

Zomato data analysis project

```
In [1]: # STEP 1 :
# IMPORTING LIBRARIES
# PANDAS is used for data manipulating and analysis
# NUMPY is used for numerical operations.
# MATPLOTLIB.PYPLLOT and SEABORN are used for data visualization
```

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

```
In [4]: # STEP 2:
# CREATW THE DATA FRAME
```

```
In [8]: dataframe=pd.read_csv("Zomato data .csv")
print(dataframe)
```

	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	
..	
143	Melting Melodies	No	No	3.3/5	0	
144	New Indraprasta	No	No	3.3/5	0	
145	Anna Kuteera	Yes	No	4.0/5	771	
146	Darbar	No	No	3.0/5	98	
147	Vijayalakshmi	Yes	No	3.9/5	47	

	approx_cost(for two people)	listed_in(type)
0	800	Buffet
1	800	Buffet
2	800	Buffet
3	300	Buffet
4	600	Buffet
..
143	100	Dining
144	150	Dining
145	450	Dining
146	800	Dining
147	200	Dining


[148 rows x 7 columns]

In [9]:  dataframe

Out[9]:

	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

In [10]:  # CONVERT THE DATA TYPE OF COLUMN - RATE
In [11]: 

```
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

In [13]: `dataframe.info()` *#find missing*

```
<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
```

In [14]: `# TYPE OF RESTURANT`

In [15]: `dataframe.head()` *# SHOWS STARTING 5 DATA*

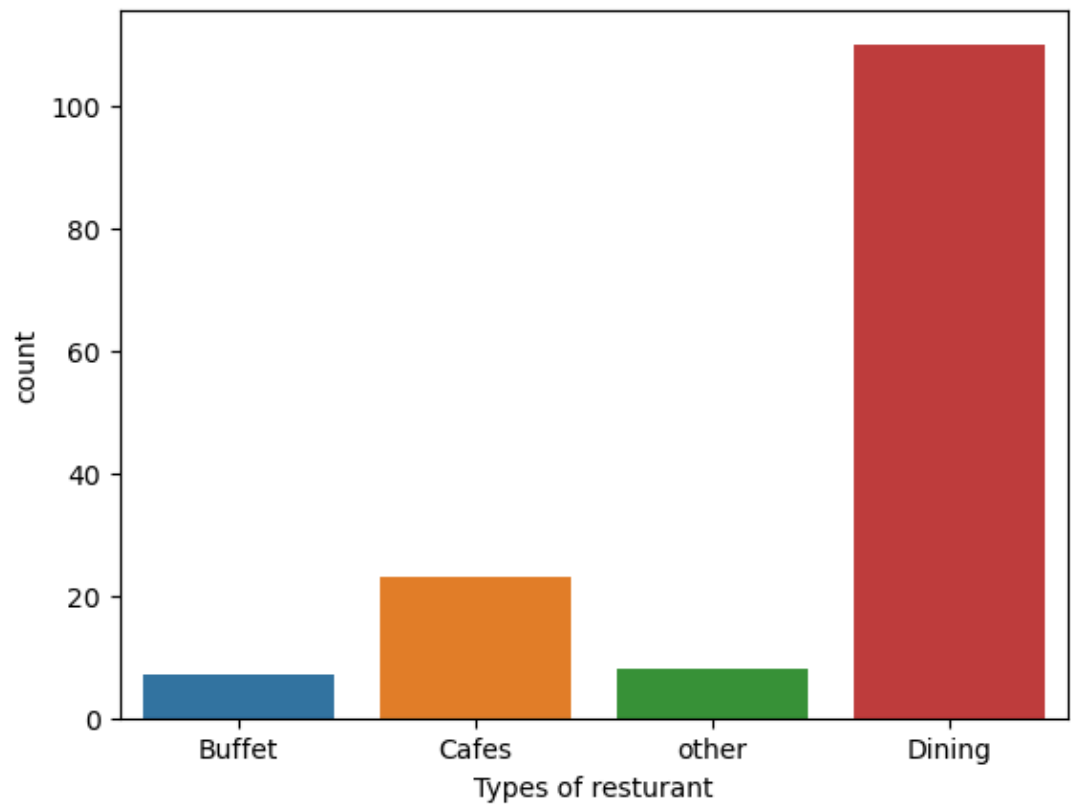
Out[15]:

	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 []: `# 1) WHAT TYPE OF RESTURANT DO THE MAJORITY OF CUSTOMERS ORDER FROM?`

```
In [17]: ▶ sns.countplot(x=dataframe['listed_in(type)'])  
          plt.xlabel("Types of resturant")
```

```
Out[17]: Text(0.5, 0, 'Types of resturant')
```

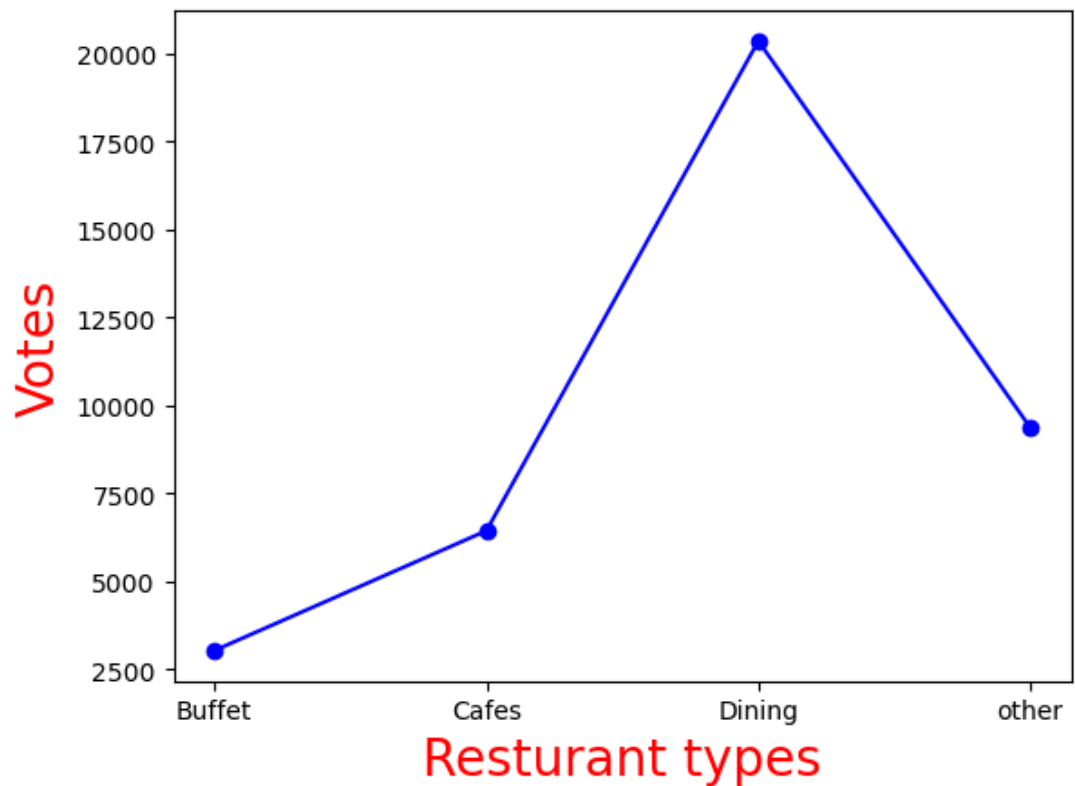


CONCLUSION = MAJORITY OF THE RESTURANT FALLS IN DINNING CATEGORY

```
In [18]: ▶ # HOW MANY VOTES HAS EACH TYPE OF RESTURANT RECEIVED FROM CUSTOMERS?
```

```
In [21]: grouped_data=dataframe.groupby('listed_in(type)')['votes'].sum()  
result=pd.DataFrame({'votes':grouped_data})  
plt.plot(result,c="blue",marker="o")  
plt.xlabel("Resturant types",c="red",size=20)  
plt.ylabel("Votes",c="red",size=20)
```

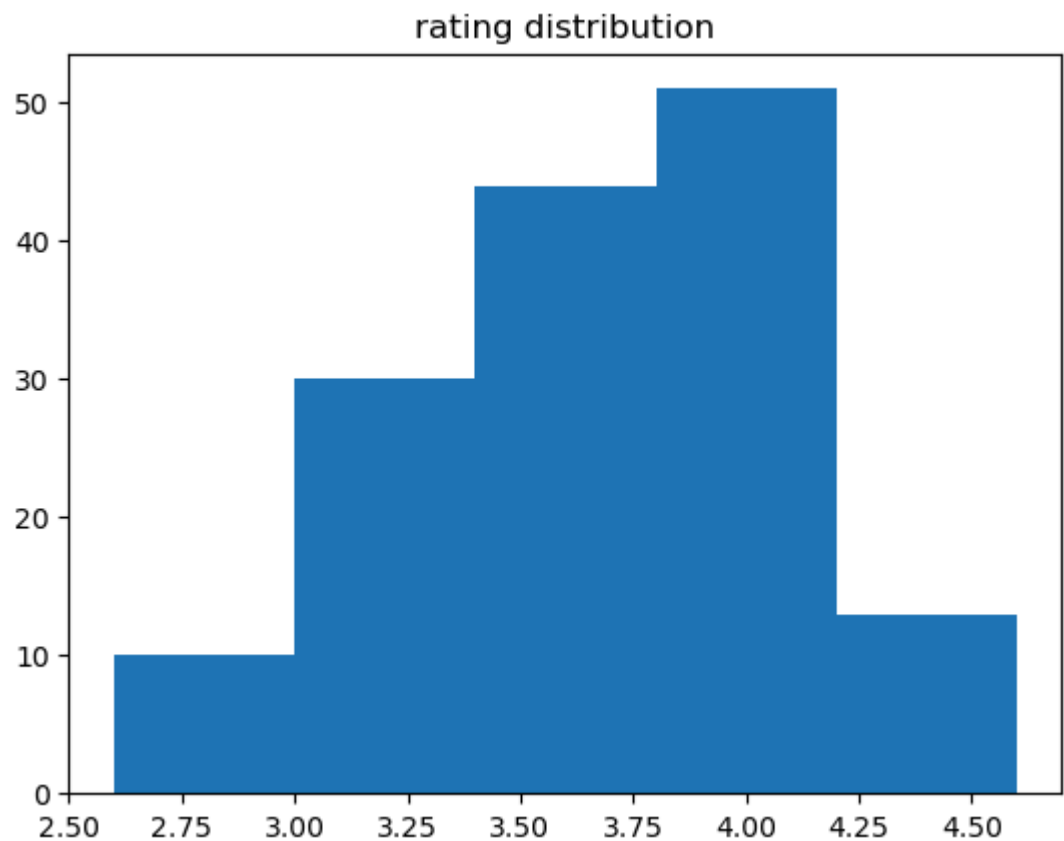
Out[21]: Text(0, 0.5, 'Votes')



CONCLUSION - DINING RESTURENTS HAS RECEIVED MAX. VOTES

```
In [22]: # WHAT ARE THE RATINGS THAT THE MAJORITY OF RESTURANTS HAVE RECEIVED ?
```

```
In [25]: ▶ plt.hist(dataframe['rate'],bins=5)  
plt.title("rating distribution")  
plt.show()
```

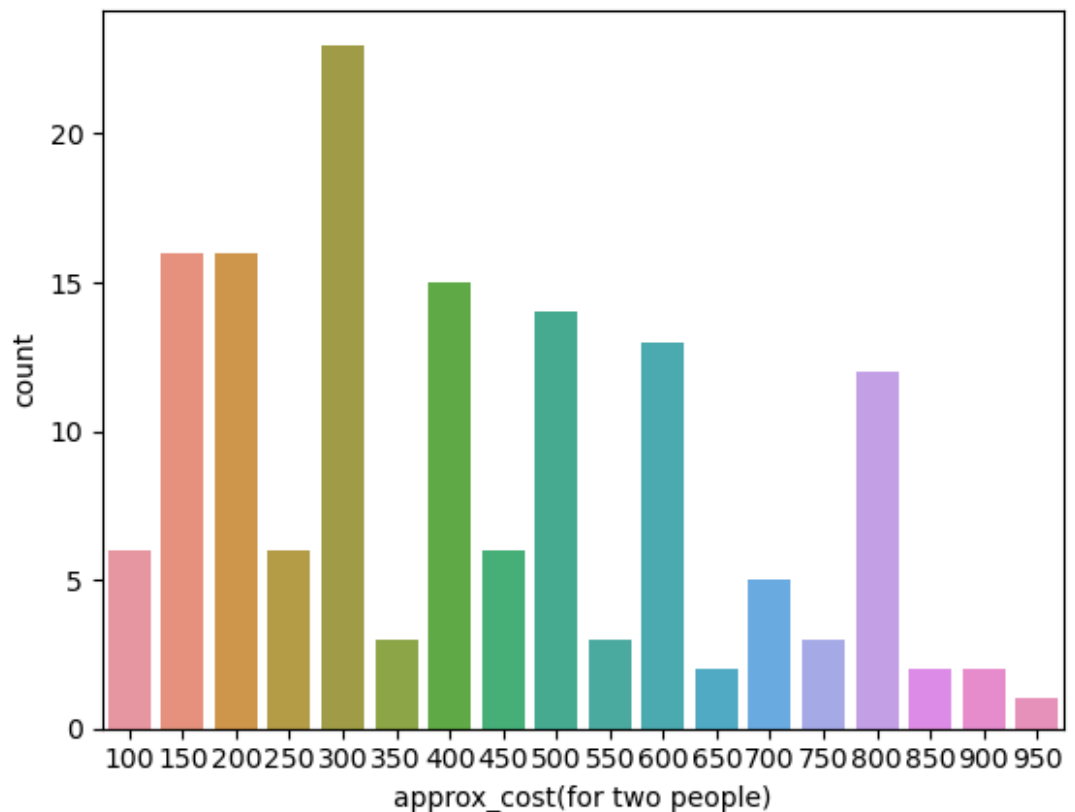


CONCLUSION - THE MAJORITY RETURANTS RECEIVED RATINGS FROM 3.5 TO 4

```
In [26]: ▶ # ZOMATO HAS OBSERVED THAT MOST COUPLES ORDER MOST OF THEIR FOOD ONLINE.
```

```
In [28]: couple_data=dataframe['approx_cost(for two people)']  
sns.countplot(x=couple_data)
```

```
Out[28]: <Axes: xlabel='approx_cost(for two people)', ylabel='count'>
```

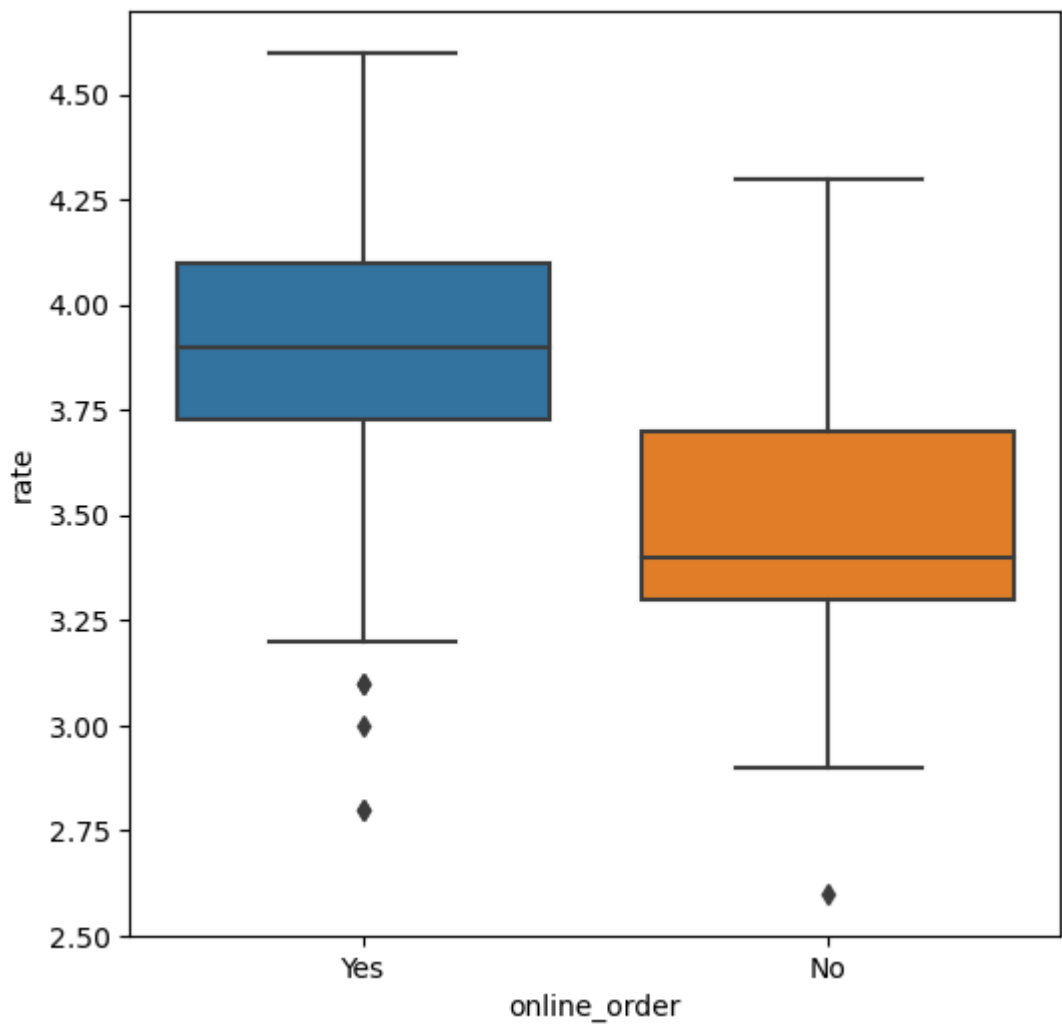


**CONCLUSION - THE MAJORITY OF COUPLES
PREFER RESTURANTS WITH AN APPROX
COST OF 300 RUPEES**

```
In [29]: # WHICH MODE (ONLINE OR OFFLINE) HAS RECIEVED THE MAXIMUM RATING ?
```

```
In [31]: ▶ plt.figure(figsize=(6,6))  
sns.boxplot(x='online_order',y='rate',data=dataframe)
```

```
Out[31]: <Axes: xlabel='online_order', ylabel='rate'>
```

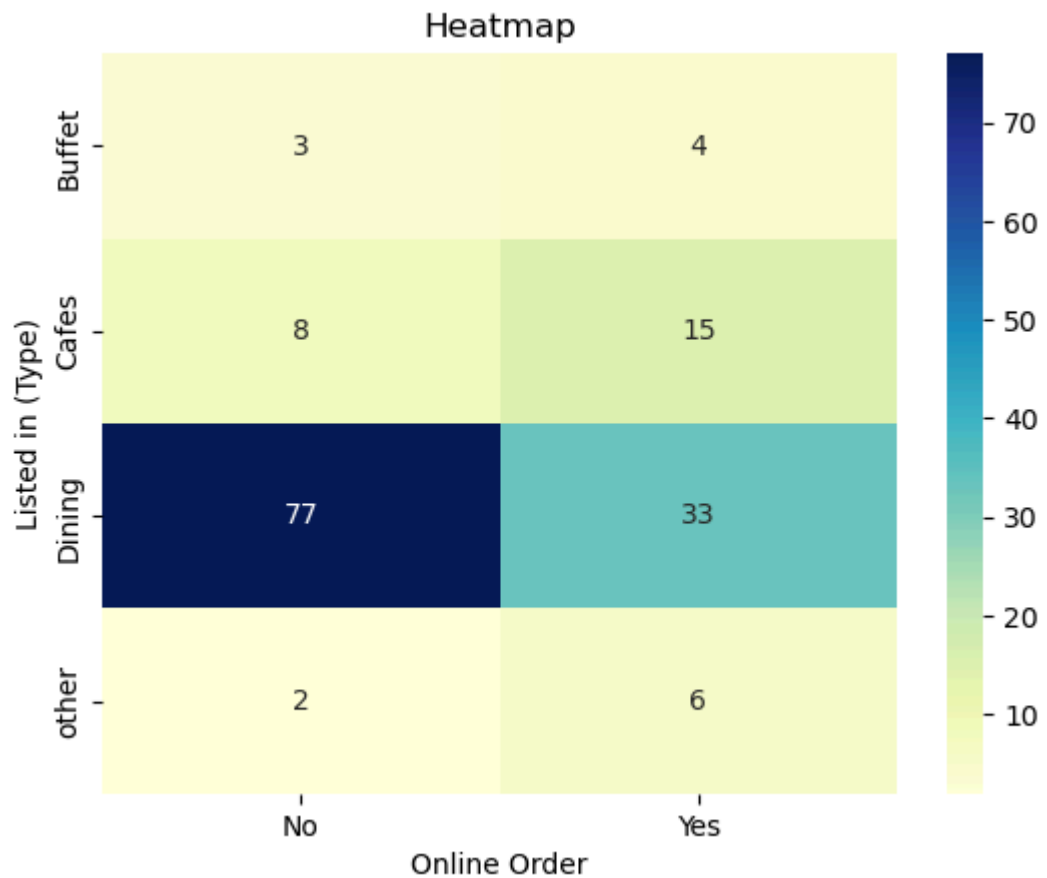


CONCLUSION - OFFLINE ORDER RECEIVED LOWER RATING IN COMPARISON TO ONLINE

```
In [32]: ▶ # WHICH TYPE OF RESTURANT RECEIVED MORE OFFLINE ORDERS , SO THAT ZOMATO
```



```
In [36]: ▶ pivot_table=dataframe.pivot_table(index='listed_in(type)',columns='online_order',aggfunc='sum')
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 - DINNING RETURANTS
ACCEPT OFFLINE ORDERS, CAFES RECEIVE
ONLINE ORDER, THIS SUGGESTS THAT
CLIENTS PREFER TO ORDERS IN PERSON
AT RESTURANTS ,BUT PREFER ONLINE
ORDERING AT CAFES**

```
In [ ]: ▶
```