Import the libraries

In [108…
```python
#Importing the Libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

import the dataset

In [95]:
```python
df=pd.read_csv('hotel_bookings.csv')
df
```

Out[95]:

	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	...	deposit_type	agent	company
0	Resort Hotel	0	342	2015	July	27	1	0	0	2	...	No Deposit	NaN	NaN
1	Resort Hotel	0	737	2015	July	27	1	0	0	2	...	No Deposit	NaN	NaN
2	Resort Hotel	0	7	2015	July	27	1	0	1	1	...	No Deposit	NaN	NaN
3	Resort Hotel	0	13	2015	July	27	1	0	1	1	...	No Deposit	304.0	NaN
4	Resort Hotel	0	14	2015	July	27	1	0	2	2	...	No Deposit	240.0	NaN
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
119385	City Hotel	0	23	2017	August	35	30	2	5	2	...	No Deposit	394.0	NaN
119386	City Hotel	0	102	2017	August	35	31	2	5	3	...	No Deposit	9.0	NaN
119387	City Hotel	0	34	2017	August	35	31	2	5	2	...	No Deposit	9.0	NaN
119388	City Hotel	0	109	2017	August	35	31	2	5	2	...	No Deposit	89.0	NaN
119389	City Hotel	0	205	2017	August	35	29	2	7	2	...	No Deposit	9.0	NaN

119390 rows × 32 columns

Exploratory Data Analysis and Data Cleaning

list of first five rows

In [96]:
```python
df.head()
```

Out[96]:

	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	...	deposit_type	agent	company	days_
0	Resort Hotel	0	342	2015	July	27	1	0	0	2	...	No Deposit	NaN	NaN	
1	Resort Hotel	0	737	2015	July	27	1	0	0	2	...	No Deposit	NaN	NaN	
2	Resort Hotel	0	7	2015	July	27	1	0	1	1	...	No Deposit	NaN	NaN	
3	Resort Hotel	0	13	2015	July	27	1	0	1	1	...	No Deposit	304.0	NaN	
4	Resort Hotel	0	14	2015	July	27	1	0	2	2	...	No Deposit	240.0	NaN	

5 rows × 32 columns

list of last five rows

In [97]:
```python
df.tail()
```

Out[97]:

	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	...	deposit_type	agent	company
119385	City Hotel	0	23	2017	August	35	30	2	5	2	...	No Deposit	394.0	NaN
119386	City Hotel	0	102	2017	August	35	31	2	5	3	...	No Deposit	9.0	NaN
119387	City Hotel	0	34	2017	August	35	31	2	5	2	...	No Deposit	9.0	NaN
119388	City Hotel	0	109	2017	August	35	31	2	5	2	...	No Deposit	89.0	NaN
119389	City Hotel	0	205	2017	August	35	29	2	7	2	...	No Deposit	9.0	NaN

5 rows × 32 columns

find the shape

In [98]:
```python
df.shape
```

Out[98]: (119390, 32)

check the data info

In [99]:
```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
```

its convert the reservatrion date in to date

In [100…
```python
df['reservation_status_date']=pd.to_datetime(df['reservation_status_date'])
```

In [8]:
```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  datetime64[ns]
dtypes: datetime64[ns](1), float64(4), int64(16), object(11)
memory usage: 29.1+ MB
```

In [101…
```python
df.describe(include = 'object')
```

Out[101…

	hotel	arrival_date_month	meal	country	market_segment	distribution_channel	reserved_room_type	assigned_room_type	deposit_type	customer_type	reservation_status
count	119390	119390	119390	118902	119390	119390	119390	119390	119390	119390	119390
unique	2	12	5	177	8	5	10	12	3	4	3
top	City Hotel	August	BB	PRT	Online TA	TA/TO	A	A	No Deposit	Transient	Check-Out
freq	79330	13877	92310	48590	56477	97870	85994	74053	104641	89613	75166

checking for nulll values

In [102…
```python
df.isna().sum()
```

Out[102…
```
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
```

remove the null missing values and null columns

In [103…
```python
df.dropna(inplace = True)
```

In [104…
```python
df.drop(columns=['agent', 'company'], inplace=True)
```

In [83]:
```python
df.isna().sum()
```

Out[83]:
```
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 [105…
```python
df = df[df['adr']<5000]
```

now we are creating new columns for easy visualization

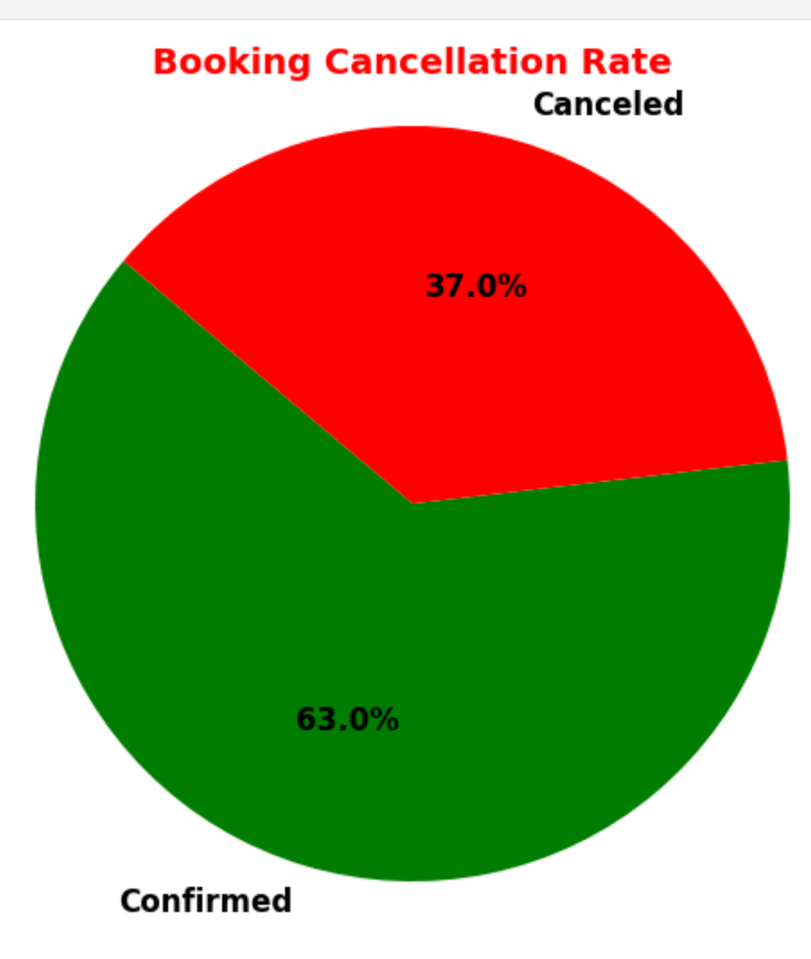
In [91]:
```python
df['cancellation_status'] = df['is_canceled'].astype(int)
```

Q1. What is the overall cancellation rate?

In [124…
```python
cancellation_counts = df['is_canceled'].value_counts(normalize=True) * 100
```

Now we plot A pie chart

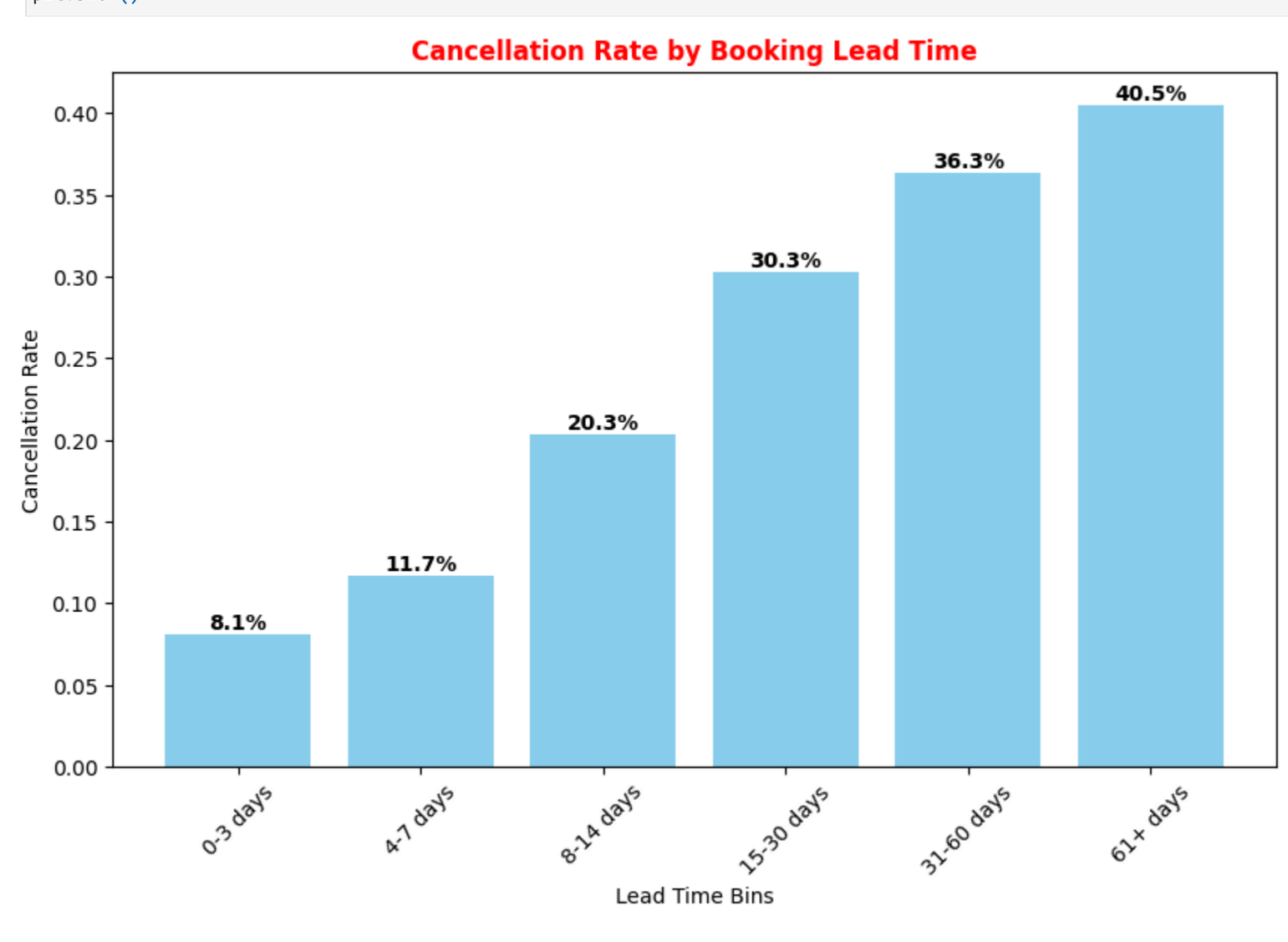
In [34]:
```python
labels = ['Confirmed', 'Canceled']
sizes = [cancellation_counts[0], cancellation_counts[1]]  # Order corresponds to the labels
plt.figure(figsize=(8, 6))
colors = ['green','red']
plt.pie(sizes, labels=labels, colors=colors, autopct='%1.1f%%',
textprops={'color': 'black', 'weight': 'bold', 'fontsize': 12},  # Set font color, weight, and size
startangle=140)
plt.title('Booking Cancellation Rate' ,color = 'red',fontweight='bold', fontsize=14)
plt.axis('equal')  # Equal aspect ratio ensures that pie chart is circular.
plt.show()
```



Qn2. How does the cancellation rate vary by lead time?

In [36]:
```python
bins = [0, 3, 7, 14, 30, 60, 100]  # Define your bins
labels = ['0-3 days', '4-7 days', '8-14 days', '15-30 days', '31-60 days', '61+ days']
df['lead_time_bins'] = pd.cut(df['lead_time'], bins=bins, labels=labels, right=False)

# Calculate cancellation rates for each bin
cancellation_rates = df.groupby('lead_time_bins', observed=False)['cancellation_status'].mean()
```

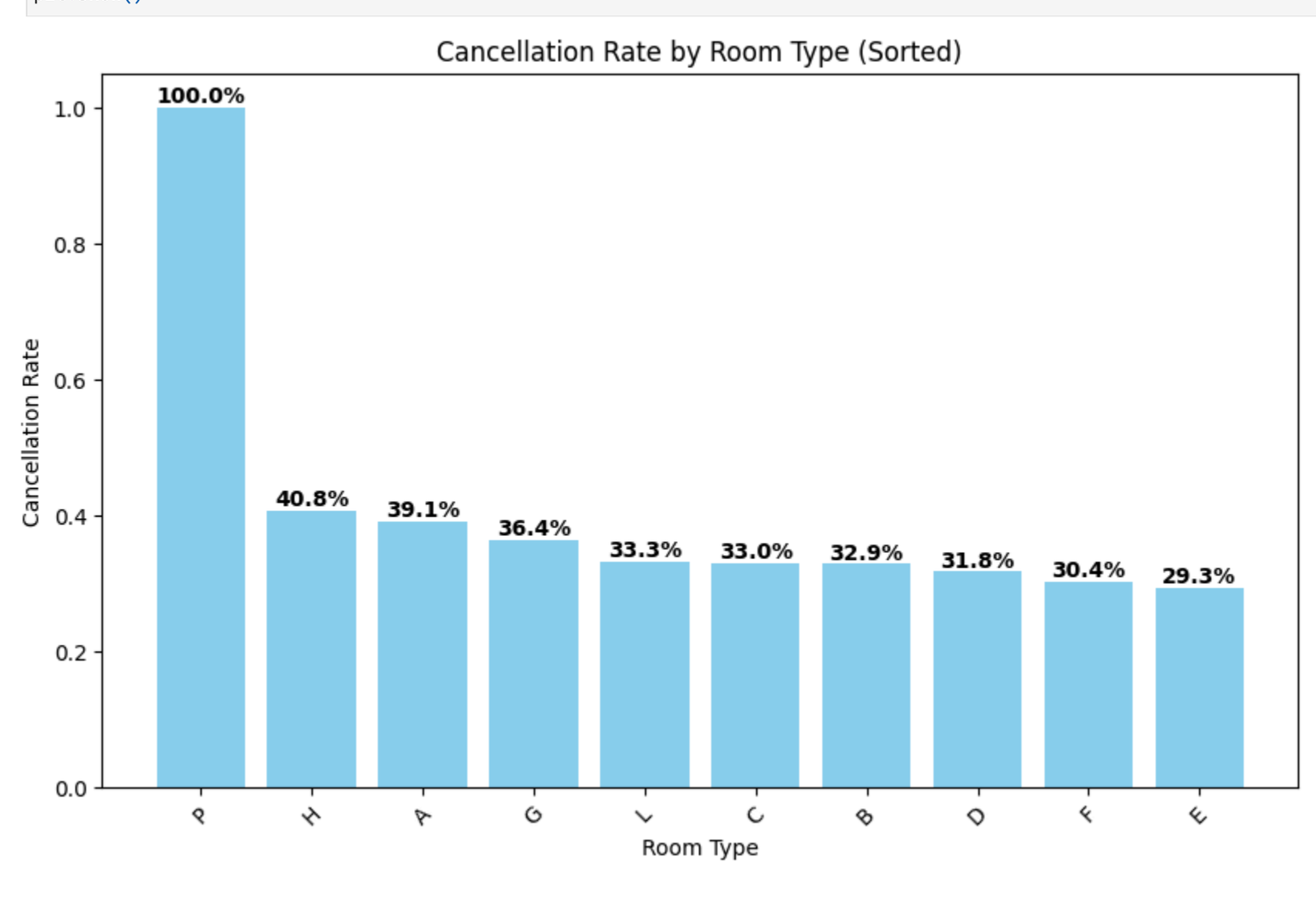
In [37]:
```python
plt.figure(figsize=(10, 6))
barplot = plt.bar(cancellation_rates.index, cancellation_rates.values, color='skyblue')

# Format bar labels as percentages
labels = [f'{value:.1%}' for value in cancellation_rates.values]

plt.bar_label(barplot, labels=labels, label_type='edge', fontweight='bold')
plt.xlabel('Lead Time Bins')
plt.ylabel('Cancellation Rate')
plt.title('Cancellation Rate by Booking Lead Time',color = 'red',fontweight = 'bold')
plt.xticks(rotation=45)
plt.show()
```



Q3. Which room types have the highest cancellation rates?

In [38]:
```python
cancellation_rates_by_room_type = df.groupby('reserved_room_type')['cancellation_status'].mean()
```

In [39]:
```python
# Sort the cancellation rates by room type in descending order
sorted_cancellation_rates = cancellation_rates_by_room_type.sort_values(ascending=False)

# Creating the bar chart with sorted data
plt.figure(figsize=(10, 6))
barplot = plt.bar(sorted_cancellation_rates.index, sorted_cancellation_rates.values, color='skyblue')

# Adding labels to the bars
plt.bar_label(barplot, labels=[f'{value:.1%}' for value in sorted_cancellation_rates.values], label_type='edge', fontweight='bold')

# Labeling the chart and axes
plt.xlabel('Room Type')
plt.ylabel('Cancellation Rate')
plt.title('Cancellation Rate by Room Type (Sorted)')
plt.xticks(rotation=45)

# Display the chart
plt.show()
```



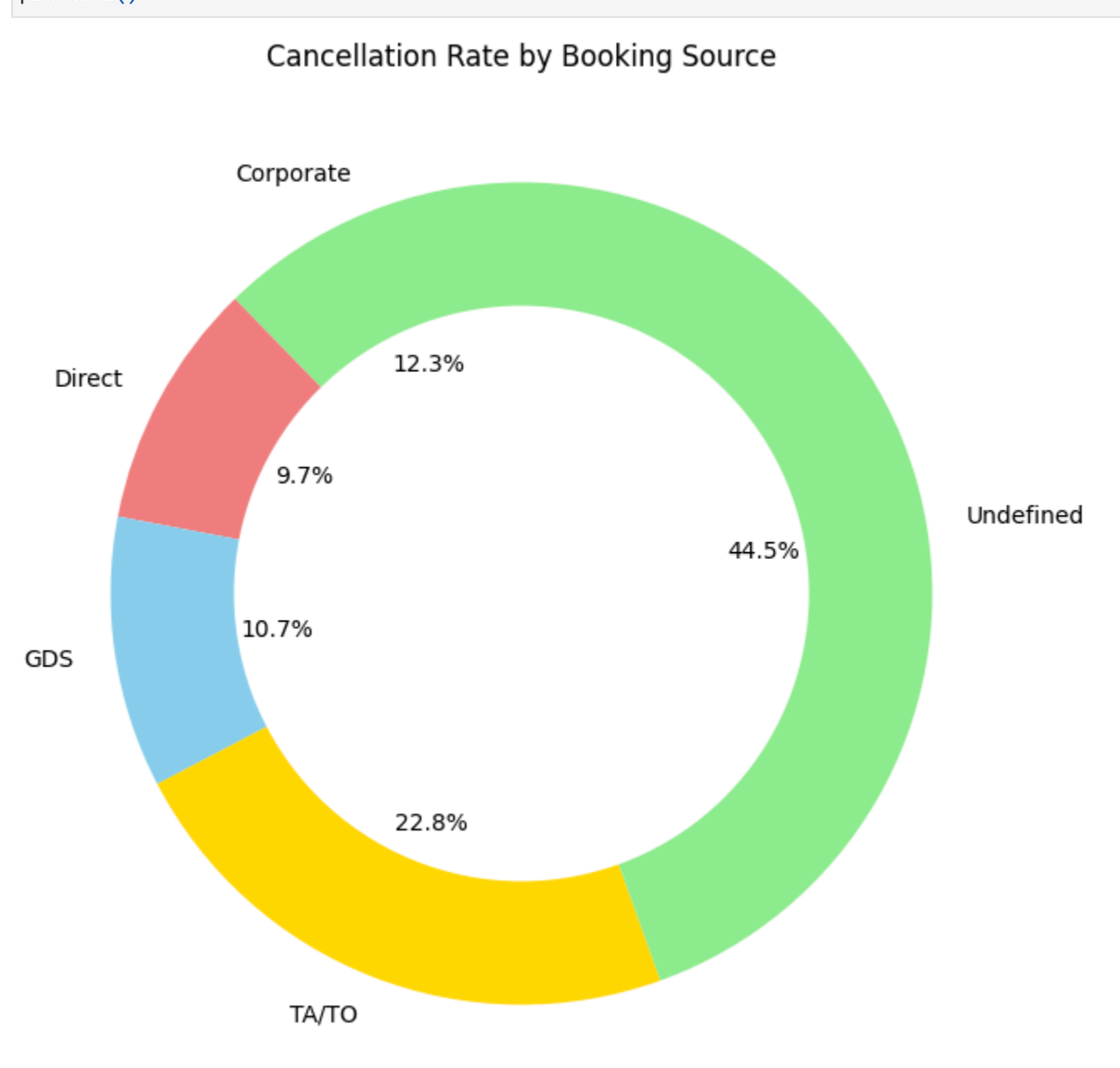
qn4. What impact does the booking source have on the cancellation rate?

In [41]:
```python
booking_sources = df['distribution_channel'].unique()
booking_sources
```

Out[41]: array(['Direct', 'Corporate', 'TA/TO', 'Undefined', 'GDS'], dtype=object)

In [42]:
```python
cancellation_rates_by_source = df.groupby('distribution_channel')['cancellation_status'].mean()
```

In [43]:
```python
# Create the donut chart
plt.figure(figsize=(8, 8))
colors = ['lightgreen', 'lightcoral', 'skyblue', 'gold']
plt.pie(cancellation_rates_by_source.values, labels=cancellation_rates_by_source.index, autopct='%1.1f%%', startangle=90, colors=colors)

# Draw a white circle in the center to make it a donut chart
centre_circle = plt.Circle((0, 0), 0.70, fc='white')
plt.gca().add_artist(centre_circle)

plt.title('Cancellation Rate by Booking Source')
plt.show()
```



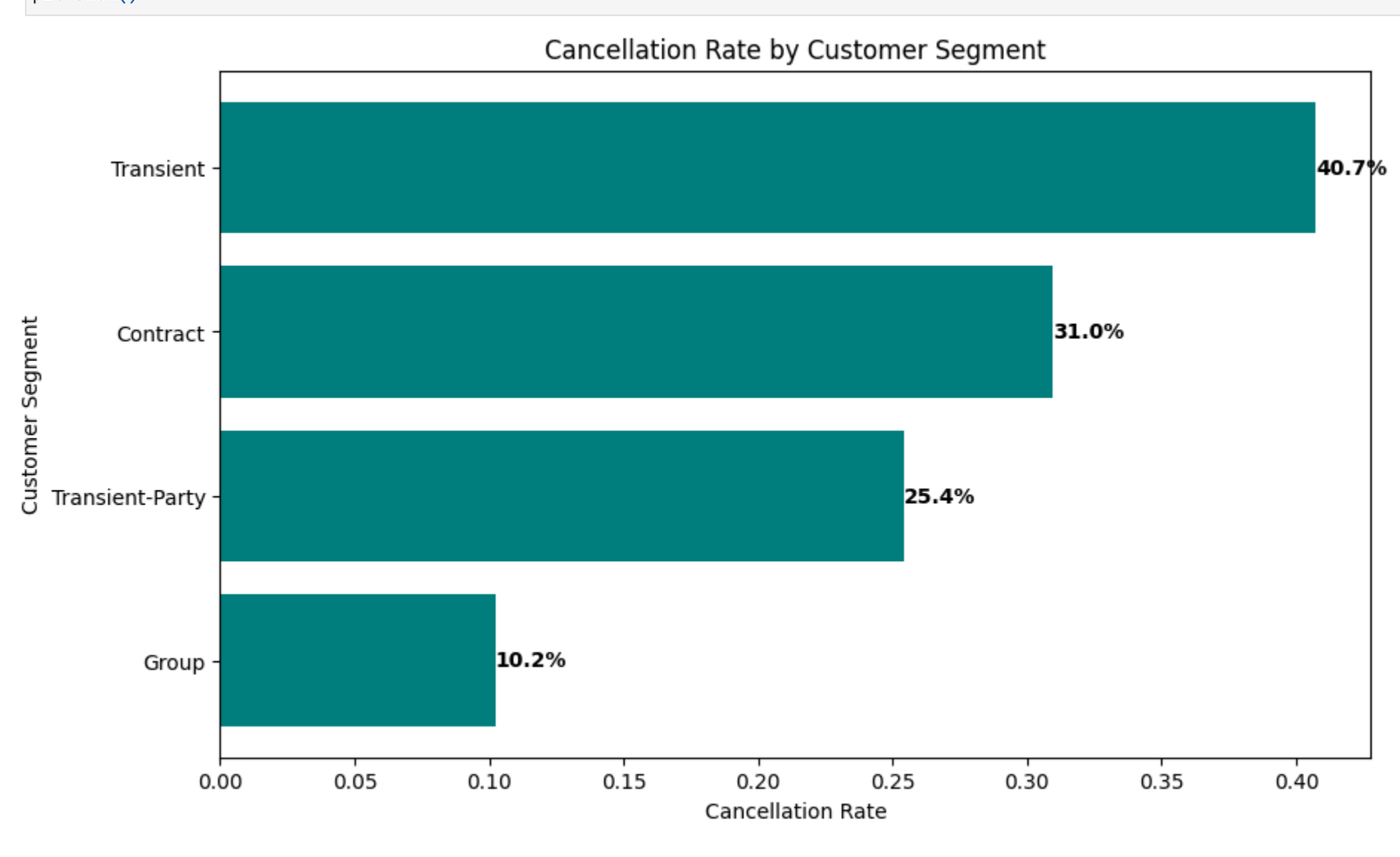
Qn5. How do cancellations differ across customer segments?

In [53]:
```python
df['customer_type'].unique()  # Check available customer segments
```

Out[53]: array(['Transient', 'Contract', 'Transient-Party', 'Group'], dtype=object)

In [54]:
```python
cancellation_rates_by_customer_type = df.groupby('customer_type')['cancellation_status'].mean()
```

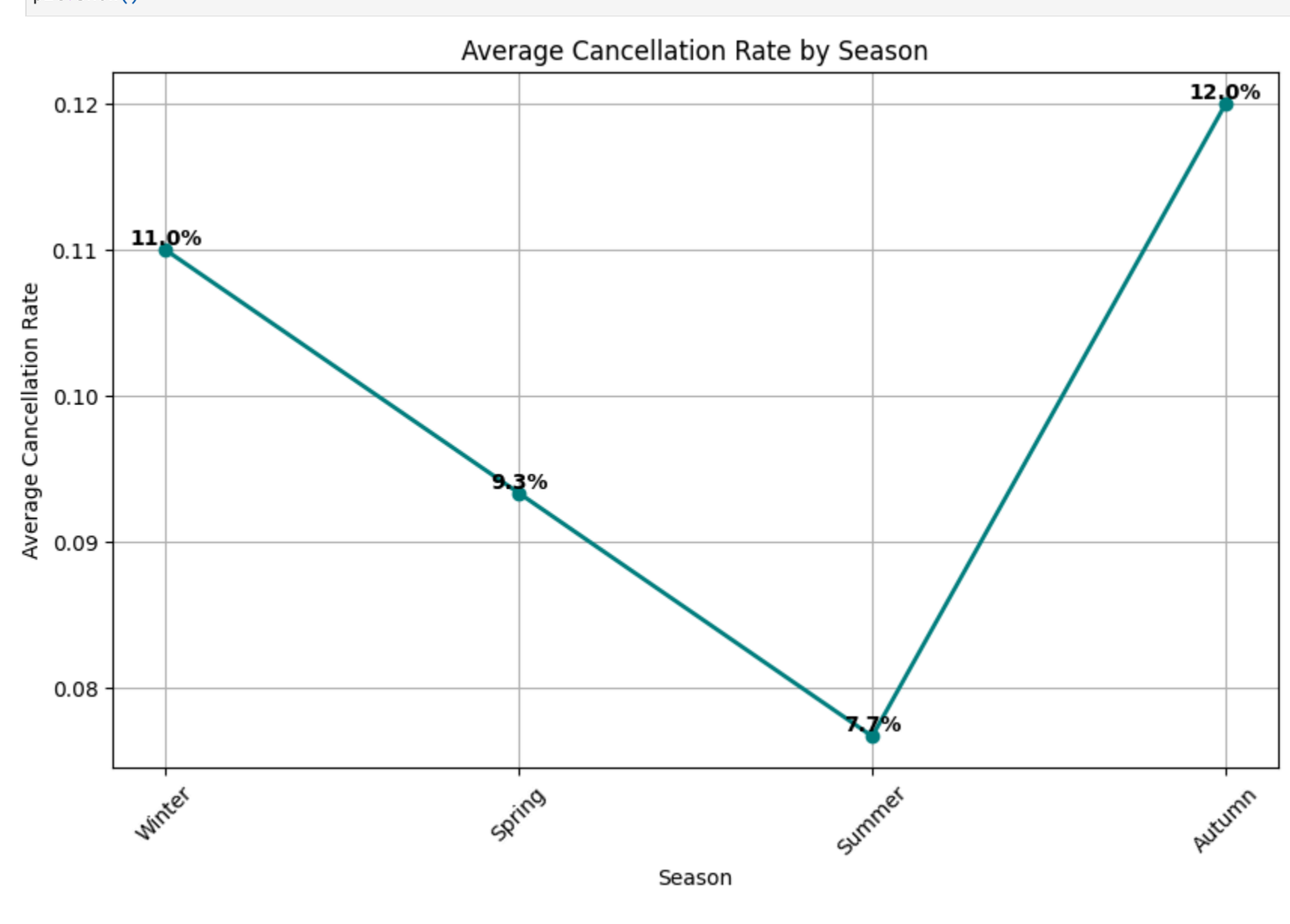
In [55]:
```python
sort_cancellation_rates_by_customer_type = cancellation_rates_by_customer_type.sort_values()
plt.figure(figsize=(10, 6))
barplot = plt.barh(sort_cancellation_rates_by_customer_type.index, sort_cancellation_rates_by_customer_type.values, color='teal')
plt.bar_label(barplot, labels=[f'{value:.1%}' for value in sort_cancellation_rates_by_customer_type.values], label_type='edge', fontweight='bold')
plt.xlabel('Cancellation Rate')
plt.ylabel('Customer Segment')
plt.title('Cancellation Rate by Customer Segment')
plt.show()
```



Qn6. Does seasonality affect cancellation rates?

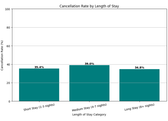
In [56]:
```python
# Group data by arrival month and calculate the average cancellation rate
monthly_cancellation_rate = df.groupby('arrival_date_month')['cancellation_status'].mean()
# Ensure months are in chronological order
import calendar
monthly_cancellation_rate.index = pd.Categorical(monthly_cancellation_rate.index, categories=calendar.month_name[1:], ordered=True)
monthly_cancellation_rate = monthly_cancellation_rate.sort_index()
```

In [57]:
```python
# Example data for monthly cancellation rates
data = {
    'January': 0.1,
    'February': 0.12,
    'March': 0.15,
    'April': 0.08,
    'May': 0.05,
    'June': 0.07,
    'July': 0.06,
    'August': 0.1,
    'September': 0.14,
    'October': 0.09,
    'November': 0.13,
    'December': 0.11
}
monthly_cancellation_rate = pd.Series(data)

# Create a mapping of months to seasons
season_map = {
    'January': 'Winter', 'February': 'Winter', 'March': 'Spring',
    'April': 'Spring', 'May': 'Spring', 'June': 'Summer',
    'July': 'Summer', 'August': 'Summer', 'September': 'Autumn',
    'October': 'Autumn', 'November': 'Autumn', 'December': 'Winter'
}

# Map months to seasons and calculate average cancellation rate by season
monthly_cancellation_rate.index = pd.Index(monthly_cancellation_rate.index.map(season_map))
seasonal_cancellation_rate = monthly_cancellation_rate.groupby(monthly_cancellation_rate.index).mean()

# Sorting seasons for proper plotting order
season_order = ['Winter', 'Spring', 'Summer', 'Autumn']
seasonal_cancellation_rate = seasonal_cancellation_rate.reindex(season_order)

# Plotting
plt.figure(figsize=(10, 6))
plt.plot(seasonal_cancellation_rate.index, seasonal_cancellation_rate.values, marker='o', color='teal', linestyle='-', linewidth=2)

# Adding labels at each point
for i, value in enumerate(seasonal_cancellation_rate.values):
    plt.text(i, value, f'{value:.1%}', ha='center', va='bottom', fontweight='bold')

# Labeling the chart
plt.xlabel('Season')
plt.ylabel('Average Cancellation Rate')
plt.title('Average Cancellation Rate by Season')
plt.xticks(rotation=45)
plt.grid(True)

# Show the chart
plt.show()
```



Qn7. 7. How does the cancellation rate vary by length of stay?

In [70]:
```python
# Create a new column for total length of stay
df.loc[:, 'total_stay'] = df['stays_in_weekend_nights'] + df['stays_in_week_nights']

# Categorize length of stay
bins = [0, 3, 7, 30]  # Define bins: 1-3 nights, 4-7 nights, 8+ nights
labels = ['Short Stay (1-3 nights)', 'Medium Stay (4-7 nights)', 'Long Stay (8+ nights)']
df.loc[:, 'stay_category'] = pd.cut(df['total_stay'], bins=bins, labels=labels, right=False)

# Calculate cancellation rates by length of stay category with observed parameter
cancellation_rates = df.groupby('stay_category', observed=True)['is_canceled'].mean() * 100  # Multiply by 100 for percentage

# Plotting
plt.figure(figsize=(10, 6))
bars = plt.bar(cancellation_rates.index, cancellation_rates.values, color='teal')
plt.bar_label(bars, labels=[f'{val:.1f}%' for val in cancellation_rates.values], label_type='edge', fontweight='bold')



# Labeling the chart
plt.xlabel('Length of Stay Category')
plt.ylabel('Cancellation Rate (%)')
plt.title('Cancellation Rate by Length of Stay')
plt.xticks(rotation=10)
plt.ylim(0, 100)  # Set y-axis limit from 0% to 100%
plt.grid(axis='y')

# Show the chart
plt.show()
```


Cancellation Rate by Length of Stay
100
80
60
40
20
0
Cancellation Rate (%)
39.4%
39.8%
39.4%
Short Stay (1-3 nights)
Medium Stay (4-7 nights)
Long Stay (8+ nights)
Length of Stay Category