Untitled4

February 6, 2024

1 Hotel Bookings EDA

Introduction: The hotel_bookings.csv file provides a comprehensive snapshot of hotel reservations, encompassing guest demographics, booking details, and reservation statuses. With variables ranging from the type of hotel to the average daily rate and reservation status, this dataset offers a wealth of insights into booking patterns and guest behavior within the hospitality industry. Through exploratory analysis, we aim to uncover key trends and patterns that can inform strategic decision-making in the dynamic world of hotel management.

step 0 : imports and reading data :

import packages :

[82]:
```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import seaborn as sns
```

load dataset :

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

step 1 : data understanding :

[46]:
```
# Increase the maximum number of displayed columns to 200 for better visibility

pd.set_option('display.max_columns', 200)
```

[47]:
```
# Display first few rows of the dataframe

df.head()
```

[47]:
```
   index         hotel  is_canceled  lead_time  arrival_date_year  \
0      0  Resort Hotel            0        342               2015
1      1  Resort Hotel            0        737               2015
2      2  Resort Hotel            0          7               2015
3      3  Resort Hotel            0         13               2015
4      4  Resort Hotel            0         14               2015

  arrival_date_month  arrival_date_week_number  arrival_date_day_of_month  \
0               July                        27                           1
1               July                        27                           1
2               July                        27                           1
3               July                        27                           1
4               July                        27                           1

   stays_in_weekend_nights  stays_in_week_nights  adults  children  babies  \
0                        0                     0       2       0.0       0
1                        0                     0       2       0.0       0
2                        0                     1       1       0.0       0
3                        0                     1       1       0.0       0
4                        0                     2       2       0.0       0

  meal country market_segment distribution_channel  is_repeated_guest  \
0   BB     PRT         Direct               Direct                  0
1   BB     PRT         Direct               Direct                  0
2   BB     GBR         Direct               Direct                  0
3   BB     GBR      Corporate            Corporate                  0
4   BB     GBR      Online TA                TA/TO                  0

   previous_cancellations  previous_bookings_not_canceled reserved_room_type  \
0                       0                               0                  C
1                       0                               0                  C
2                       0                               0                  A
3                       0                               0                  A
4                       0                               0                  A

  assigned_room_type  booking_changes deposit_type  agent  company  \
0                  C                3   No Deposit    NaN      NaN
1                  C                4   No Deposit    NaN      NaN
2                  C                0   No Deposit    NaN      NaN
3                  A                0   No Deposit  304.0      NaN
4                  A                0   No Deposit  240.0      NaN

   days_in_waiting_list customer_type   adr  required_car_parking_spaces  \
0                     0     Transient   0.0                            0
1                     0     Transient   0.0                            0
2                     0     Transient  75.0                            0
3                     0     Transient  75.0                            0
4                     0     Transient  98.0                            0

   total_of_special_requests reservation_status reservation_status_date
0                          0          Check-Out                01-07-15
1                          0          Check-Out                01-07-15
2                          0          Check-Out                02-07-15
3                          0          Check-Out                02-07-15
4                          1          Check-Out                03-07-15
```

[48]:

```
#get the shape of the dataframe (rows and columns)

df.shape
```

[48]:

```
(119390, 33)
```

[49]:

```
#gather basic information about the data

df.info()
```

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

```
dtypes: float64(4), int64(17), object(12)
memory usage: 30.1+ MB
```

```
[50]: #gather descriptive statistics about the data

      df.describe()
```

```
[50]:              index    is_canceled      lead_time  arrival_date_year  \
      count  119390.000000  119390.000000  119390.000000      119390.000000
      mean    59694.500000       0.370416     104.011416        2016.156554
      std     34465.068657       0.482918     106.863097           0.707476
      min         0.000000       0.000000       0.000000        2015.000000
      25%     29847.250000       0.000000      18.000000        2016.000000
      50%     59694.500000       0.000000      69.000000        2016.000000
      75%     89541.750000       1.000000     160.000000        2017.000000
      max    119389.000000       1.000000     737.000000        2017.000000

             arrival_date_week_number  arrival_date_day_of_month  \
      count             119390.000000              119390.000000
      mean                  27.165173                  15.798241
      std                   13.605138                   8.780829
      min                    1.000000                   1.000000
      25%                   16.000000                   8.000000
      50%                   28.000000                  16.000000
      75%                   38.000000                  23.000000
      max                   53.000000                  31.000000

             stays_in_weekend_nights  stays_in_week_nights         adults  \
      count            119390.000000         119390.000000  119390.000000
      mean                  0.927599              2.500302       1.856403
      std                   0.998613              1.908286       0.579261
      min                   0.000000              0.000000       0.000000
      25%                   0.000000              1.000000       2.000000
      50%                   1.000000              2.000000       2.000000
      75%                   2.000000              3.000000       2.000000
      max                  19.000000             50.000000      55.000000

                  children         babies  is_repeated_guest  \
      count  119386.000000  119390.000000      119390.000000
      mean        0.103890       0.007949           0.031912
      std         0.398561       0.097436           0.175767
      min         0.000000       0.000000           0.000000
      25%         0.000000       0.000000           0.000000
      50%         0.000000       0.000000           0.000000
      75%         0.000000       0.000000           0.000000
      max        10.000000      10.000000           1.000000
```

```
       previous_cancellations  previous_bookings_not_canceled  \
count           119390.000000                   119390.000000
mean                 0.087118                        0.137097
std                  0.844336                        1.497437
min                  0.000000                        0.000000
25%                  0.000000                        0.000000
50%                  0.000000                        0.000000
75%                  0.000000                        0.000000
max                 26.000000                       72.000000

       booking_changes          agent      company  days_in_waiting_list  \
count    119390.000000  103050.000000  6797.000000         119390.000000
mean          0.221124      86.693382   189.266735              2.321149
std           0.652306     110.774548   131.655015             17.594721
min           0.000000       1.000000     6.000000              0.000000
25%           0.000000       9.000000    62.000000              0.000000
50%           0.000000      14.000000   179.000000              0.000000
75%           0.000000     229.000000   270.000000              0.000000
max          21.000000     535.000000   543.000000            391.000000

                 adr  required_car_parking_spaces  total_of_special_requests
count  119390.000000                119390.000000              119390.000000
mean      101.831122                     0.062518                   0.571363
std        50.535790                     0.245291                   0.792798
min        -6.380000                     0.000000                   0.000000
25%        69.290000                     0.000000                   0.000000
50%        94.575000                     0.000000                   0.000000
75%       126.000000                     0.000000                   1.000000
max      5400.000000                     8.000000                   5.000000
```

[51]:
```
#display all column name
df.columns
```

[51]:
```
Index(['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')
```

[52]:

```
#display the data types of columns in the DataFrame

df.dtypes
```

[52]:

```
index                              int64
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           object
dtype: object
```

step 2 : Data Analysis and Visualization :

[55]:

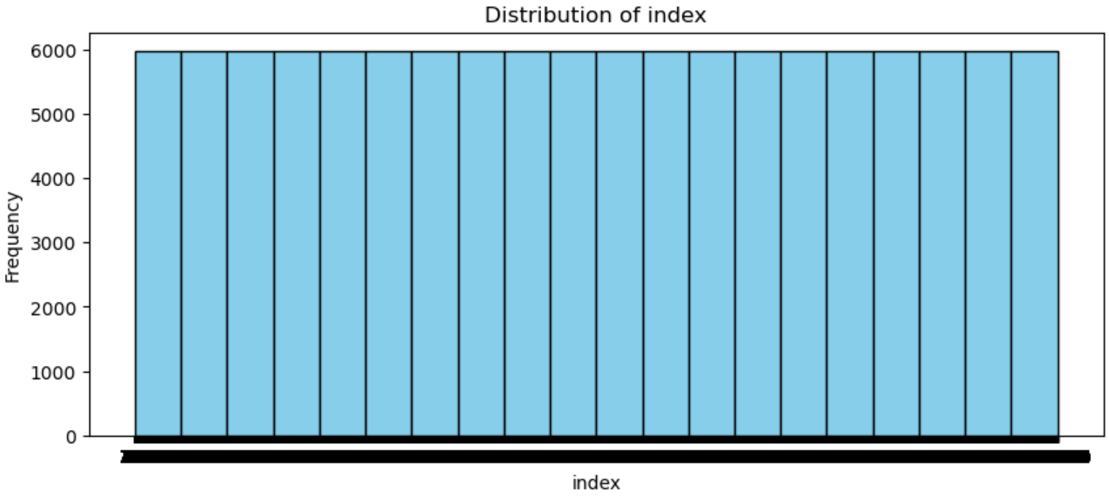
```
#calculat the percentage of cancelled and not cancelled bookings

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

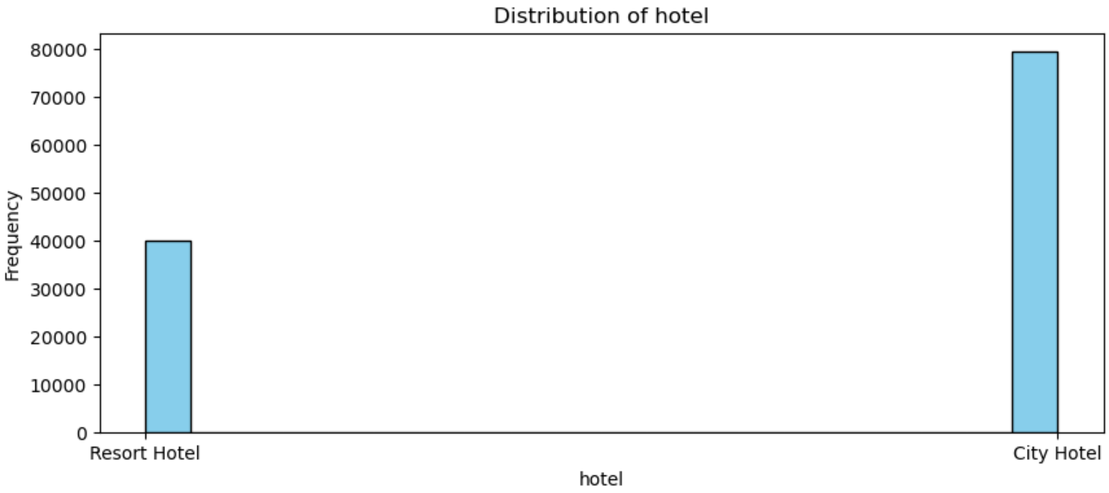
[55]:

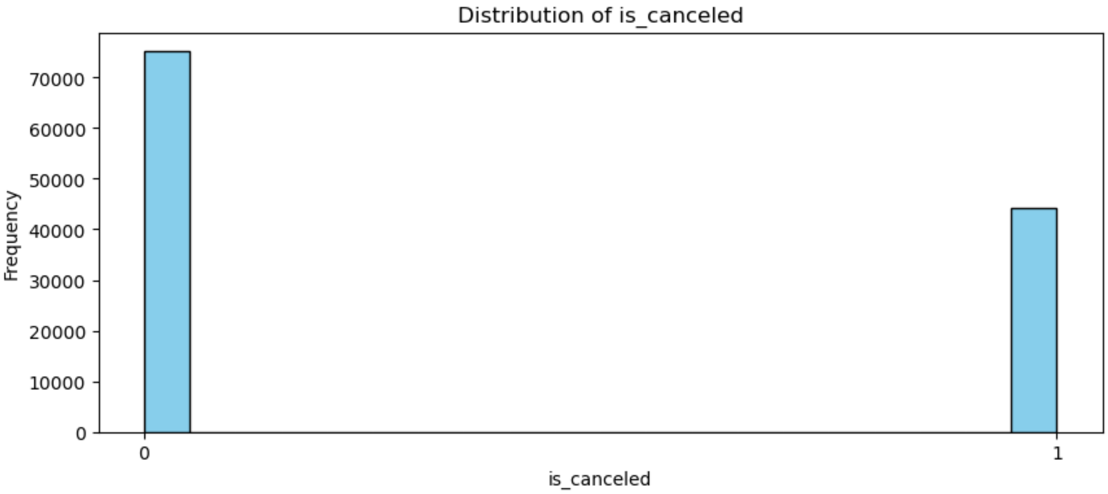
```
is_canceled
0    0.629589
```

```
1    0.370411
Name: proportion, dtype: float64
```

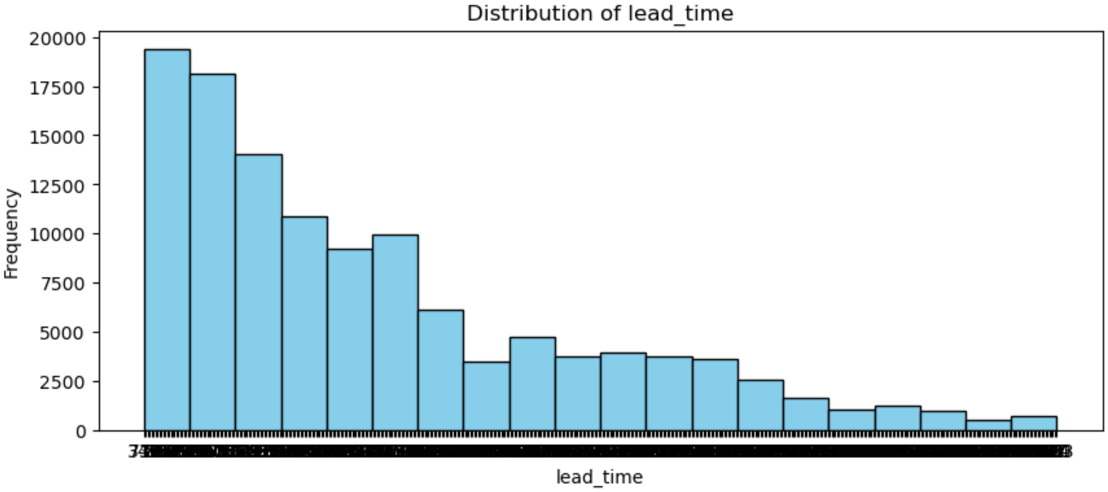
[ ]:
```
#creating histograms to visualize the distribution of values. Each histogram is␣
↪labeled with the column name and displays the frequency of values.

for col in df.columns:
    df[col] = df[col].astype(str)
    plt.figure(figsize=(10, 4))
    plt.hist(df[col], bins=20, color='skyblue', edgecolor='black')
    plt.title(f'Distribution of {col}')
    plt.xlabel(col)
    plt.ylabel('Frequency')
    plt.show()
```



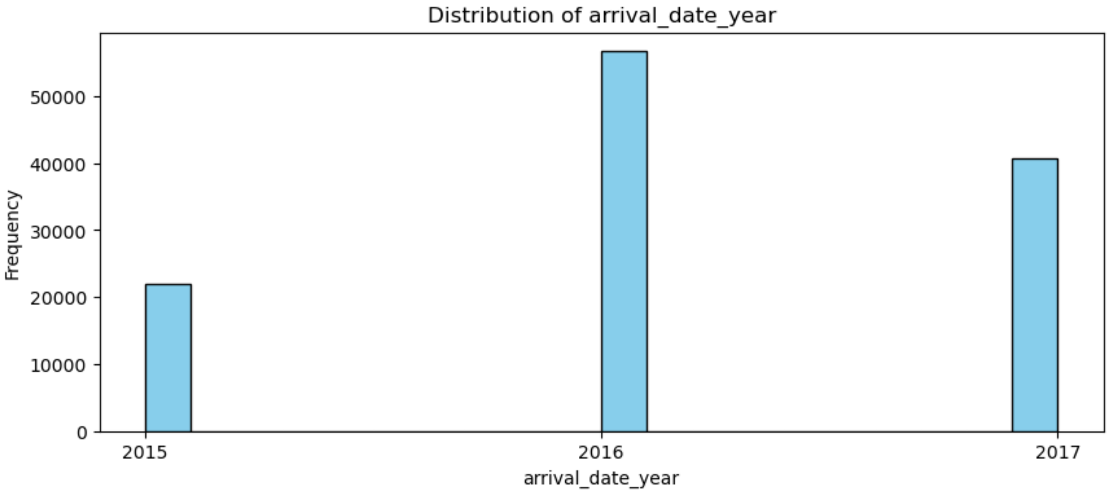


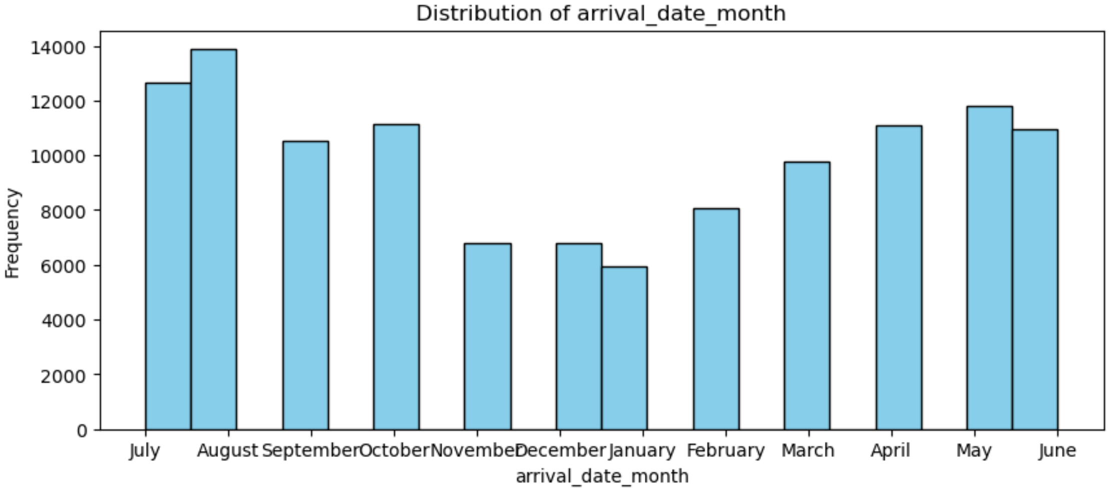


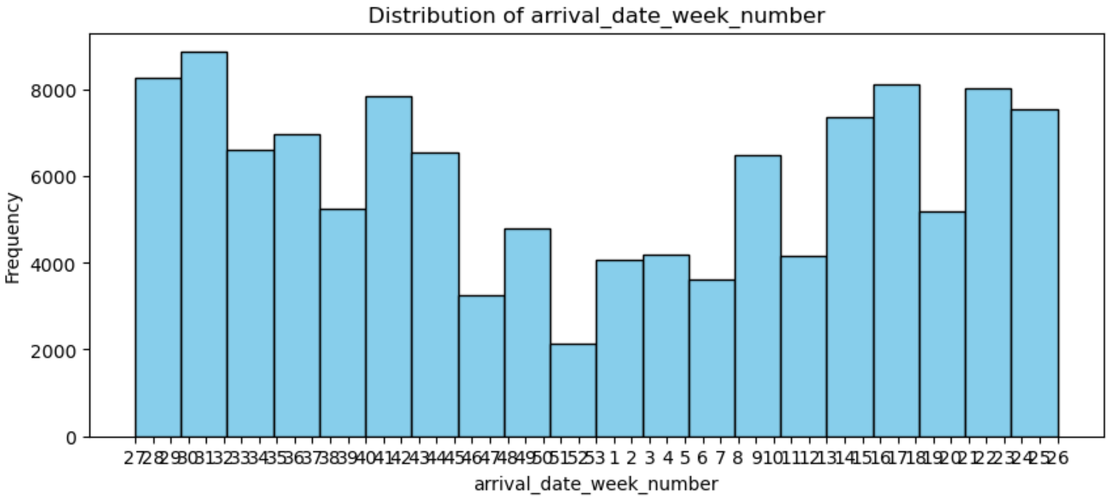
Distribution of is_canceled



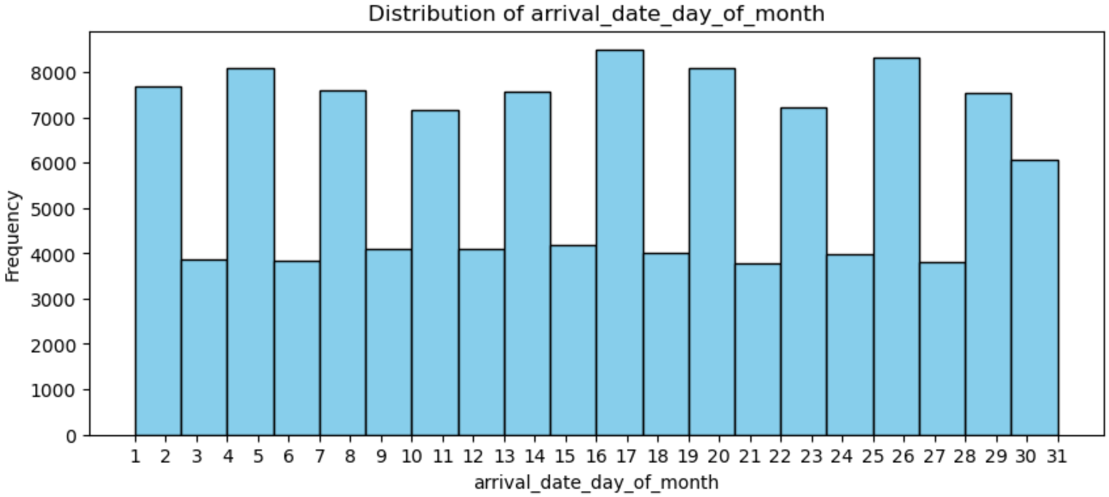
Distribution of lead_time


Distribution of arrival_date_year
Frequency
0
10000
20000
30000
40000
50000
2015
2016
2017
arrival_date_year

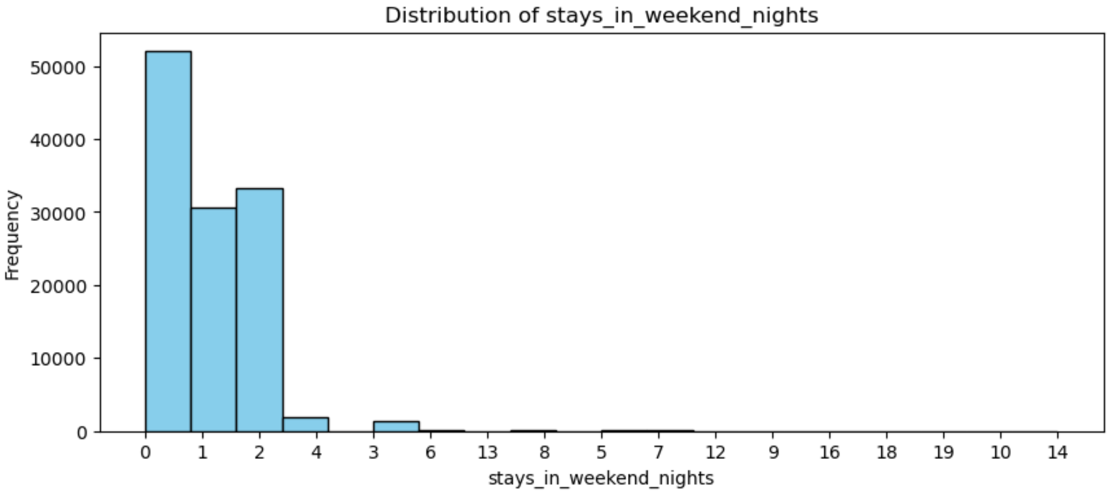

Distribution of arrival_date_month
Frequency
0
2000
4000
6000
8000
10000
12000
14000
July
August
September
October
November
December
January
February
March
April
May
June
arrival_date_month

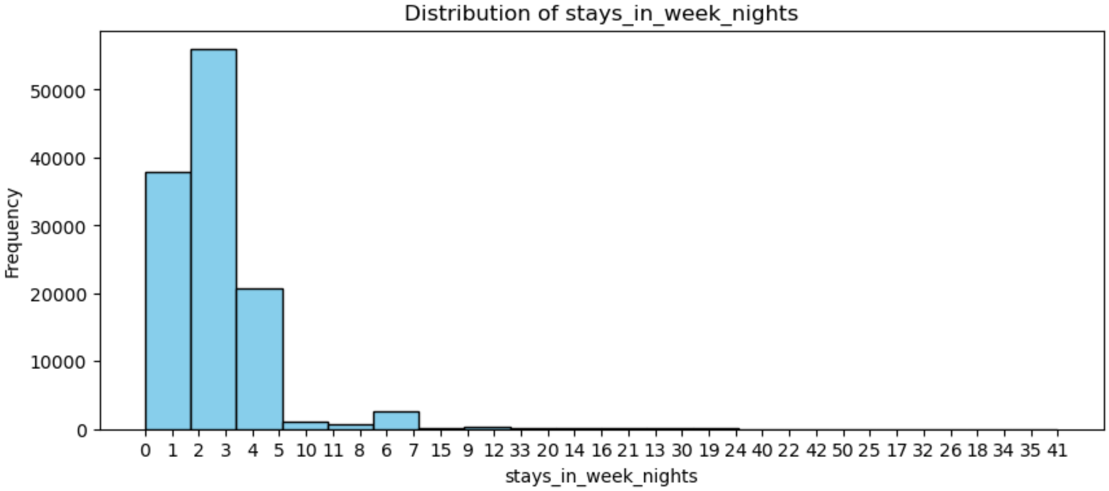


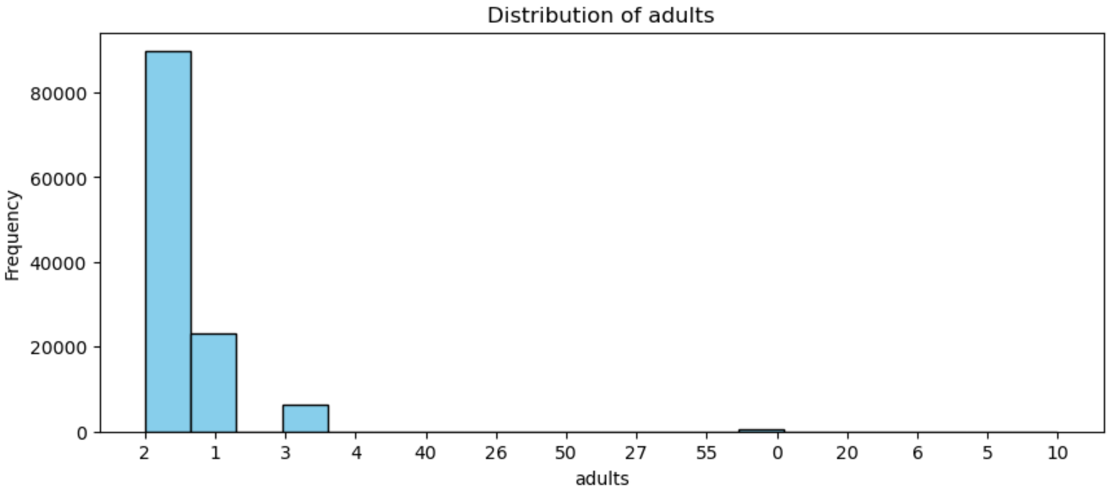
Distribution of arrival_date_week_number
Frequency
0
2000
4000
6000
8000
27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26
arrival_date_week_number

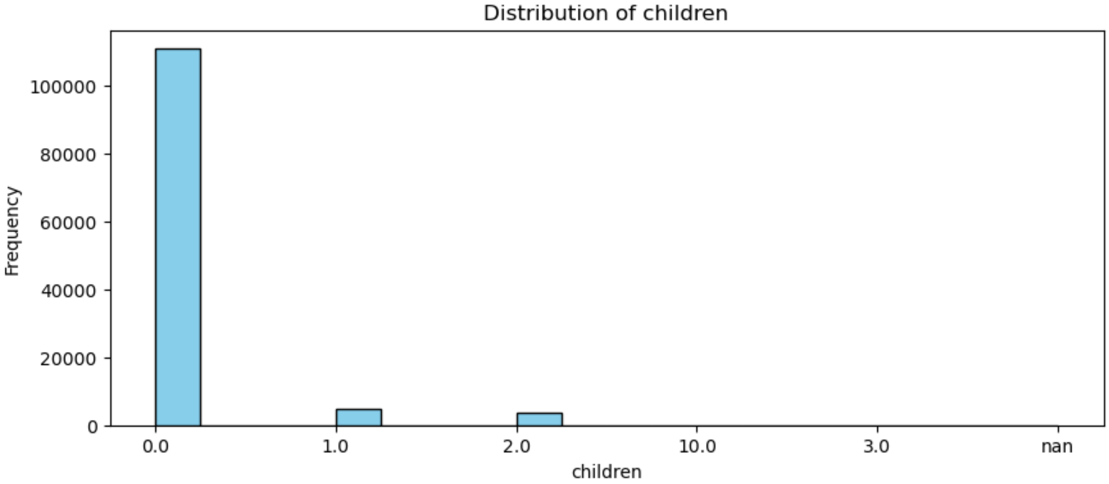


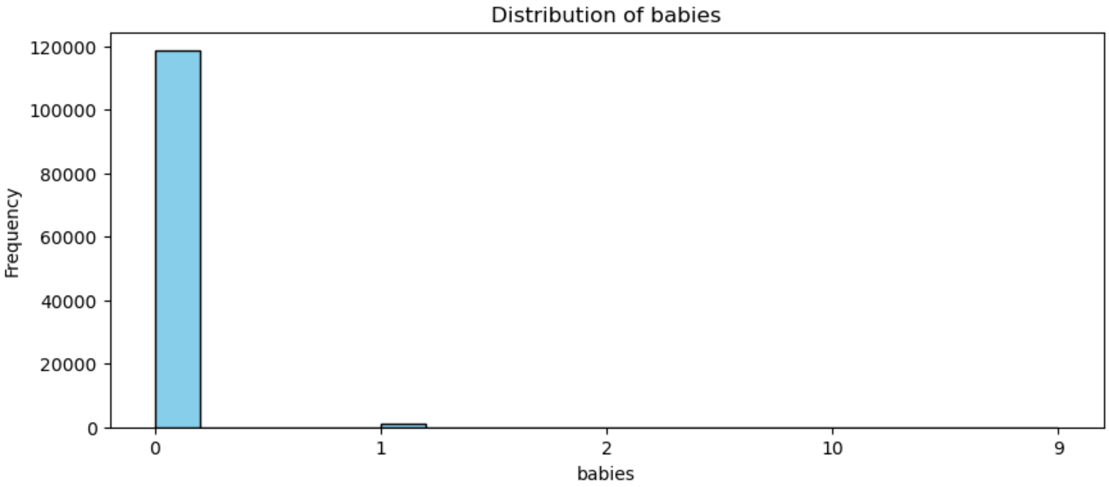
Distribution of arrival_date_day_of_month
Frequency
0
1000
2000
3000
4000
5000
6000
7000
8000
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
arrival_date_day_of_month

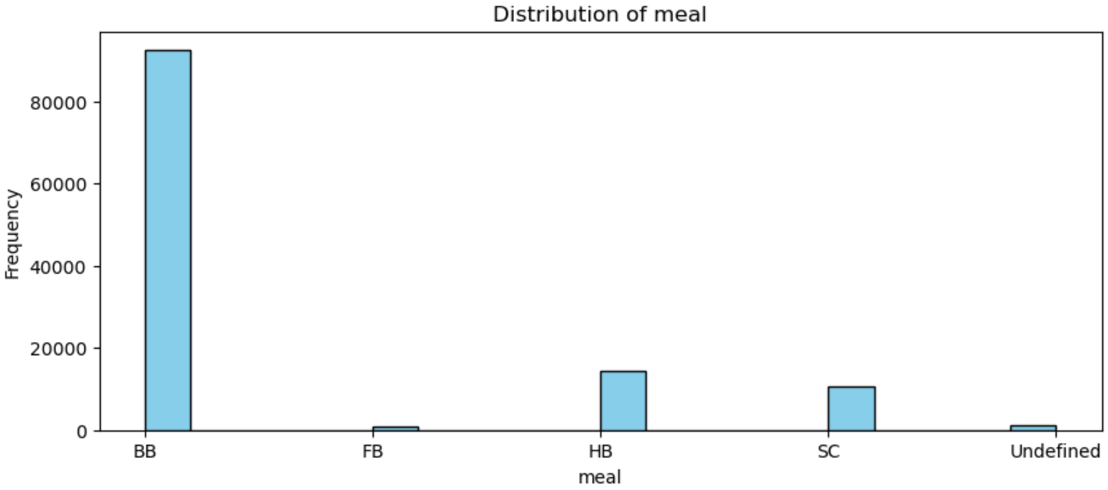

Distribution of stays_in_weekend_nights
Frequency
0
10000
20000
30000
40000
50000
0
1
2
4
3
6
13
8
5
7
12
9
16
18
19
10
14
stays_in_weekend_nights

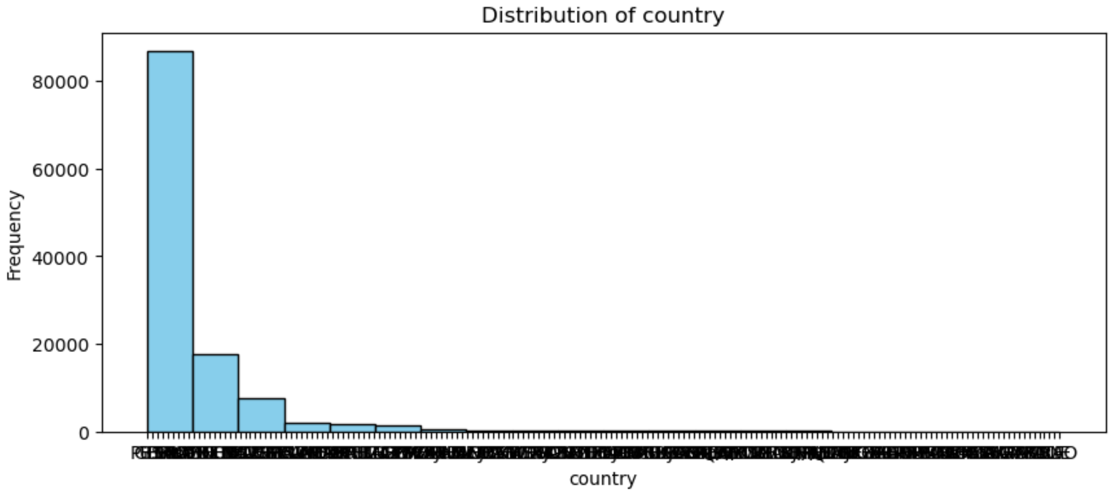

Distribution of stays_in_week_nights
Frequency
0
10000
20000
30000
40000
50000
0 1 2 3 4 5 10 11 8 6 7 15 9 12 33 20 14 16 21 13 30 19 24 40 22 42 50 25 17 32 26 18 34 35 41
stays_in_week_nights

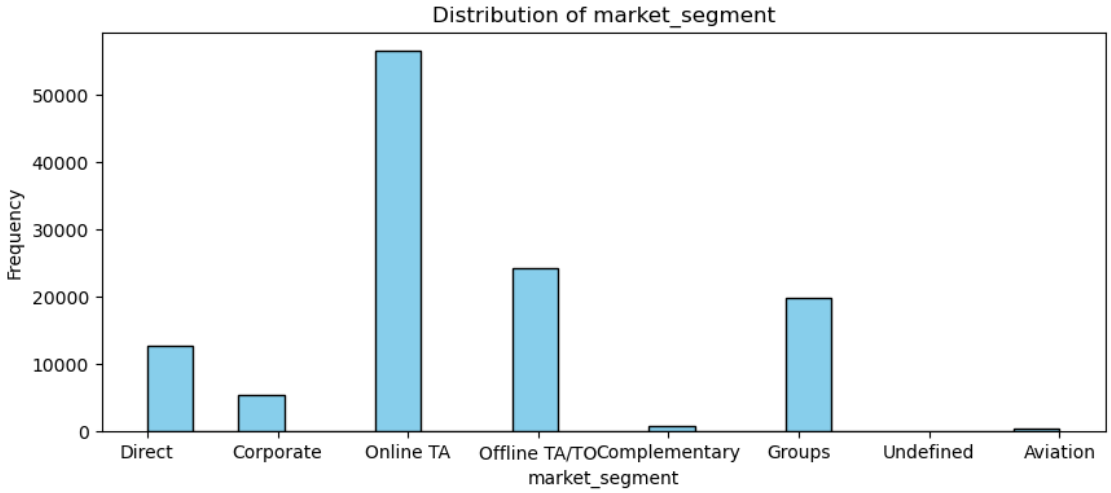

Distribution of adults
Frequency
80000
60000
40000
20000
0
2
1
3
4
40
26
50
27
55
0
20
6
5
10
adults

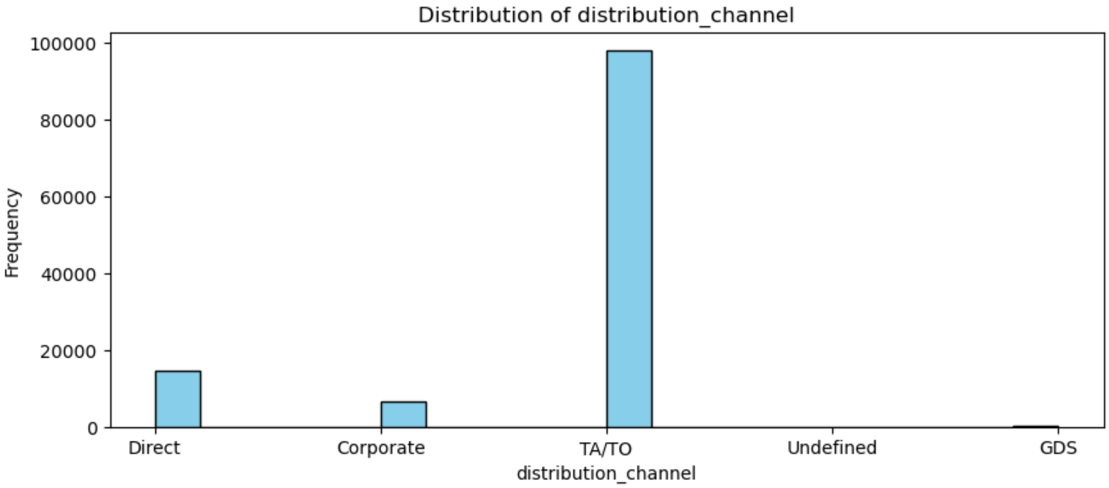

Distribution of children
Frequency
100000
80000
60000
40000
20000
0
0.0
1.0
2.0
10.0
3.0
nan
children

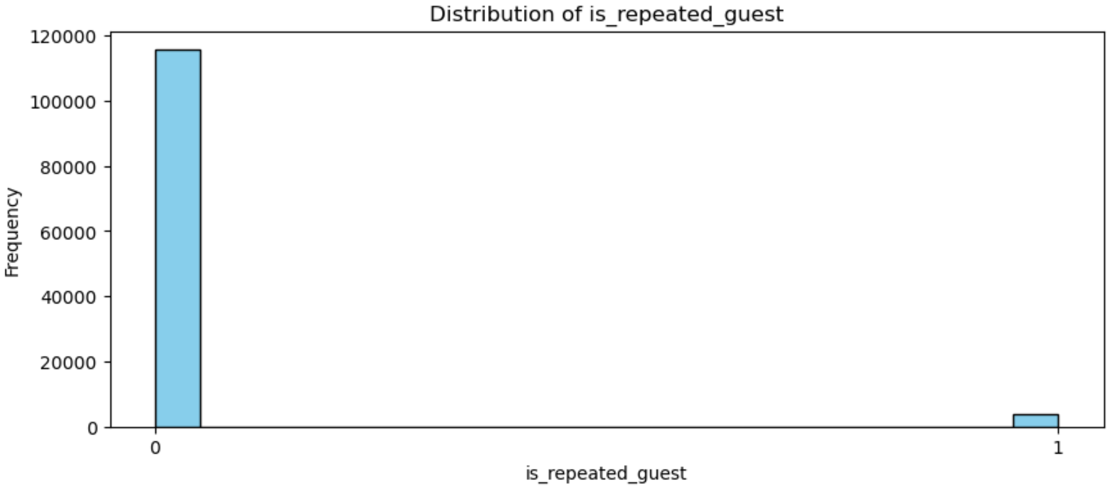

Distribution of babies
Frequency
120000
100000
80000
60000
40000
20000
0
0
1
2
10
9
babies

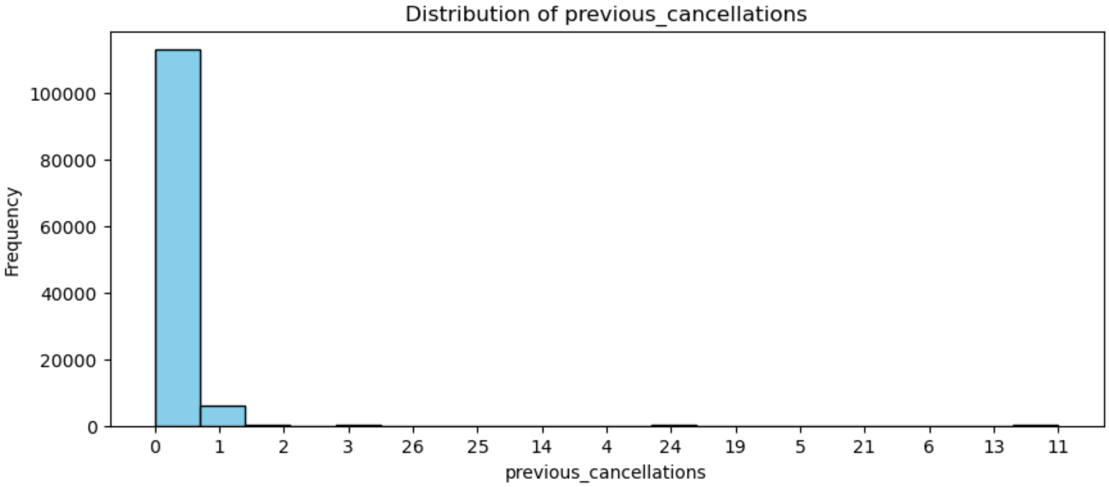

Distribution of meal
Frequency
80000
60000
40000
20000
0
BB
FB
HB
SC
Undefined
meal

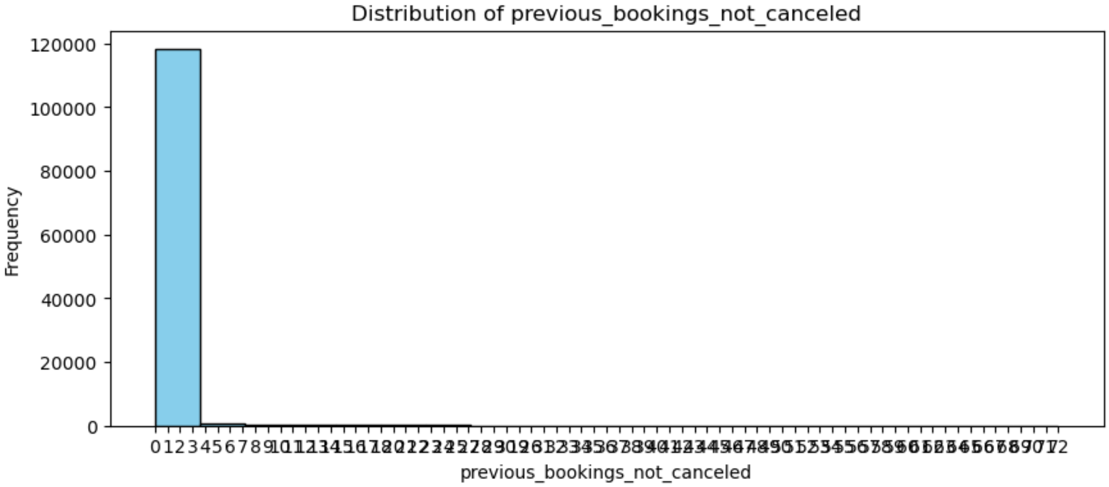

Distribution of country
Frequency
0
20000
40000
60000
80000
country

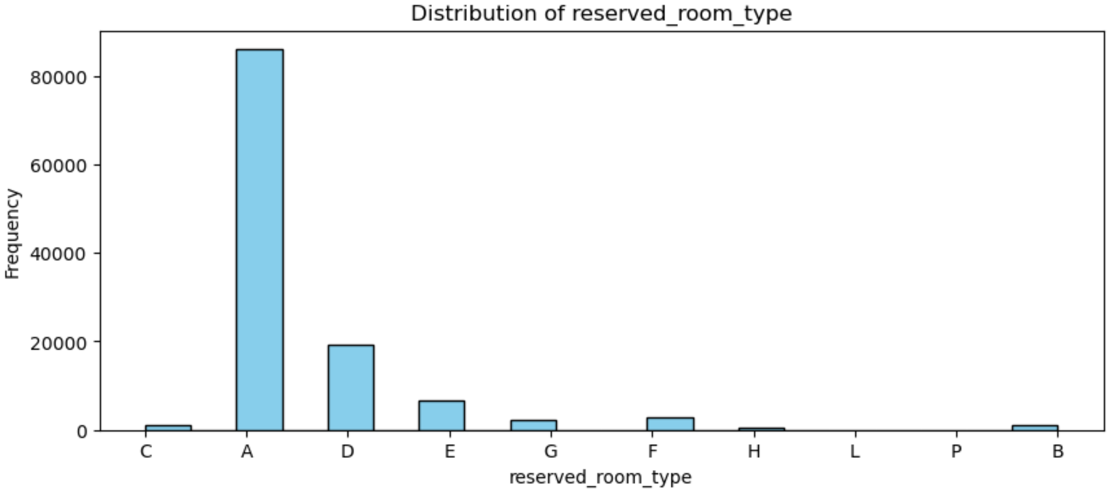

Distribution of market_segment
Frequency
0
10000
20000
30000
40000
50000
Direct
Corporate
Online TA
Offline TA/TO
Complementary
Groups
Undefined
Aviation
market_segment

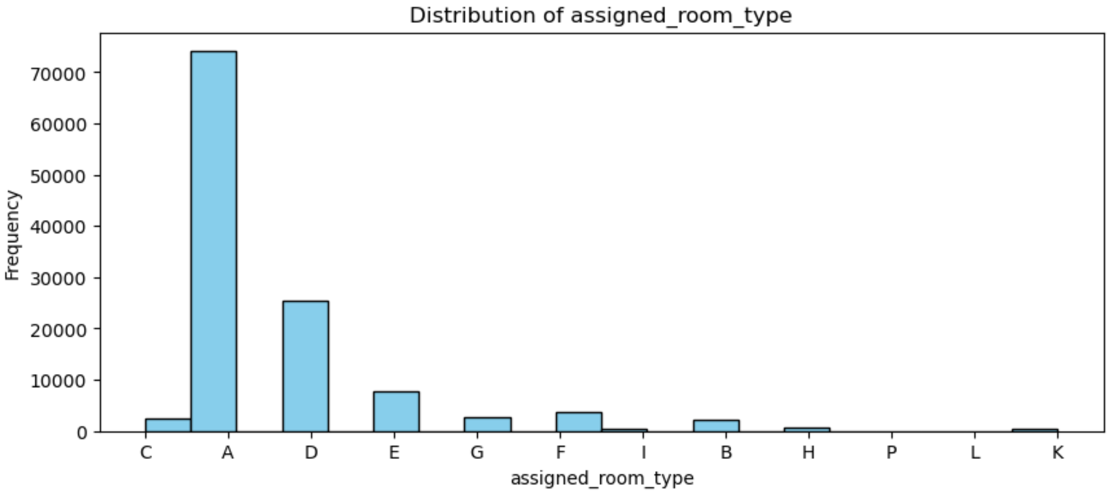

Distribution of distribution_channel
Frequency
0
20000
40000
60000
80000
100000
Direct
Corporate
TA/TO
Undefined
GDS
distribution_channel

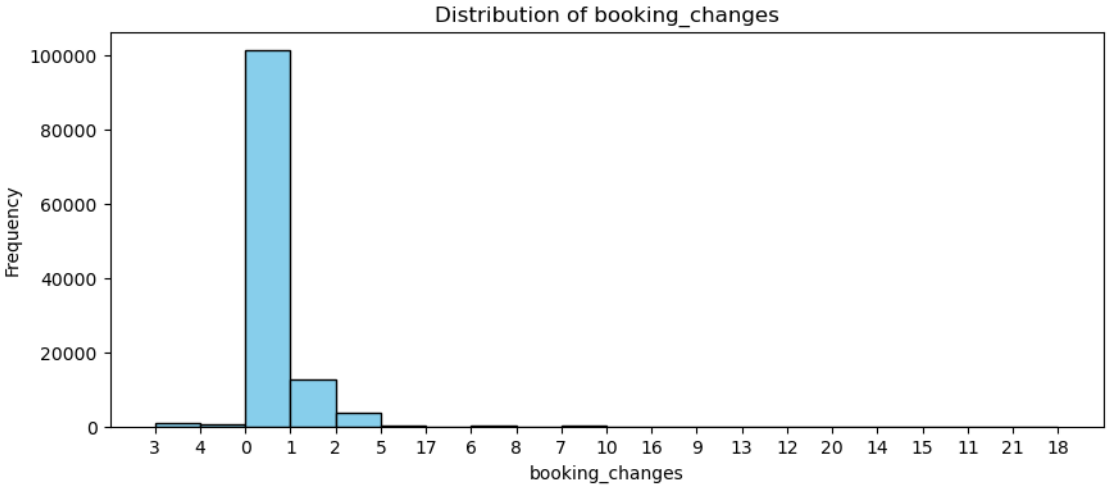

Distribution of is_repeated_guest
Frequency
0
20000
40000
60000
80000
100000
120000
0
1
is_repeated_guest

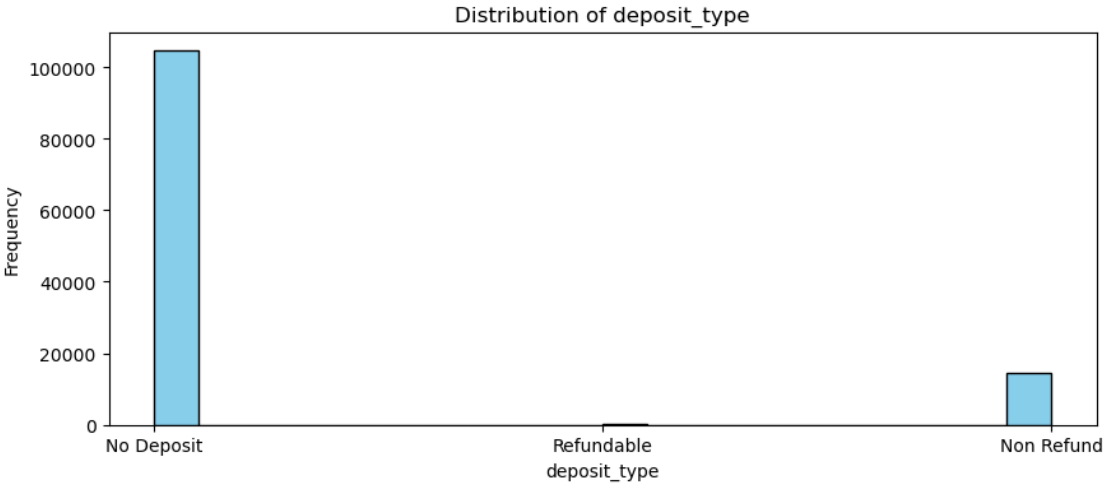

Distribution of previous_cancellations
Frequency
0
20000
40000
60000
80000
100000
0
1
2
3
26
25
14
4
24
19
5
21
6
13
11
previous_cancellations

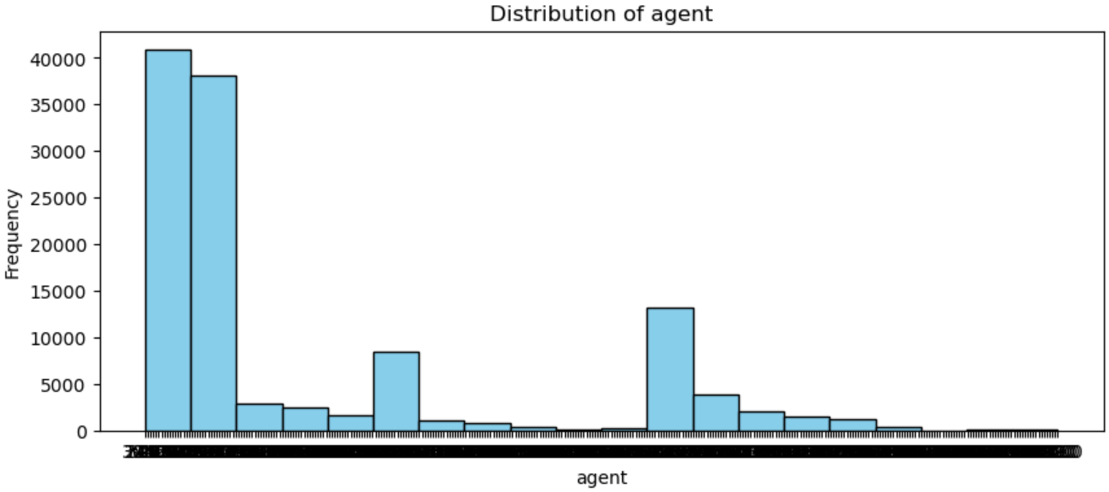

Distribution of previous_bookings_not_canceled
Frequency
120000
100000
80000
60000
40000
20000
0
0 1 2 3 4 5 6 7 8 9
72
previous_bookings_not_canceled

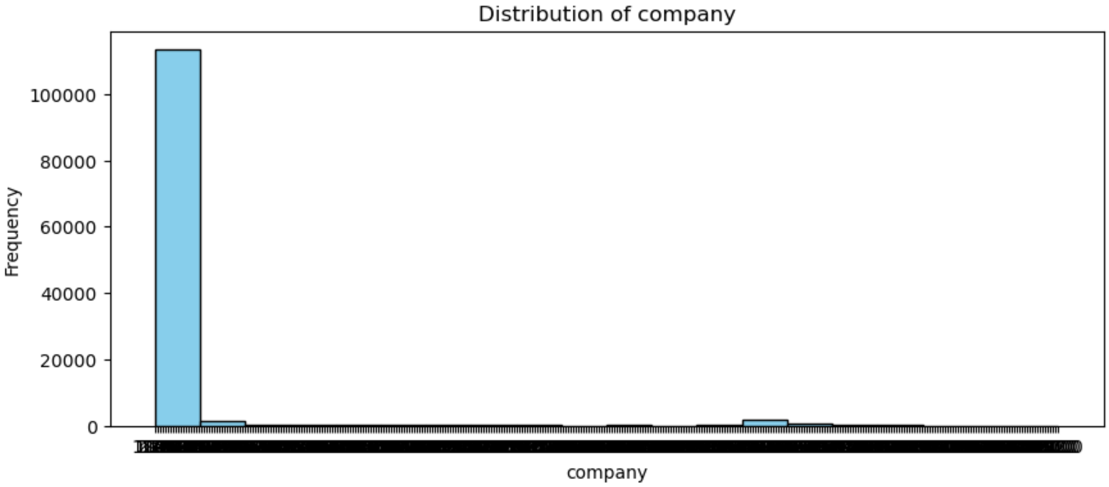

Distribution of reserved_room_type
Frequency
80000
60000
40000
20000
0
C
A
D
E
G
F
H
L
P
B
reserved_room_type

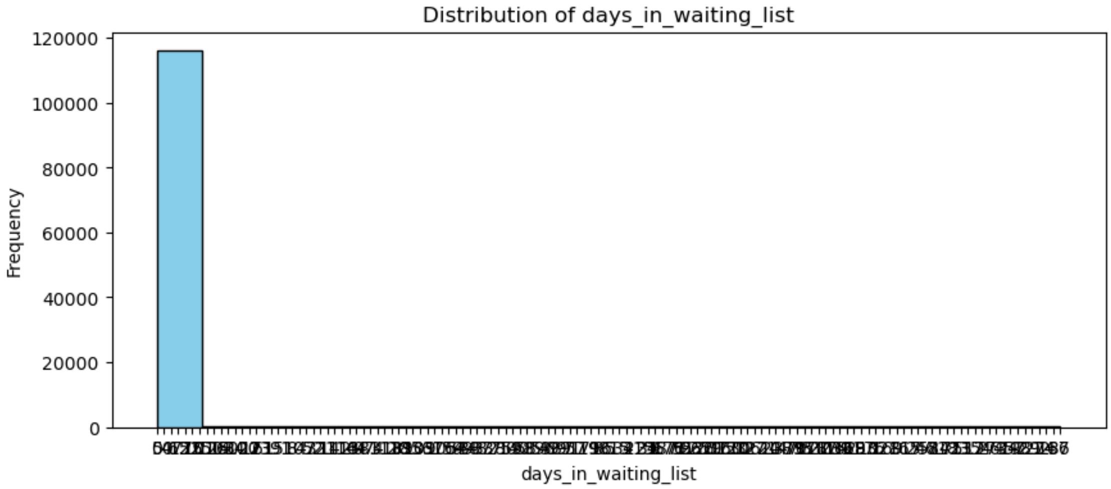

Distribution of assigned_room_type
Frequency
0
10000
20000
30000
40000
50000
60000
70000
C
A
D
E
G
F
I
B
H
P
L
K
assigned_room_type

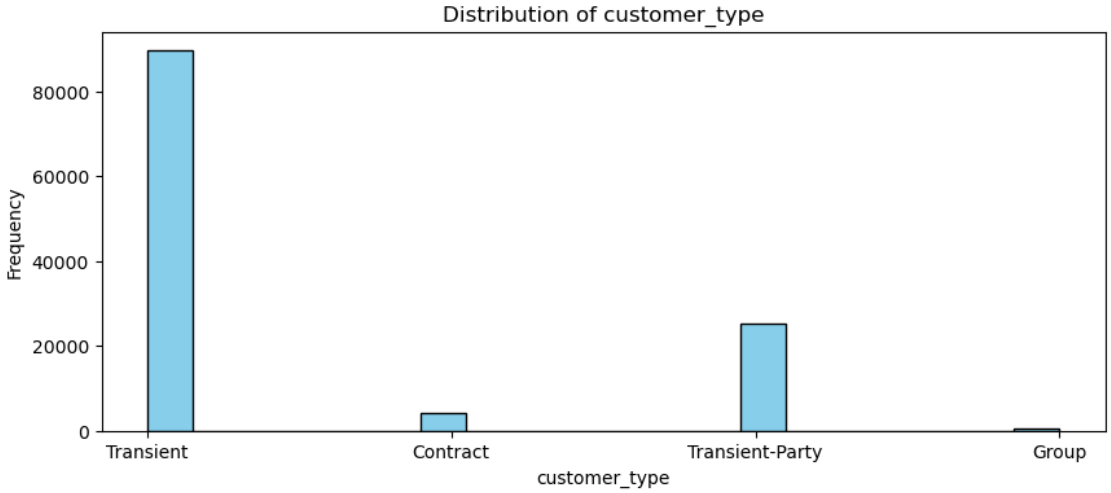

Distribution of booking_changes
Frequency
0
20000
40000
60000
80000
100000
3
4
0
1
2
5
17
6
8
7
10
16
9
13
12
20
14
15
11
21
18
booking_changes


Distribution of deposit_type
Frequency
0
20000
40000
60000
80000
100000
No Deposit
Refundable
Non Refund
deposit_type

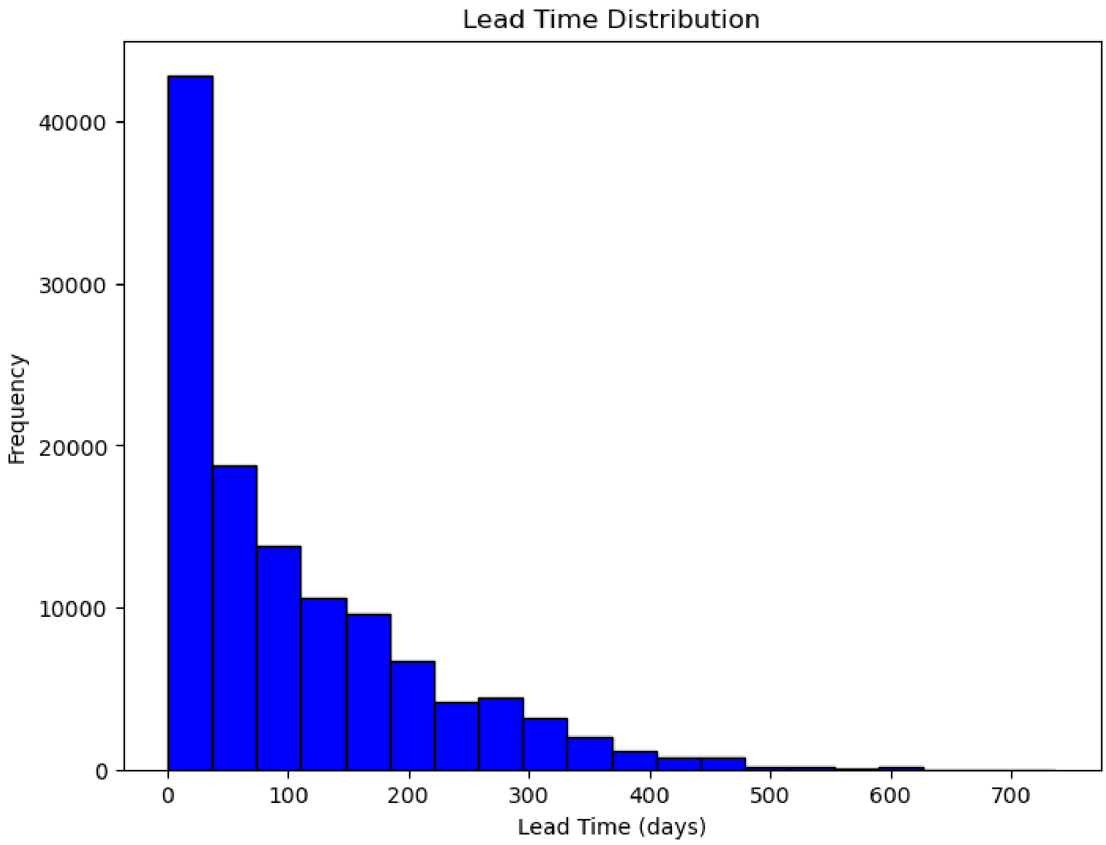

Distribution of agent
Frequency
40000
35000
30000
25000
20000
15000
10000
5000
0
agent


Distribution of company
Frequency
100000
80000
60000
40000
20000
0
company



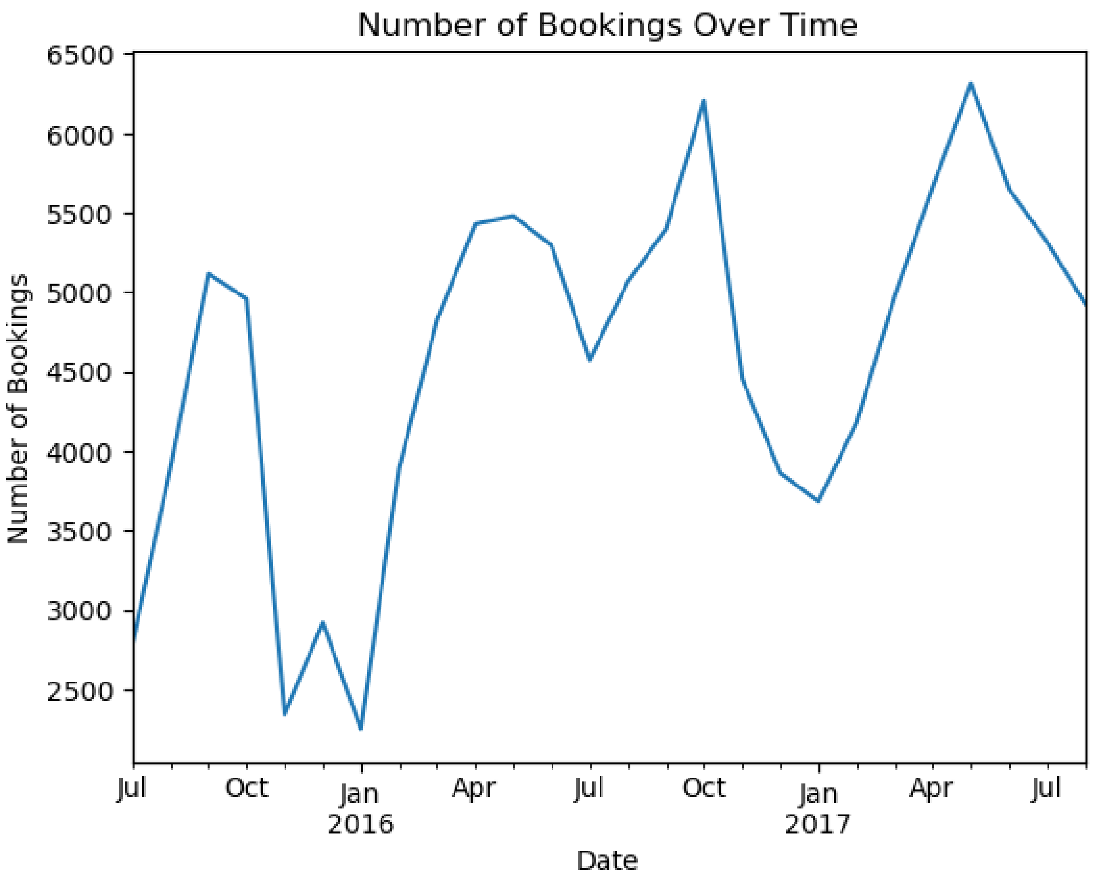


[57]:
```
# histogram show the distribution of lead time, representing the number of days␣
  ↪between booking and arrival.

plt.figure(figsize=(8, 6))
plt.hist(df['lead_time'], bins=20, color='blue', edgecolor='black')
plt.title('Lead Time Distribution')
plt.xlabel('Lead Time (days)')
plt.ylabel('Frequency')
plt.show()
```



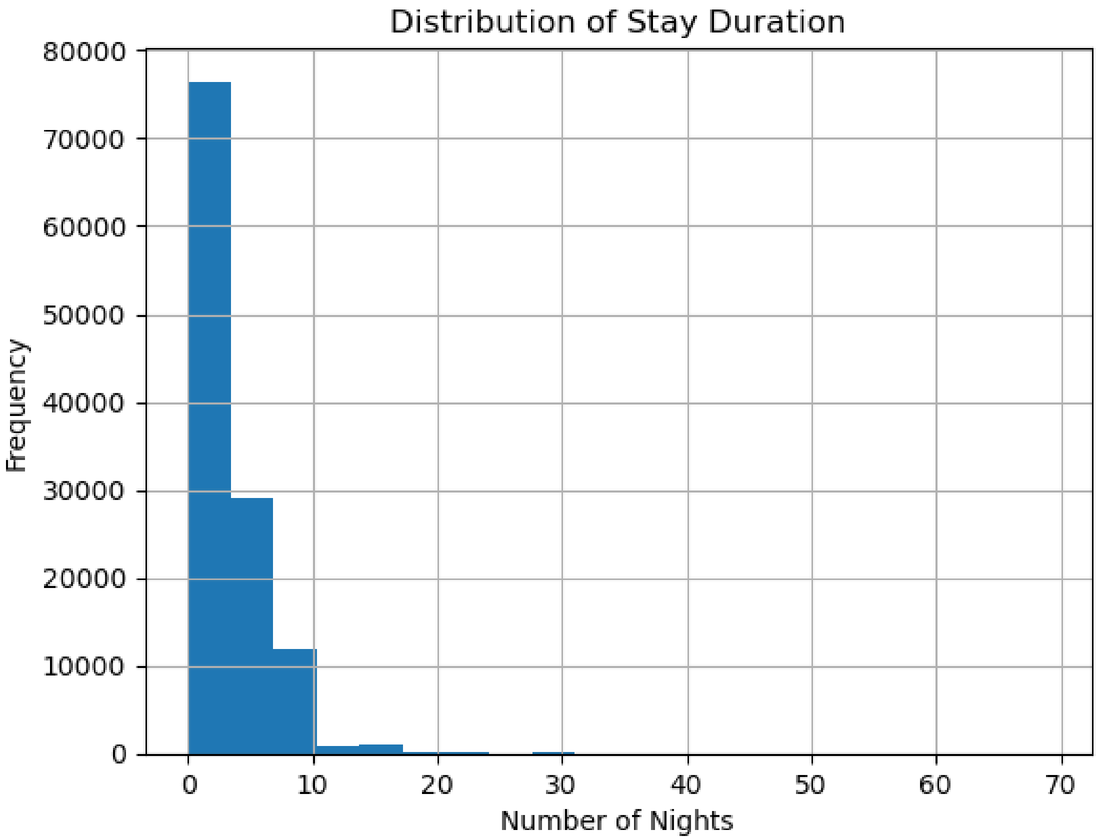
[58]:
```
# number of bookings over time.

df['arrival_date'] = pd.to_datetime(df['arrival_date_year'].astype(str) + '-' +␣
↪df['arrival_date_month'] + '-' + df['arrival_date_day_of_month'].astype(str))
df.set_index('arrival_date', inplace=True)
df.resample('M')['hotel'].count().plot(kind='line')
plt.title('Number of Bookings Over Time')
plt.ylabel('Number of Bookings')
plt.xlabel('Date')
```

[58]:
```
Text(0.5, 0, 'Date')
```



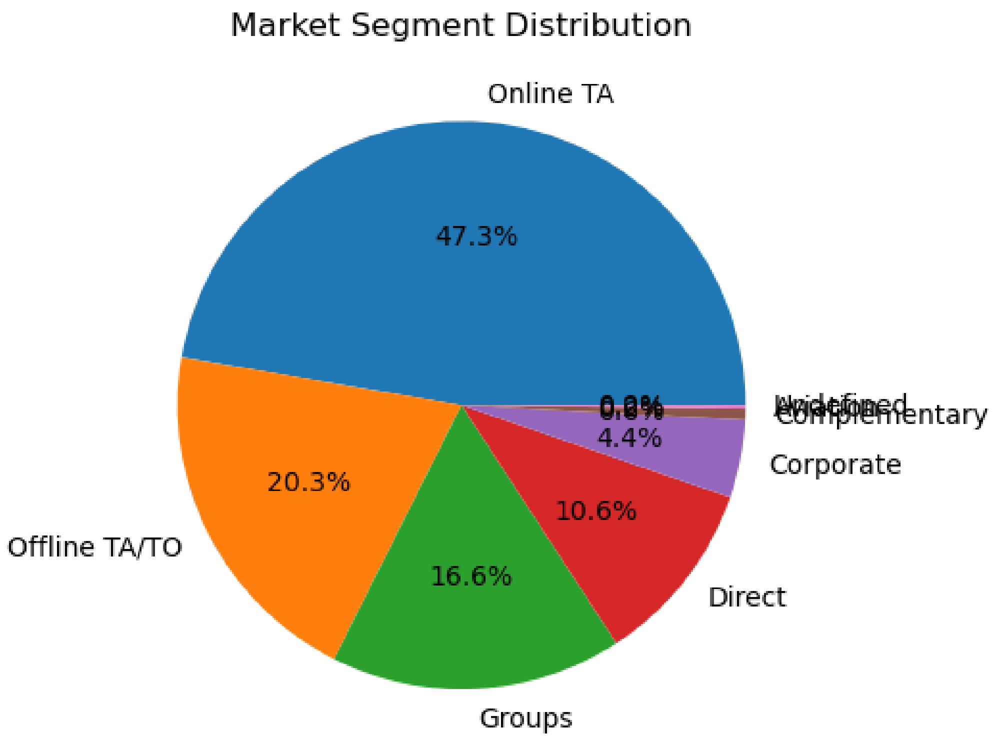
[59]:
```
# stay duration.

df['total_stay'] = df['stays_in_weekend_nights'] + df['stays_in_week_nights']
df['total_stay'].hist(bins=20)
plt.title('Distribution of Stay Duration')
plt.xlabel('Number of Nights')
plt.ylabel('Frequency')
```

[59]:
```
Text(0, 0.5, 'Frequency')
```



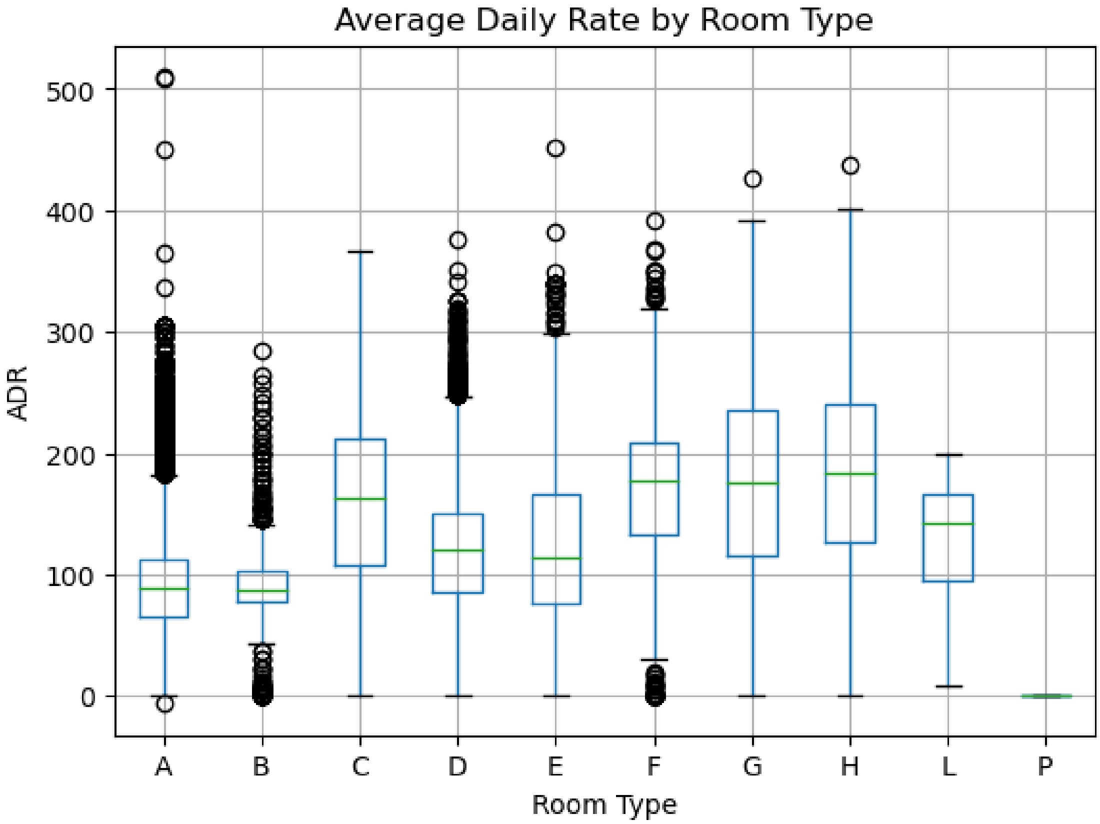
[60]:
```
# market Segment distribution.

df['market_segment'].value_counts().plot(kind='pie', autopct='%1.1f%%')
plt.title('Market Segment Distribution')
plt.ylabel('')
```

[60]:
```
Text(0, 0.5, '')
```

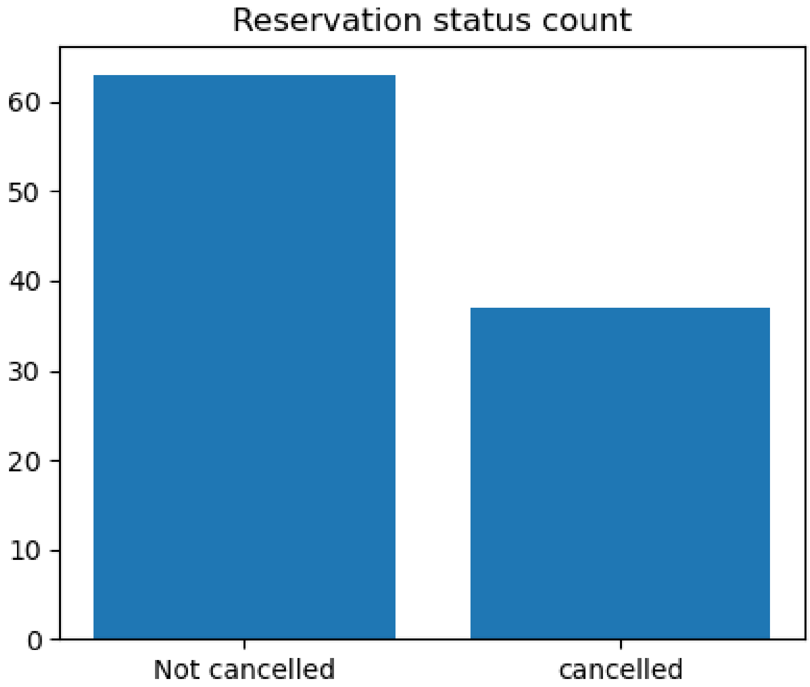


```
[61]: # average daily rate (ADR) by room type.

      df.boxplot(column='adr', by='reserved_room_type')
      plt.title('Average Daily Rate by Room Type')
      plt.xlabel('Room Type')
      plt.ylabel('ADR')
      plt.suptitle('')  # removes the default title
```

```
[61]: Text(0.5, 0.98, '')
```

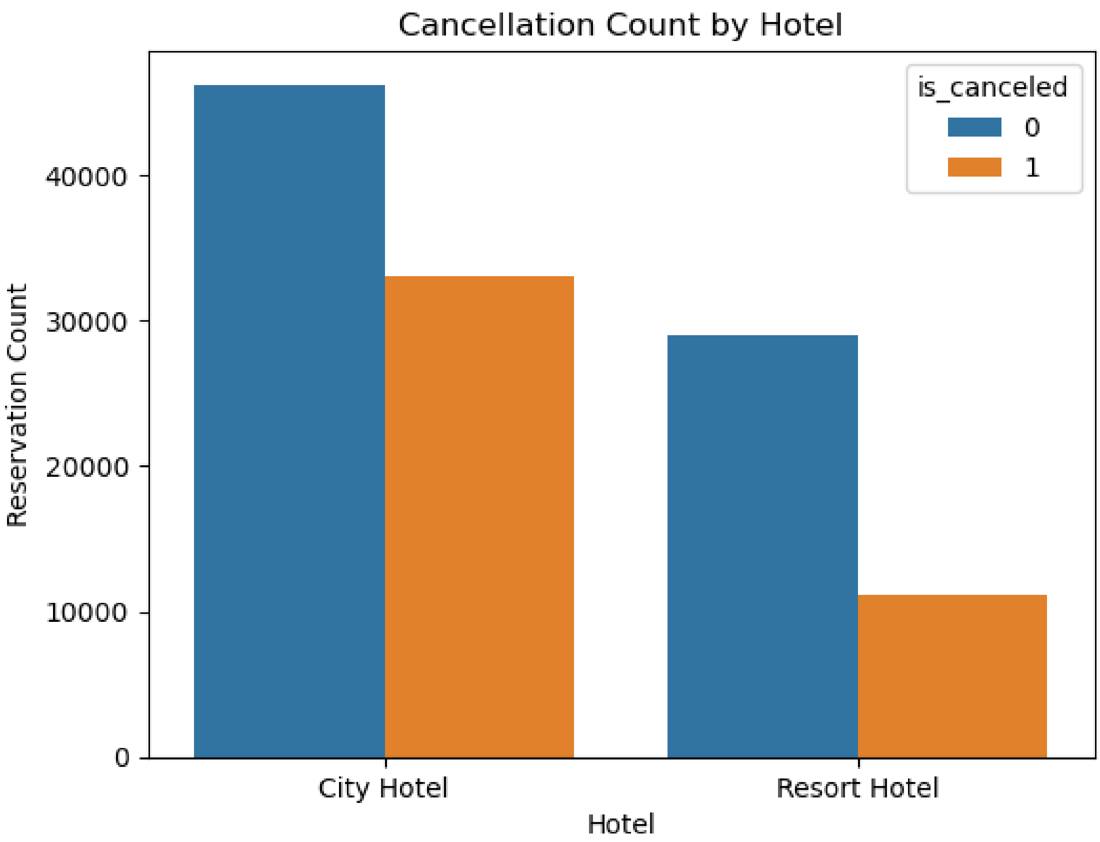


[62]:
```
# the count of reservation statuses, distinguishing between cancelled and not␣
↪cancelled bookings.

plt.figure(figsize=(5,4))
plt.title("Reservation status count")
plt.bar(["Not cancelled","cancelled"],df['is_canceled'].value_counts(normalize␣
↪=True).mul(100))
plt.show()
```



the accompanying bar graph shows the percentage of reservations that are cancelled and those that are not. it is obvious that there are still significant number of reservations that have not been cancelled. there are still 37% of clients who cancelled their reservations, which has significant impact on the hotels earnings.

[63]:
```
# calculate the cancellation count for each hotel
cancellation_count = df.groupby('hotel')['is_canceled'].value_counts().
 ↪reset_index(name='cancellation_count')

# bar plot of cancellation count by hotel
sns.barplot(data=cancellation_count, x='hotel', y='cancellation_count',␣
 ↪hue='is_canceled')
plt.title('Cancellation Count by Hotel')
plt.xlabel('Hotel')
plt.ylabel('Reservation Count')
plt.show()
```



```
[64]: # resort hotel.

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

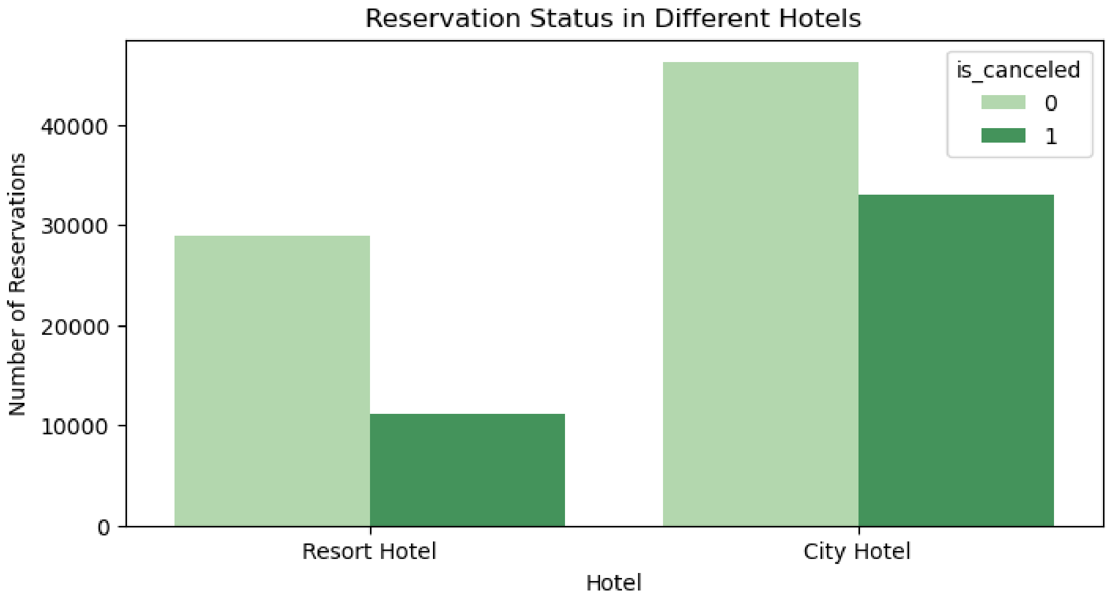
```
[64]: is_canceled
      0    0.722366
      1    0.277634
      Name: proportion, dtype: float64
```

```
[65]: # city hotel.

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

```
[65]: is_canceled
      0    0.582738
      1    0.417262
      Name: proportion, dtype: float64
```

[91]:

```python
# countplot to visualize the reservation status in each hotel.

plt.figure(figsize=(8, 4))
sns.countplot(data=df, x='hotel', hue='is_canceled', palette='Greens')
plt.title('Reservation Status in Different Hotels')
plt.xlabel('Hotel')
plt.ylabel('Number of Reservations')
plt.show()
```



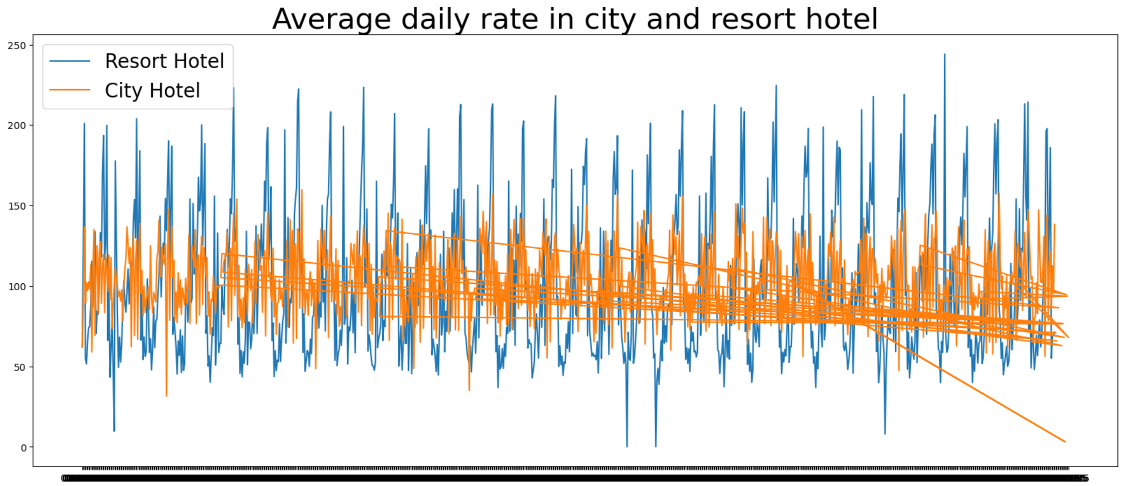
in comparison to resort hotels, city hotels have more bookings. it’s possible that resort hotels are more expensive then those in cities.

[67]:

```python
# calculate the mean ADR for each date.

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

[68]:

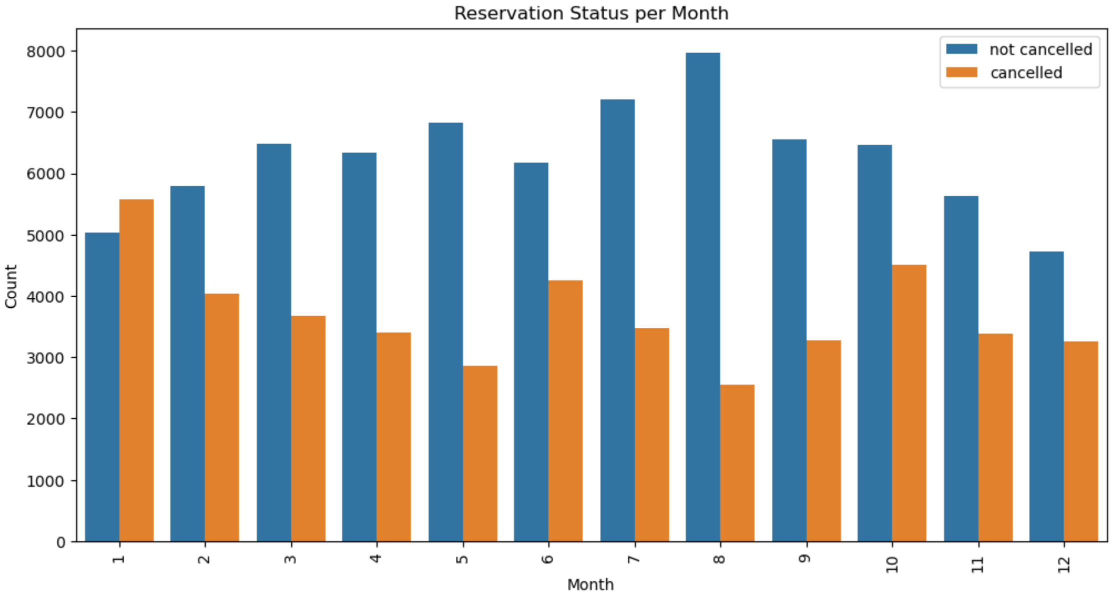
```python
#  a line plot to compare the ADR between city hotel and resort hotel over time.

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 city hotel is less then 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.

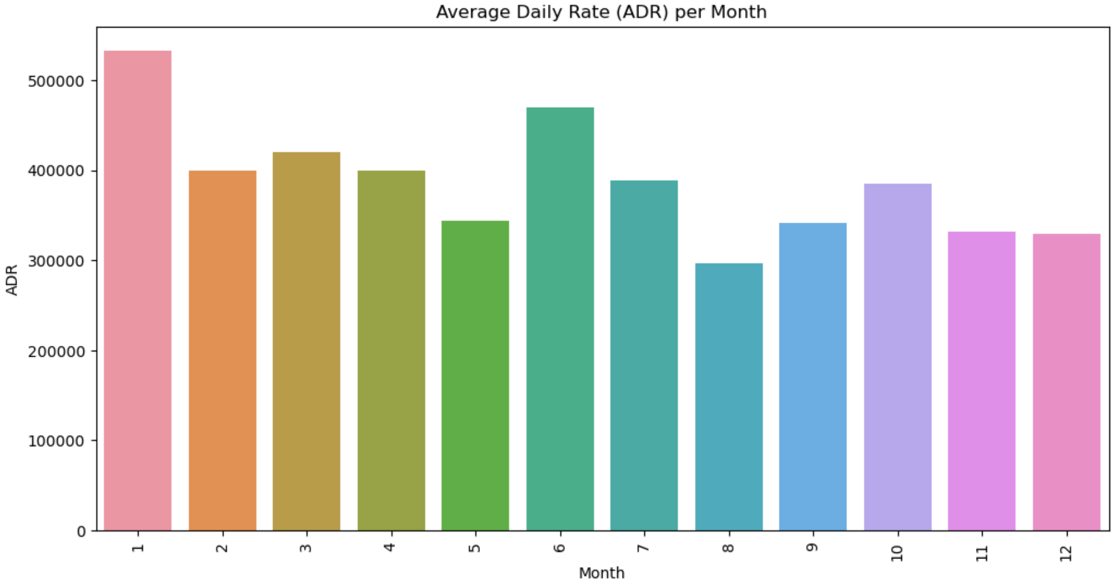
[69]:
```
# converts the 'reservation_status_date' column to datetime format and extracts␣
↪the month component into a new 'month' column, create a countplot to␣
↪visualize the reservation status (cancelled or not cancelled) per month.

df['reservation_status_date'] = pd.to_datetime(df['reservation_status_date'])
df['month'] = df['reservation_status_date'].dt.month
plt.figure(figsize=(12, 6))
sns.countplot(data=df, x='month', hue='is_canceled')
plt.title('Reservation Status per Month')
plt.xlabel('Month')
plt.ylabel('Count')
plt.legend(['not cancelled','cancelled'])
plt.xticks(rotation=90)
plt.show()
```

```
C:\Users\user\AppData\Local\Temp\ipykernel_22180\1978012023.py:1: UserWarning:
Could not infer format, so each element will be parsed individually, falling
back to `dateutil`. To ensure parsing is consistent and as-expected, please
specify a format.
  df['reservation_status_date'] = pd.to_datetime(df['reservation_status_date'])
```



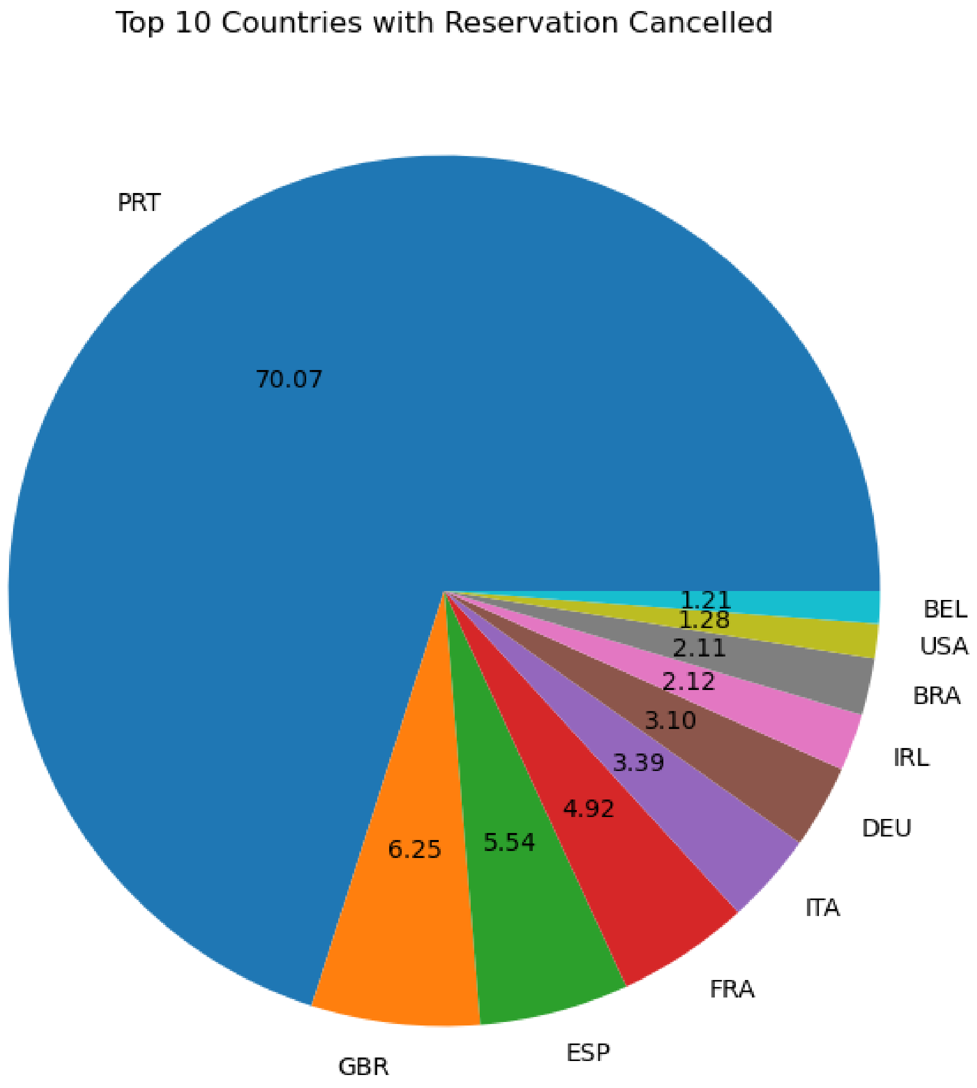
we have create the grouped bar graph to analyze the months with the highest and lowest reservation levels according to reservation status. as can be seen, both the number of confirmed reservations and the number of cancelled reservations are largest in the month of August whereas January is the month with the most cancelled reservations.

[70]:
```
# calculate the ADR per month.
adr_per_month = df[df['is_canceled']==1].groupby('month')['adr'].sum().
↪reset_index()

# create a bar plot of ADR per month
plt.figure(figsize=(12, 6))
sns.barplot(data=adr_per_month, x='month', y='adr')
plt.title('Average Daily Rate (ADR) per Month')
plt.xlabel('Month')
plt.ylabel('ADR')
plt.xticks(rotation=90)
plt.show()
```



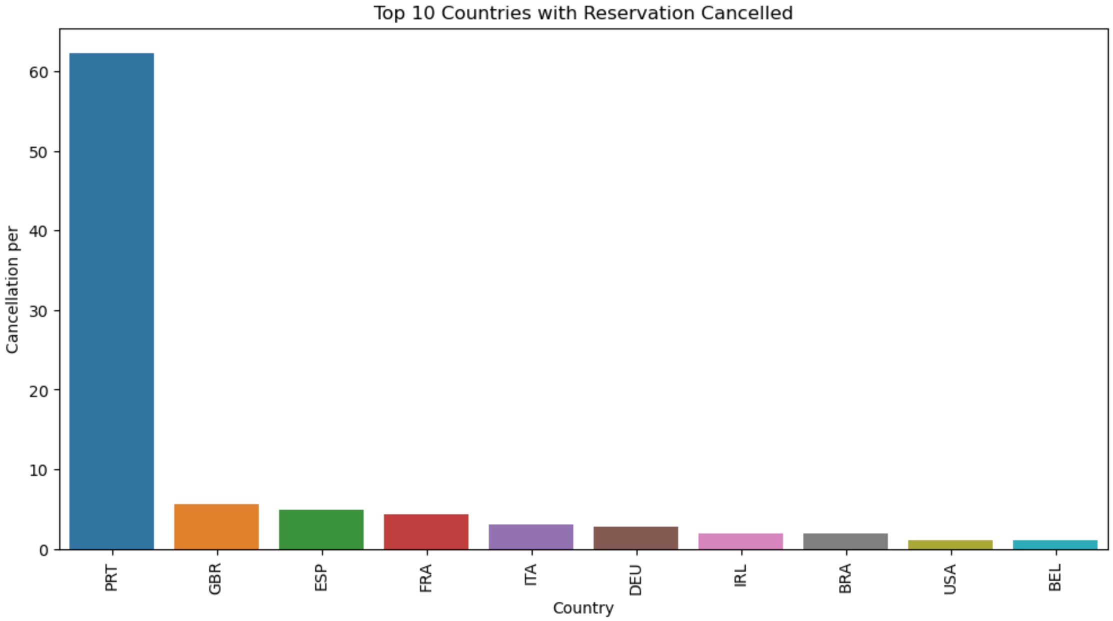
this bar graph demonstrates that cancellations are most common when prices are greatest and are least common when they are lowest. Therefore, the cost of the accommodation is solely responsible for the cancellation.

[71]:
```
# calculates the count of cancelled bookings for each country and selects the␣
↪top 10

cancaled_data =df[df['is_canceled']==1]
top_10_countries=cancaled_data['country'].value_counts()[:10]
plt.figure(figsize=(8, 8))
plt.title('Top 10 Countries with Reservation Cancelled')
plt.pie(top_10_countries, autopct='%.2f',labels=top_10_countries.index)
plt.show()
```



now, let's see which country has the highest reservation canceled. the top country is Portugal with the highest number of cancellations.

[72]:
```
cancaled_data =df[df['is_canceled']==1]
top_10_countries=cancaled_data['country'].value_counts(normalize=True).
  ↪mul(100)[:10]
plt.figure(figsize=(12, 6))
sns.barplot(x=top_10_countries.index, y=top_10_countries.values)
plt.title('Top 10 Countries with Reservation Cancelled')
plt.xlabel('Country')
plt.ylabel('Cancellation per')
plt.xticks(rotation=90)
plt.show()
```



[73]:
```
#  count of bookings for each market segment.

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

[73]:
```
market_segment
Online TA        0.473050
Offline TA/TO    0.202850
Groups           0.165937
Direct           0.105588
Corporate        0.044351
Complementary    0.006223
Aviation         0.001985
Undefined        0.000017
Name: proportion, dtype: float64
```

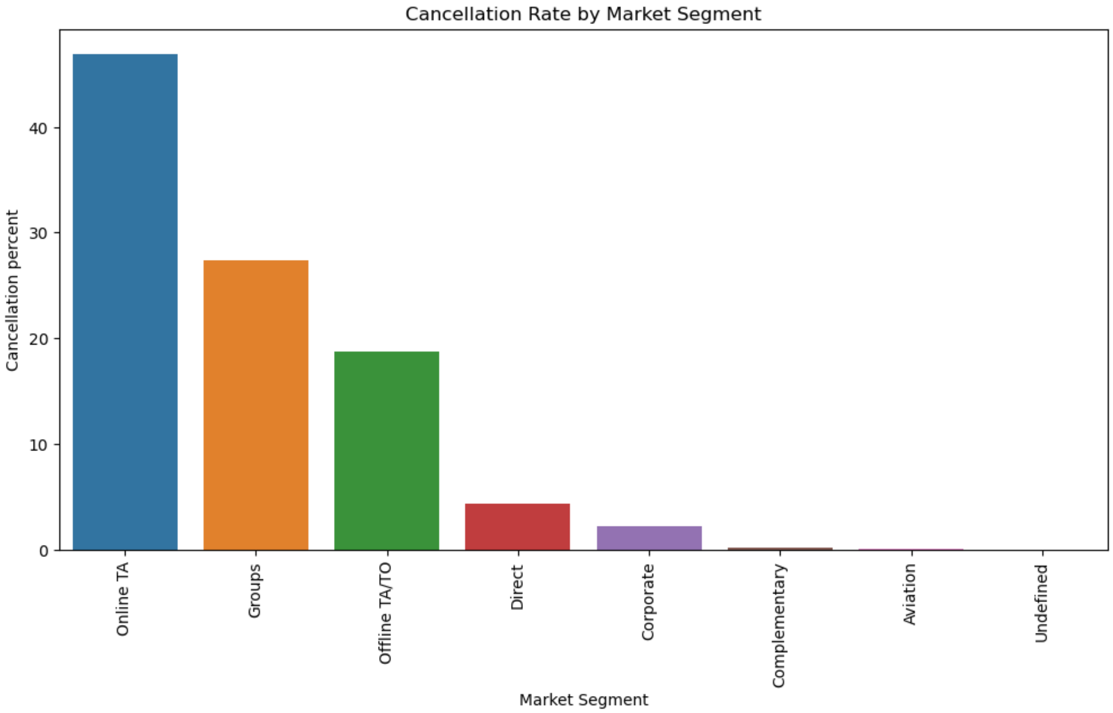
[74]:
```
# count of cancelled bookings for each market segment.

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

[74]:
```
market_segment
Online TA        0.468964
Groups           0.273545
Offline TA/TO    0.187911
Direct           0.043733
```

```
Corporate        0.022432
Complementary    0.002193
Aviation         0.001176
Undefined        0.000045
Name: proportion, dtype: float64
```

[75]:

```
# analyze cancellation rate by market segment.

df_market_segment = cancaled_data['market_segment'].
 ↪value_counts(normalize=True).mul(100)
plt.figure(figsize=(12, 6))
sns.barplot(x=df_market_segment.index,  y=df_market_segment.values)
plt.title('Cancellation Rate by Market Segment')
plt.xlabel('Market Segment')
plt.ylabel('Cancellation percent')
plt.xticks(rotation=90)
plt.show()
```



most cancellation are coming from online travel agencies 46% and then followed by groups which is 18%.

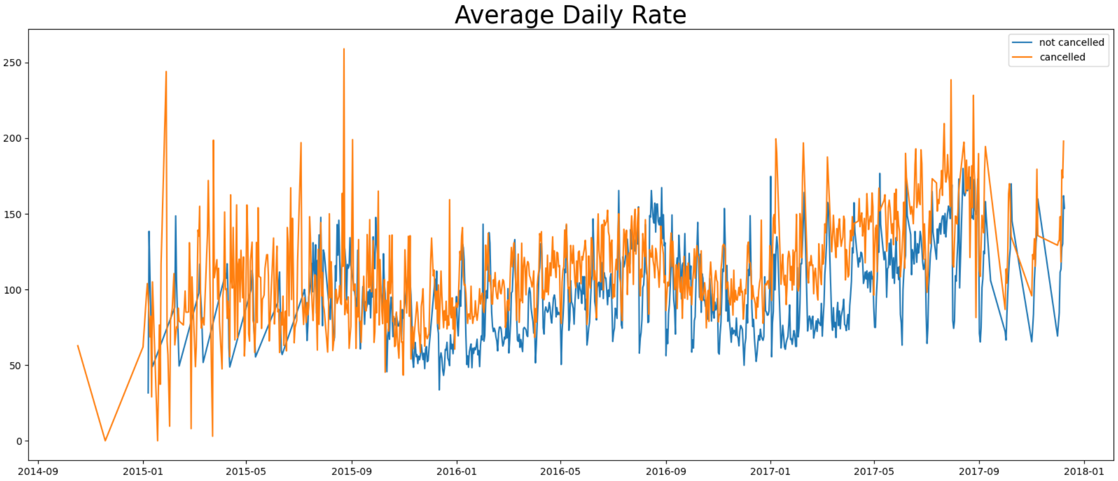
[76]:

```
# calculate the mean of ADR for both cancelled and not cancelled bookings,␣
 ↪create a line plot to visualize the ADR trends over time
```

```
cancaled_df_adr = cancaled_data.groupby('reservation_status_date')[['adr']].
 ↪mean()
cancaled_df_adr.reset_index(inplace=True)
cancaled_df_adr.sort_values('reservation_status_date', inplace=True)

not_cancaled_data =df[df['is_canceled'] ==0]
not_cancaled_df_adr = not_cancaled_data.
 ↪groupby('reservation_status_date')[['adr']].mean()
not_cancaled_df_adr.reset_index(inplace=True)
not_cancaled_df_adr.sort_values('reservation_status_date', inplace=True)

plt.figure(figsize=(20,8))
plt.title('Average Daily Rate', fontsize=25)
plt.plot(not_cancaled_df_adr['reservation_status_date'],␣
 ↪not_cancaled_df_adr['adr'], label='not cancelled')
plt.plot(cancaled_df_adr['reservation_status_date'], cancaled_df_adr['adr'],␣
 ↪label='cancelled')
plt.legend()
```

[76]:

```
<matplotlib.legend.Legend at 0x1f50ae2ac90>
```

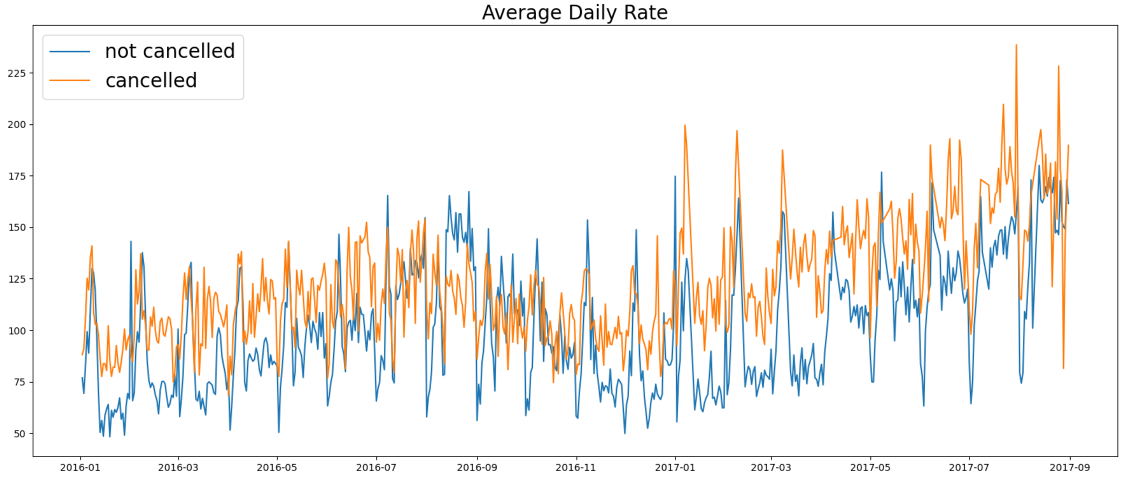


[77]:

```
# filter the data for both cancelled and not cancelled bookings to include only␣
 ↪entries between January 2016 and September 2017.

cancaled_df_adr =␣
 ↪cancaled_df_adr[(cancaled_df_adr['reservation_status_date']>'2016') &␣
 ↪(cancaled_df_adr['reservation_status_date']<'2017-09')]
not_cancaled_df_adr =␣
 ↪not_cancaled_df_adr[(not_cancaled_df_adr['reservation_status_date']>'2016')␣
 ↪& (not_cancaled_df_adr['reservation_status_date']<'2017-09')]
```

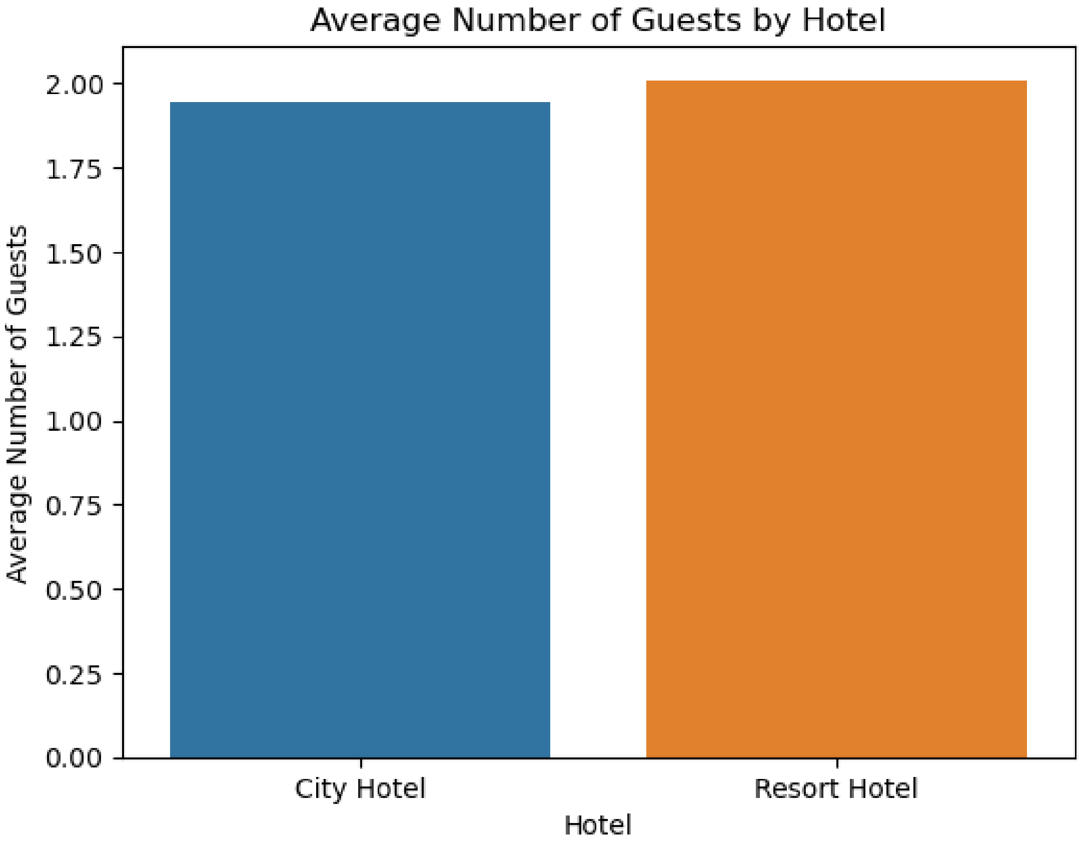
```
[78]: plt.figure(figsize=(20,8))
      plt.title('Average Daily Rate', fontsize=20)
      plt.plot(not_canceled_df_adr['reservation_status_date'],␣
        ↪not_canceled_df_adr['adr'], label='not cancelled')
      plt.plot(canceled_df_adr['reservation_status_date'], canceled_df_adr['adr'],␣
        ↪label='cancelled')
      plt.legend(fontsize=20)
```

```
[78]: <matplotlib.legend.Legend at 0x1f50c228d10>
```



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

```
[79]: # calculate average number of guests (adults, children, babies) per booking.

      df['total_guests'] = df['adults'] + df['children'] + df['babies']
      average_guests = df.groupby('hotel')['total_guests'].mean().reset_index()
      average_guests.columns = ['hotel', 'average_guests']

      # bar plot of average number of guests by hotel.

      sns.barplot(data=average_guests, x='hotel', y='average_guests')
      plt.title('Average Number of Guests by Hotel')
      plt.xlabel('Hotel')
      plt.ylabel('Average Number of Guests')
      plt.show()
```



Suggestion: Adjust pricing strategies: Offer targeted discounts based on location to curb cancellations.

Weekend/holiday discounts: Provide competitive rates during peak times to reduce cancellations, especially in resort hotels.

January campaigns: Launch marketing initiatives with attractive offers to combat high cancellation rates during this month.

Enhance quality and service: Improve hotel standards, particularly in regions like Portugal, to foster guest satisfaction and lower cancellation rates.