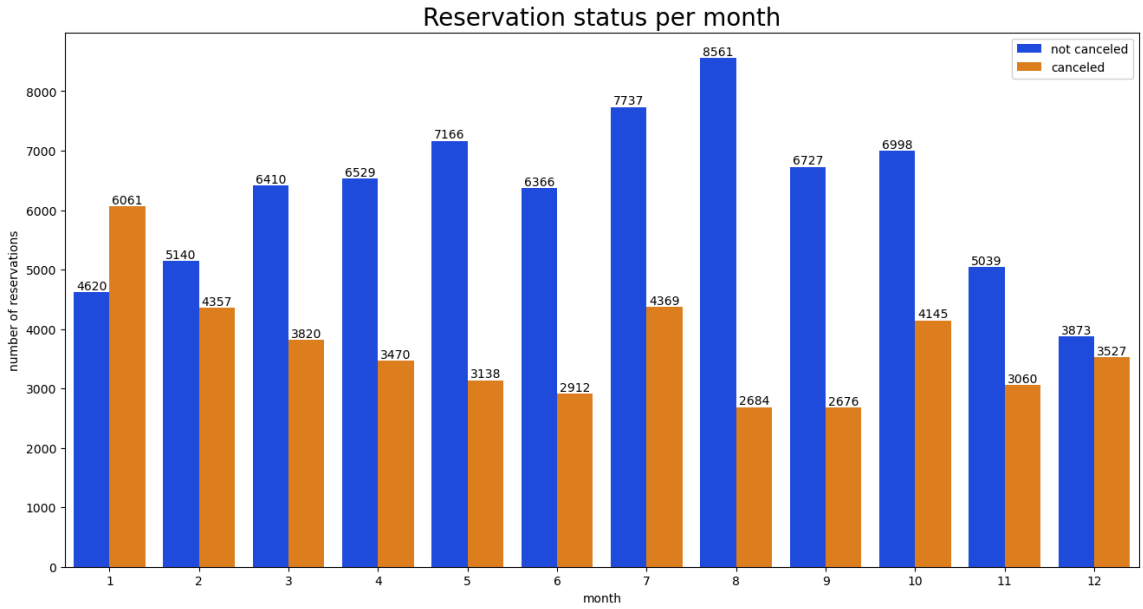
In [20]:
```
resort_hotel = resort_hotel.groupby('reservation_status_date')[['adr']].mea
city_hotel = city_hotel.groupby('reservation_status_date')[['adr']].mean()
```

In [21]:
```
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()
```



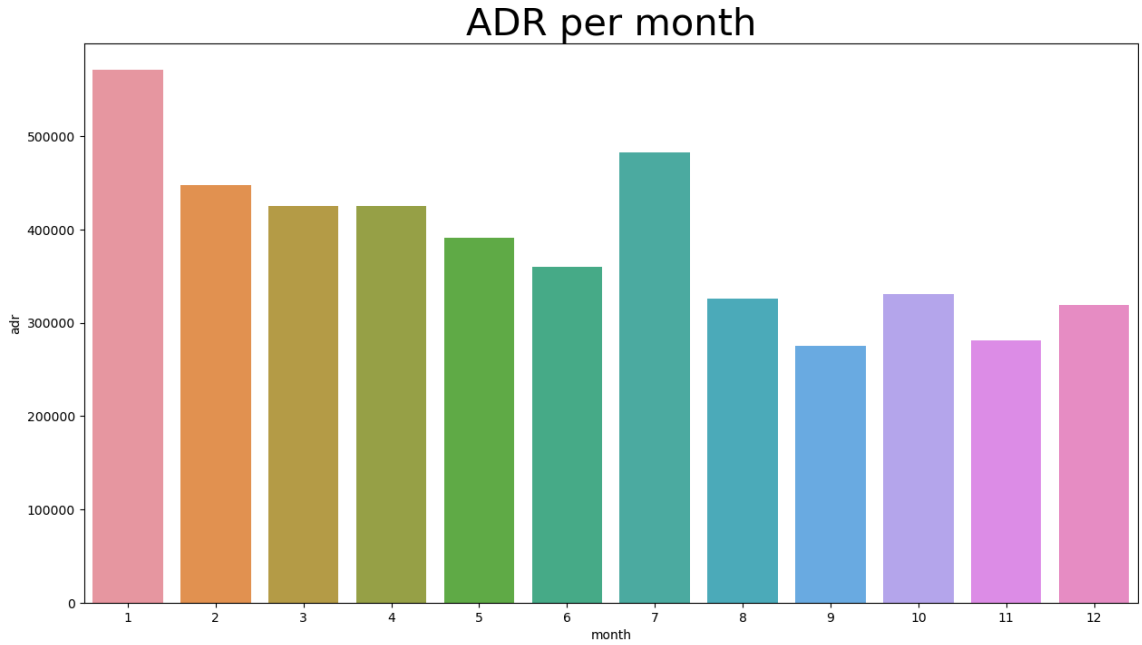
- The average daily rate (ADR) for both city and resort hotels fluctuates throughout the year.
- The line graph above shows that, on certain days, the average daily rate for a city hotel is less than that of a resort hotel, and on other days, it is even less. It goes without saying that weekends and holidays may see a rise in resort hotel rates.

In [22]:
```
Hotel_Booking['month'] = Hotel_Booking['reservation_status_date'].dt.month
plt.figure(figsize = (16,8))
ax1 = sns.countplot(x = 'month', hue = 'is_canceled', data = Hotel_Booking,
for bars in ax1.containers:
    ax1.bar_label(bars)
legend_labels,_ = ax1. get_legend_handles_labels()
ax1.legend(bbox_to_anchor=(1,1))
plt.title('Reservation status per month', size = 20)
plt.xlabel('month')
plt.ylabel('number of reservations')
plt.legend(['not canceled', 'canceled'])
plt.show()
```



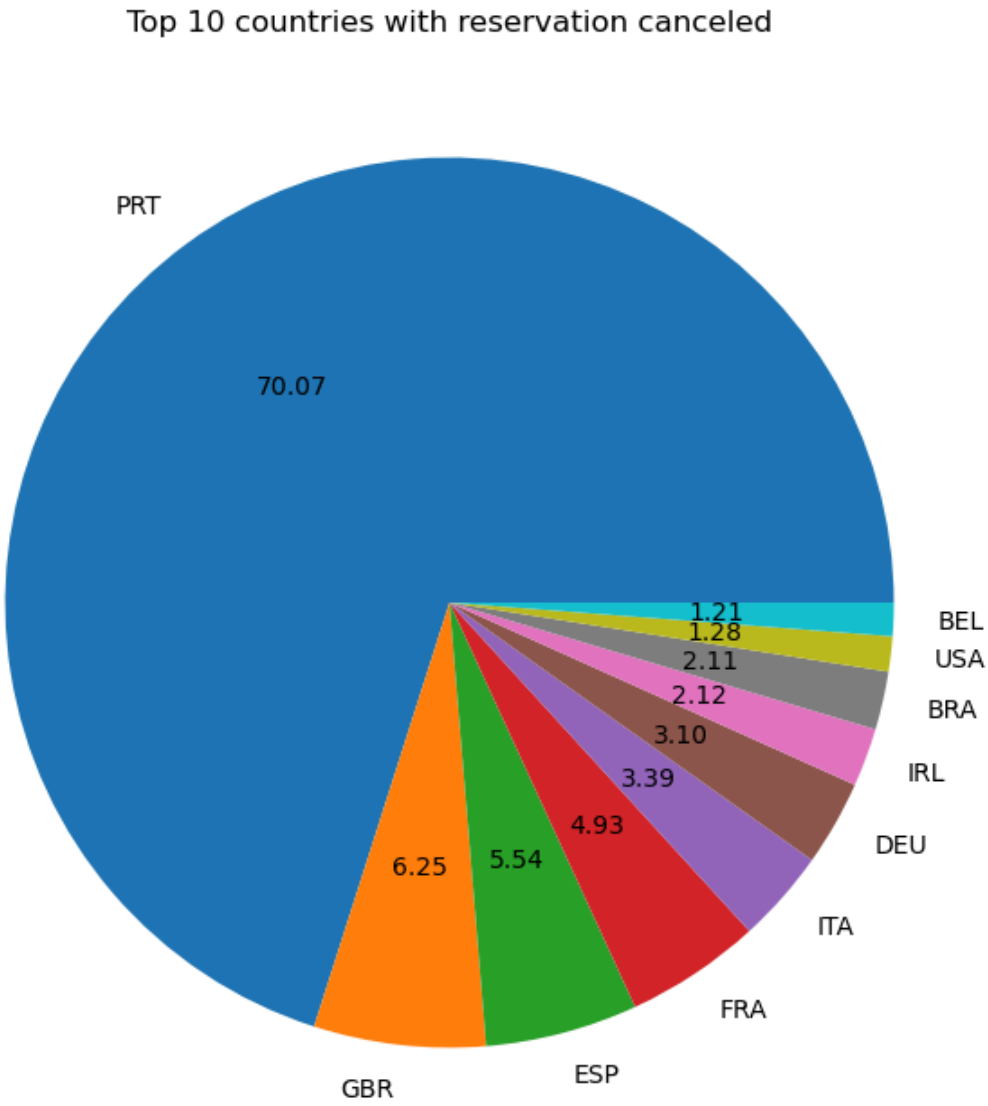
- As can be seen, both the number of confirmed reservations and the number of canceled reservations are largest in the month of August. whereas January is the month with the most canceled reservations.

In [23]:
```
plt.figure(figsize=(15, 8))
plt.title('ADR per month', fontsize=30)
ax2=sns.barplot(x='month', y='adr', data=Hotel_Booking[Hotel_Booking['is_ca
plt.show()
```



- This bar graph demonstrates that cancellations are most common when prices are greatest and are least common when they are lowest. Therefore, the cost of the accommodation is solely responsible for the cancellation.

In [24]:
```
cancelled_data = Hotel_Booking[Hotel_Booking['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()
```



- The top country is Portugal with the highest number of cancellations 70.07% .

In [25]:
```
Hotel_Booking['market_segment'].value_counts()
```

Out[25]:
```
market_segment
Online TA        56476
Offline TA/TO    24218
Groups           19811
Direct           12605
Corporate         5295
Complementary      743
Aviation           237
Name: count, dtype: int64
```

- online Travel agent are Highest number of booking (49834)
- second Highest is offline travel agent 13838

In [26]:
```
Hotel_Booking['market_segment'].value_counts(normalize = True)
```

Out[26]:
```
market_segment
Online TA        0.473058
Offline TA/TO    0.202856
Groups           0.165942
Direct           0.105583
Corporate        0.044352
Complementary    0.006224
Aviation         0.001985
Name: proportion, dtype: float64
```

- In overall percentage 58.70 % Hotel booking through online travel agent

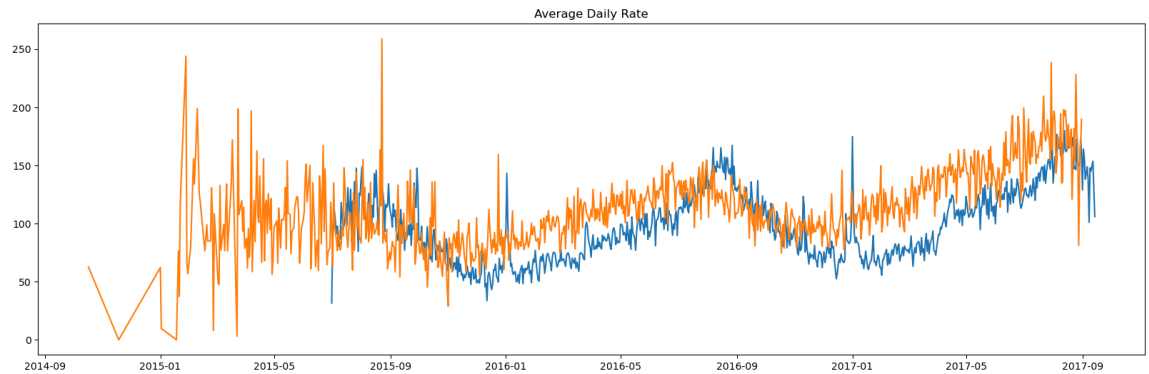
In [27]:
```
cancelled_data['market_segment'].value_counts(normalize = True)
```

Out[27]:
```
market_segment
Online TA        0.468984
Groups           0.273570
Offline TA/TO    0.187928
Direct           0.043714
Corporate        0.022434
Complementary    0.002194
Aviation         0.001176
Name: proportion, dtype: float64
```

- In Hotel Booking Cancelation is high online travel agent

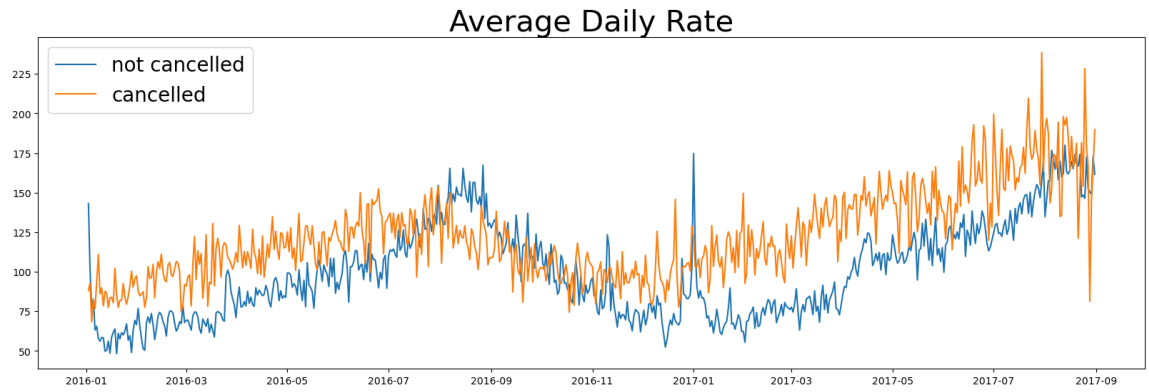
In [28]:
```
cancelled_df_adr = cancelled_data.groupby('reservation_status_date')[['adr'
cancelled_df_adr.reset_index(inplace = True)
cancelled_df_adr.sort_values('reservation_status_date', inplace = True)

not_cancelled_data = Hotel_Booking[Hotel_Booking['is_canceled'] == 0]
not_cancelled_df_adr = not_cancelled_data.groupby('reservation_status_date'
not_cancelled_df_adr.reset_index(inplace = True)
not_cancelled_df_adr.sort_values('reservation_status_date', inplace = True)

plt.figure(figsize = (20,6))
plt.title('Average Daily Rate')
plt.plot(not_cancelled_df_adr['reservation_status_date'],not_cancelled_df_a
plt.plot(cancelled_df_adr['reservation_status_date'],cancelled_df_adr['adr'
```

Out[28]:
```
[<matplotlib.lines.Line2D at 0x24096c5c310>]
```



In [29]:
```
cancelled_df_adr = cancelled_df_adr[(cancelled_df_adr['reservation_status_da
not_cancelled_df_adr = not_cancelled_df_adr[(not_cancelled_df_adr['reservat
```

In [30]:
```
plt.figure(figsize = (20,6))
plt.title('Average Daily Rate', fontsize = 30)
plt.plot(not_cancelled_df_adr['reservation_status_date'],not_cancelled_df_a
plt.plot(cancelled_df_adr['reservation_status_date'],cancelled_df_adr['adr'
plt.legend(fontsize = 20)
plt.show()
```



- reservations are canceled when the average daily rate is higher than when it is not canceled. It clearly proves all the above analysis, that the higher price leads to higher cancellation.

Type *Markdown* and LaTeX: α^2

In [ ]:

In [ ]: