

## Import necessary Python libraries

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

## Create the data frame

```
In [2]: 1 dataframe = pd.read_csv("Zomato data .csv")
        2 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 Udipi 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

## Convert the data type of the “rate” column to float and remove the denominator

```
In [3]: 1 def handleRate(value):
2         value=str(value).split('/')
3         value=value[0];
4         return float(value)
5
6         dataframe['rate']=dataframe['rate'].apply(handleRate)
7         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 Udipi 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

## Summary of the data frame

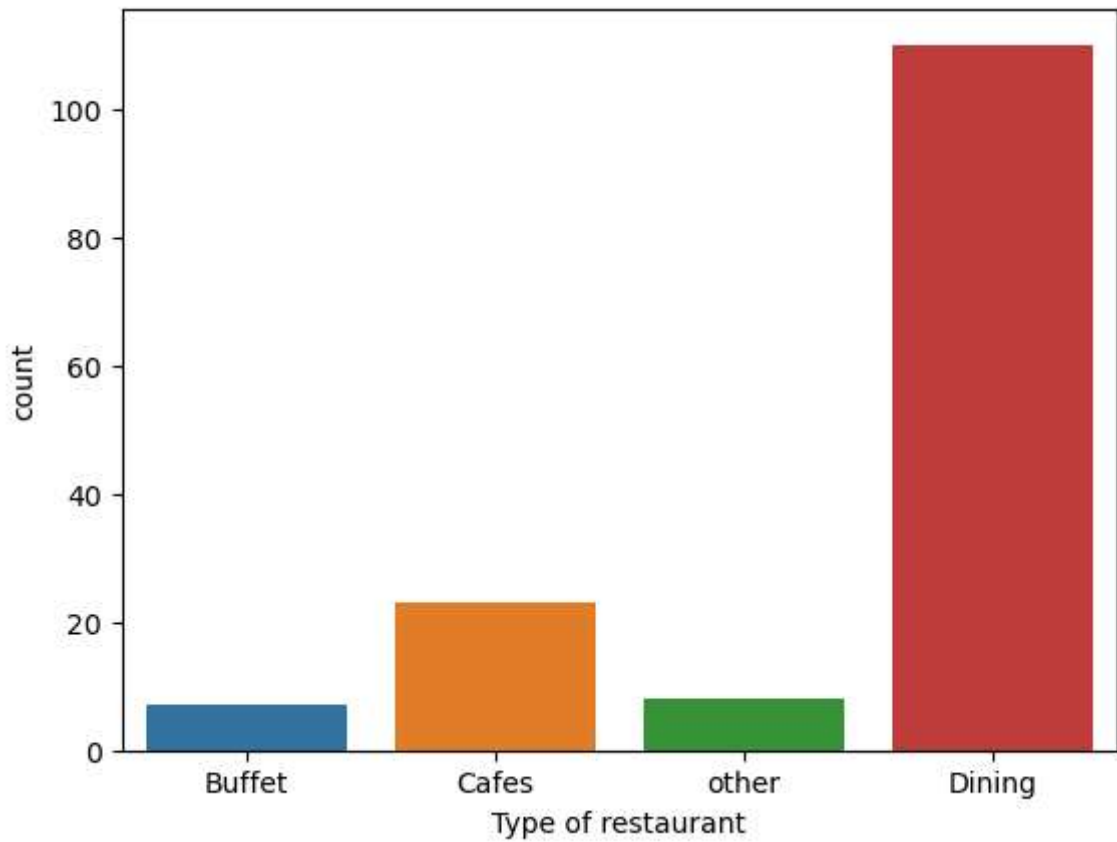
```
In [4]: 1 dataframe.info()
```

```
<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
```

## Exploring the listed\_in (type) column

```
In [5]: 1 sns.countplot(x=dataframe['listed_in(type)'])  
        2 plt.xlabel("Type of restaurant")
```

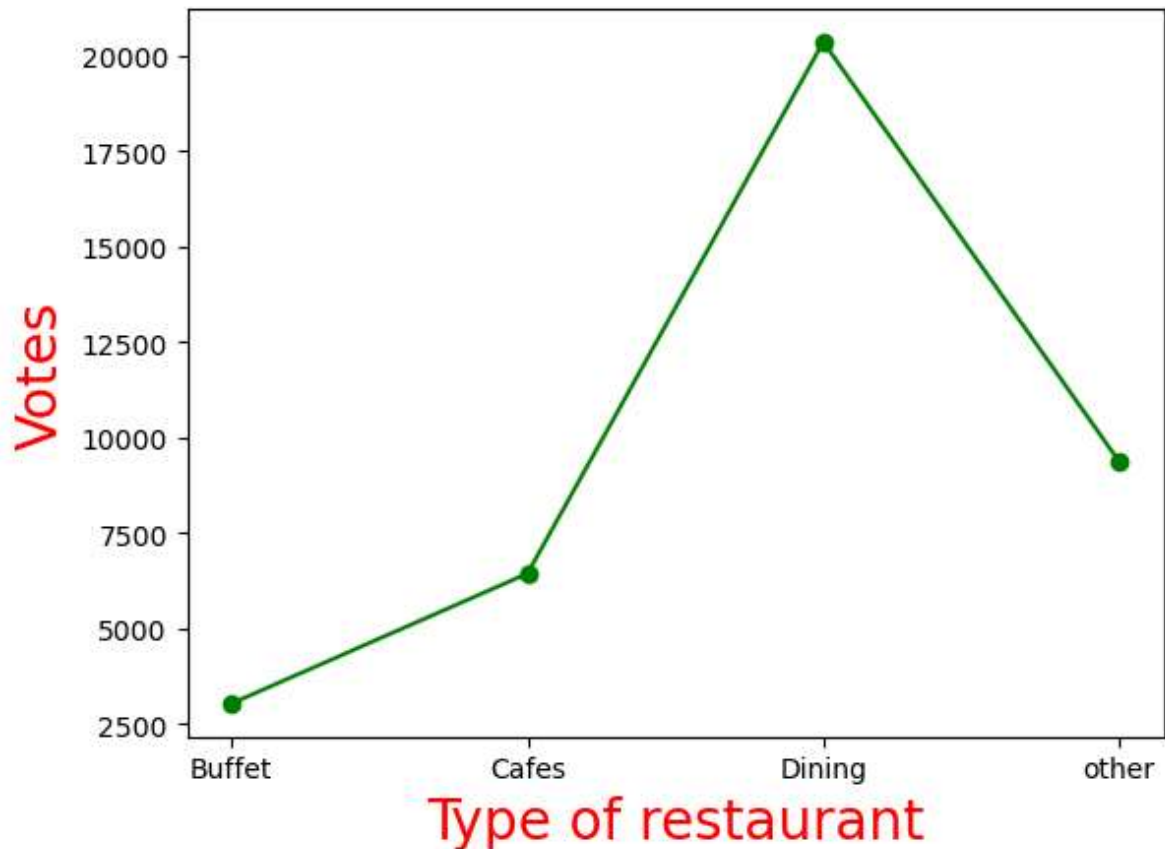
Out[5]: Text(0.5, 0, 'Type of restaurant')



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

```
In [6]: 1 grouped_data = dataframe.groupby('listed_in(type)')['votes'].sum()
2 result = pd.DataFrame({'votes': grouped_data})
3 plt.plot(result, c="green", marker="o")
4 plt.xlabel("Type of restaurant", c="red", size=20)
5 plt.ylabel("Votes", c="red", size=20)
```

Out[6]: Text(0, 0.5, 'Votes')



Conclusion: Dining restaurants are preferred by a larger number of individuals.

## Determining the restaurant's name that received the maximum votes based on a given dataframe

```
In [7]: 1 max_votes = dataframe['votes'].max()
2 restaurant_with_max_votes = dataframe.loc[dataframe['votes'] == max_votes]
3
4 print("Restaurant(s) with the maximum votes:")
5 print(restaurant_with_max_votes)
```

Restaurant(s) with the maximum votes:

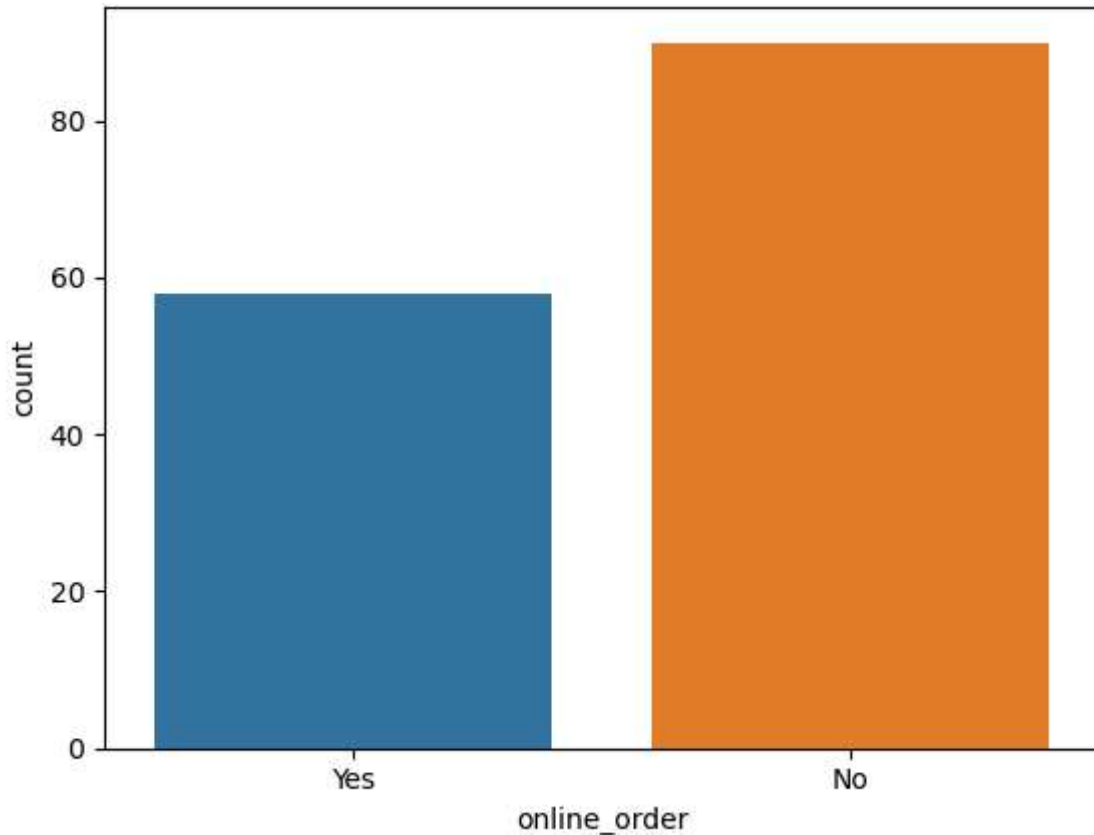
38 Empire Restaurant

Name: name, dtype: object

## Exploring the online\_order column

```
In [9]: 1 sns.countplot(x=dataframe['online_order'])
```

```
Out[9]: <Axes: xlabel='online_order', ylabel='count'>
```

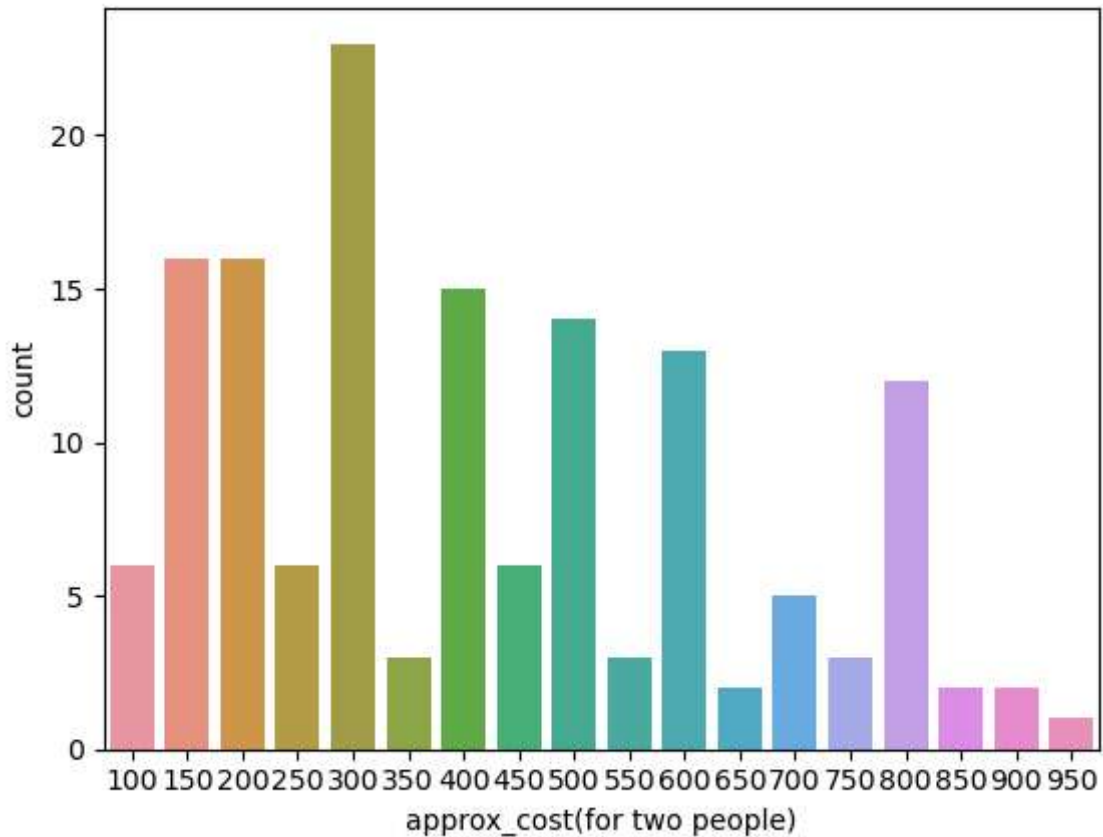


Conclusion: This suggests that a majority of the restaurants do not accept online orders.

## Exploring the approx\_cost(for two people) column

```
In [10]: 1 couple_data=dataframe['approx_cost(for two people)']  
        2 sns.countplot(x=couple_data)
```

Out[10]: <Axes: xlabel='approx\_cost(for two people)', ylabel='count'>



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

```
In [ ]: 1
```