

zomato



PYTHON ANALYSIS





Zomato data analysis project

Importing Libraries

```
In [28]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

Create the data frame

```
In [29]: dataframe = pd.read_csv("C:/Users/Dell Laptop/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
```

```
In [30]: dataframe
```

Out[30]:

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

148 rows × 7 columns

Convert the date type of column - rate

In [31]:

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

dataframe['rate']=dataframe['rate'].apply(handelRate)
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

Summary of the data frame

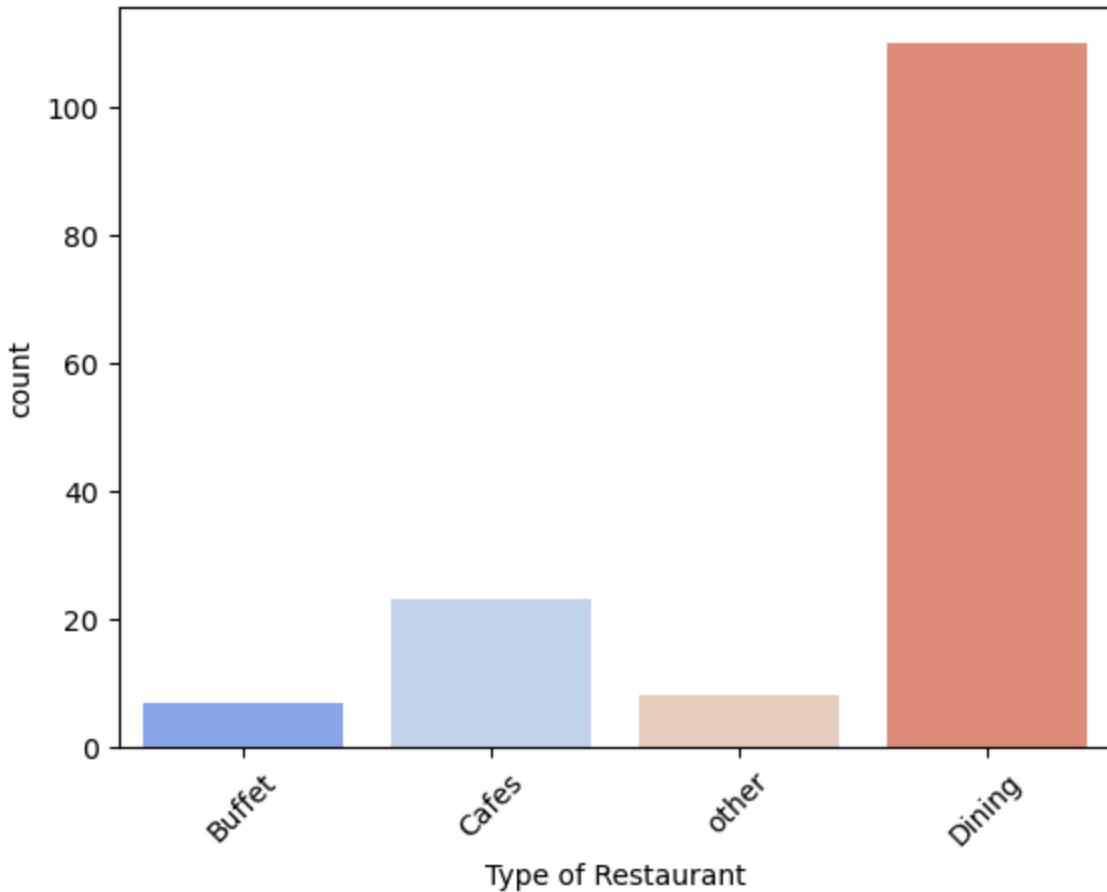
```
In [32]: dataframe.info()
```

```
<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
```

Type of restaurant

```
In [33]: sns.countplot(
    x='listed_in(type)',
    hue='listed_in(type)',
    data=dataframe,
    palette='coolwarm',
    legend=False
)

plt.xlabel("Type of Restaurant")
plt.xticks(rotation=45)
plt.show()
```



Conclusion - majority of the restaurant fall in dinning Category

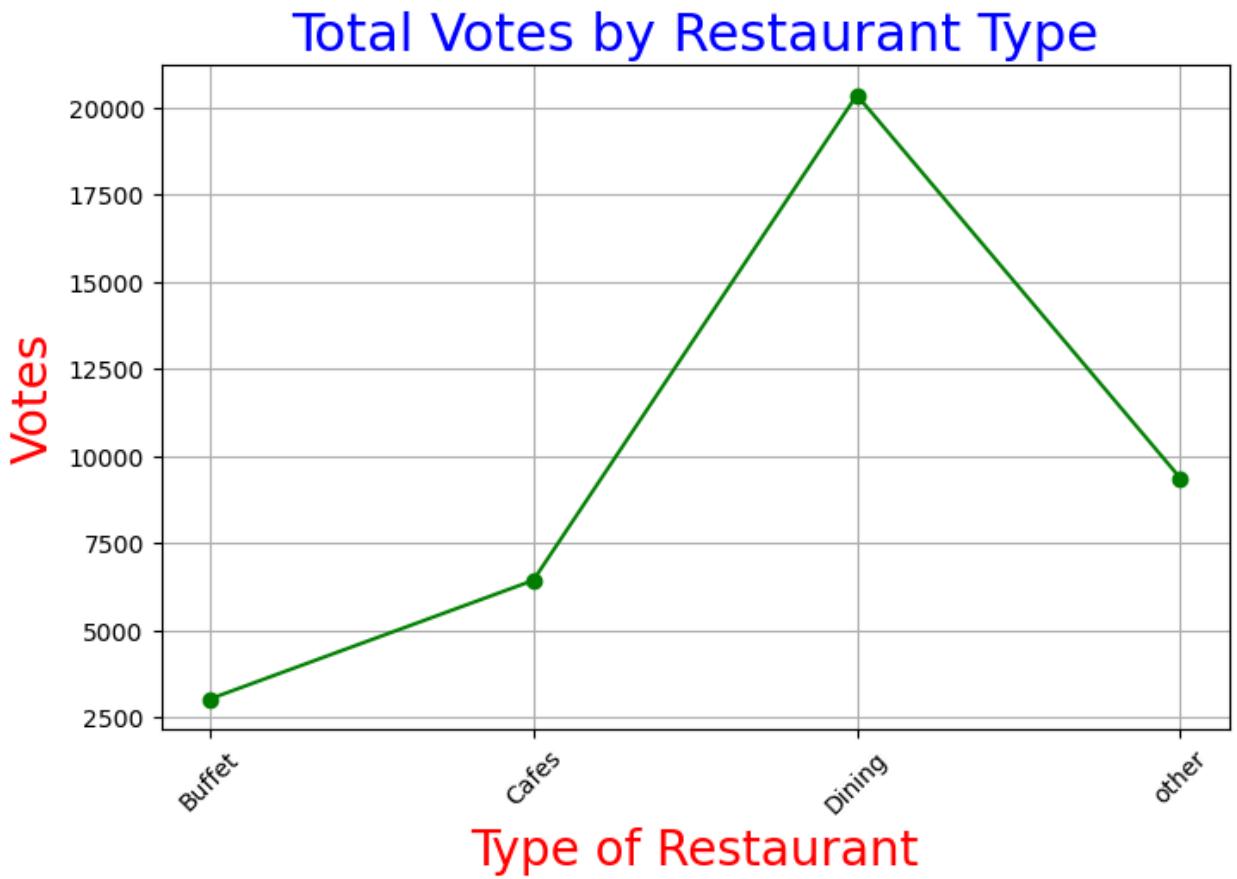
```
In [40]: grouped_data = dataframe.groupby('listed_in(type)')['votes'].sum()

result = pd.DataFrame({'Type': grouped_data.index, 'votes': grouped_data.values})

plt.figure(figsize=(8,5))
plt.plot(result['Type'], result['votes'], color="green", marker="o")

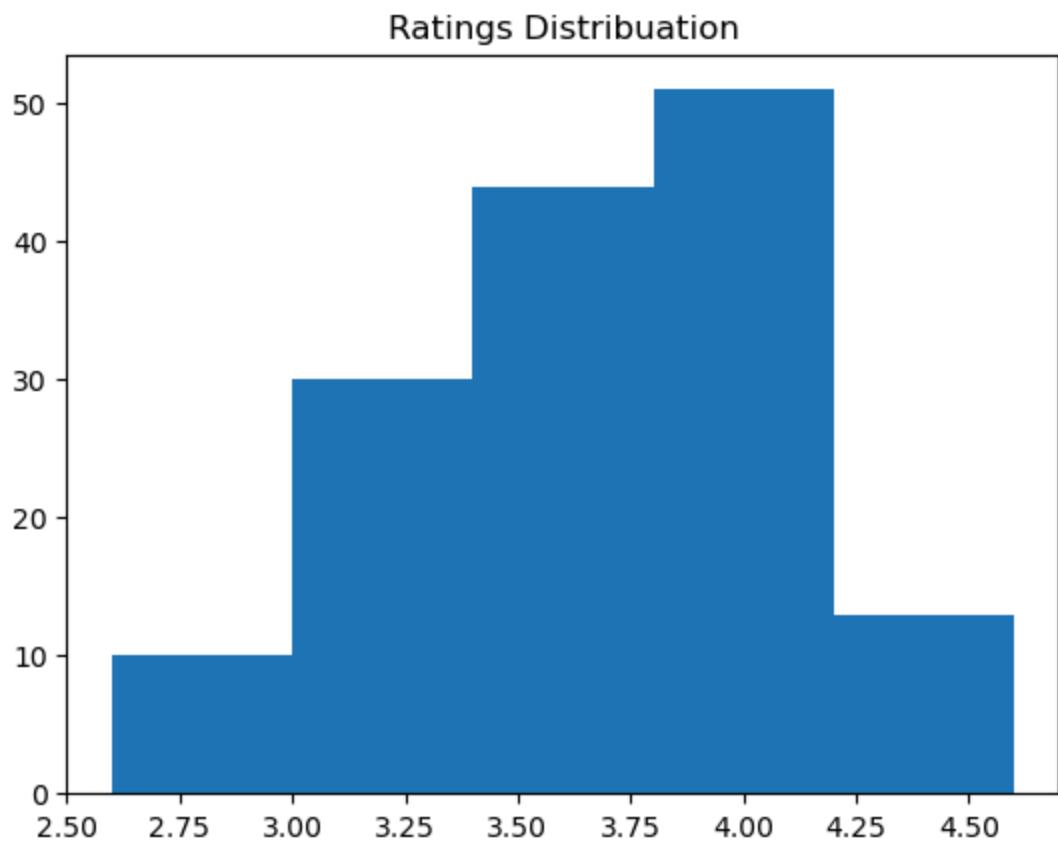
plt.xlabel("Type of Restaurant", color="red", fontsize=20)
plt.ylabel("Votes", color="red", fontsize=20)
plt.title("Total Votes by Restaurant Type", color="blue", fontsize=22)

plt.xticks(rotation=45)
plt.grid(True)
plt.show()
```



Majority of the restaurant received rating

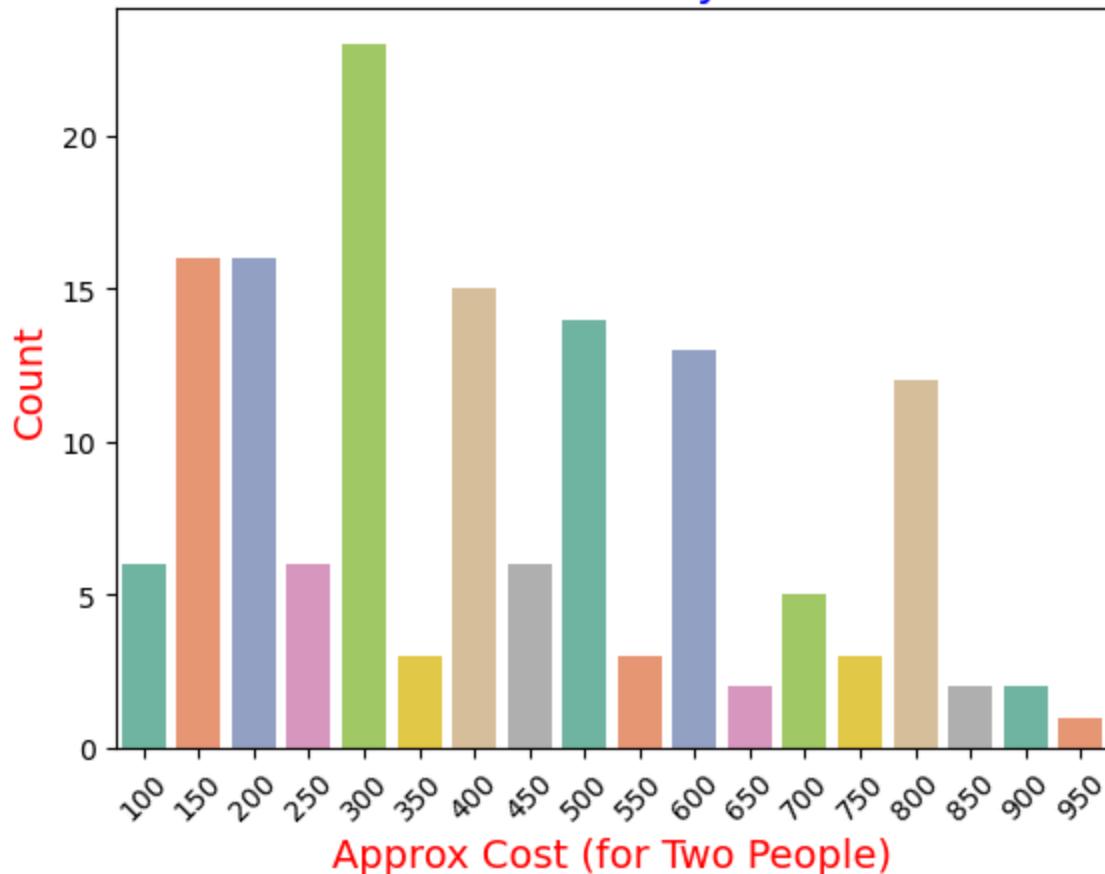
```
In [42]: plt.hist(dataframe['rate'],bins=5)
plt.title("Ratings Distribution")
plt.show()
```



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

```
In [47]: sns.countplot(  
    x='approx_cost(for two people)',  
    hue='approx_cost(for two people)',  
  
    data=dataframe,  
    palette='Set2',  
    legend=False  
)  
  
plt.xlabel("Approx Cost (for Two People)", color="red", fontsize=14)  
plt.ylabel("Count", color="red", fontsize=14)  
plt.title("Restaurants by Cost", color="blue", fontsize=16)  
plt.xticks(rotation=45)  
plt.show()
```

Restaurants by Cost



Which mode receives maximum rating

```
In [49]: plt.figure(figsize=(6,6))
sns.boxplot(
    x='online_order',
    y='rate',
    data=dataframe,
    palette='Set2',
    width=0.5,
    fliersize=5
)

plt.title("Restaurant Ratings vs Online Order Availability", color="blue", fontcolor="blue")
plt.xlabel("Online Order", color="red", fontsize=12)
plt.ylabel("Ratings", color="red", fontsize=12)
plt.grid(True, linestyle='--', alpha=0.4)
plt.show()
```

```
C:\Users\Dell Laptop\AppData\Local\Temp\ipykernel_7764\4236540427.py:2: FutureWarning:
```

```
Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign the `x` variable to `hue` and set `legend=False` for the same effect.
```

```
sns.boxplot(
```

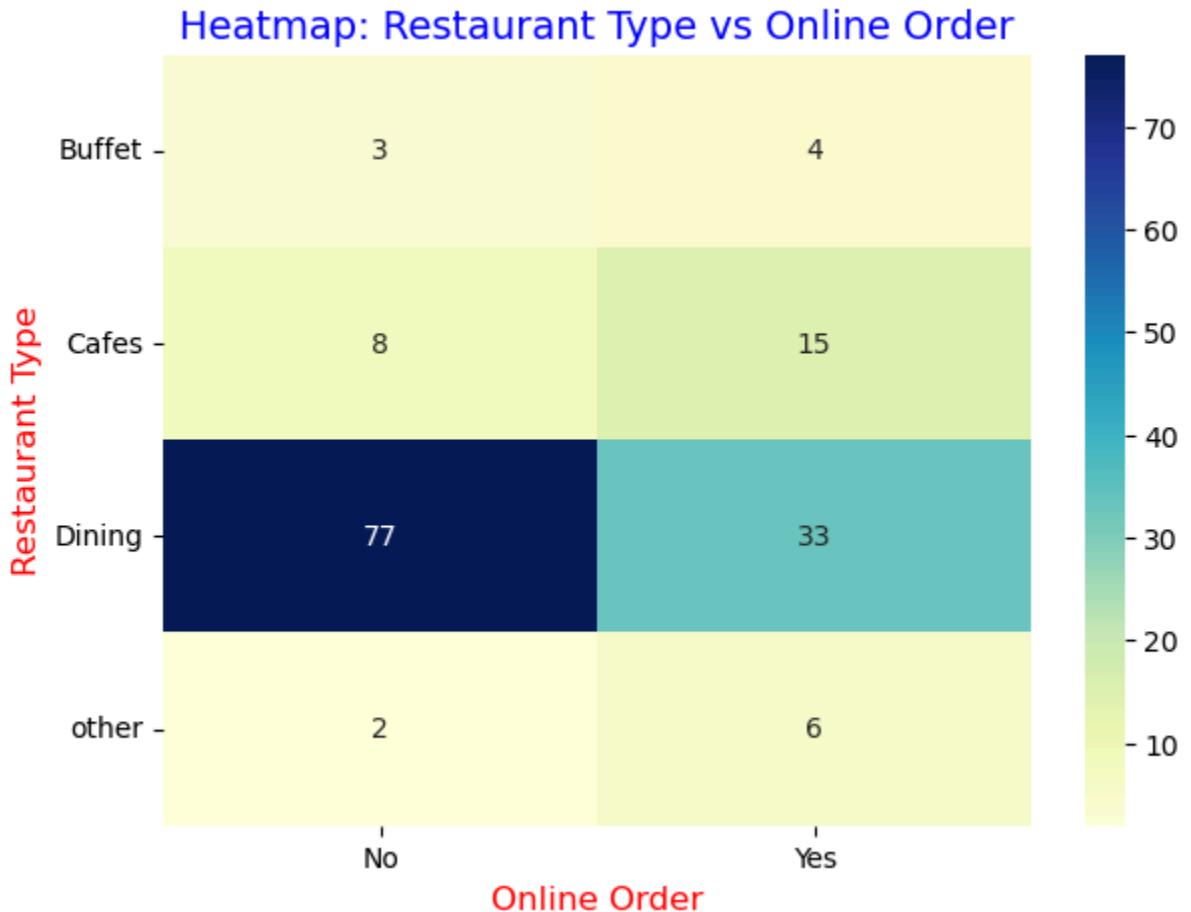


conclusion - offline order received lower rating in comparison to online order

```
In [51]: pivot_table = dataframe.pivot_table(  
    index='listed_in(type)',  
    columns='online_order',  
    aggfunc='size',  
    fill_value=0  
)  
  
plt.figure(figsize=(7,5))
```

```
sns.heatmap(pivot_table, annot=True, cmap="YlGnBu", fmt='d')

plt.title("Heatmap: Restaurant Type vs Online Order", fontsize=14, color="blue")
plt.xlabel("Online Order", fontsize=12, color="red")
plt.ylabel("Restaurant Type", fontsize=12, color="red")
plt.xticks(rotation=0)
plt.yticks(rotation=0)
plt.show()
```



In []: