Zomato data analysis

Step 1 - importing libraries

Double-click (or enter) to edit

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

Step 2 - create a dataframe

```
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]
```

```
dataframe
```

	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

Next steps: Generate code with dataframe | View recommended plots | New interactive sheet

```
dataframe.head()
```

	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

Next steps: Generate code with dataframe | View recommended plots | New interactive sheet

Covert thev data type of Column - Rate

```
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
```

```
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
```

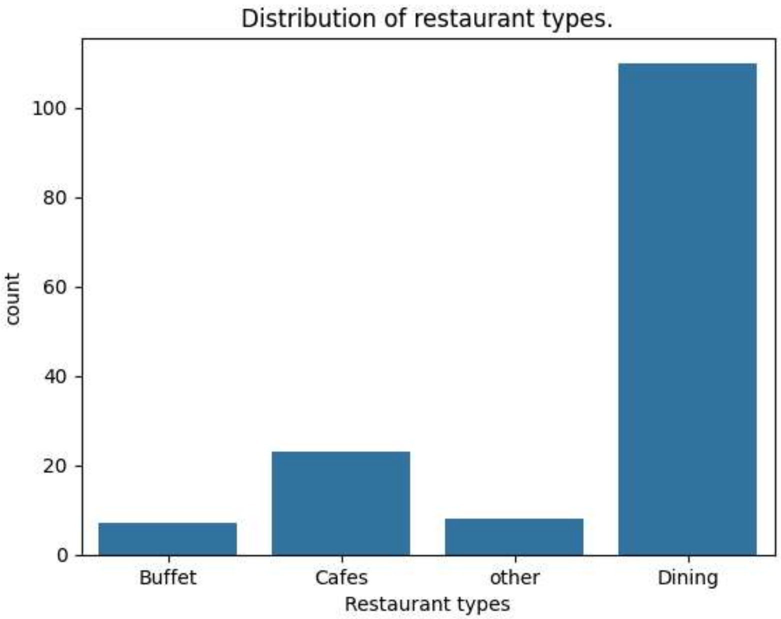
Analysis 1:Type of Restaurant

```
dataframe.head()
```

	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

Next steps: Generate code with dataframe   View recommended plots   New interactive sheet

```
sns.countplot(x=dataframe['listed_in(type)'])
plt.xlabel("Restaurant types")
plt.title("Distribution of restaurant types.")
plt.show()
```



Conclusion - Majority of the restaurant fall in the dinning category

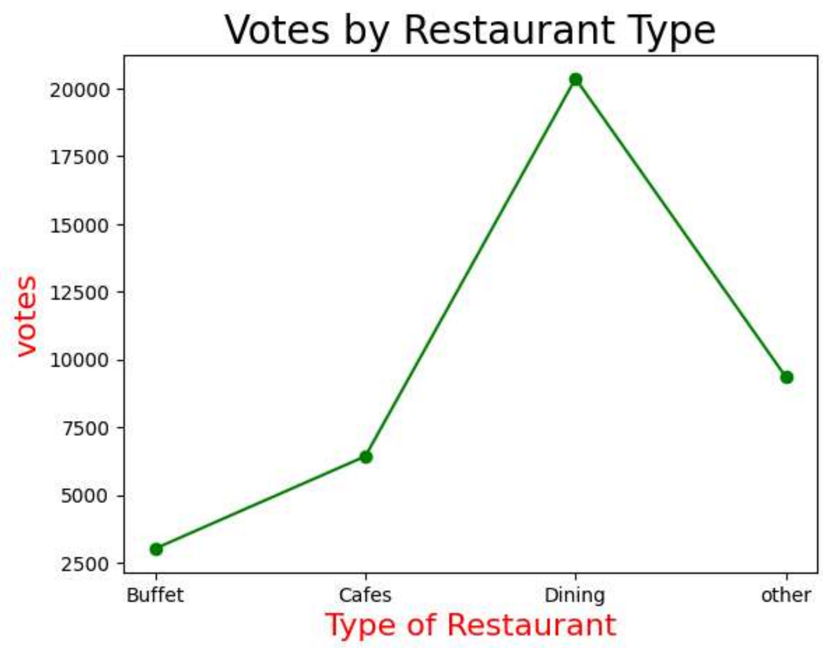
Analysis 2:Votes by Restaurant Type

```
dataframe.head()
```

	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

Next steps: Generate code with dataframe | View recommended plots | New interactive sheet

```
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=16)
plt.ylabel("votes", c="red", size=16)
plt.title("Votes by Restaurant Type",size=20)
plt.show()
```



Conclusion - Dinning restaurants has recieved maximum votes

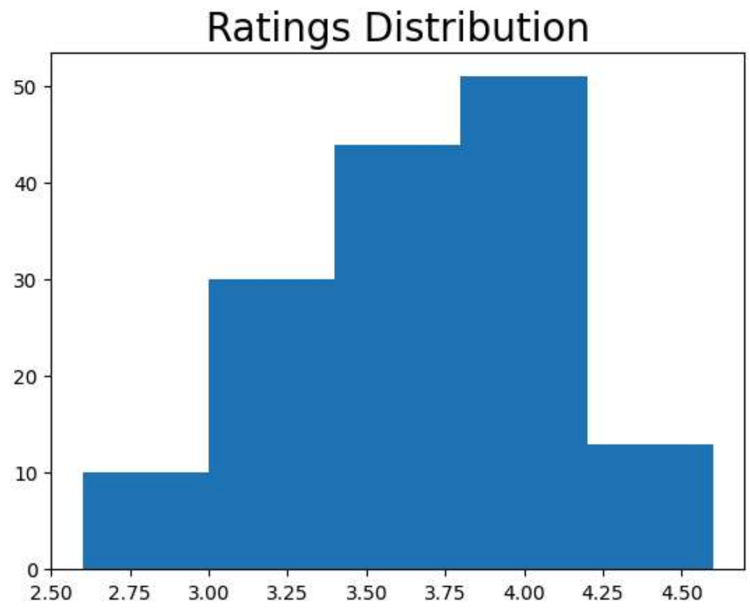
Analysis 3:Ratings Distribution

```
dataframe.head()
```

	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

Next steps: Generate code with `dataframe` View recommended plots New interactive sheet

```
plt.hist(dataframe['rate'], bins= 5)
plt.title("Ratings Distribution",size=20)
plt.show()
```



Conclusion - Majority restaurant received ratings from 3.5 - 4

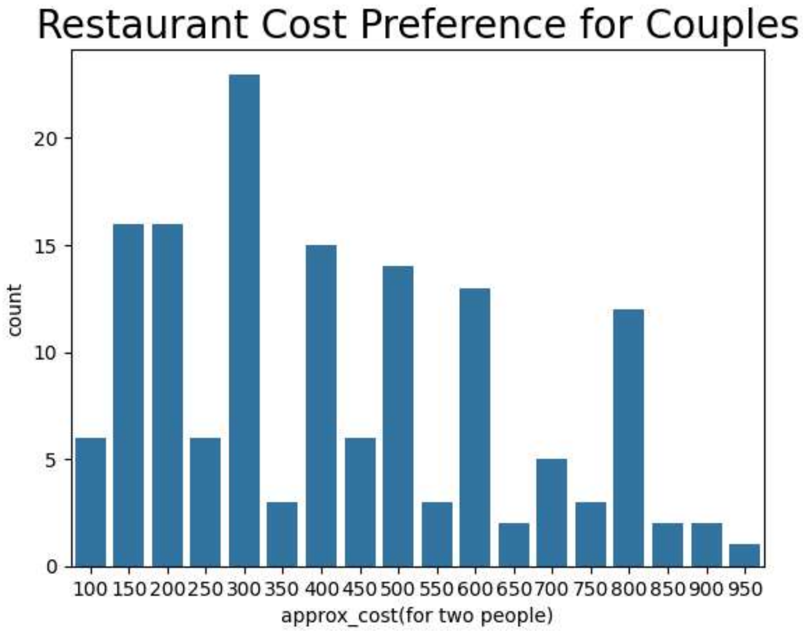
Analysis 4: Restaurant Cost Preference for Couples

```
dataframe.head()
```

	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

Next steps: Generate code with dataframe | View recommended plots | New interactive sheet

```
couple_data = dataframe['approx_cost(for two people)']
sns.countplot(x=couple_data)
plt.title("Restaurant Cost Preference for Couples",size=20)
plt.show()
```



Conclusion - majority of couples prefer restaurants with an approximate cost of 300 rupees

Analysis 5: Online vs. Offline Ratings

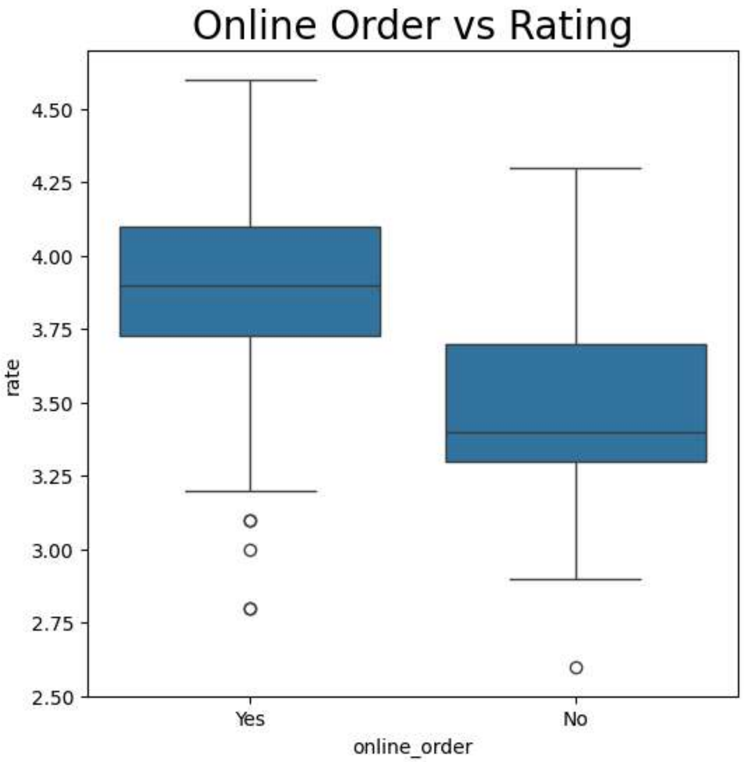
Which mode recieves maximum rating

```
dataframe.head()
```

	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

Next steps: Generate code with dataframe | View recommended plots | New interactive sheet

```
plt.figure(figsize=(6,6))
sns.boxplot(x= 'online_order',y='rate',data=dataframe)
plt.title("Online Order vs Rating",size=20)
plt.show()
```



Conclusion - Offline order received lower rating compared to online rating

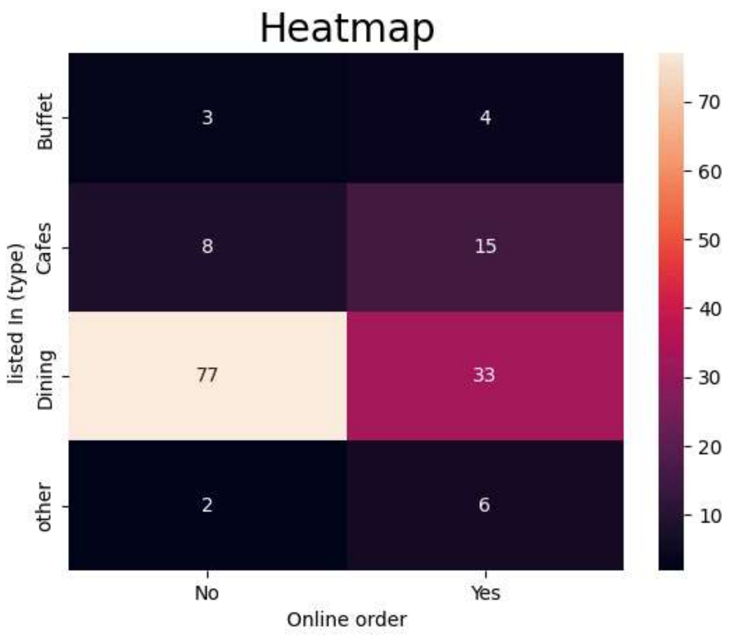
Analysis 6: Online Orders by Restaurant Type

```
dataframe.head()
```

	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

Next steps: Generate code with `dataframe` View recommended plots New interactive sheet

```
pivot_table = dataframe.pivot_table(index= 'listed_in(type)',columns='online_order', aggfunc='size', fill_value=0)
sns.heatmap(pivot_table, annot=True,fmt='d')
plt.xlabel("Online order")
plt.ylabel("listed In (type)")
plt.title("Heatmap",size=20)
plt.show()
```



Conclusion - Dinning restaurants primarily accept offline orders, whereas cafes primarily recieve online orders. This suggests that clients prefers orders in person at restaurants, but prefer online ordering at cafes.