

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np
from datetime import datetime

from pandas.plotting import scatter_matrix
import warnings
warnings.filterwarnings("ignore")
```


Dicionário

- `hotel` - dois tipos de hotel, Resort Hotel e City Hotel.
- `is_canceled` - indica se a reserva foi cancelada (1) ou não (0)
- `lead_time` - número de dias transcorridos entre a data de entrada da reserva no sistema e a data de chegada ao hotel
- `arrival_date_year` - ano da data de chegada
- `arrival_date_month` - mês da data de chegada
- `arrival_date_week_number` - número da semana da data de chegada
- `arrival_date_day_of_month` - dia do mês da data de chegada
- `stays_in_weekend_nights` - número de noites de fim de semana (sábado ou domingo) em que o hóspede ficou ou reservou para ficar no hotel.
- `stays_in_week_nights` - número de noites da semana (segunda a sexta) em que o hóspede ficou ou reservou para ficar no hotel.
- `adults` - número de adultos
- `children` - número de crianças
- `babies` - número de bebês
- `meal` - tipo de refeição reservada.
- `country` - país de origem
- `market_segment` - segmento de mercado
- `distribution_channel` - canal de distribuição da reserva.
- `is_repeated_guest` - valor indicando se o nome da reserva foi de um convidado repetido (1) ou não (0)
- `previous_cancellations` - número de reservas anteriores que foram canceladas pelo cliente antes da reserva atual
- `previous_bookings_not_canceled` - número de reservas anteriores que não foram canceladas pelo cliente antes da reserva atual

- `reserved_room_type` - código do tipo de quarto reservado.
- `assigned_room_type` - código para o tipo de quarto designado para a reserva.
- `booking_changes` - número de mudanças/alterações feitas na reserva desde o momento em que a reserva foi inserida no sistema até o momento do check-in ou cancelamento.
- `deposit_type` - indicação sobre se o cliente fez um depósito para garantir a reserva.
- `agent` - ID da agência de viagens que fez a reserva
- `company` - ID da empresa/entidade que fez a reserva ou responsável pelo pagamento da reserva.
- `days_in_waiting_list` - Número de dias em que a reserva estava na lista de espera antes de ser confirmada ao cliente
- `customer_type` - tipo da reserva.
- `adr` - taxa diária média
- `required_car_parking_spaces` - número de vagas de estacionamento necessárias para o cliente
- `total_of_special_requests` - número de pedidos especiais feitos pelo cliente
- `reservation_status` - status da reserva
- `reservation_status_date` - data na qual o último status foi definido.

✓ Preparação e Limpeza Dataset

```
df = pd.read_csv('hotel_bookings.csv')
df.head()
```



	hotel	is_canceled	lead_time	arrival_date_year	arrival_date_month	arr:
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 x 32 columns

```
print('Total de variáveis:', df.shape[1])
print('Total de entradas:', df.shape[0])
```



```
Total de variáveis: 32
Total de entradas: 119390
```

```
df['is_canceled'].mean()
```



```
0.37041628277075134
```

```
pd.DataFrame({'tipo de dados':df.dtypes,
              'dados_ausentes(%)': (df.isnull().sum()/df.shape[0])*100,
              'valores unicos': df.nunique()})
```

```
df.drop(columns = ['agent', 'company'], inplace=True)
```

```
pd.DataFrame({'tipo de dados':df.dtypes,
              'dados_ausentes(%)': (df.isnull().sum()/df.shape[0])*100,
              'valores unicos': df.nunique()})
```

```
df = df.dropna(subset=['children', 'country']).reset_index(drop=True)
```

```
df['total_days'] = df['stays_in_week_nights'] + df['stays_in_weekend_nights']
df.drop(columns = ['stays_in_week_nights', 'stays_in_weekend_nights'], inplace=True)
df.head()
```

```
↵
```

	hotel	is_canceled	lead_time	arrival_date_year	arrival_date_month	arr:
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 x 29 columns

```
filtro = (df['babies'] == 0) & (df['children'] == 0) & (df['adults'] == 0)
df = df[~filtro]
```

```
df.drop_duplicates(inplace=True)
```

```
df.reset_index(drop=True, inplace=True)
```

df



	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
2	Resort Hotel	0	7	2015	July
3	Resort Hotel	0	13	2015	July
4	Resort Hotel	0	14	2015	July
...
86748	City Hotel	0	23	2017	August
86749	City Hotel	0	102	2017	August
86750	City Hotel	0	34	2017	August
86751	City Hotel	0	109	2017	August
86752	City Hotel	0	205	2017	August


86753 rows × 29 columns

#substituindo os codigos

#replace Undefined, BB, FB, HB, SC to its meaning.

```
df['meal'].replace(['Undefined', 'BB', 'FB', 'HB', 'SC'],  
                  [ 'No Meal', 'Breakfast', 'Full Board', 'Half Board', 'No Me  
inplace = True)
```

```
countries_code = pd.read_csv('https://raw.githubusercontent.com/luke/ISO-3166-
countries_code.head()
```



	name	alpha- 2	alpha- 3	country- code	iso_3166- 2	region	sub- region	intermedi re
0	Afghanistan	AF	AFG	4	ISO 3166- 2:AF	Asia	Southern Asia	
1	Åland Islands	AX	ALA	248	ISO 3166- 2:AX	Europe	Northern Europe	
2	ISO 3166- 2:...	...	Southern	

```
dict_country= {}
for i,j in zip(countries_code['alpha-3'], countries_code['name']):
    dict_country[i] = j
```

```
df['country'].replace(dict_country, inplace= True)
```

```
df['country'].replace('United Kingdom of Great Britain and Northern Ireland', '

```

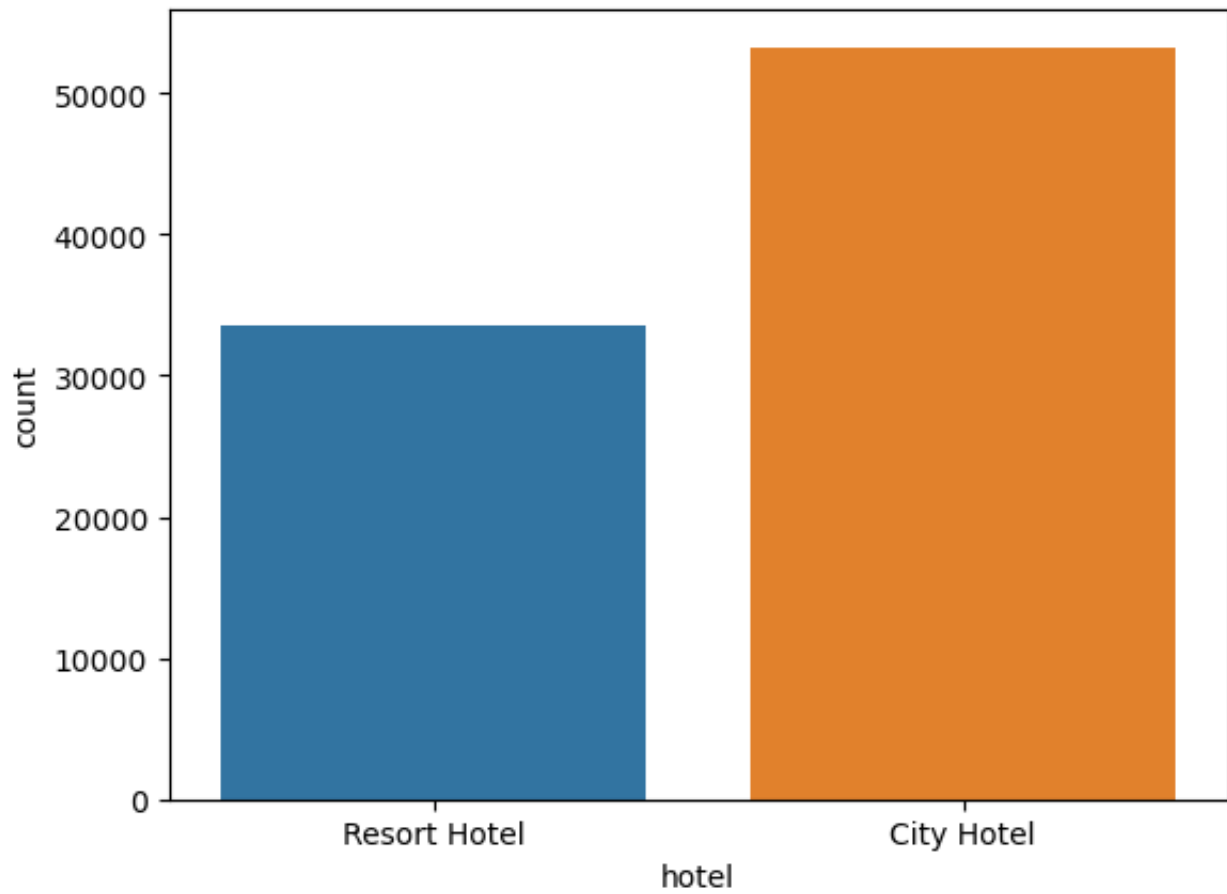
```
#is_canceled
confirmed = df.loc[df['is_canceled'] == 0]
```

✓ Análise Exploratória e Visualização

✓ DF

✓ Reservas realizadas

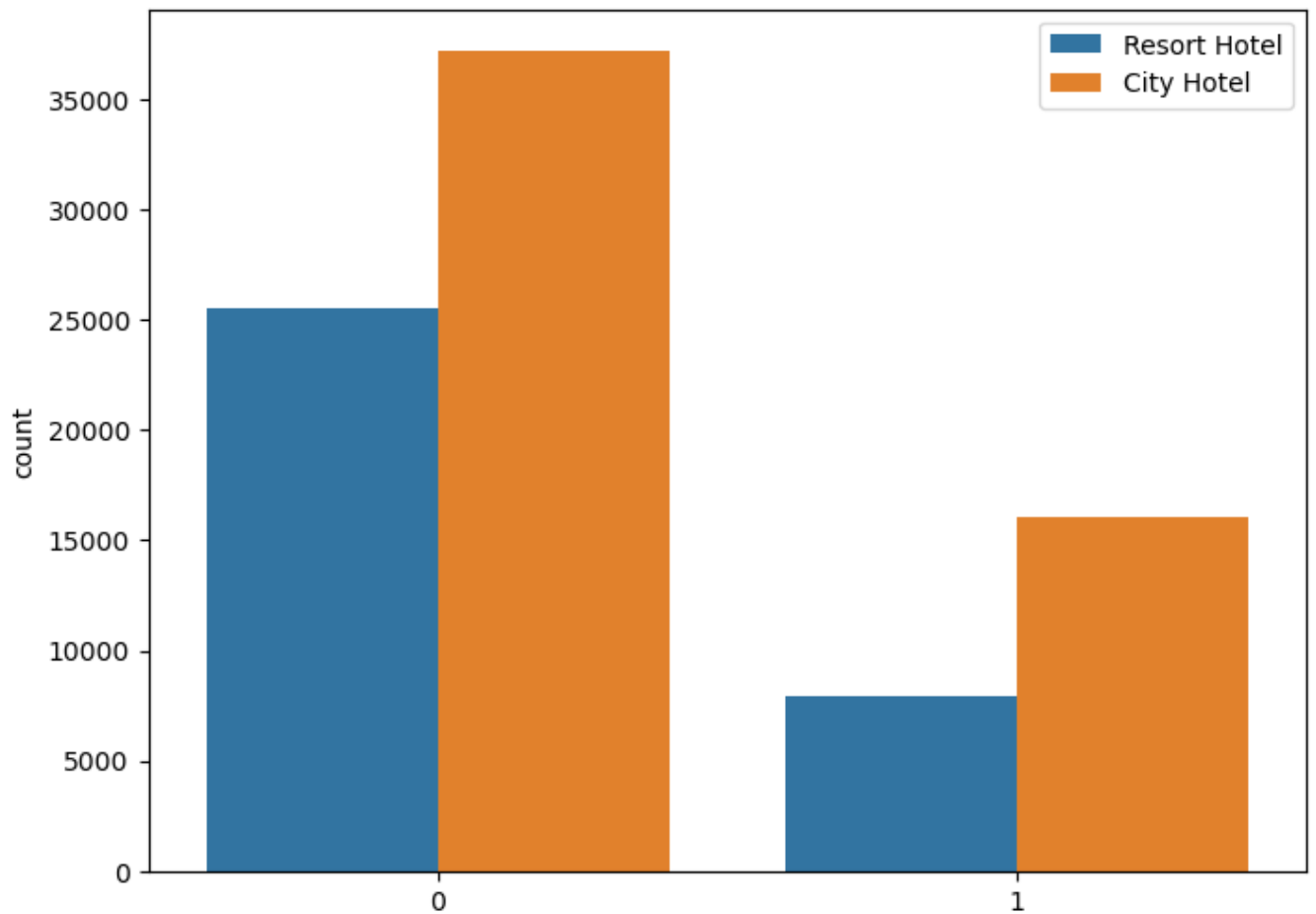
```
sns.countplot(x='hotel', data = df);
```



✓ Quantidade de cancelamento

```
plt.figure(figsize=(8,6))
sns.countplot(x = df['is_canceled'], hue = 'hotel', data = df)
plt.xlabel(' ');
plt.legend(title = '')
```

 <matplotlib.legend.Legend at 0x7fae09337820>

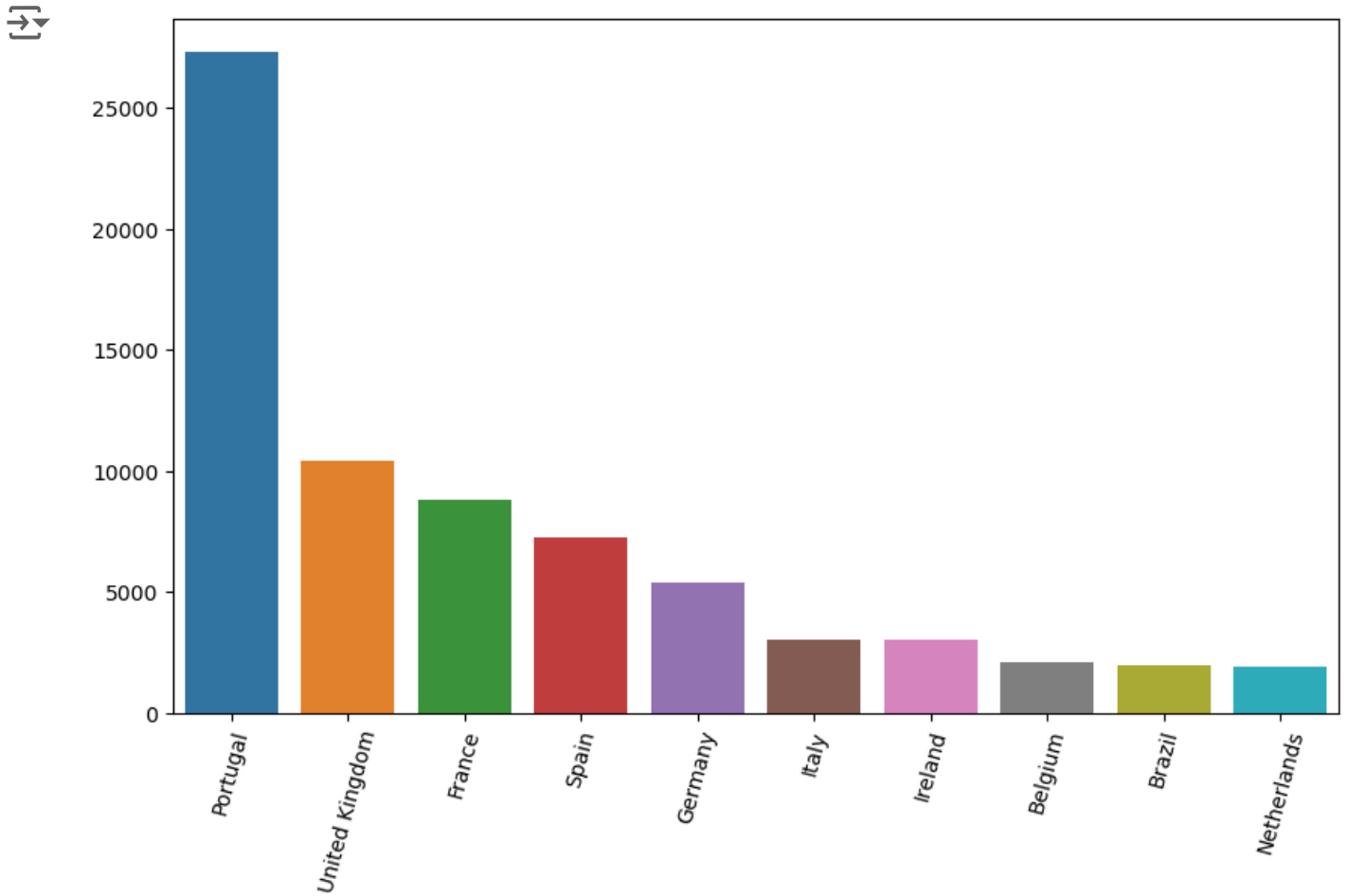


✓ De onde são os hóspedes?


```
# quantidade de reservas por país  
# os dez maiores  
df['country'].value_counts().iloc[0:10]
```

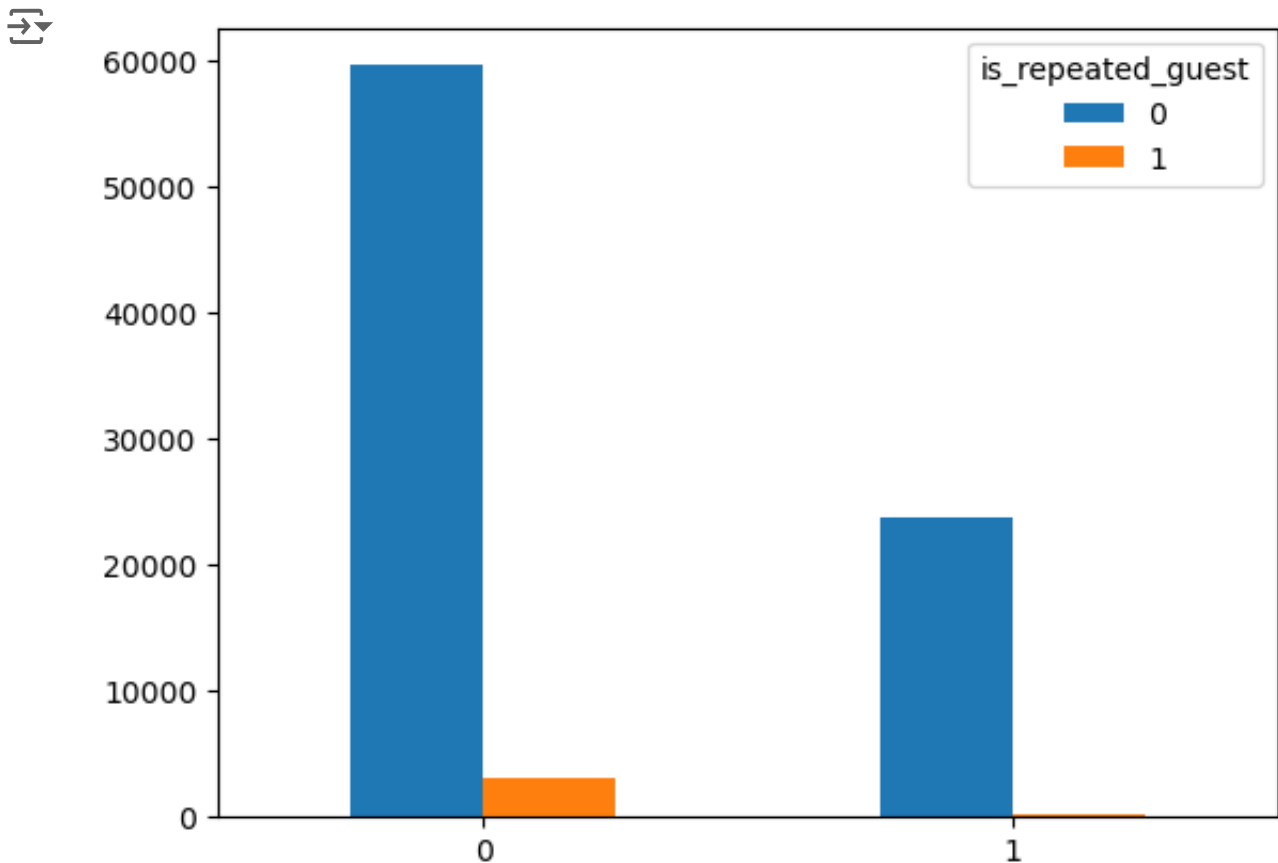
```
↔ Portugal      27338  
   United Kingdom 10422  
   France        8823  
   Spain         7242  
   Germany       5383  
   Italy         3060  
   Ireland       3015  
   Belgium       2081  
   Brazil        1991  
   Netherlands   1910  
Name: country, dtype: int64
```

```
plt.figure(figsize=(10,6))
sns.barplot(x = df['country'].value_counts().iloc[0:10].index, y = df['country']
plt.xticks(rotation = 75);
```



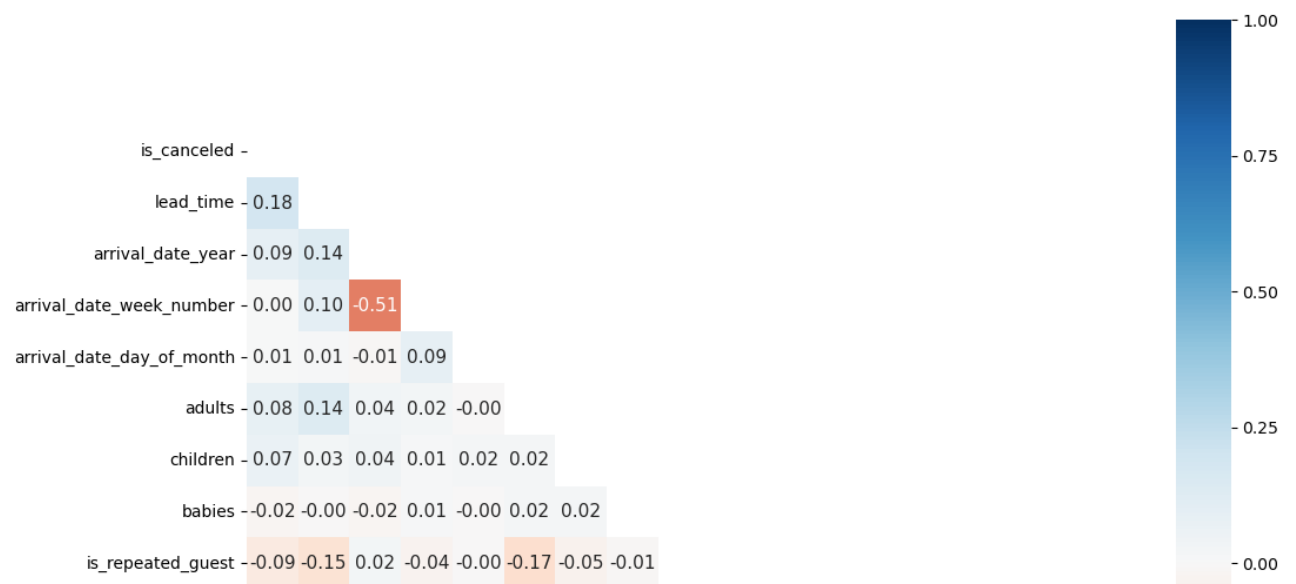
✓ Efeito do hospede repetido nos cancelamentos

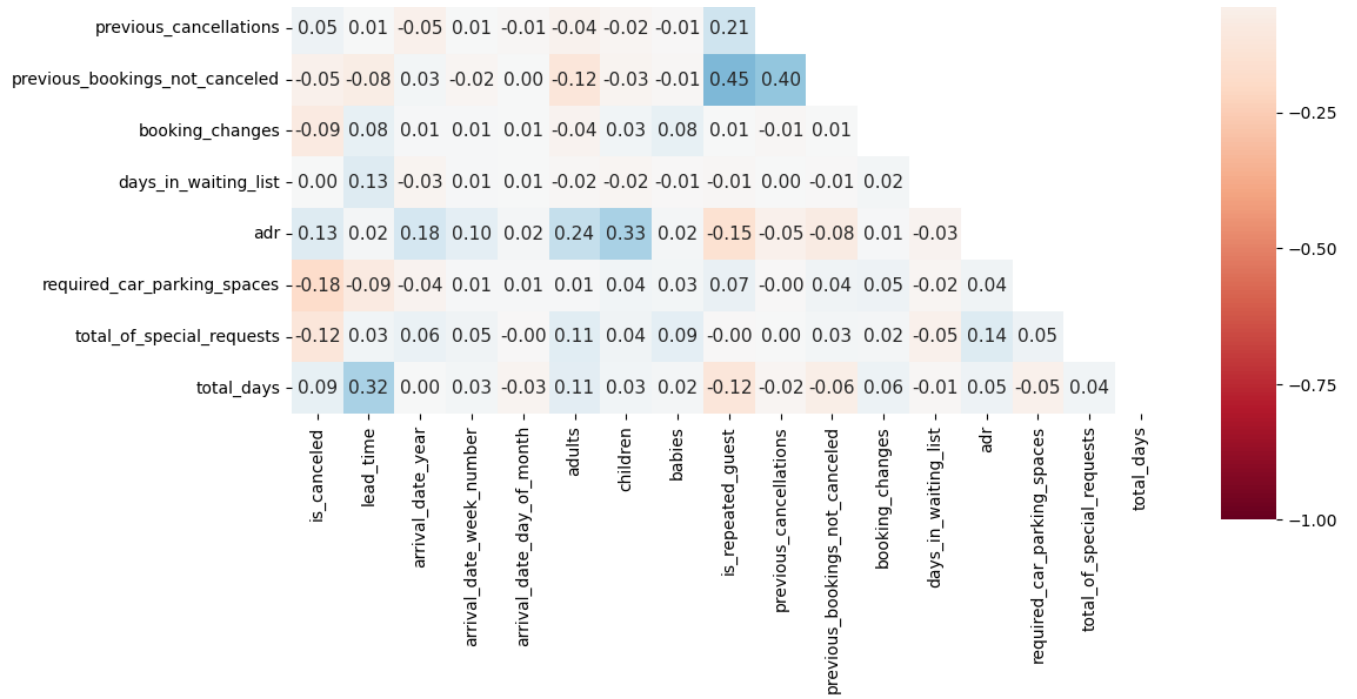
```
df.groupby('is_canceled')['is_repeated_guest'].value_counts().unstack().plot(kind='bar')
plt.xlabel('')
plt.xticks(rotation = 0)
plt.tight_layout;
```



```
plt.figure(figsize=(12,12))
mask = np.triu(np.ones_like(df.corr()))
sns.heatmap(df.corr(), cmap = 'RdBu', linecolor= 'white', annot=True,fmt='.2f',
```

<Axes: >



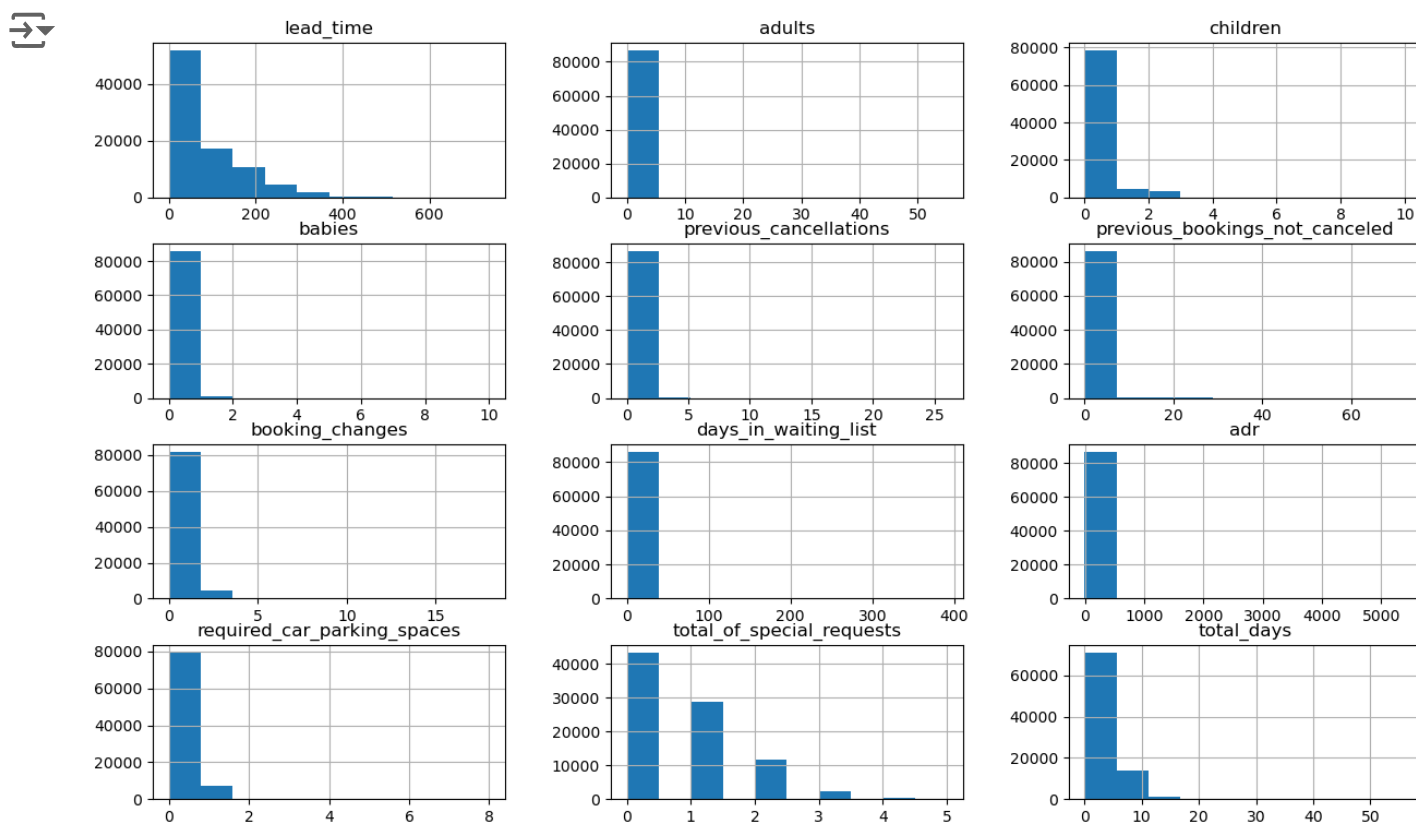


```
df.describe()
```




	is_canceled	lead_time	arrival_date_year	arrival_date_week_number
count	86753.000000	86753.000000	86753.000000	86753.000000
mean	0.276325	80.278803	2016.211900	26.838830
std	0.447182	86.108773	0.685937	13.649083
min	0.000000	0.000000	2015.000000	1.000000
25%	0.000000	12.000000	2016.000000	16.000000
50%	0.000000	50.000000	2016.000000	27.000000
75%	1.000000	125.000000	2017.000000	37.000000
max	1.000000	737.000000	2017.000000	53.000000

```
df.hist(['lead_time', 'adults', 'children', 'babies', 'previous_cancellations',  
        figsize=(15,9),);
```



```
df.describe()
```



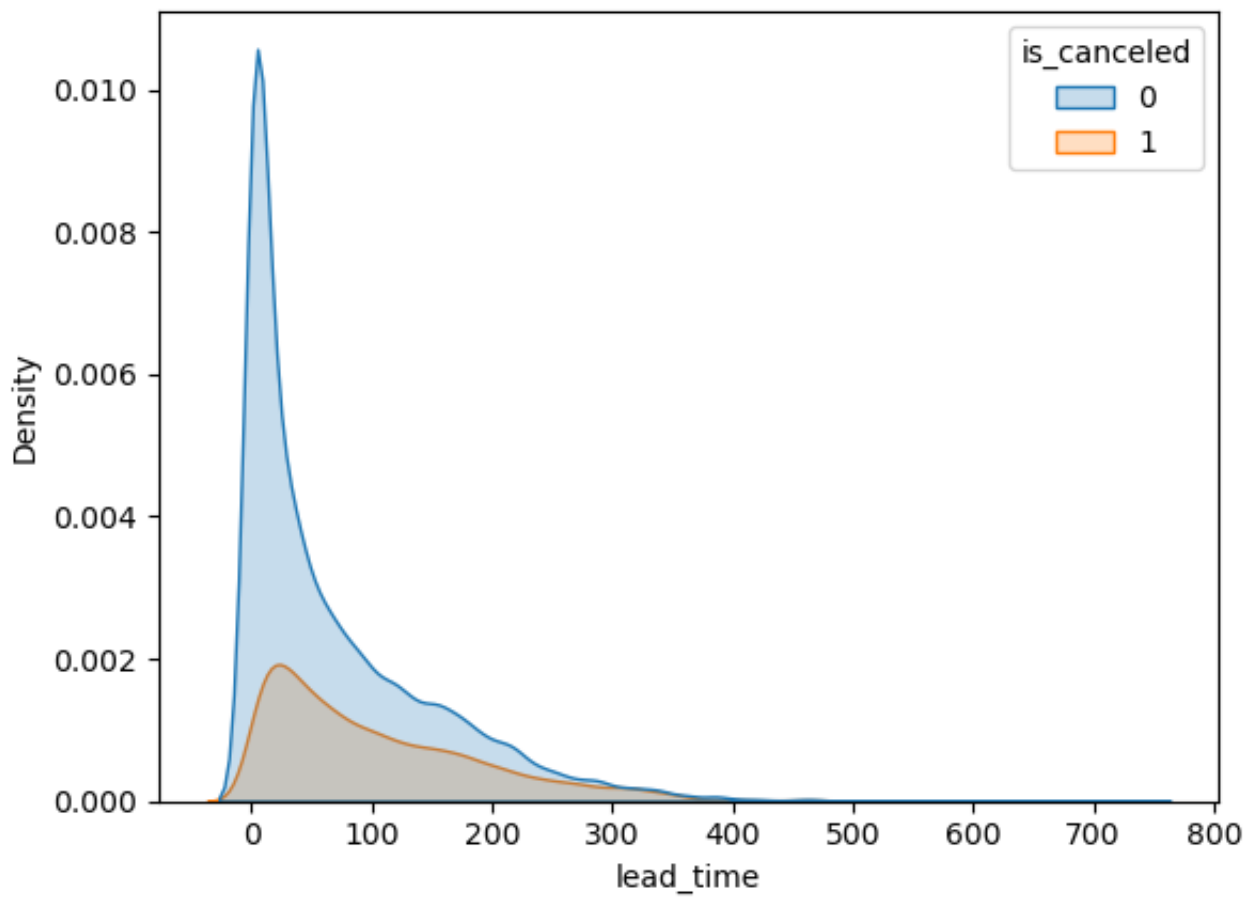
	is_canceled	lead_time	arrival_date_year	arrival_date_week_number
count	86753.000000	86753.000000	86753.000000	86753.000000
mean	0.276325	80.278803	2016.211900	26.838830
std	0.447182	86.108773	0.685937	13.649083
min	0.000000	0.000000	2015.000000	1.000000
25%	0.000000	12.000000	2016.000000	16.000000
50%	0.000000	50.000000	2016.000000	27.000000
75%	1.000000	125.000000	2017.000000	37.000000
max	1.000000	737.000000	2017.000000	53.000000

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 86753 entries, 0 to 86752
Data columns (total 29 columns):
#   Column                                     Non-Null Count  Dtype
---  -
0   hotel                                     86753 non-null  object
1   is_canceled                             86753 non-null  int64
2   lead_time                               86753 non-null  int64
3   arrival_date_year                       86753 non-null  int64
4   arrival_date_month                     86753 non-null  object
5   arrival_date_week_number               86753 non-null  int64
6   arrival_date_day_of_month              86753 non-null  int64
7   adults                                  86753 non-null  int64
8   children                                86753 non-null  float64
9   babies                                  86753 non-null  int64
10  meal                                    86753 non-null  object
11  country                                86753 non-null  object
12  market_segment                         86753 non-null  object
13  distribution_channel                   86753 non-null  object
14  is_repeated_guest                     86753 non-null  int64
15  previous_cancellations                 86753 non-null  int64
16  previous_bookings_not_canceled         86753 non-null  int64
17  reserved_room_type                    86753 non-null  object
18  assigned_room_type                    86753 non-null  object
19  booking_changes                        86753 non-null  int64
20  deposit_type                           86753 non-null  object
21  days_in_waiting_list                  86753 non-null  int64
22  customer_type                          86753 non-null  object
23  adr                                    86753 non-null  float64
24  required_car_parking_spaces            86753 non-null  int64
25  total_of_special_requests              86753 non-null  int64
26  reservation_status                    86753 non-null  object
27  reservation_status_date                86753 non-null  object
28  total_days                            86753 non-null  int64
dtypes: float64(2), int64(15), object(12)
memory usage: 19.2+ MB
```



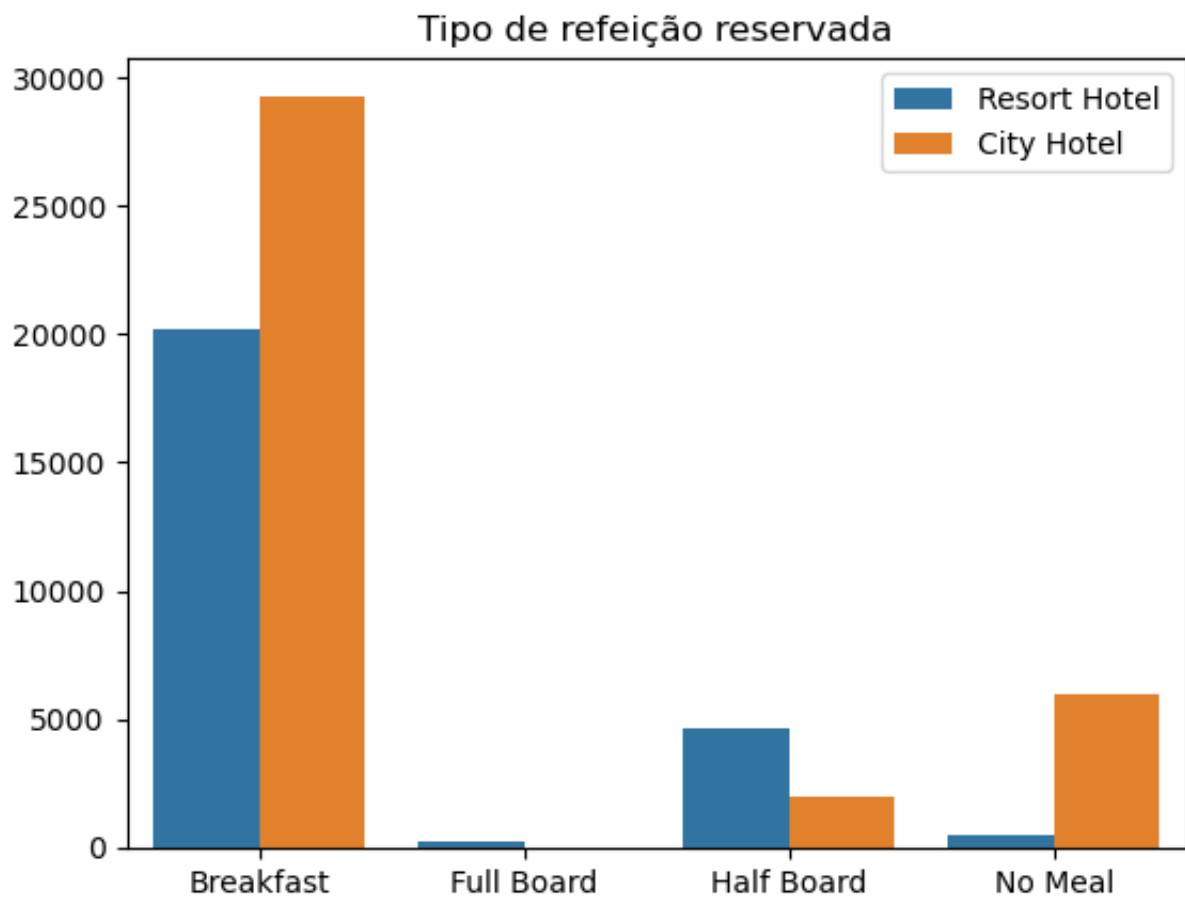
```
# Tempo de espera e cancelamento  
sns.kdeplot(data=df, x='lead_time', hue = 'is_canceled', shade=True);
```



✓ Confirmed

✓ Refeição realizada

```
ax = sns.countplot(data = confirmed , x = 'meal', hue = 'hotel')
ax.set_xlabel(' ')
ax.set_ylabel(' ')
plt.legend(loc = 'upper right')
ax.set_title('Tipo de refeição reservada')
plt.tight_layout;
```



✓ Meses com mais reservas

```
confirmed.groupby('arrival_date_year')['arrival_date_month'].value_counts()
```

```
↔ arrival_date_year  arrival_date_month
2015                September      2255
                  October         2233
                  August          1863
                  December        1582
                  November        1397
2016                July          1147
                  October        3000
                  August         2965
                  March          2906
                  May            2823
                  September       2759
                  July           2738
                  April          2731
                  June           2593
                  November       2482
                  February       2240
                  December       2107
2017                January       1514
                  May            3065
                  July           2931
                  June           2799
                  August         2771
                  April          2729
                  March          2699
                  February       2374
                  January        2078
Name: arrival_date_month, dtype: int64
```

```
month = ['January', 'February', 'March', 'April', 'May', 'June', 'July', 'August', 'September', 'October', 'November', 'December']
```

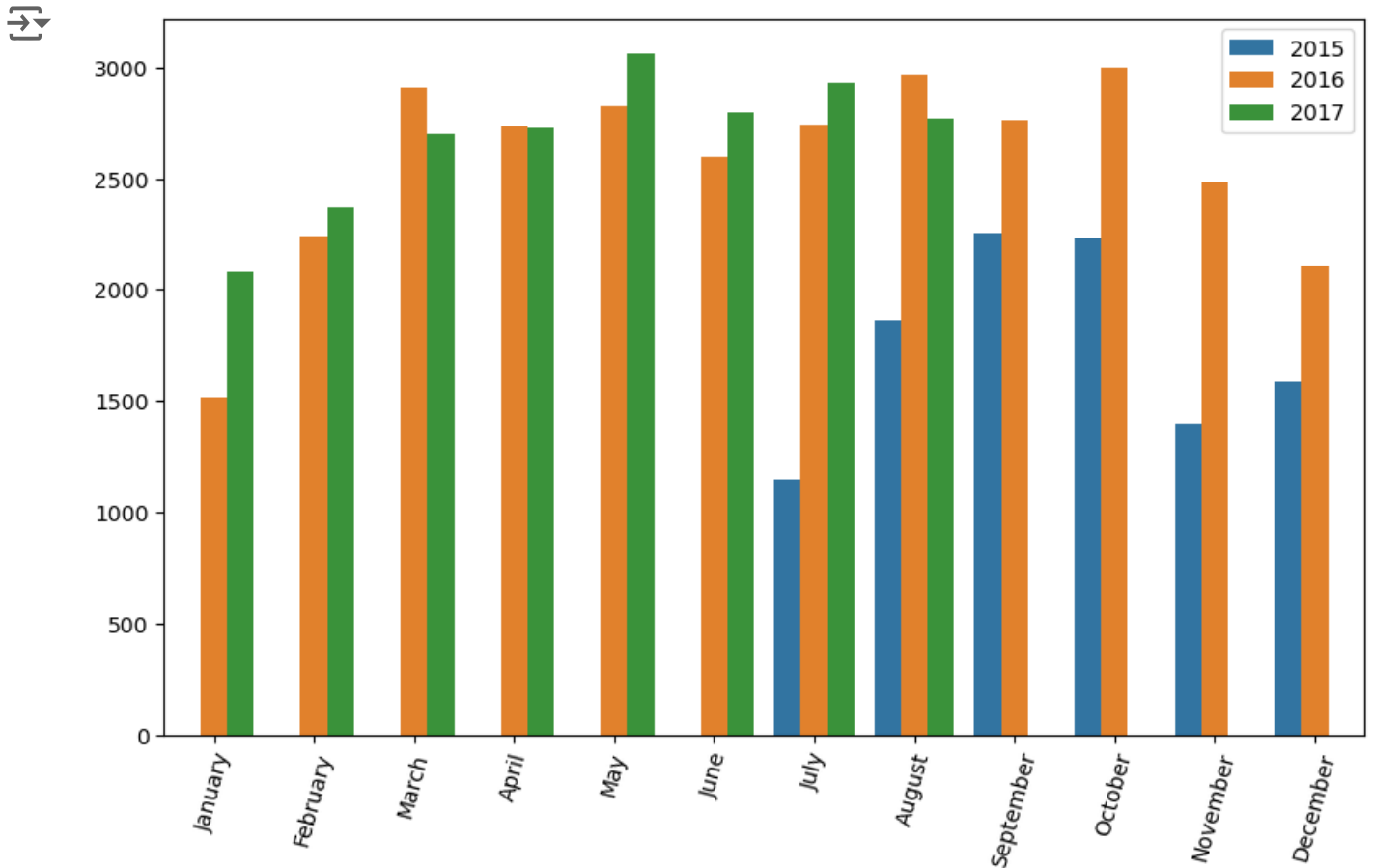
```
confirmed.head()
```



	hotel	is_canceled	lead_time	arrival_date_year	arrival_date_month	arr:
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 × 29 columns

```
plt.figure(figsize=(10,6))
sns.countplot(data = confirmed, x = 'arrival_date_month', hue = 'arrival_date_y')
plt.xticks(rotation = 75);
plt.legend(title = '')
plt.xlabel('')
plt.ylabel('')
plt.tight_layout;
```



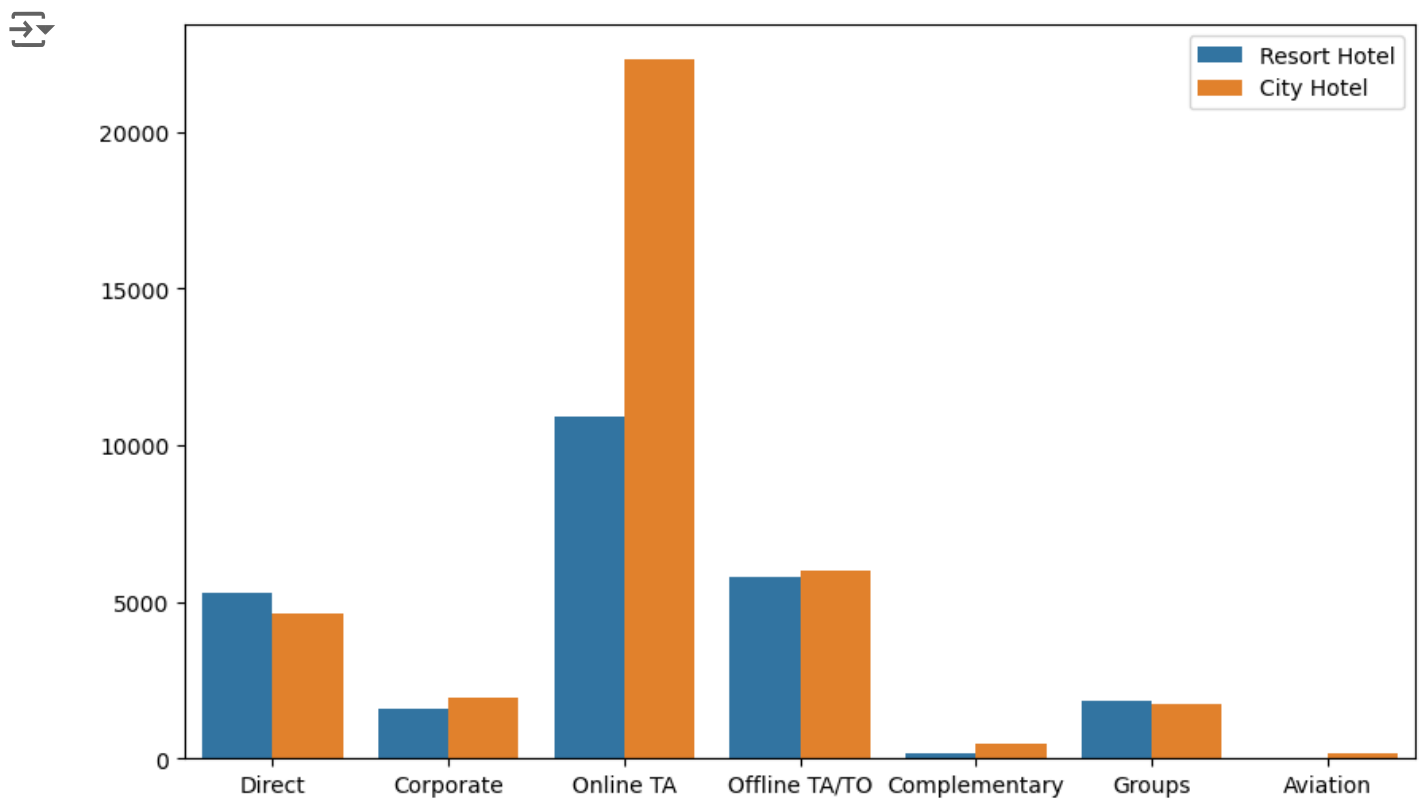
✓ Quanto tempo dura a hospedagem?

```
#round(confirmed.groupby(['arrival_date_year', 'hotel'])['total_days'].mean(),  
print('O tempo médio das hospedagens é de: \n Em 2015, {:.2f} noites \n Em 2016
```

```
↗ 0 tempo médio das hospedagens é de:  
   Em 2015, 3.70 noites  
   Em 2016, 3.38 noites  
   Em 2017, 3.55 noites
```

▼ Reservas realizadas por segmentos

```
plt.figure(figsize=(10,6))  
sns.countplot(data = confirmed, x = 'market_segment', hue = 'hotel')  
plt.legend(title = '')  
plt.xlabel('')  
plt.ylabel('')  
plt.tight_layout;
```



✓ Hotel com mais tempo gasto

```
confirmed.groupby('hotel')['total_days'].sum()
```

```
↗ hotel
City Hotel      111102
Resort Hotel    108052
Name: total_days, dtype: int64
```

✓ Canceled

✓ Mês com o maior número de cancelamentos

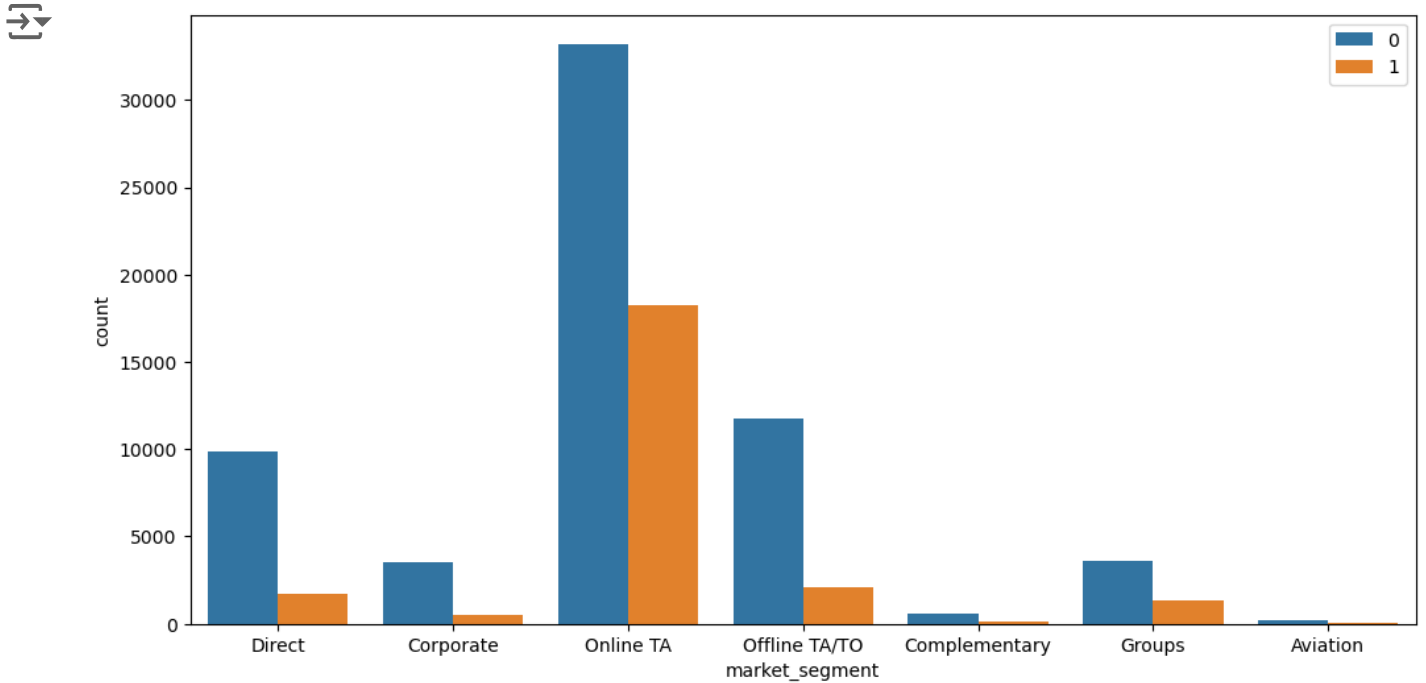
```
canceled = df.loc[df['is_canceled'] == 'canceled']
```

```
canceled.groupby('arrival_date_year')['arrival_date_month'].value_counts()
```

```
↗ Series([], Name: arrival_date_month, dtype: int64)
```

✓ Efeito dos depósitos nos cancelamentos por segmentos

```
## Numeros absolutos
plt.figure(figsize=(12,6))
sns.countplot(data = df,
              x = 'market_segment',
              hue = 'is_canceled',
              )
plt.legend(loc = 'upper right')
plt.tight_layout;
```




```

# print(' Aviation \n Total Reservas: {} \n Porcentagem de cancelamentos: {}
# Complementary \n Total de Reservas: {} \n Porcentagem de cancelamentos: {}
# Corporate \n Total Reservas: {} \n Porcentagem de cancelamentos: {} \n\n \
# Direct \n Total de Reservas: {} \n Porcentagem de cancelamentos: {} \n\n \
# Groups \n Total Reservas: {} \n Porcentagem de cancelamentos: {} \n\n \
# Offline TA/T0 \n Total de Reservas: {} \n Porcentagem de cancelamentos: {}
# Online TA \n Total Reservas: {} \n Porcentagem de cancelamentos: {} \n\n \
# Undefined \n Total de Reservas: {}'.format((df.groupby('market_segment')['is
# round((df.groupby('market_segment
# (df.groupby('market_segment')['i
# round((df.groupby('market_segment
# (df.groupby('market_segment')
# round((df.groupby('market_segment
# (df.groupby('market_segme
# round((df.groupby('market_segment
# (df.groupby('market_segme
# round((df.groupby('market_segment
# (df.groupby('market_segme
# round((df.groupby('market_segment
# (df.groupby('market_segme
# round((df.groupby('market_segment
# (df.groupby('market_segme

```

```
## Numeros absolutos
plt.figure(figsize=(12,6))
sns.countplot(data = df,
              x = 'market_segment',
              hue = 'is_canceled')
```

 <Axes: xlabel='market_segment', ylabel='count'>

