

Zomato data analysis

June 4, 2024

```
[ ]: Zomato Sales Analysis Projects
```

```
[1]: # .To address our analysis, we need to respond to the subsequent inquiries:

# 1 ..Do a greater number of restaurants provide online delivery as opposed to
↳offline services?
# 2 ..Which types of restaurants are the most favored by the general public?
# 3 ..What price range is preferred by couples for their dinner at restaurants?
```

```
[ ]: Step 1 : importing necessary libraries----
```

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

```
[5]: Step 2: Create the data frame.
Download the file containing the data.
```

```
[14]: dataframe = pd.read_csv("C:\\Users\\hp\\Downloads\\Zomato data .csv")
print(dataframe.head())
```

	name	online_order	book_table	rate	votes	\
0	Jalsa	Yes	Yes	4.1/5	775	
1	Spice Elephant	Yes	No	4.1/5	787	
2	San Churro Cafe	Yes	No	3.8/5	918	
3	Addhuri Udupi Bhojana	No	No	3.7/5	88	
4	Grand Village	No	No	3.8/5	166	

	approx_cost(for two people)	listed_in(type)
0	800	Buffet
1	800	Buffet
2	800	Buffet
3	300	Buffet
4	600	Buffet

```
[15]: #Before proceeding, let's convert the data type of the "rate" column to float
↳and remove the denominator
```

```
def handleRate(value):
    value=str(value).split('/')
    value=value[0];
    return float(value)

dataframe['rate']=dataframe['rate'].apply(handleRate)
print(dataframe.head())
```

	name	online_order	book_table	rate	votes	\
0	Jalsa	Yes	Yes	4.1	775	
1	Spice Elephant	Yes	No	4.1	787	
2	San Churro Cafe	Yes	No	3.8	918	
3	Addhuri Udupi Bhojana	No	No	3.7	88	
4	Grand Village	No	No	3.8	166	

	approx_cost(for two people)	listed_in(type)
0	800	Buffet
1	800	Buffet
2	800	Buffet
3	300	Buffet
4	600	Buffet

[10]: *# To obtain a summary of the data frame, we can use the following code:-*

```
dataframe.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 148 entries, 0 to 147
Data columns (total 7 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   name                                  148 non-null    object
1   online_order                         148 non-null    object
2   book_table                           148 non-null    object
3   rate                                 148 non-null    float64
4   votes                                148 non-null    int64
5   approx_cost(for two people)          148 non-null    int64
6   listed_in(type)                      148 non-null    object
dtypes: float64(1), int64(2), object(4)
memory usage: 8.2+ KB
```

[20]: *# Let's explore the listed_in (type of restaurant) column.*

```

import matplotlib.pyplot as plt
import pandas as pd

# Assuming 'dataframe' is your DataFrame and 'listed_in(type)' is the column
# you're interested in.
data = dataframe['listed_in(type)']

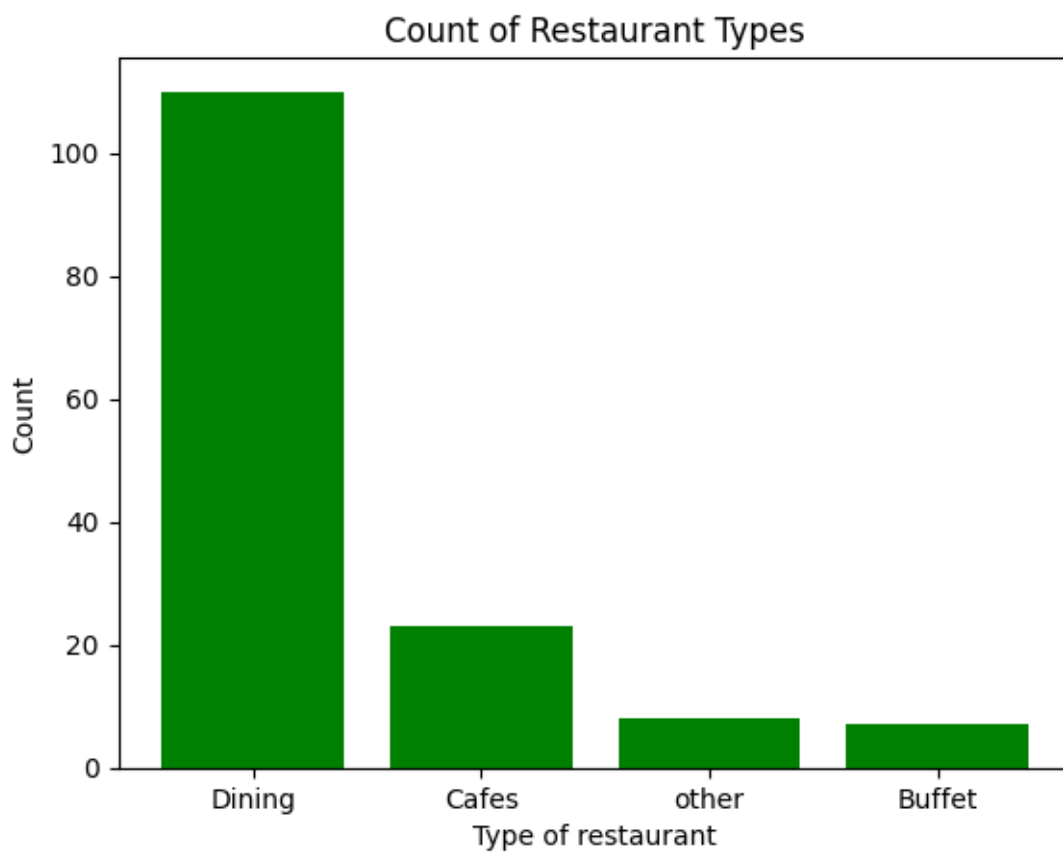
# Step 1: Count the occurrences of each category
counts = data.value_counts()

# Step 2: Plot these counts using Matplotlib
plt.bar(counts.index, counts.values,color="green")

plt.xlabel('Type of restaurant')
plt.ylabel('Count')
plt.title('Count of Restaurant Types')

plt.show()

```

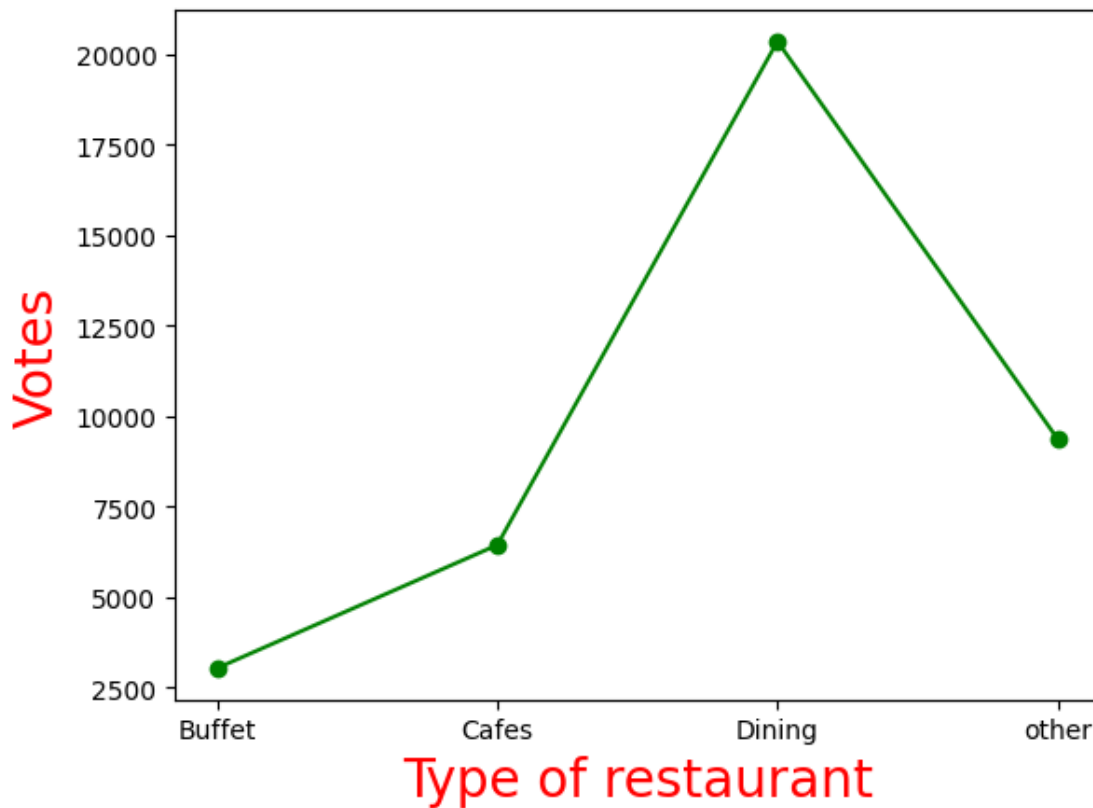


```
[26]: ## Conclusion: The majority of the restaurants fall into the dining category
```

```
[36]: #Q.. which restaurant are preferred by the peoples
```

```
grouped_data = dataframe.groupby('listed_in(type)')['votes'].sum()
result = pd.DataFrame({'votes': grouped_data})
plt.plot(result, c="green", marker="o")
plt.xlabel("Type of restaurant", c="red", size=20)
plt.ylabel("Votes", c="red", size=20)
```

```
[36]: Text(0, 0.5, 'Votes')
```



```
[37]: # Conclusion: Dining restaurants are preferred by a larger number of
↳ individuals.
```

```
[39]: # Now we will determine the restaurant's name that received the maximum votes
↳ based on a given dataframe
```

```
max_votes = dataframe['votes'].max()
restaurant_with_max_votes = dataframe.loc[dataframe['votes'] == max_votes,
↳ 'name']
```

```
print("Restaurant(s) with the maximum votes:")
print(restaurant_with_max_votes)
```

Restaurant(s) with the maximum votes:

38 Empire Restaurant

Name: name, dtype: object

```
[31]: # Let's explore the online_order column.

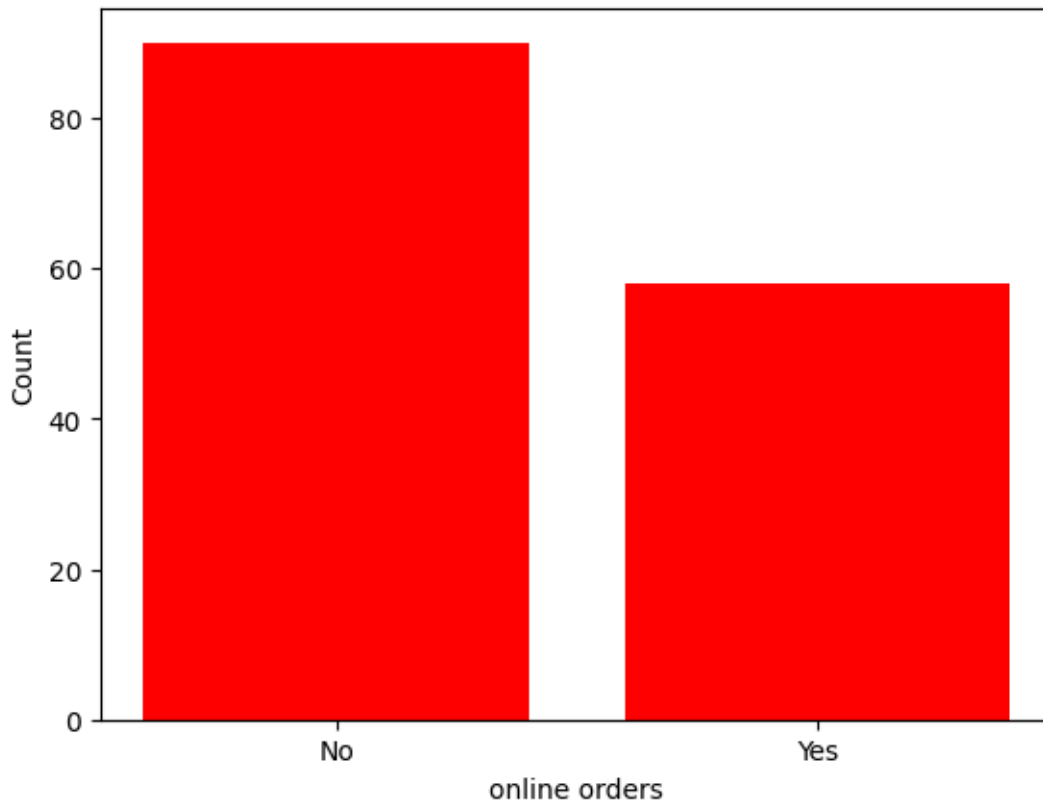
data = dataframe['online_order']

# Step 1: Count the occurrences of each category
counts = data.value_counts()

# Step 2: Plot these counts using Matplotlib
plt.bar(counts.index, counts.values,color="r")

plt.xlabel('online orders')
plt.ylabel('Count')

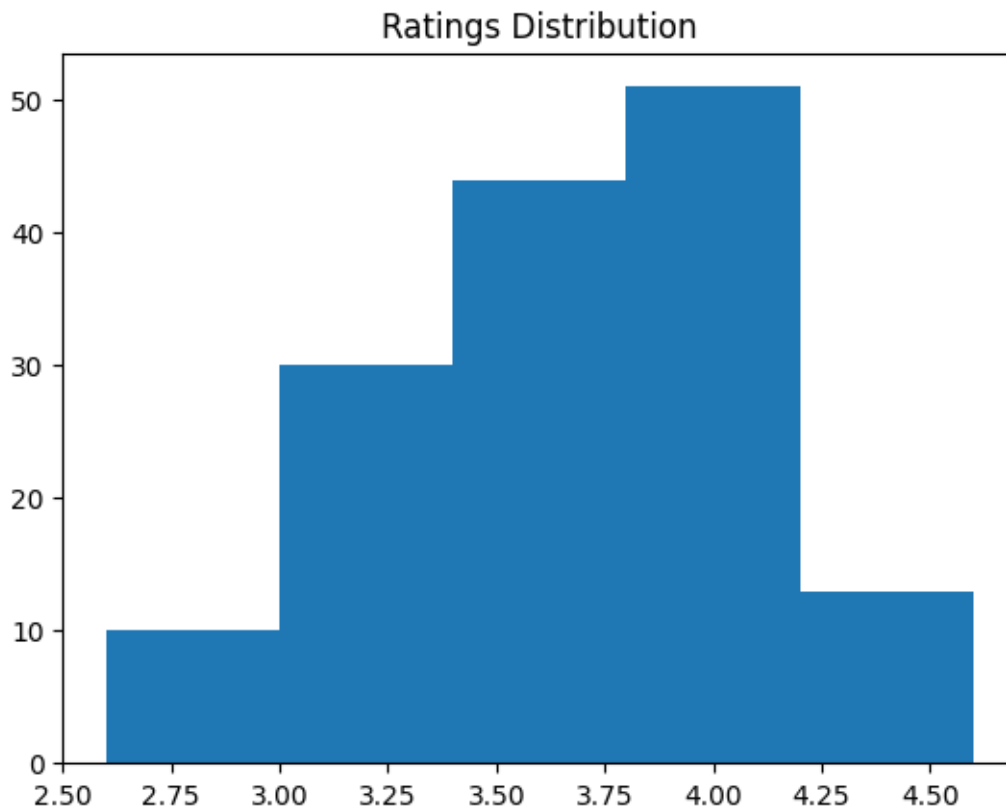
plt.show()
```



```
[43]: tfgrfgt ## Conclusion: This suggests that a majority of the restaurants do not  
      ↪ accept online orders.
```

```
[16]: # Let's explore the rate column.
```

```
plt.hist(dataframe['rate'],bins=5)  
plt.title("Ratings Distribution")  
plt.show()
```



```
[45]: # Conclusion: The majority of restaurants received ratings ranging from 3.5  
      ↪ to 4
```

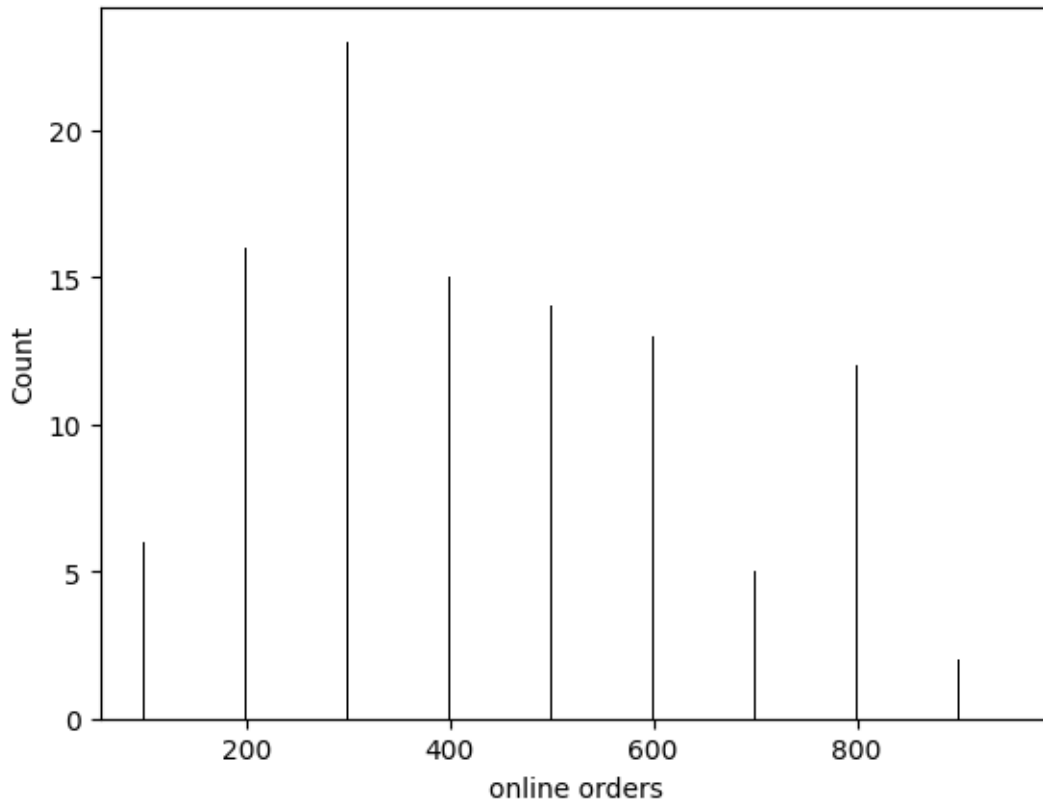
```
[35]: # Let's explore the approx_cost(for two people) column
```

```
couple_data=dataframe['approx_cost(for two people)']  
  
counts = couple_data.value_counts()
```

```
plt.bar(counts.index, counts.values, color="k")

plt.xlabel('online orders')
plt.ylabel('Count')

plt.show()
```

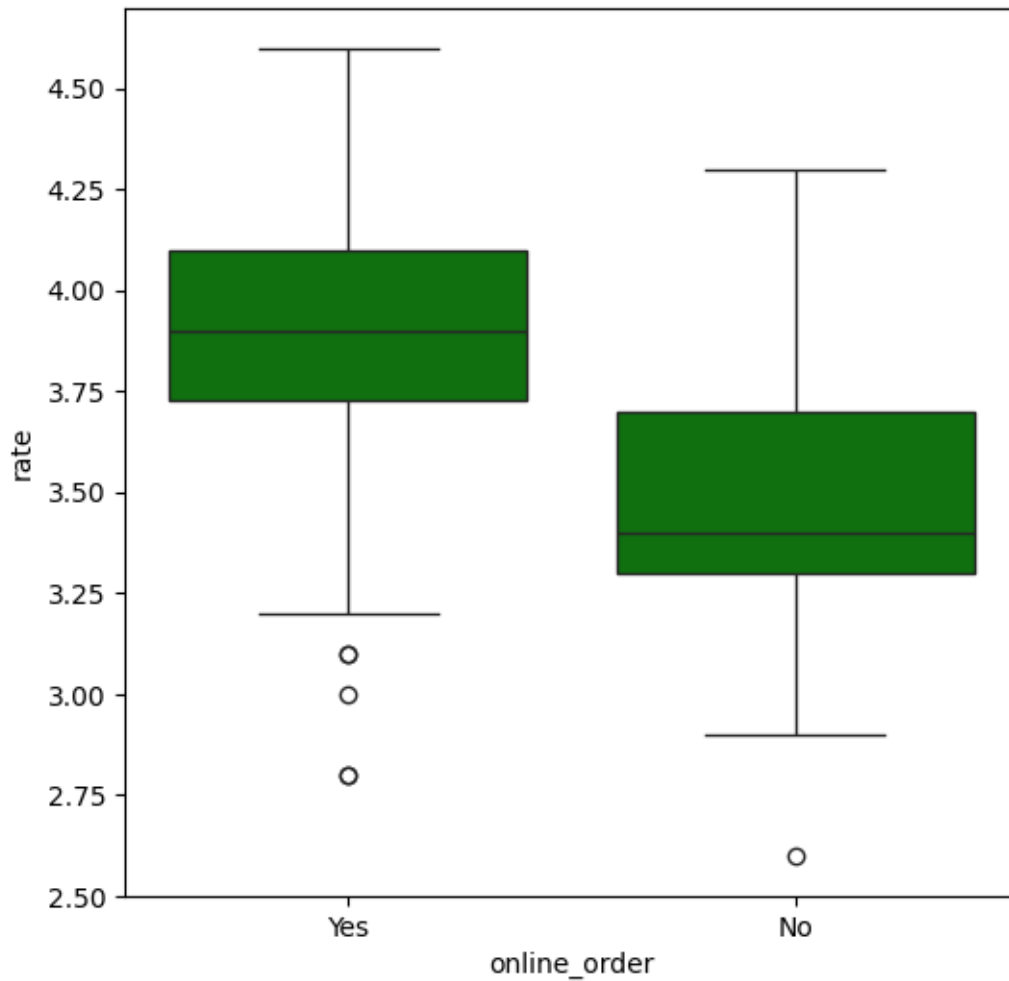


[47]: *# Conclusion: The majority of couples prefer restaurants with an approximate*
↪ cost of 300 rupees

[37]: *# Now we will examine whether online orders receive higher ratings than*
↪ offline orders

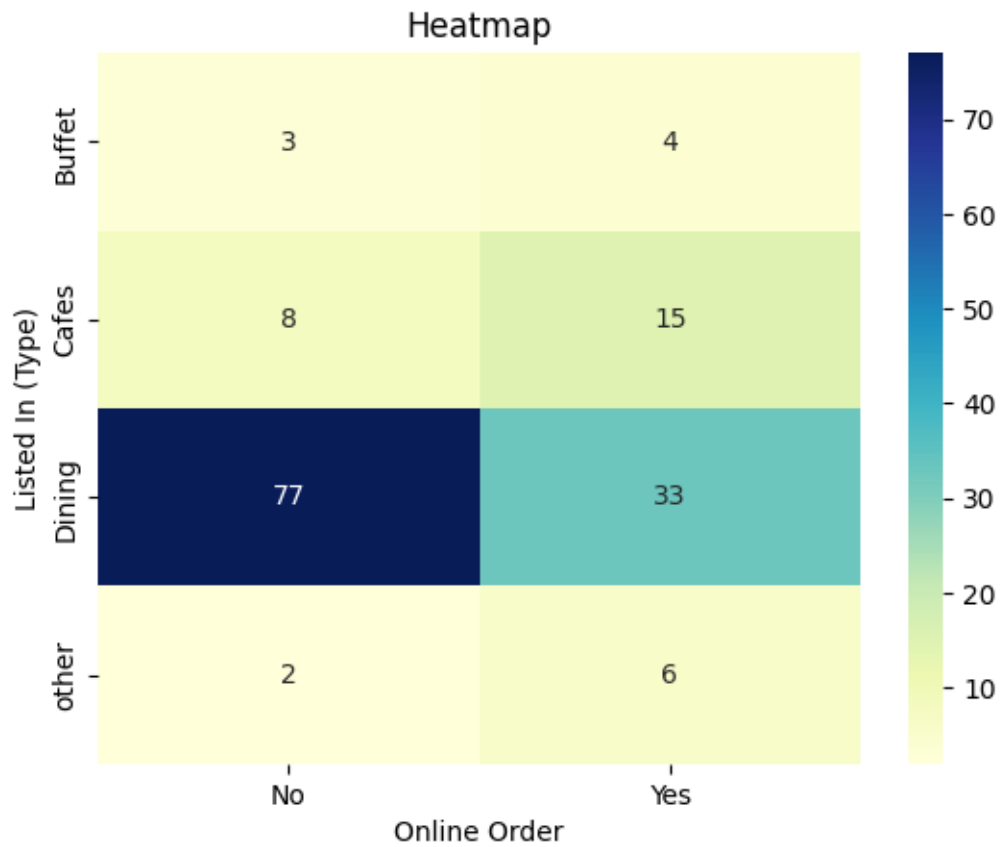
```
plt.figure(figsize = (6,6))
sns.boxplot(x = 'online_order', y = 'rate', data = dataframe, color='green')
```

[37]: <Axes: xlabel='online_order', ylabel='rate'>



```
[51]: ## CONCLUSION: Offline orders received lower ratings in comparison
      ## to online orders, which obtained excellent ratings.
```

```
[52]: pivot_table = dataframe.pivot_table(index='listed_in(type)',
      ↪columns='online_order', aggfunc='size', fill_value=0)
sns.heatmap(pivot_table, annot=True, cmap="YlGnBu", fmt='d')
plt.title("Heatmap")
plt.xlabel("Online Order")
plt.ylabel("Listed In (Type)")
plt.show()
```

[]: CONCLUSION: Dining restaurants primarily accept offline orders, whereas cafes primarily receive online orders. This suggests that clients prefer to place orders **in** person at restaurants, but prefer online ordering at cafes.