Zomato has an average of 17.5 million monthly transacting customers for its food delivery business.e average monthly active food delivery restaurant partners on Zomato's platform have also increased by 8.7% year-on-year, from 208,000 to 226,000.You are working in a data-driven role at Zomato. You have a dataset of customers. As a data professional, you need to analyze the data, perform EDA (Exploratory Data Analysis) and visualization, and answer the following questions:

- What type of restaurant do the majority of customers order from?
- How many votes has each type of restaurant received from customers?
- What are the ratings that the majority of restaurants have received?
- Zomato has observed that most couples order most of their food online. What is their

average spending on each order?

- Which mode (online or offline) has received the maximum rating?
- Which type of restorent recieved more offline orders, so that Zomato can p customers with some good offers?

Zomato Data Analysis Using Python

Step 1: Import necessary Python libraries.

In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

- pandas is used for data manipulation and analysis.
- numpy is used for numerical operations.
- matplotlib.pyplot and seaborn are used for data visualization.

Step 2: Create the data frame.

In [2]:
```
dataframe = pd.read_csv(r"C:\Users\Prathamesh Jadhav\Documents\PROJECTS\ZOMATO DATA ANYLICES\
dataframe
```

Out[2]:

	name	online_order	book_table	rate	votes	approx_cost(for two people)	listed_in(type)
0	Jalsa	Yes	Yes	4.1/5	775	800	Buffet
1	Spice Elephant	Yes	No	4.1/5	787	800	Buffet
2	San Churro Cafe	Yes	No	3.8/5	918	800	Buffet
3	Addhuri Udupi Bhojana	No	No	3.7/5	88	300	Buffet
4	Grand Village	No	No	3.8/5	166	600	Buffet
...	...	...	...	...	...	...	...
143	Melting Melodies	No	No	3.3/5	0	100	Dining
144	New Indraprasta	No	No	3.3/5	0	150	Dining
145	Anna Kuteera	Yes	No	4.0/5	771	450	Dining
146	Darbar	No	No	3.0/5	98	800	Dining
147	Vijayalakshmi	Yes	No	3.9/5	47	200	Dining

148 rows × 7 columns

Convert the Data type of - 'rate' Column.

In [3]:
```
def handel_rate(values):
    value = str(values).split('/')
    value = value[0]
    return float(value)

dataframe['rate'] = dataframe['rate'].apply(handel_rate)
dataframe
```

Out[3]:

	name	online_order	book_table	rate	votes	approx_cost(for two people)	listed_in(type)
0	Jalsa	Yes	Yes	4.1	775	800	Buffet
1	Spice Elephant	Yes	No	4.1	787	800	Buffet
2	San Churro Cafe	Yes	No	3.8	918	800	Buffet
3	Addhuri Udupi Bhojana	No	No	3.7	88	300	Buffet
4	Grand Village	No	No	3.8	166	600	Buffet
...	...	...	...	...	...	...	...
143	Melting Melodies	No	No	3.3	0	100	Dining
144	New Indraprasta	No	No	3.3	0	150	Dining
145	Anna Kuteera	Yes	No	4.0	771	450	Dining
146	Darbar	No	No	3.0	98	800	Dining
147	Vijayalakshmi	Yes	No	3.9	47	200	Dining

148 rows × 7 columns

summary of the data frame

In [4]:
```
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
```

Conclusion - There is no NULL value in dataframe.

Type of Resturant

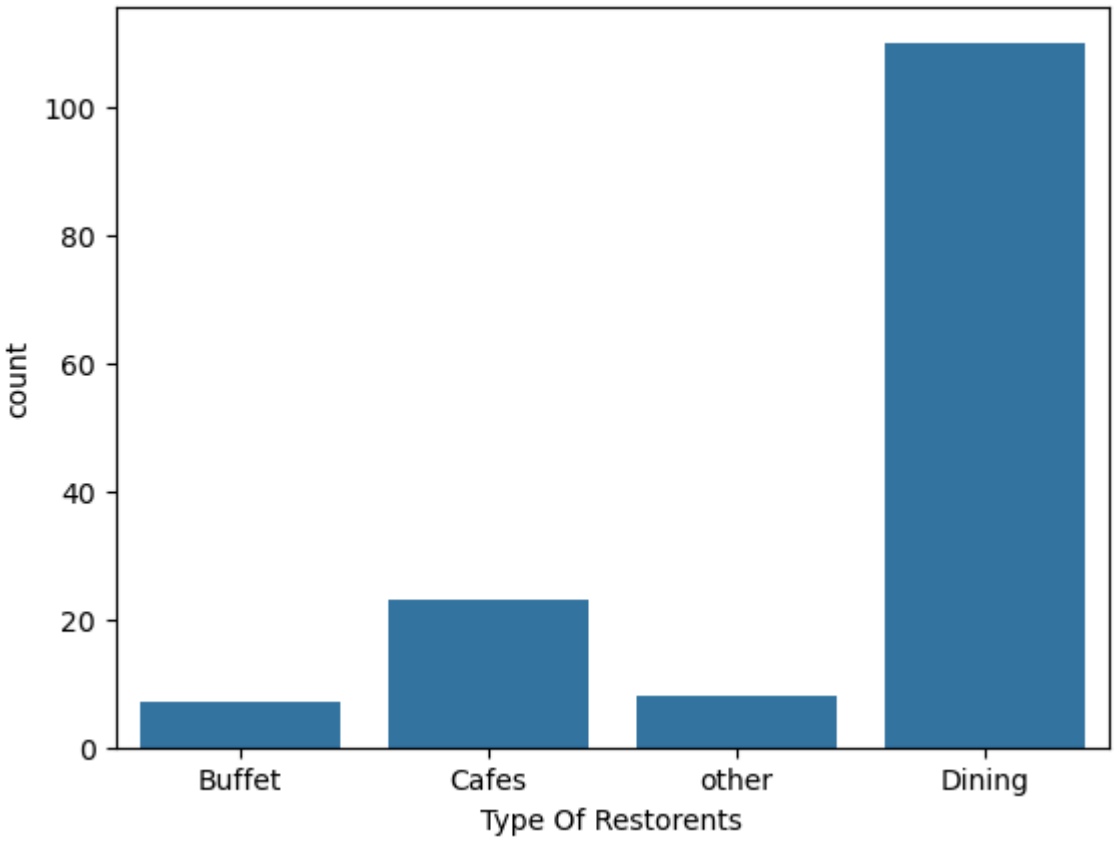
In [5]:
```
dataframe.head()
```

Out[5]:

	name	online_order	book_table	rate	votes	approx_cost(for two people)	listed_in(type)
0	Jalsa	Yes	Yes	4.1	775	800	Buffet
1	Spice Elephant	Yes	No	4.1	787	800	Buffet
2	San Churro Cafe	Yes	No	3.8	918	800	Buffet
3	Addhuri Udupi Bhojana	No	No	3.7	88	300	Buffet
4	Grand Village	No	No	3.8	166	600	Buffet

In [6]:
```
sns.countplot(x=dataframe['listed_in(type)'])
plt.xlabel('Type Of Restorents')
```

Out[6]:
```
Text(0.5, 0, 'Type Of Restorents')
```



Conclusion: The majority of the restaurants fall into the dining category.

'Dining' restaurants are preferred by a larger number of individuals.

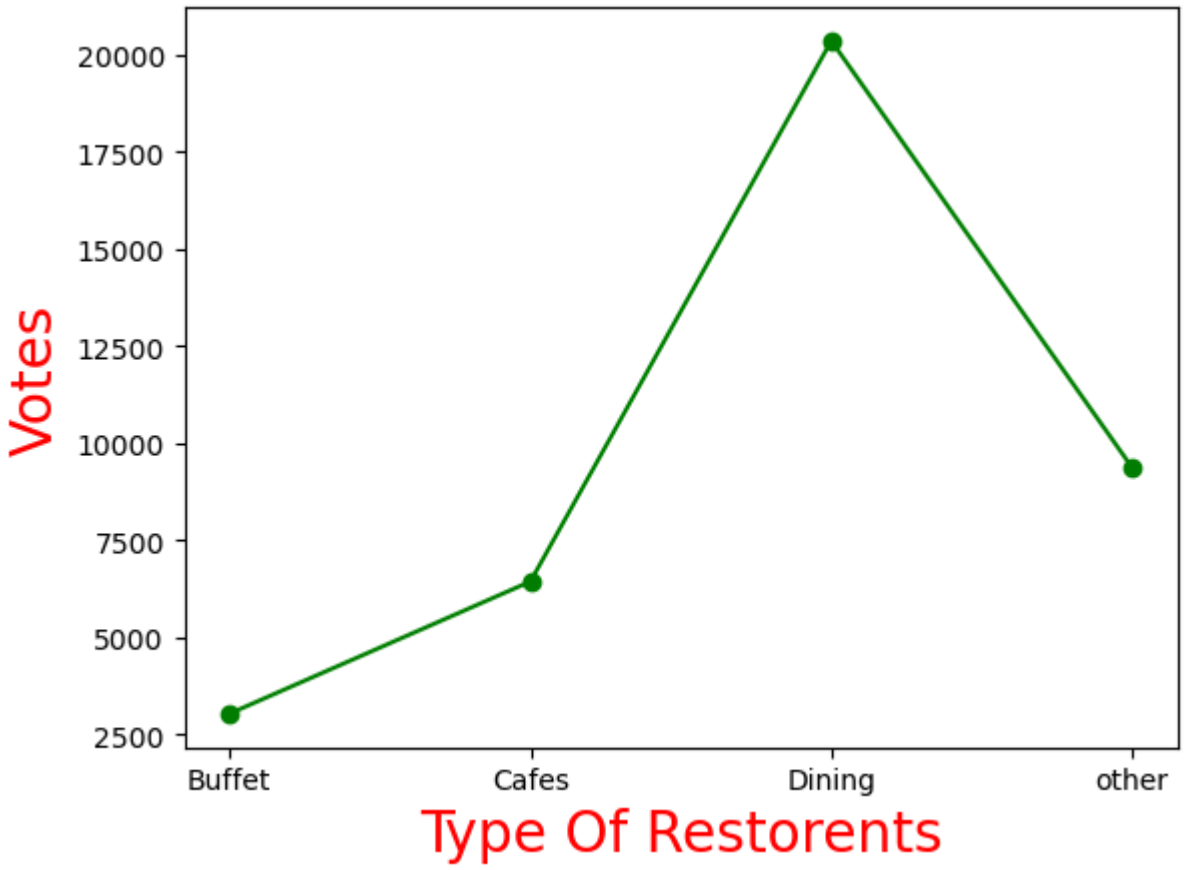
In [7]:
```
dataframe.head()
```

Out[7]:

	name	online_order	book_table	rate	votes	approx_cost(for two people)	listed_in(type)
0	Jalsa	Yes	Yes	4.1	775	800	Buffet
1	Spice Elephant	Yes	No	4.1	787	800	Buffet
2	San Churro Cafe	Yes	No	3.8	918	800	Buffet
3	Addhuri Udupi Bhojana	No	No	3.7	88	300	Buffet
4	Grand Village	No	No	3.8	166	600	Buffet

In [8]:
```
gropued_data = dataframe.groupby('listed_in(type)')['votes'].sum()
result = pd.DataFrame({'votes' : gropued_data})
plt.plot(result, c='green', marker='o')
plt.xlabel('Type Of Restorents', c='red', size=20)
plt.ylabel('Votes', c='red', size=20)
```

Out[8]:
```
Text(0, 0.5, 'Votes')
```



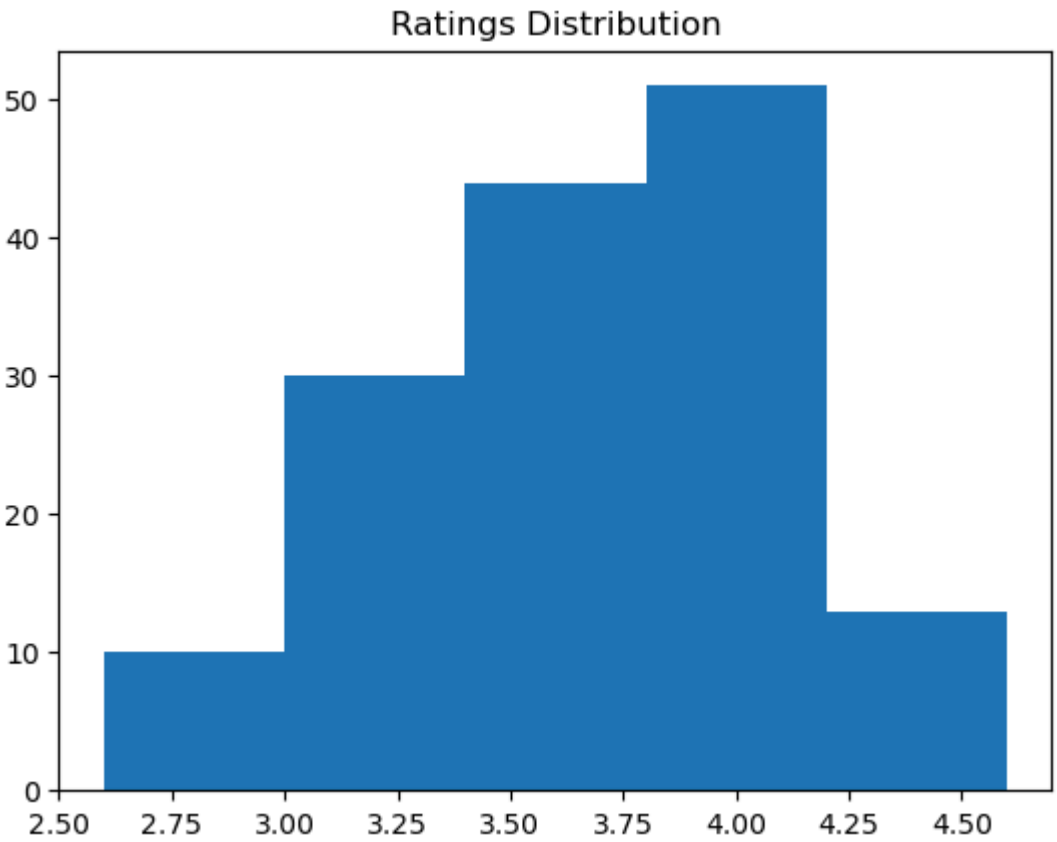
Conclution - 'Dining' restorents has recieved maximum votes

In [9]:
```
dataframe.head()
```

Out[9]:

	name	online_order	book_table	rate	votes	approx_cost(for two people)	listed_in(type)
0	Jalsa	Yes	Yes	4.1	775	800	Buffet
1	Spice Elephant	Yes	No	4.1	787	800	Buffet
2	San Churro Cafe	Yes	No	3.8	918	800	Buffet
3	Addhuri Udupi Bhojana	No	No	3.7	88	300	Buffet
4	Grand Village	No	No	3.8	166	600	Buffet

In [10]:
```
plt.hist(dataframe['rate'], bins=5)
plt.title("Ratings Distribution")
plt.show()
```



Conclusion: The majority of restaurants received ratings ranging from 3.5 to 4.

The majority of couples prefer restaurants with an approximate cost of 300 rupees.

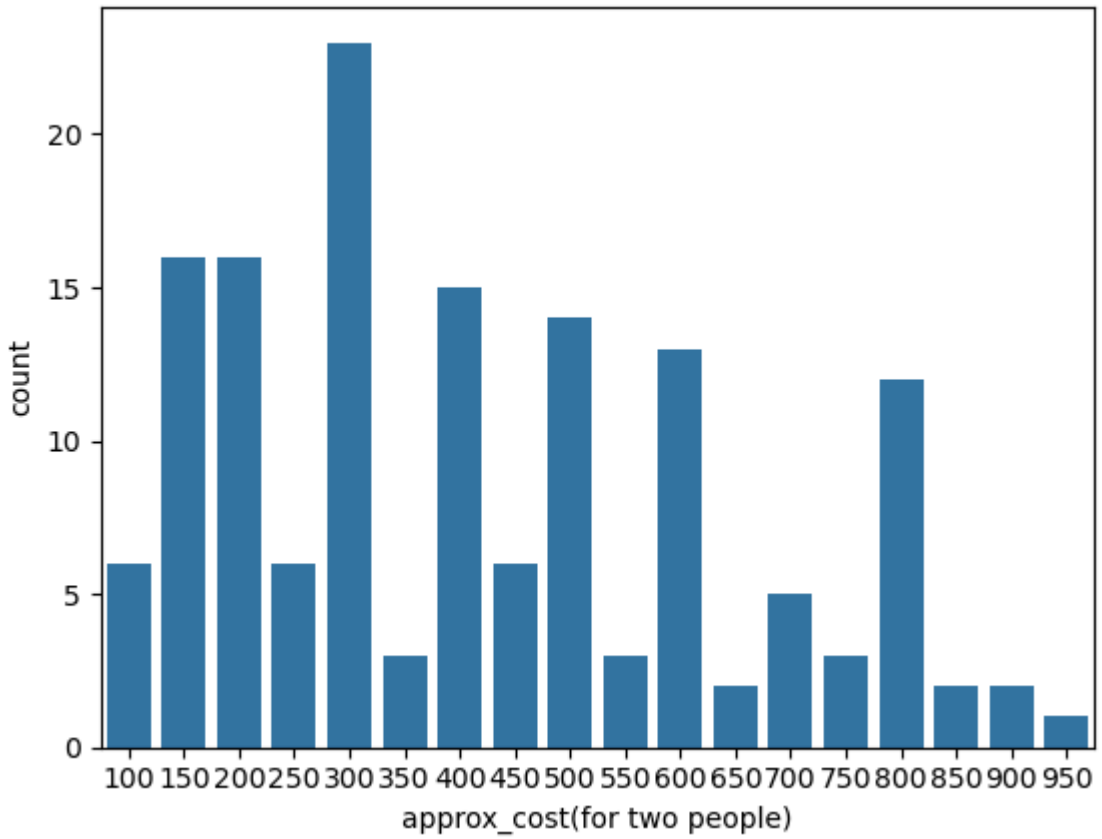
In [11]:
```
dataframe.head()
```

Out[11]:

	name	online_order	book_table	rate	votes	approx_cost(for two people)	listed_in(type)
0	Jalsa	Yes	Yes	4.1	775	800	Buffet
1	Spice Elephant	Yes	No	4.1	787	800	Buffet
2	San Churro Cafe	Yes	No	3.8	918	800	Buffet
3	Addhuri Udupi Bhojana	No	No	3.7	88	300	Buffet
4	Grand Village	No	No	3.8	166	600	Buffet

In [12]:
```
couple_data = dataframe['approx_cost(for two people)']
sns.countplot(x=couple_data)
```

Out[12]:
```
<Axes: xlabel='approx_cost(for two people)', ylabel='count'>
```

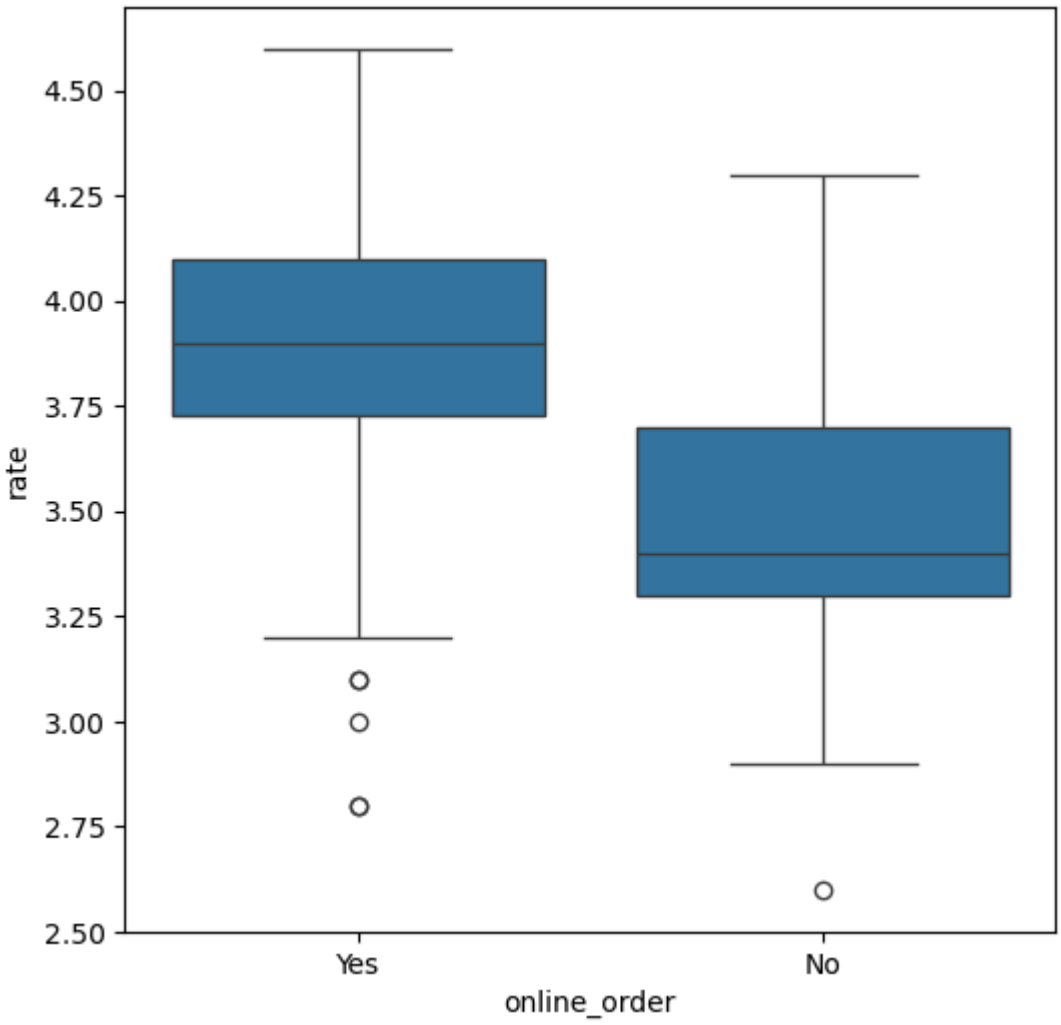


Conclusion - the majority of couples preferr resturants with an approximate cost of 30 rupees

whether online orders receive higher ratings than offline orders.

In [13]:
```
plt.figure(figsize=(6,6))
sns.boxplot(x='online_order', y = 'rate', data = dataframe)
```

Out[13]:
```
<Axes: xlabel='online_order', ylabel='rate'>
```



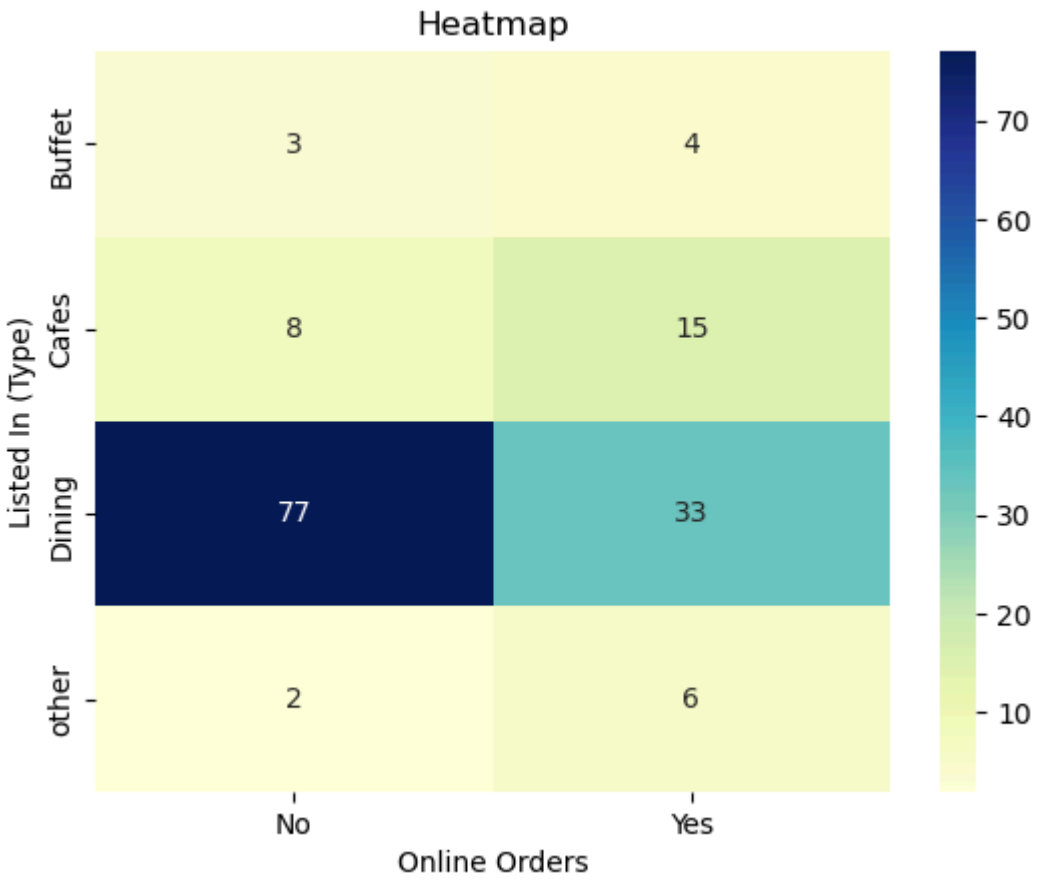
Conclusion - offline order received lower rating in comparison to online order

In [14]:
```
dataframe.head()
```

Out[14]:

	name	online_order	book_table	rate	votes	approx_cost(for two people)	listed_in(type)
0	Jalsa	Yes	Yes	4.1	775	800	Buffet
1	Spice Elephant	Yes	No	4.1	787	800	Buffet
2	San Churro Cafe	Yes	No	3.8	918	800	Buffet
3	Addhuri Udupi Bhojana	No	No	3.7	88	300	Buffet
4	Grand Village	No	No	3.8	166	600	Buffet

In [15]:
```
pivot_table = dataframe.pivot_table(index='listed_in(type)', columns='online_order', aggfunc=
sns.heatmap(pivot_table, annot=True, cmap='YlGnBu', fmt='d')
plt.title('Heatmap')
plt.xlabel('Online Orders')
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.