HotelBookingDemand

July 18, 2022

[1]:
```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
sns.set(style = "whitegrid")
```

[2]:
```python
df = pd.read_csv("hotel_bookings.csv")
```

[3]:
```python
print(df.shape)
df.columns
```

```
(119390, 32)
```

[3]:
```
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')
```

[4]:
```python
df.info()
```

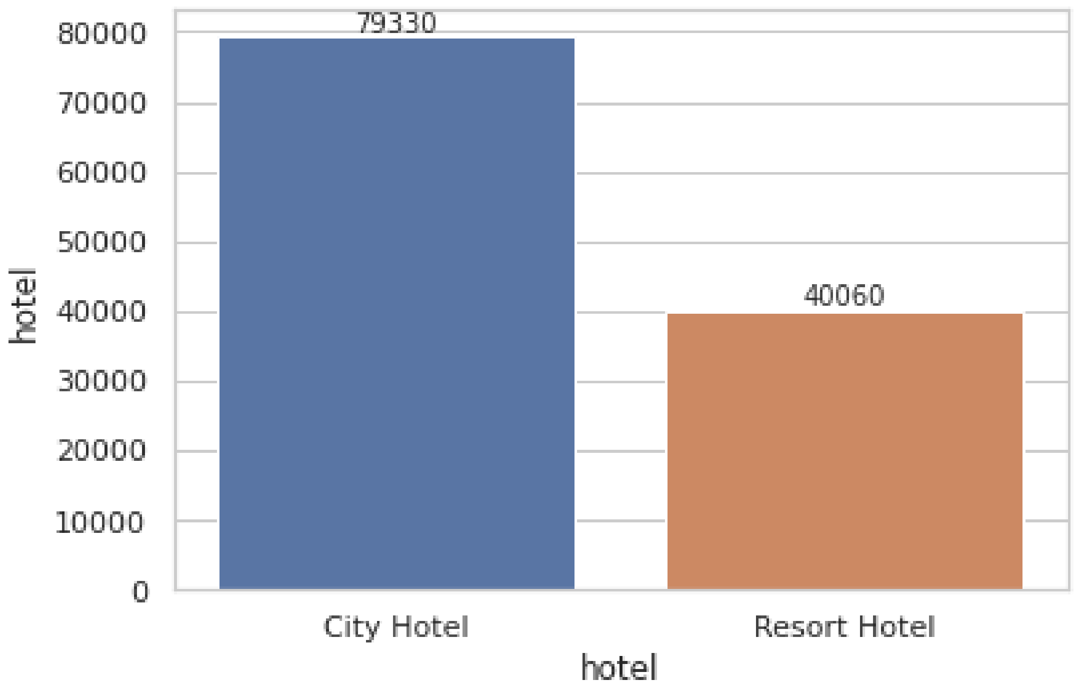
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 119390 entries, 0 to 119389
Data columns (total 32 columns):
 #   Column                          Non-Null Count   Dtype
---  ------                          --------------   -----
 0   hotel                           119390 non-null  object
 1   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
```

[5]:
```
print("Active Reservation status values:␣
 ↪",df[df['is_canceled']==0]['reservation_status'].unique())
print("Cancelled Resrv. status values:
 ↪",df[df['is_canceled']==1]['reservation_status'].unique())
```

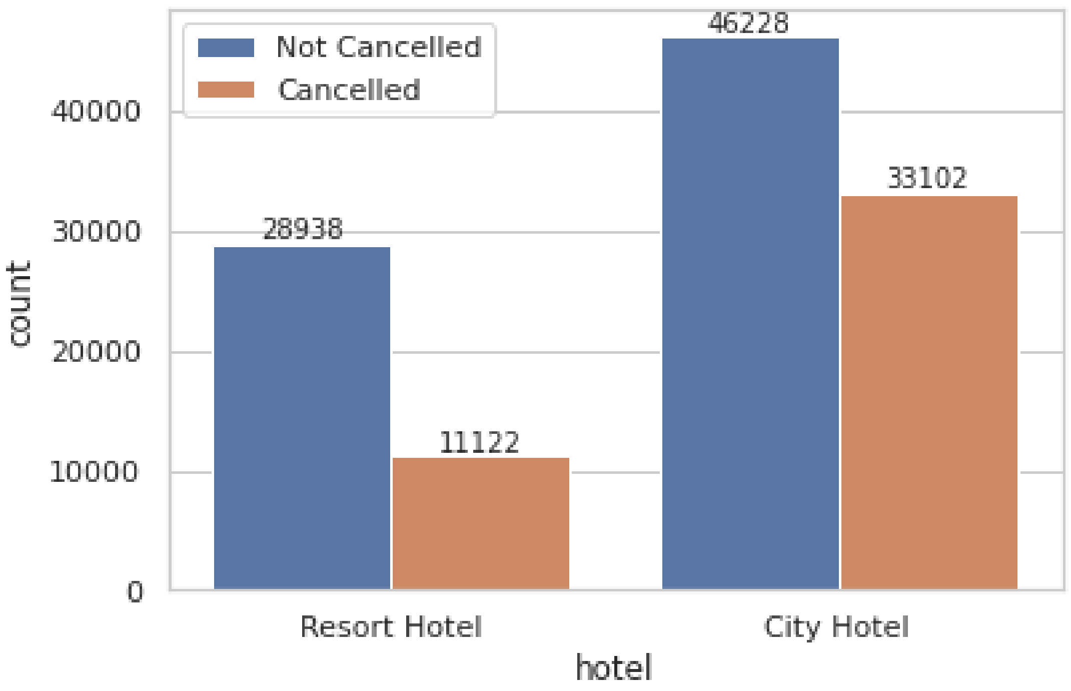
```
Active Reservation status values:  ['Check-Out']
Cancelled Resrv. status values: ['Canceled' 'No-Show']
```

[6]:
```
#Bookings by different types of hotels

d = df.groupby('hotel')['hotel'].count()
ax = sns.barplot(x=d.index, y=d)
sns.__version__
ax.bar_label(ax.containers[0])
```

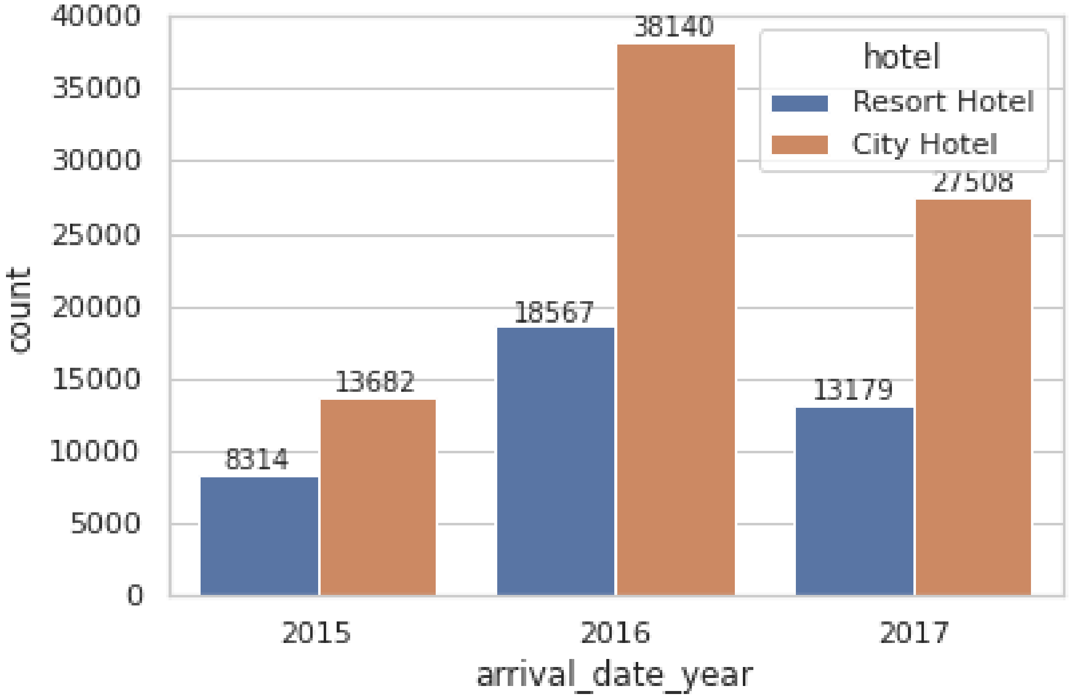
[6]:
```
[Text(0, 0, '79330'), Text(0, 0, '40060')]
```



[75]:
```python
## Counts of Cancelled vs Not-Cancelled Bookings in different types of hotels
ax = sns.countplot(x='hotel', hue='is_canceled', data=df)
plt.legend(['Not Cancelled', 'Cancelled'])
for container in ax.containers:
    ax.bar_label(container)
```



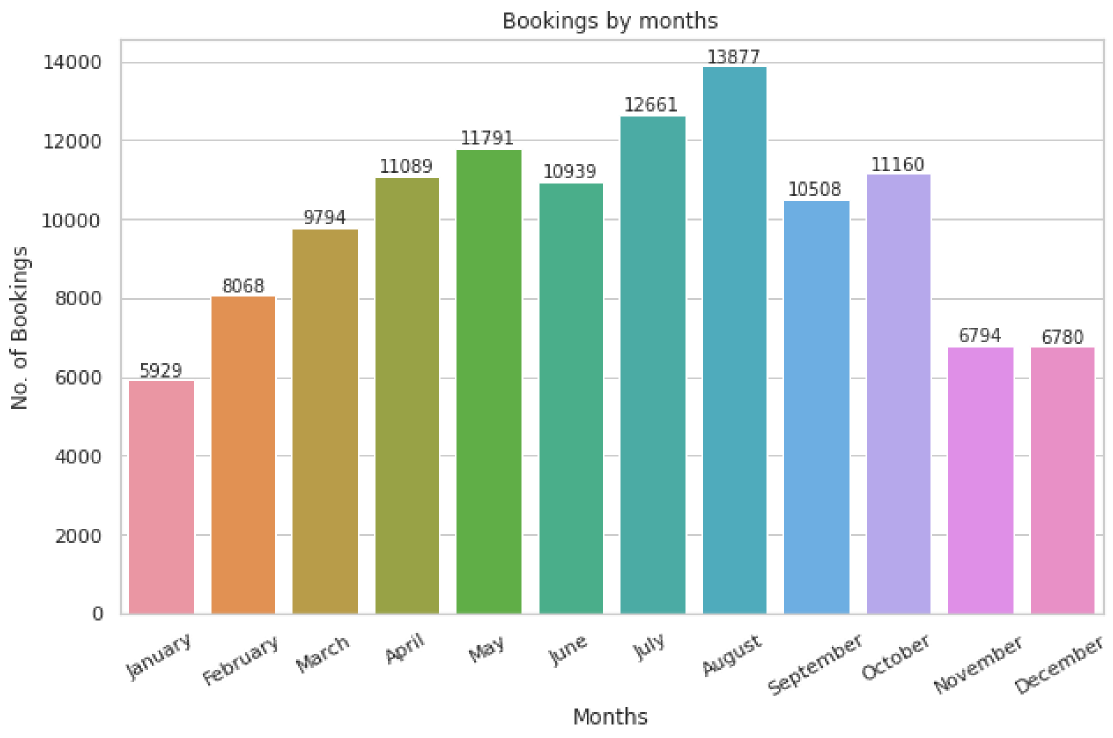
```
[76]: ax = sns.countplot(x='arrival_date_year', hue='hotel', data=df)
      for container in ax.containers:
          ax.bar_label(container)
```



1 No. of bookings by months of year

[77]:
```
months = ["January", "February", "March", "April", "May", "June", "July",␣
↪"August", "September", "October", "November", "December"]
```

[79]:
```
d = df.groupby("arrival_date_month")["arrival_date_month"].count()
plt.figure(figsize=(10,6))
ax = sns.barplot(x=d.index, y=d, order=months)
p = plt.xticks(rotation=30)
plt.xlabel("Months")
plt.ylabel("No. of Bookings")
plt.title("Bookings by months")
for container in ax.containers:
    ax.bar_label(container)
```

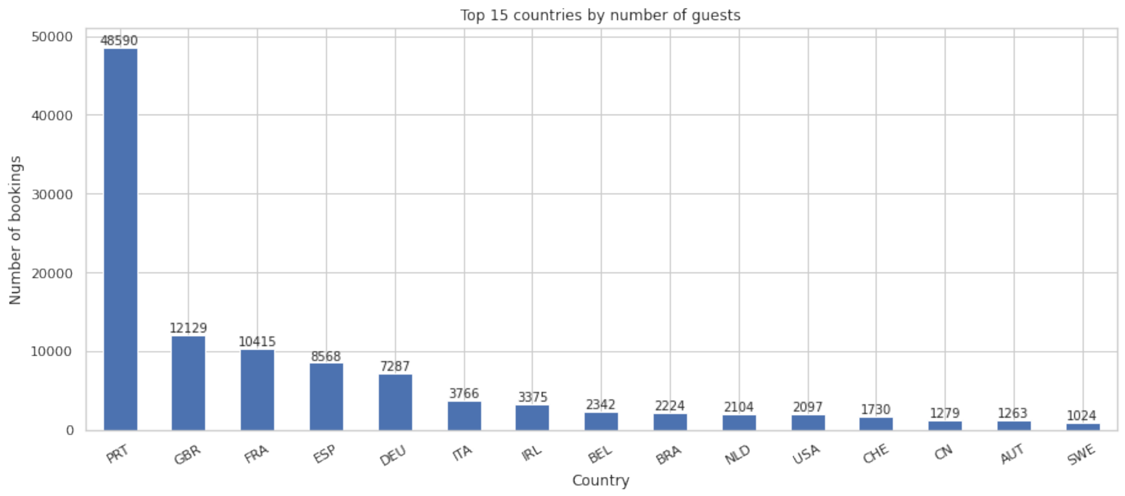


[12]:
```
df['country'].unique()
d = df['country'].value_counts()
d.describe()
```

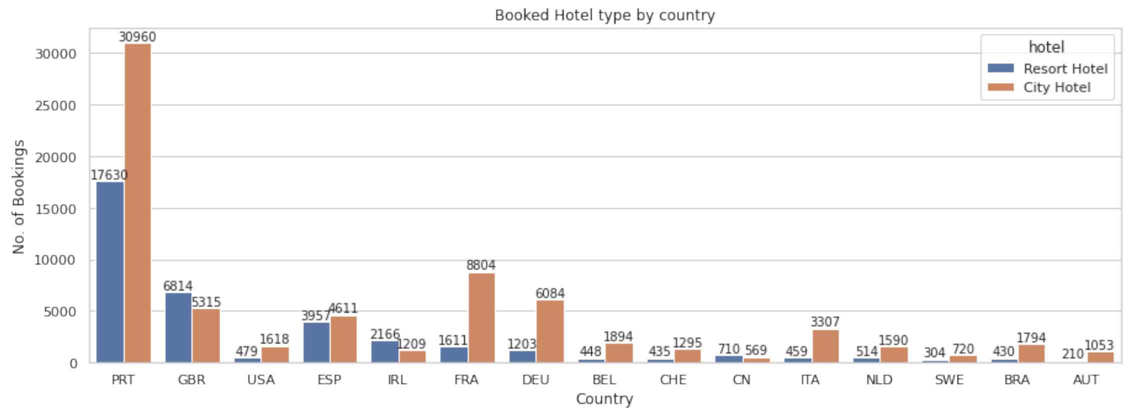
[12]:
```
count        177.000000
mean         671.762712
std         3931.154035
min            1.000000
25%            2.000000
50%           12.000000
75%           74.000000
max        48590.000000
Name: country, dtype: float64
```

2 Top 15 countries by number of bookings

[82]:
```python
plt.figure(figsize=(15,6))
ax = d.sort_values(ascending=False)[:15].plot(kind='bar')
p = plt.xticks(rotation=30)
plt.xlabel("Country")
plt.ylabel("Number of bookings")
plt.title("Top 15 countries by number of guests")
for container in ax.containers:
    ax.bar_label(container)
#The country of Portugal (PRT) has significantly higher number of bookings␣
 ↪compared to any other countries.
```



[80]:
```python
d = df['country'].value_counts().sort_values(ascending=False)[:15]
plt.figure(figsize=(15,5))
ax = sns.countplot(x='country', hue='hotel', data=df[df['country'].isin(d.
 ↪index)])
plt.xlabel("Country")
plt.ylabel("No. of Bookings")
plt.title("Booked Hotel type by country")
for container in ax.containers:
    ax.bar_label(container)
```



[15]:
```
#Guests form Portugal (PRT) from where most bookings are made prefers City␣
↪Hotels over Resort Hotels. Whereas guests from Britain (GBR), country with␣
↪second highest bookings prefers Resort Hotels more.
```

[16]:
```
from plotly import express as px
```

[17]:
```
new_dataset=df.
↪drop(['company','arrival_date_week_number','lead_time','days_in_waiting_list','agent'],axis
new_dataset.head(5)
```

[17]:
```
          hotel  is_canceled  arrival_date_year arrival_date_month  \
0  Resort Hotel            0               2015               July
1  Resort Hotel            0               2015               July
2  Resort Hotel            0               2015               July
3  Resort Hotel            0               2015               July
4  Resort Hotel            0               2015               July

   arrival_date_day_of_month  stays_in_weekend_nights  stays_in_week_nights  \
0                          1                        0                     0
1                          1                        0                     0
2                          1                        0                     1
3                          1                        0                     1
4                          1                        0                     2

   adults  children  babies  … reserved_room_type assigned_room_type  \
0       2       0.0       0  …                  C                  C
1       2       0.0       0  …                  C                  C
2       1       0.0       0  …                  A                  C
```

```
3       1       0.0       0  …                  A                   A
4       2       0.0       0  …                  A                   A

  booking_changes deposit_type  customer_type   adr  \
0               3   No Deposit      Transient   0.0
1               4   No Deposit      Transient   0.0
2               0   No Deposit      Transient  75.0
3               0   No Deposit      Transient  75.0
4               0   No Deposit      Transient  98.0

   required_car_parking_spaces total_of_special_requests reservation_status  \
0                            0                         0          Check-Out
1                            0                         0          Check-Out
2                            0                         0          Check-Out
3                            0                         0          Check-Out
4                            0                         1          Check-Out

   reservation_status_date
0               2015-07-01
1               2015-07-01
2               2015-07-02
3               2015-07-02
4               2015-07-03

[5 rows x 27 columns]
```

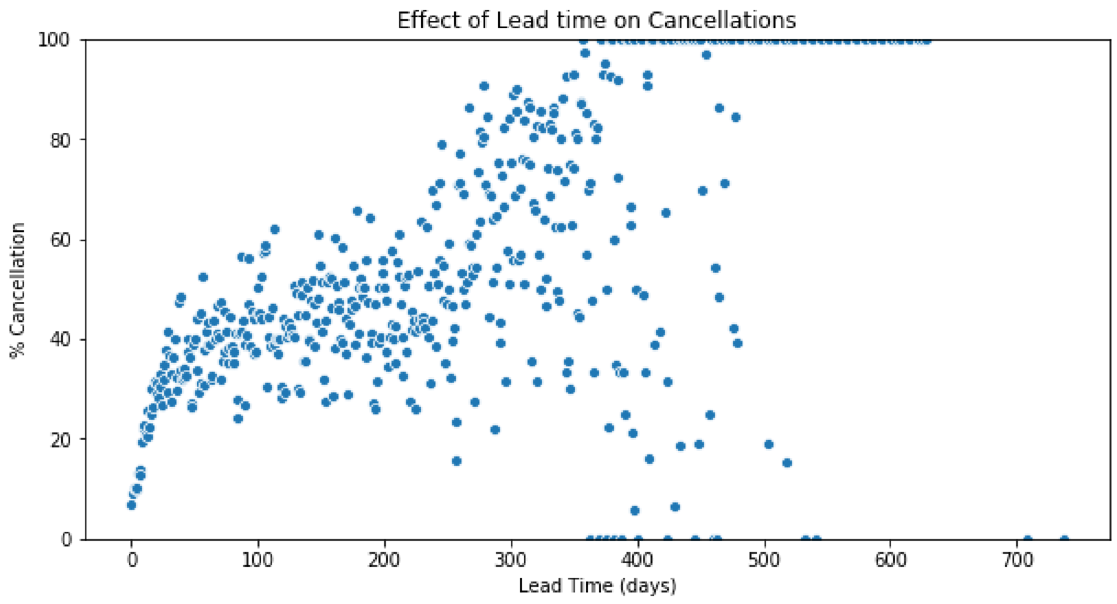
[18]:
```
#Extract the total country data
country_data=new_dataset['country'].value_counts().to_frame().reset_index()
country_data.rename(columns={'index':'country','country':
 ↪'guest_count'},inplace=True)
print(country_data)
```

```
    country  guest_count
0       PRT        48590
1       GBR        12129
2       FRA        10415
3       ESP         8568
4       DEU         7287
..      …            …
172     BHS            1
173     BFA            1
174     ASM            1
175     MRT            1
176     CYM            1

[177 rows x 2 columns]
```

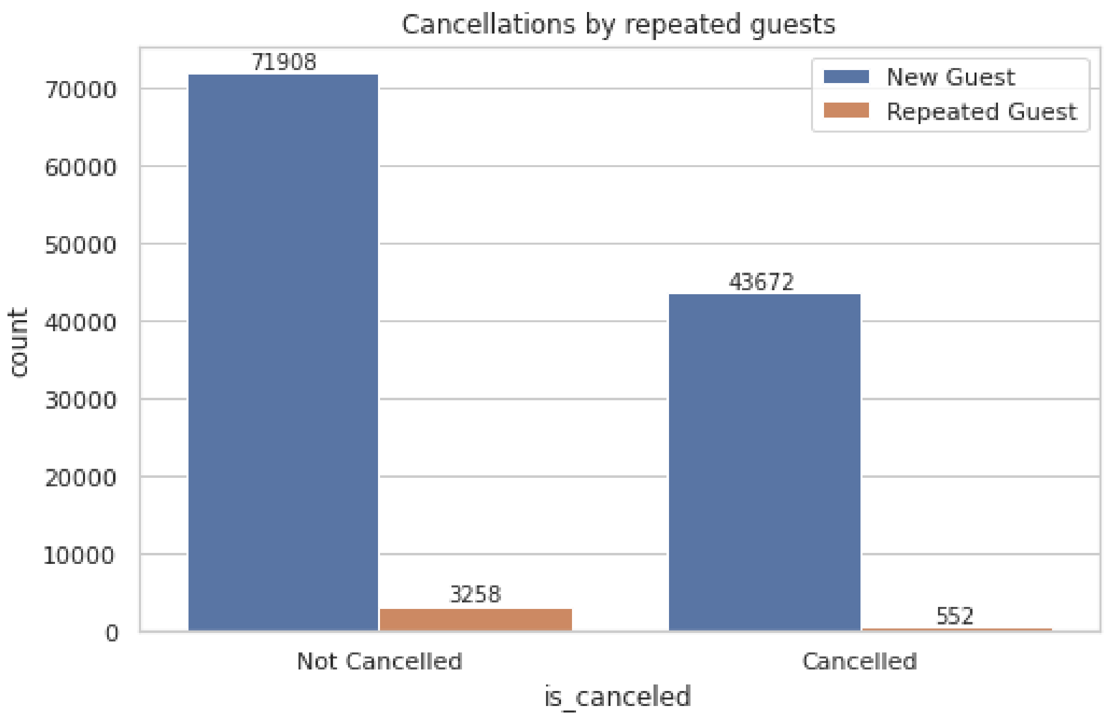
[19]:
```
#Calculate percentage of bookings across countries
country_data['%guest_across_countries']=np.
 ↪round(country_data['guest_count']*100/country_data['guest_count'].sum(),2)
#Plot the map
map=px.choropleth(country_data,
                  locations=country_data['country'],
                  color=country_data["%guest_across_countries"],
                  hover_name=country_data['country'],
                  color_continuous_scale=px.colors.sequential.Agsunset,
                              title="Total Bookings Across Countries")
map.show()
```

[20]:
```
d = df.groupby('lead_time').agg({'is_canceled':'sum', 'hotel':'count'}).
 ↪reset_index().rename(columns={'is_canceled':'cancelled_bookings', 'hotel':
 ↪'total_bookings'})
d['cancellation_percentage'] = (d['cancelled_bookings']/d['total_bookings'])*100
d.head()
plt.figure(figsize=(10,5))
sns.scatterplot(x='lead_time', y='cancellation_percentage', data=d)
plt.ylim((0,100))
plt.xlabel("Lead Time (days)")
plt.ylabel("% Cancellation")
plt.title("Effect of Lead time on Cancellations")
```

[20]:
```
Text(0.5, 1.0, 'Effect of Lead time on Cancellations')
```



[83]:
```python
plt.figure(figsize=(8,5))
ax = sns.countplot(x = "is_canceled", hue = 'is_repeated_guest', data = df)
plt.legend(['New Guest', 'Repeated Guest'])
plt.xticks(ticks=[0,1], labels=['Not Cancelled', 'Cancelled'])
plt.title("Cancellations by repeated guests")
for container in ax.containers:
    ax.bar_label(container)
```



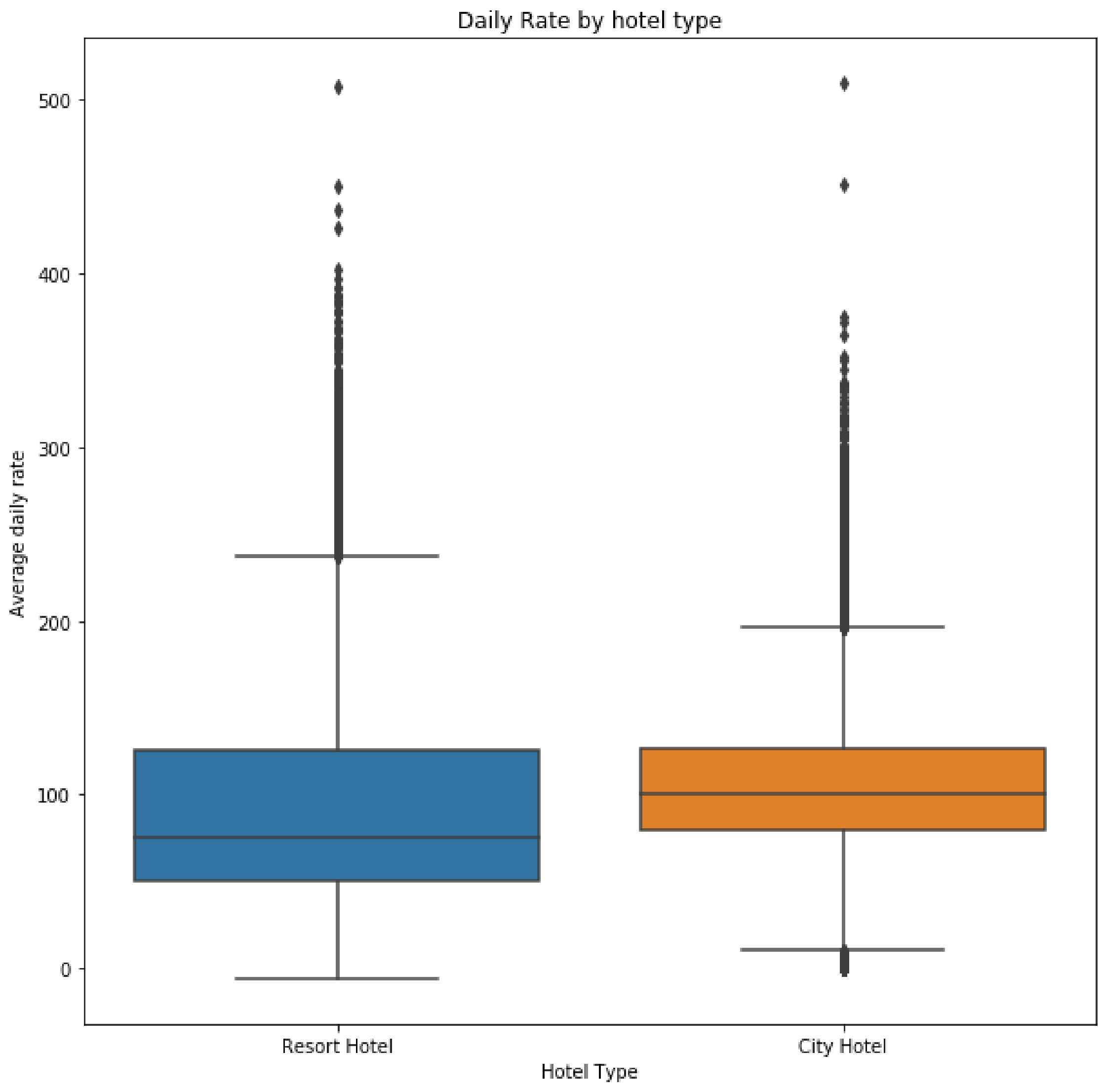
[22]:
```python
df['adr'].describe()
```

[22]:
```
count    119390.000000
mean        101.831122
std          50.535790
min          -6.380000
25%          69.290000
50%          94.575000
75%         126.000000
max        5400.000000
Name: adr, dtype: float64
```

[23]:
```python
plt.figure(figsize=(10,10))
df2 = df.drop(df[df['adr']==5400].index, axis=0, inplace=False)  # Removed an␣
 ↪extreme outlier (adr=5400) that made boxplot very squeezed to view
sns.boxplot(x='hotel', y='adr', data = df2)
```

```
plt.ylabel('Average daily rate')
plt.xlabel("Hotel Type")
plt.title("Daily Rate by hotel type")
```

[23]:
```
Text(0.5, 1.0, 'Daily Rate by hotel type')
```

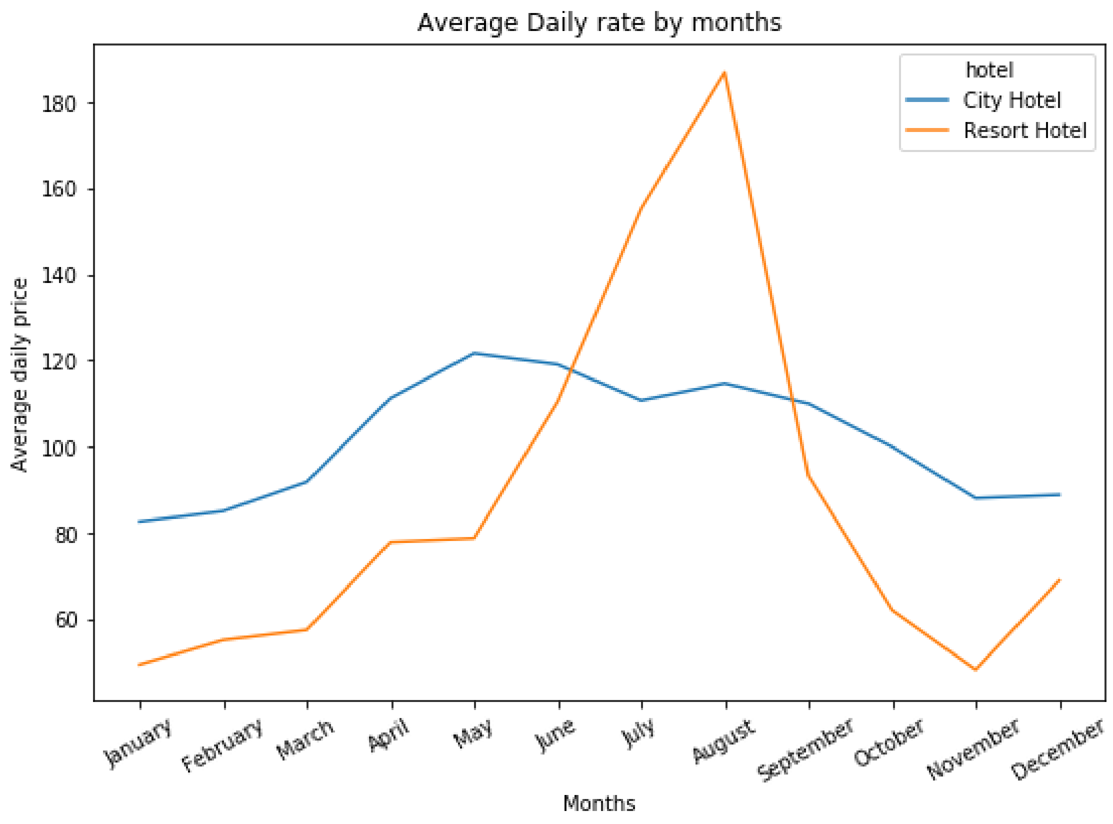


[24]:
```
d = df2.groupby(['hotel', 'arrival_date_month'])['adr'].mean().reset_index()
d['arrival_date_month'] = pd.Categorical(d['arrival_date_month'],␣
 ↪categories=months, ordered=True)
d.sort_values('arrival_date_month', inplace=True)
```

[25]:
```
plt.figure(figsize=(9,6))
sns.lineplot(x='arrival_date_month', y='adr', hue='hotel', data=d)
```

```
plt.ylabel("Average daily price")
plt.xlabel("Months")
p = plt.xticks(rotation=30)
plt.title("Average Daily rate by months")
```

[25]:
```
Text(0.5, 1.0, 'Average Daily rate by months')
```



2.1 Average Daily Rate trend over three years

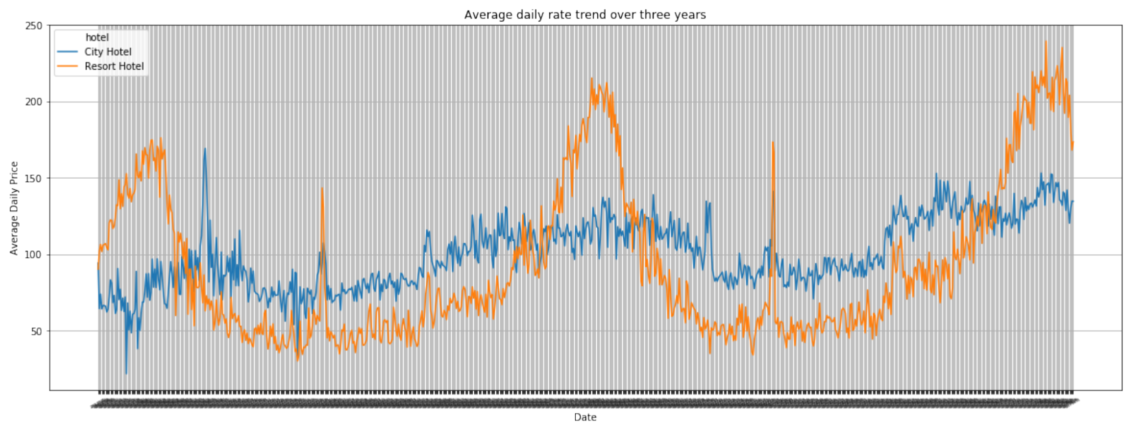
[26]:
```
def get_month(x):
    pre = ''
    if months.index(x)<9:
        pre = '0'
    return pre+str(months.index(x)+1)

def get_day(x):
    pre = ''
    if x<10:
        pre = '0'
    return pre+str(x)
```

```
[27]: df2['arrival_date'] = df2.arrival_date_year.apply(lambda x: str(x))+"-"+df2.
       ↪arrival_date_month.apply(get_month)+"-"+df2.arrival_date_day_of_month.
       ↪apply(get_day)
      df2.head(2)
```

```
[27]:           hotel  is_canceled  lead_time  arrival_date_year arrival_date_month  \
      0  Resort Hotel            0        342               2015               July
      1  Resort Hotel            0        737               2015               July

         arrival_date_week_number  arrival_date_day_of_month  \
      0                        27                          1
      1                        27                          1

         stays_in_weekend_nights  stays_in_week_nights  adults  …  agent  company  \
      0                        0                     0       2  …    NaN      NaN
      1                        0                     0       2  …    NaN      NaN

        days_in_waiting_list customer_type  adr required_car_parking_spaces  \
      0                    0     Transient  0.0                           0
      1                    0     Transient  0.0                           0

         total_of_special_requests  reservation_status  reservation_status_date  \
      0                          0           Check-Out               2015-07-01
      1                          0           Check-Out               2015-07-01

        arrival_date
      0   2015-07-01
      1   2015-07-01

      [2 rows x 33 columns]
```

```
[28]: d = df2.groupby(['hotel','arrival_date'])['adr'].mean().reset_index().
       ↪sort_values('arrival_date')
      # fig = plt.figure(figsize=(20,7))
      fig, ax = plt.subplots(figsize=(20, 7))
      sns.lineplot(x='arrival_date', y='adr', hue='hotel', data=d)
      plt.xlabel("Date")
      plt.ylabel("Average Daily Price")
      plt.grid()
      # fig.autofmt_xdate()
      p = plt.xticks(rotation=30)
      ax.tick_params(axis='x', labelsize=3)
      plt.title("Average daily rate trend over three years")
```

```
[28]: Text(0.5, 1.0, 'Average daily rate trend over three years')
```



[38]:
```
df2
```

[38]:
```
                hotel  is_canceled  lead_time  arrival_date_year  \
0        Resort Hotel            0        342               2015
1        Resort Hotel            0        737               2015
2        Resort Hotel            0          7               2015
3        Resort Hotel            0         13               2015
4        Resort Hotel            0         14               2015
…                 …          …          …                 …
119385     City Hotel            0         23               2017
119386     City Hotel            0        102               2017
119387     City Hotel            0         34               2017
119388     City Hotel            0        109               2017
119389     City Hotel            0        205               2017

       arrival_date_month  arrival_date_week_number  \
0                    July                        27
1                    July                        27
2                    July                        27
3                    July                        27
4                    July                        27
…                  …                        …
119385             August                        35
119386             August                        35
119387             August                        35
119388             August                        35
119389             August                        35

        arrival_date_day_of_month  stays_in_weekend_nights  \
0                               1                        0
1                               1                        0
2                               1                        0
```

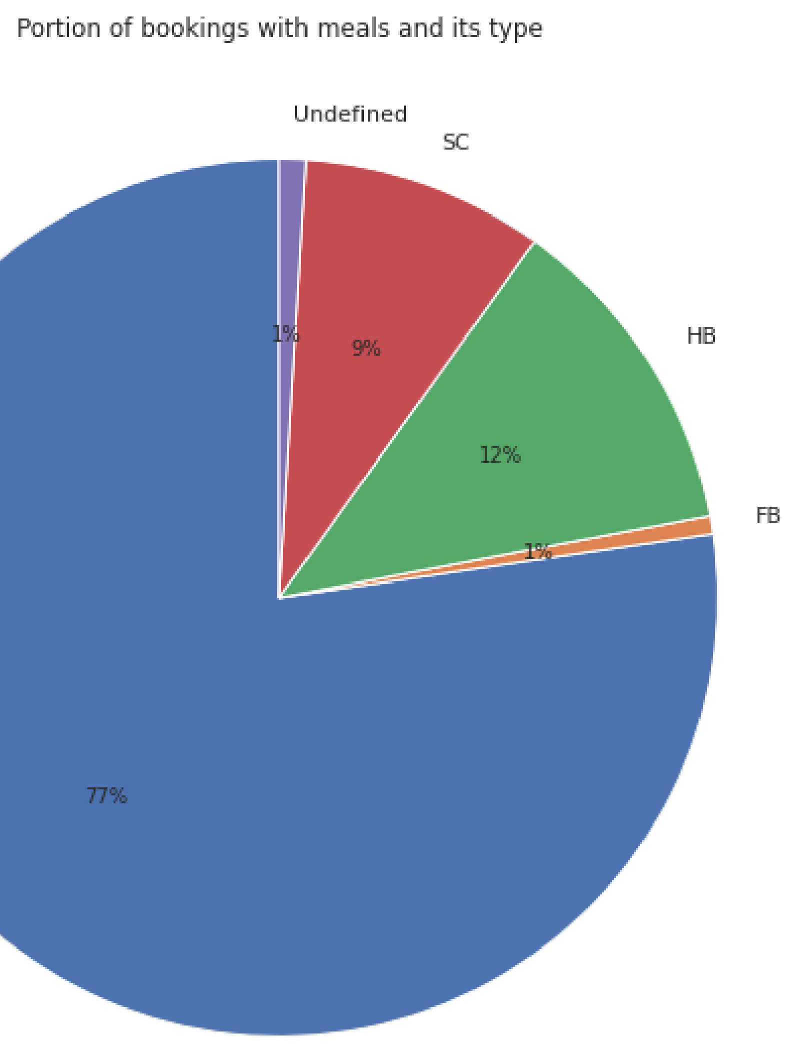
```
3                             1                        0
4                             1                        0
…                          …                      …
119385                       30                        2
119386                       31                        2
119387                       31                        2
119388                       31                        2
119389                       29                        2

        stays_in_week_nights  adults  …  agent  company  \
0                          0       2  …    NaN      NaN
1                          0       2  …    NaN      NaN
2                          1       1  …    NaN      NaN
3                          1       1  …  304.0      NaN
4                          2       2  …  240.0      NaN
…                        …     …   …    …      …
119385                     5       2  …  394.0      NaN
119386                     5       3  …    9.0      NaN
119387                     5       2  …    9.0      NaN
119388                     5       2  …   89.0      NaN
119389                     7       2  …    9.0      NaN

       days_in_waiting_list customer_type     adr required_car_parking_spaces  \
0                         0     Transient    0.00                           0
1                         0     Transient    0.00                           0
2                         0     Transient   75.00                           0
3                         0     Transient   75.00                           0
4                         0     Transient   98.00                           0
…                       …           …     …                         …
119385                    0     Transient   96.14                           0
119386                    0     Transient  225.43                           0
119387                    0     Transient  157.71                           0
119388                    0     Transient  104.40                           0
119389                    0     Transient  151.20                           0

        total_of_special_requests  reservation_status  \
0                               0           Check-Out
1                               0           Check-Out
2                               0           Check-Out
3                               0           Check-Out
4                               1           Check-Out
…                            …              …
119385                          0           Check-Out
119386                          2           Check-Out
119387                          4           Check-Out
119388                          0           Check-Out
119389                          2           Check-Out
```

```
        reservation_status_date arrival_date
0                    2015-07-01   2015-07-01
1                    2015-07-01   2015-07-01
2                    2015-07-02   2015-07-01
3                    2015-07-02   2015-07-01
4                    2015-07-03   2015-07-01
…                           …          …
119385               2017-09-06   2017-08-30
119386               2017-09-07   2017-08-31
119387               2017-09-07   2017-08-31
119388               2017-09-07   2017-08-31
119389               2017-09-07   2017-08-29

[119389 rows x 33 columns]
```

2.2 Bữa ăn

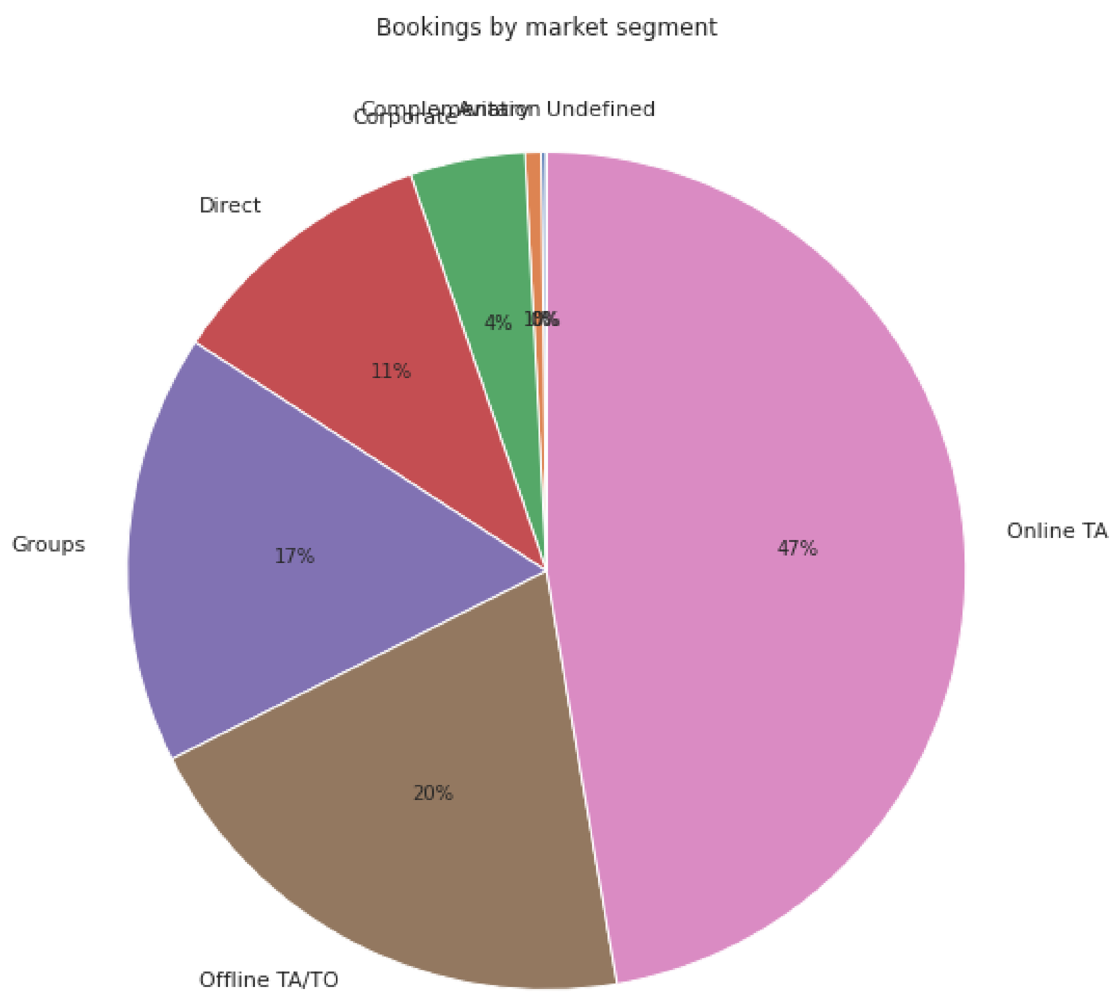
[73]:
```
d = df['meal'].value_counts()
d = d.sort_index()
plt.figure(figsize=(10,10))
p = plt.pie(d, labels=d.index, autopct="%.0f%%", startangle=90)
plt.title("Portion of bookings with meals and its type")
```

[73]:
```
Text(0.5, 1.0, 'Portion of bookings with meals and its type')
```



```
[45]: d = df['market_segment'].value_counts()
      d = d.sort_index()
      plt.figure(figsize=(10,10))
      explode = (0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0)
      p = plt.pie(d, labels=d.index, autopct="%.0f%%", explode = explode,␣
       ↪startangle=90)
      plt.title("Bookings by market segment")
```

```
[45]: Text(0.5, 1.0, 'Bookings by market segment')
```



```
[40]: d.index
```

```
[40]: Index(['Aviation', 'Complementary', 'Corporate', 'Direct', 'Groups',
             'Offline TA/TO', 'Online TA', 'Undefined'],
            dtype='object')
```

```
[17]: df['market_segment'].value_counts()
```

```
[17]: Online TA        56477
      Offline TA/TO    24219
      Groups           19811
      Direct           12606
      Corporate         5295
```

```
Complementary      743
Aviation           237
Undefined            2
Name: market_segment, dtype: int64
```

[65]:
```
otherSegment = {
    "Online TA" : "Online TA",
    "Offline TA/TO": "Offline TA/TO",
    "Groups": "Groups",
    "Direct": "Direct",
    "Corporate": "Corporate",
    "Complementary": "Others",
    "Aviation": "Others",
    "Undefined": "Others",
}
```

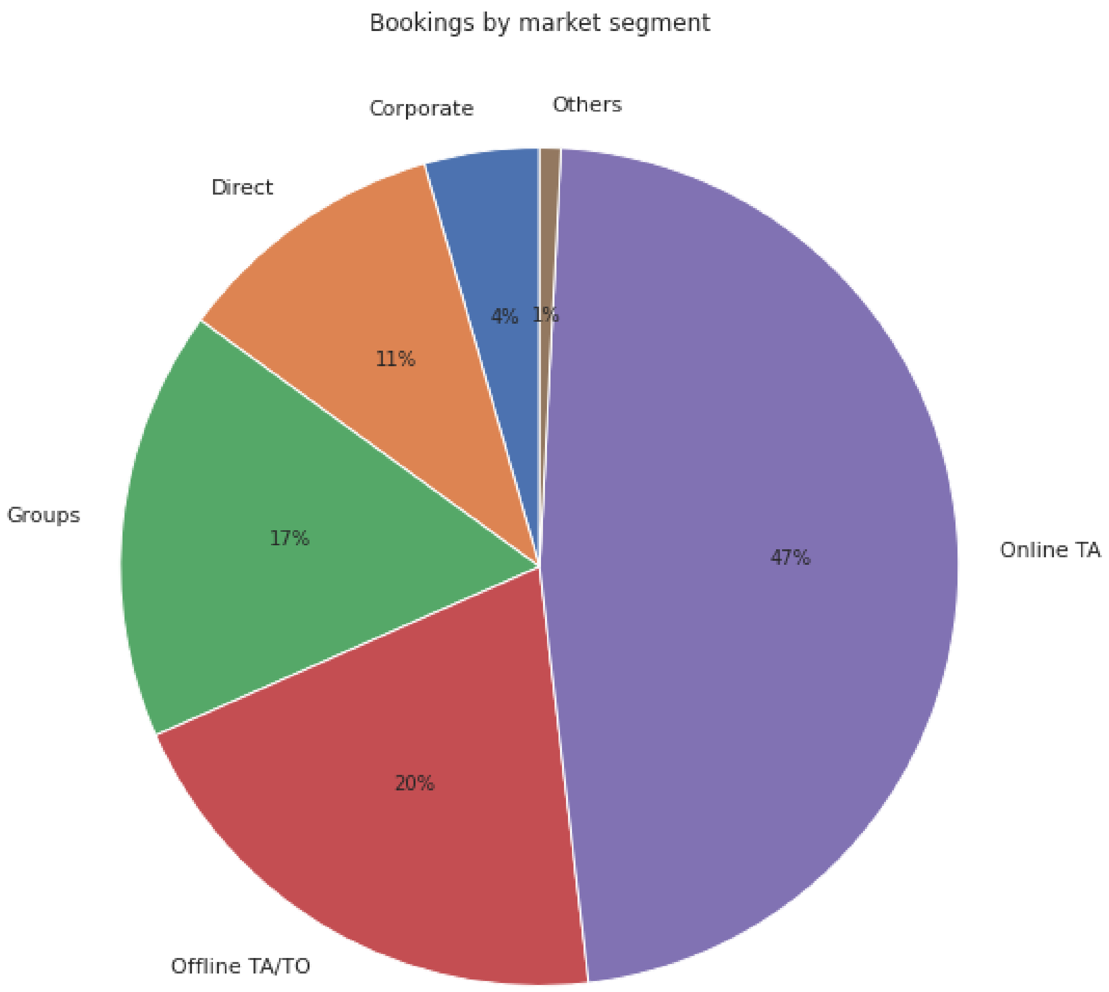
[66]:
```
df['market_segment']
```

[66]:
```
0                 Direct
1                 Direct
2                 Direct
3              Corporate
4              Online TA
                 …
119385    Offline TA/TO
119386        Online TA
119387        Online TA
119388        Online TA
119389        Online TA
Name: market_segment, Length: 119390, dtype: object
```

[67]:
```
data = df['market_segment'].map(otherSegment)
```

[70]:
```
data = data.value_counts()
data = data.sort_index()
plt.figure(figsize=(10,10))
#explode = (0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0)
p = plt.pie(data, labels=data.index, autopct="%.0f%%", startangle=90)
plt.title("Bookings by market segment")
```

[70]:
```
Text(0.5, 1.0, 'Bookings by market segment')
```



[69]:
```
data.value_counts()
```

[69]:
```
Online TA        56477
Offline TA/TO    24219
Groups           19811
Direct           12606
Corporate         5295
Others             982
Name: market_segment, dtype: int64
```

[68]:
```
data
```

```
[68]: 0                 Direct
      1                 Direct
      2                 Direct
      3              Corporate
      4              Online TA
                       …
      119385     Offline TA/TO
      119386         Online TA
      119387         Online TA
      119388         Online TA
      119389         Online TA
      Name: market_segment, Length: 119390, dtype: object
```

```
[ ]:
```