Hotel booking EDA Project

Bussiness Problem

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

Importing Libraries

In [3]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings('ignore')
```

Loading the Dataset

In [4]:
```
df=pd.read_csv('hotel_bookings 2.csv')
```

Exploratory Data Analysis and Data Cleaning

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

Out[5]:

	hotel	is_canceled	lead_time	arrival_date_year	arrival_date_month	arriva
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 [6]:
```
df.tail()
```

Out[6]:

	hotel	is_canceled	lead_time	arrival_date_year	arrival_date_month
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

5 rows × 32 columns

In [7]:
```
print("Number of Rows:",df.shape[0])
print("Number of Columns:",df.shape[1])
```

```
Number of Rows: 119390
Number of Columns: 32
```

In [8]:
```
df.columns
```

Out[8]:
```
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 [9]:
```
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
```

In [10]:
```
# Assuming 'df' is your DataFrame
df['reservation_status_date'] = pd.to_datetime(df['reservation_status_date']

# Verify the changes
print(df.dtypes)
```

```
hotel                              object
is_canceled                         int64
lead_time                           int64
arrival_date_year                   int64
arrival_date_month                 object
arrival_date_week_number            int64
arrival_date_day_of_month           int64
stays_in_weekend_nights             int64
stays_in_week_nights                int64
adults                              int64
children                          float64
babies                              int64
meal                               object
country                            object
market_segment                     object
distribution_channel               object
is_repeated_guest                   int64
previous_cancellations              int64
previous_bookings_not_canceled      int64
reserved_room_type                 object
assigned_room_type                 object
booking_changes                     int64
deposit_type                       object
agent                             float64
company                           float64
days_in_waiting_list                int64
customer_type                      object
adr                               float64
required_car_parking_spaces         int64
total_of_special_requests           int64
reservation_status                 object
reservation_status_date     datetime64[ns]
dtype: object
```

In [11]:
```
df.describe(include='object')
```

Out[11]:

	hotel	arrival_date_month	meal	country	market_segment	distrib
count	119390	119390	119390	118902	119390	
unique	2	12	5	177	8	
top	City Hotel	August	BB	PRT	Online TA	
freq	79330	13877	92310	48590	56477	

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

```
    print(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']
--------------------------------------------------
deposit_type
['No Deposit' 'Refundable' 'Non Refund']
--------------------------------------------------
customer_type
['Transient' 'Contract' 'Transient-Party' 'Group']
--------------------------------------------------
reservation_status
['Check-Out' 'Canceled' 'No-Show']
--------------------------------------------------
```

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

Out[13]:
```
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 [14]:
```
df.drop(['company','agent'],axis=1,inplace=True)
df.dropna(inplace=True)
```

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

Out[15]:
```
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 [16]:
```
df.describe()
```

Out[16]:

	is_canceled	lead_time	arrival_date_year	arrival_date_week_num
count	118898.000000	118898.000000	118898.000000	118898.000
mean	0.371352	104.311435	2016.157656	27.166
min	0.000000	0.000000	2015.000000	1.000
25%	0.000000	18.000000	2016.000000	16.000
50%	0.000000	69.000000	2016.000000	28.000
75%	1.000000	161.000000	2017.000000	38.000
max	1.000000	737.000000	2017.000000	53.000
std	0.483168	106.903309	0.707459	13.589

In [17]:
```
df=df[df['adr']<5000]
```

In [18]:
```
df.describe()
```

Out[18]:

	is_canceled	lead_time	arrival_date_year	arrival_date_week_num
count	118897.000000	118897.000000	118897.000000	118897.000
mean	0.371347	104.312018	2016.157657	27.166
min	0.000000	0.000000	2015.000000	1.000
25%	0.000000	18.000000	2016.000000	16.000
50%	0.000000	69.000000	2016.000000	28.000
75%	1.000000	161.000000	2017.000000	38.000
max	1.000000	737.000000	2017.000000	53.000
std	0.483167	106.903570	0.707462	13.589

Data Analysis and Visualisation

In [34]:
```
df.columns
```

Out[34]:
```
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',
       'days_in_waiting_list', 'customer_type', 'adr',
       'required_car_parking_spaces', 'total_of_special_requests',
       'reservation_status', 'reservation_status_date', 'month'],
      dtype='object')
```

In [19]:
```
cancelled_perc = df['is_canceled'].value_counts(normalize=True)
```

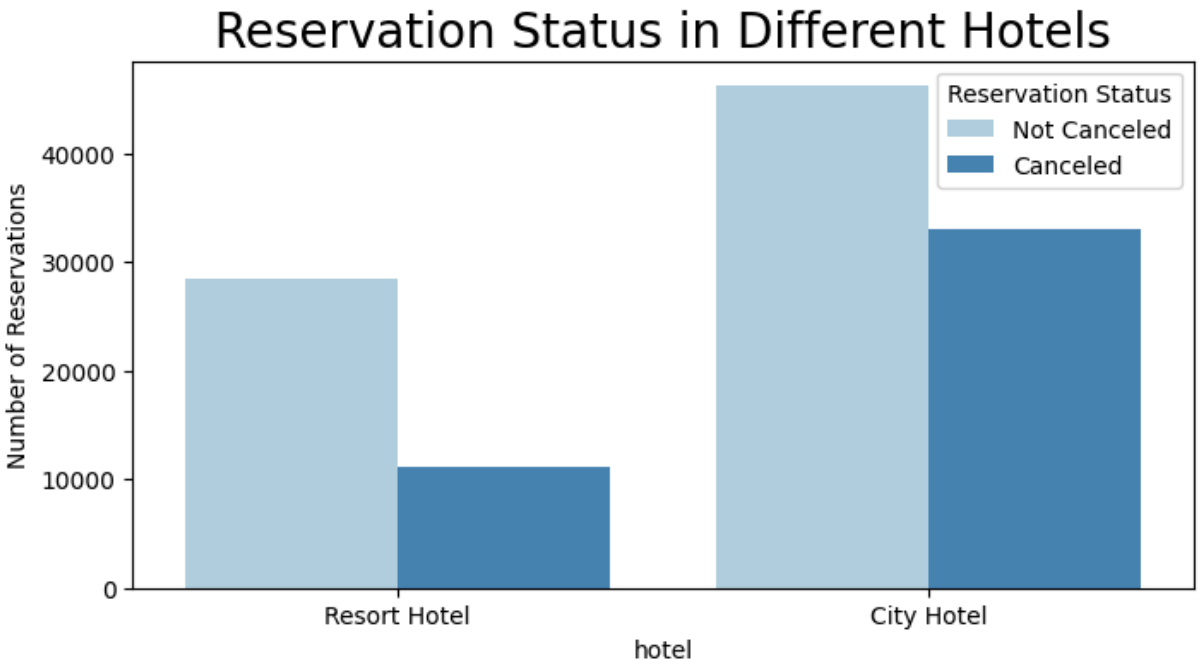
In [20]:
```
print(cancelled_perc)
```

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

In [21]:
```
plt.figure(figsize=(5,4))
plt.title('Reservation status count')
plt.bar(['Not canceled','Canceled'],df['is_canceled'].value_counts(),color='
plt.show()
```

The accompanying bar graph shows the percentage of reservations that are canceled and those that are not. there are still a significant number of reservations that have not been canceled. There are still 37% of clients who canceled their reservations, which has a significant impact on the hotels's earnings.

In [22]:
```
# Assuming 'df' is your DataFrame
plt.figure(figsize=(8, 4))
ax1 = sns.countplot(x='hotel', hue='is_canceled', data=df, palette='Blues')
ax1.legend(title='Reservation Status', loc='upper right', labels=['Not Cance

plt.title('Reservation Status in Different Hotels', size=20)
plt.xlabel('hotel')
plt.ylabel('Number of Reservations')

plt.show()
```



In comparision to resort hotels, city hotels have more bookings. its possible that resort hotels are more expensive than those in cities.

In [23]:
```
resort_hotel = df[df['hotel']=='Resort Hotel']
cancellation_distribution =resort_hotel['is_canceled'].value_counts(normaliz
print(cancellation_distribution)
```

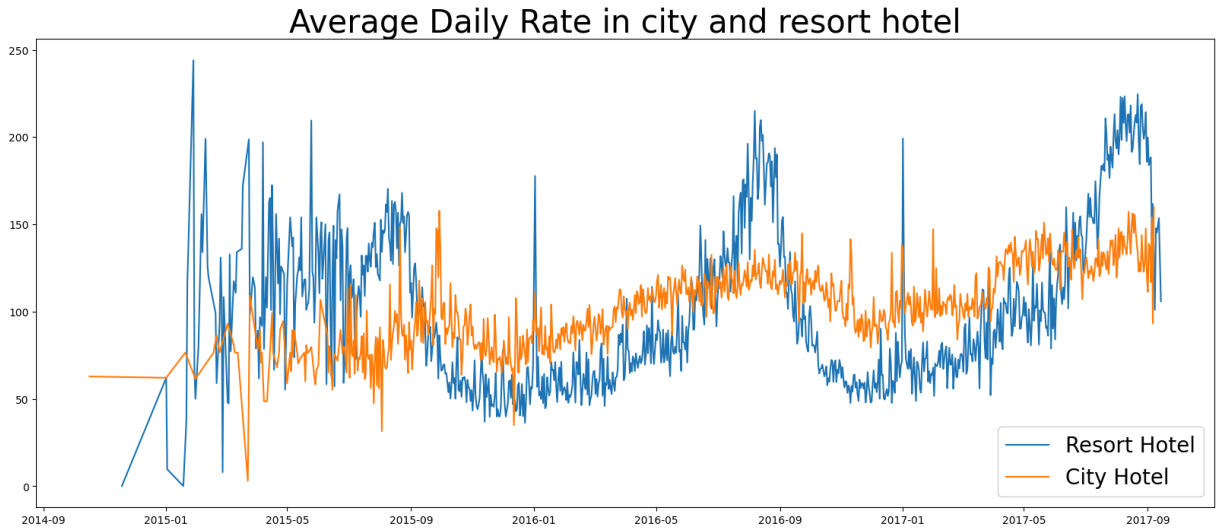
```
is_canceled
0    0.72025
1    0.27975
Name: proportion, dtype: float64
```

In [24]:
```
city_hotel = df[df['hotel'] == 'City Hotel']
cancellation_distribution = city_hotel['is_canceled'].value_counts(normalize

print(cancellation_distribution)
```

```
is_canceled
0    0.582918
1    0.417082
Name: proportion, dtype: float64
```

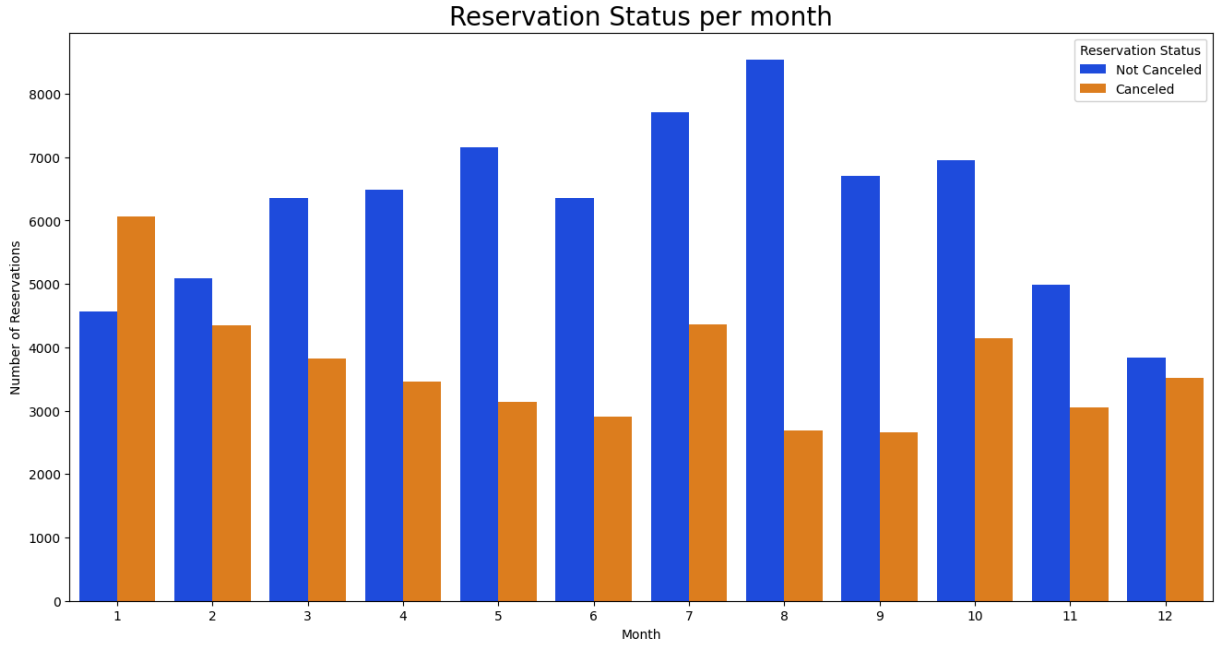
In [32]:
```
resort_hotel = resort_hotel.groupby('reservation_status_date')[['adr']].mean
city_hotel = city_hotel.groupby('reservation_status_date')[['adr']].mean()
```

In [33]:
```
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 line graph above shows that, on certain days, the average daily rate for the 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.

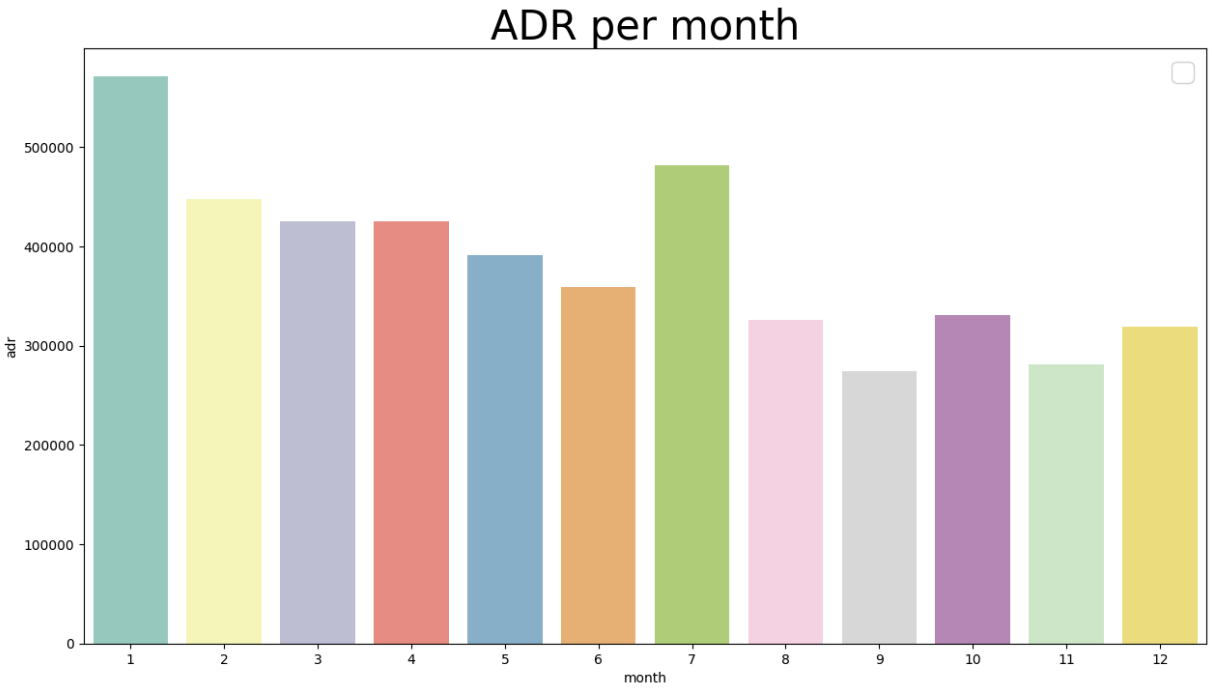
In [25]:
```
df['month']=df['reservation_status_date'].dt.month
plt.figure(figsize=(16,8))
ax1=sns.countplot(x='month',hue='is_canceled',data=df,palette='bright')
ax1.legend(title='Reservation Status', loc='upper right', labels=['Not Cance
plt.title('Reservation Status per month', size=20)
plt.xlabel('Month')
plt.ylabel('Number of Reservations')

plt.show()
```



We have developed the grouped bar graph to analyze the months with the highest and lowest reservation levels according to reservation status. as can be seen, both the number of comfirmed reservations and the number of canceled reservations are largest in the month of august. where as january is the month with the most canceled reservations

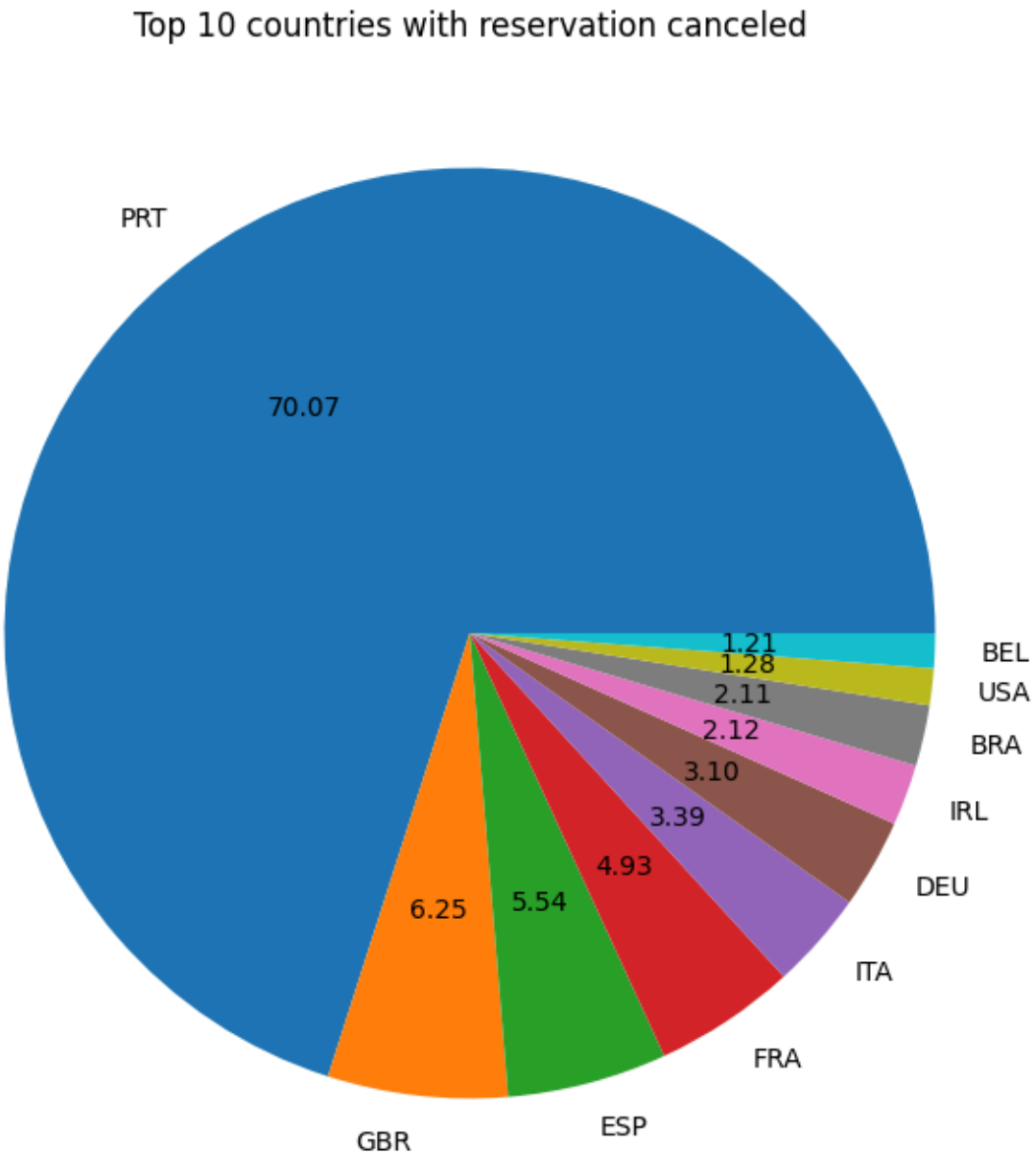
In [26]:
```
# Assuming 'df' is your DataFrame
df['reservation_status_date'] = pd.to_datetime(df['reservation_status_date']
df['month'] = df['reservation_status_date'].dt.month

plt.figure(figsize=(15, 8))
plt.title('ADR per month', fontsize=30)

# Using a Seaborn color palette ('Set3') for different colors
sns.barplot(x='month', y='adr', data=df[df['is_canceled'] == 1].groupby('mor

plt.legend(fontsize=20)
plt.show()
```

```
No artists with labels found to put in legend.  Note that artists whose labe
l start with an underscore are ignored when legend() is called with no argum
ent.
```



The bar graph demostrates that cancellations are most common when prices are greatest and are least common when prices are greatest and are least common when they are lowest. therefore, the cost of the accommodation is solely responsible for the canceelation.

In [27]:
```
cancelled_data=df[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()
```



The top country is portugal with the highest number of cancellations.

In [28]:
```
df['market_segment'].value_counts()
```

Out[28]:
```
market_segment
Online TA        56402
Offline TA/TO    24159
Groups           19806
Direct           12448
Corporate         5111
Complementary      734
Aviation           237
Name: count, dtype: int64
```

In [29]:
```
df['market_segment'].value_counts(normalize=True)
```

Out[29]:
```
market_segment
Online TA        0.474377
Offline TA/TO    0.203193
Groups           0.166581
Direct           0.104696
Corporate        0.042987
Complementary    0.006173
Aviation         0.001993
Name: proportion, dtype: float64
```

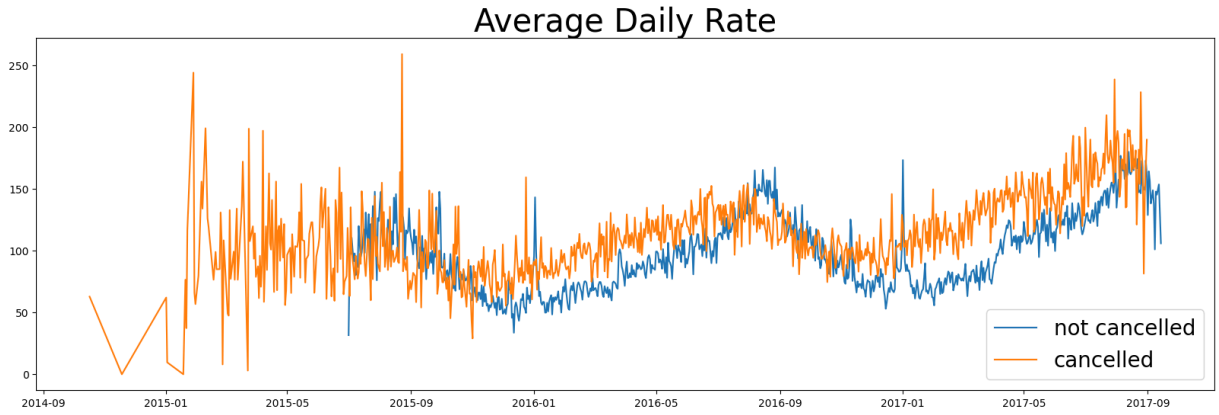
In [30]:
```
cancelled_data['market_segment'].value_counts(normalize=True)
```

Out[30]:
```
market_segment
Online TA        0.469696
Groups           0.273985
Offline TA/TO    0.187466
Direct           0.043486
Corporate        0.022151
Complementary    0.002038
Aviation         0.001178
Name: proportion, dtype: float64
```

Above the analysis , we found guests are visiting the hotels and making reservations. Around 46% of the clients come from travel agencies, where as 27% come from groups. only 4% of clients book hotels directly by visiting them and making reservations.

In [31]:
```
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 = df[df['is_canceled']==0]
not_cancelled_df_adr = not_cancelled_data.groupby('reservation_status_date')
not_cancelled_df_adr.reset_index(inplace=True)
lled_df_adr.sort_values('reservation_status_date',inplace = True)
```

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



As seen in the graph, 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.

Suggestions

1.Cancellation rates rise as the price does. in order to prevent cancellations of reservations, hotels could work on their pricing strategies and try to lower the rate of specific hotels based on locations. they can also provide some discounts to the consumers.

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

3. In the month of january, hotel can start compaigns or marketing with a reasonable amount to increase their revenue as the cancellation 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 [illegible]ation rate.

In [ ]: