

# Exploratory Data Analysis using Python

Dataset contains information about restaurants in Bangalore working with Zomato.

Following task will be performed on the data:

- 1) Data Cleaning
- 2) Data Visualization

## 1) Data Cleaning

### Import Libraries

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

### Read Data

```
In [3]: zomato = pd.read_csv(r'C:\Users\patil\Desktop\Project 1\zomato.csv\zomato.csv')
zomato.head(3)
```

Out[3]:

		url	address	name	online_order	book_table	rate
0		https://www.zomato.com/bangalore/jalsa-banasha...	942, 21st Main Road, 2nd Stage, Bananashankari, ...	Jalsa	Yes	Yes	4.1/5
1		https://www.zomato.com/bangalore/spice-elephan...	2nd Floor, 80 Feet Road, Near Big Bazaar, 6th ...	Spice Elephant	Yes	No	4.1/5
2		https://www.zomato.com/SanchurroBangalore?cont...	1112, Next to KIMS Medical College, 17th Cross...	San Churro Cafe	Yes	No	3.8/5

Delete unnecessary columns

```
In [4]: zomato.drop(columns = ['url','address','phone','dish_liked','reviews_list','menu_item'])
```

```
In [5]: zomato.head(1)
```

Out[5]:

	name	online_order	book_table	rate	votes	location	rest_type	cuisines	approx_cost(for two people)
0	Jalsa	Yes	Yes	4.1/5	775	Banashankari	Casual Dining	North Indian, Mughlai, Chinese	800

## Renaming Columns

```
In [6]: zomato.rename(columns = {'name':'restaurant'})
```

```
, 'rate':'rating'  
, 'approx_cost(for two people)':'cost'  
, 'listed_in(type)':'type'  
, 'listed_in(city)':'city'  
, inplace=True)
```

```
In [7]: zomato.head(1)
```

Out[7]:

	restaurant	online_order	book_table	rating	votes	location	rest_type	cuisines	cost	type
0	Jalsa	Yes	Yes	4.1/5	775	Banashankari	Casual Dining	North Indian, Mughlai, Chinese	800	Buffet

## Dropping NaN values

```
In [8]: zomato.isna().value_counts()
```

```
Out[8]:   restaurant  online_order  book_table  rating  votes  location  rest_type  cuisines  c
          ost      type      city
          False     False      False      False    False    False    False    False    False    F
          alse     False     False      43533
          alse     False     False      7615
          rue     False     False      247
          alse     False     False      149
          rue     False     False      71
          alse     False     False      55
          rue     False     False      21
          alse     False     False      11
          alse     False     False      8
          rue     False     False      3
          rue     False     False      2
          rue     False     False      2
          dtype: int64
```

```
In [9]: zomato.dropna(inplace=True)
```

```
In [10]: zomato.isna().value_counts()
```

```
Out[10]:   restaurant  online_order  book_table  rating  votes  location  rest_type  cuisines  c
          ost      type      city
          False     False      False      False    False    False    False    False    False    F
          alse     False     False      43533
          dtype: int64
```

## Cleaning Individual Columns

Cleaning column *restaurant* : replace disturbed characters

```
In [11]: zomato['restaurant'].unique()
```

```
Out[11]: array(['Jalsa', 'Spice Elephant', 'San Churro Cafe', ...,
               'The Nest - The Den Bengaluru', 'Nawabs Empire',
               'SeeYa Restaurant'], dtype=object)
```

```
In [12]: zomato['restaurant'] = zomato['restaurant'].str.replace('[Ã][^C]+', '', regex=True)
```

```
In [13]: zomato['restaurant'].unique()
```

```
Out[13]: array(['Jalsa', 'Spice Elephant', 'San Churro Cafe', ...,
               'The Nest - The Den Bengaluru', 'Nawabs Empire',
               'SeeYa Restaurant'], dtype=object)
```

Check column *online\_order*

```
In [14]: zomato['online_order'].unique()  
Out[14]: array(['Yes', 'No'], dtype=object)
```

Check column *book\_table*

```
In [15]: zomato['book_table'].unique()  
Out[15]: array(['Yes', 'No'], dtype=object)
```

Check column *rating*: removing /5 (out of 5) value ,replace NEW with '-' (dash), change datatype to float

```
In [16]: zomato['rating'].unique()  
Out[16]: array(['4.1/5', '3.8/5', '3.7/5', '3.6/5', '4.6/5', '4.0/5', '4.2/5',  
               '3.9/5', '3.1/5', '3.0/5', '3.2/5', '3.3/5', '2.8/5', '4.4/5',  
               '4.3/5', 'NEW', '2.9/5', '3.5/5', '2.6/5', '3.8 /5', '3.4/5',  
               '4.5/5', '2.5/5', '2.7/5', '4.7/5', '2.4/5', '2.2/5', '2.3/5',  
               '3.4 /5', '-', '3.6 /5', '4.8/5', '3.9 /5', '4.2 /5', '4.0 /5',  
               '4.1 /5', '3.7 /5', '3.1 /5', '2.9 /5', '3.3 /5', '2.8 /5',  
               '3.5 /5', '2.7 /5', '2.5 /5', '3.2 /5', '2.6 /5', '4.5 /5',  
               '4.3 /5', '4.4 /5', '4.9/5', '2.1/5', '2.0/5', '1.8/5', '4.6 /5',  
               '4.9 /5', '3.0 /5', '4.8 /5', '2.3 /5', '4.7 /5', '2.4 /5',  
               '2.1 /5', '2.2 /5', '2.0 /5', '1.8 /5'], dtype=object)
```

```
In [17]: replace = lambda x: x.replace('/5', "")  
list = []  
for a in map(replace,zomato['rating']):  
    if a!='NEW' and a!='-':  
        var = float(a)  
    list.append(var)  
zomato['rating']=list
```

```
In [18]: zomato['rating'].dtype  
Out[18]: dtype('float64')
```

Check column *votes*

```
In [19]: zomato['votes'].isnull().value_counts()  
Out[19]: False    43533  
Name: votes, dtype: int64
```

Check column *location*

```
In [20]: zomato['location'].unique()
```

```
Out[20]: array(['Banashankari', 'Basavanagudi', 'Mysore Road', 'Jayanagar',
   'Kumaraswamy Layout', 'Rajarajeshwari Nagar', 'Vijay Nagar',
   'Uttarahalli', 'JP Nagar', 'South Bangalore', 'City Market',
   'Bannerghatta Road', 'BTM', 'Kanakapura Road', 'Bommanahalli',
   'CV Raman Nagar', 'Electronic City', 'Wilson Garden',
   'Shanti Nagar', 'Koramangala 5th Block', 'Richmond Road', 'HSR',
   'Marathahalli', 'Koramangala 7th Block', 'Bellandur',
   'Sarjapur Road', 'Whitefield', 'East Bangalore',
   'Old Airport Road', 'Indiranagar', 'Koramangala 1st Block',
   'Frazer Town', 'MG Road', 'Brigade Road', 'Lavelle Road',
   'Church Street', 'Ulsoor', 'Residency Road', 'Shivajinagar',
   'Infantry Road', 'St. Marks Road', 'Cunningham Road',
   'Race Course Road', 'Commercial Street', 'Vasanth Nagar', 'Domlur',
   'Koramangala 8th Block', 'Ejipura', 'Jeevan Bhima Nagar',
   'Old Madras Road', 'Seshadripuram', 'Kammanahalli',
   'Koramangala 6th Block', 'Majestic', 'Langford Town',
   'Central Bangalore', 'Sanjay Nagar', 'Brookefield',
   'ITPL Main Road, Whitefield', 'Varthur Main Road, Whitefield',
   'Koramangala 2nd Block', 'Koramangala 3rd Block',
   'Koramangala 4th Block', 'Koramangala', 'Hosur Road',
   'Rajajinagar', 'RT Nagar', 'Banaswadi', 'North Bangalore',
   'Nagawara', 'Hennur', 'Kalyan Nagar', 'HBR Layout',
   'Rammurthy Nagar', 'Thippasandra', 'Kaggadasapura', 'Hebbal',
   'Kengeri', 'New BEL Road', 'Sankey Road', 'Malleshwaram',
   'Sadashiv Nagar', 'Basaveshwara Nagar', 'Yeshwantpur',
   'West Bangalore', 'Magadi Road', 'Yelahanka', 'Sahakara Nagar',
   'Jalahalli', 'Nagarbhavi', 'Peenya', 'KR Puram'], dtype=object)
```

Check column *rest\_type*

```
In [21]: zomato['rest_type'].isnull().value_counts(), zomato['rest_type'].unique()
```

```
Out[21]: (False    43533
          Name: rest_type, dtype: int64,
          array(['Casual Dining', 'Cafe', 'Casual Dining', 'Quick Bites',
                  'Casual Dining, Cafe', 'Cafe', 'Quick Bites, Cafe',
                  'Cafe, Quick Bites', 'Delivery', 'Mess', 'Dessert Parlor',
                  'Bakery, Dessert Parlor', 'Pub', 'Bakery', 'Takeaway, Delivery',
                  'Fine Dining', 'Beverage Shop', 'Sweet Shop', 'Bar',
                  'Dessert Parlor, Sweet Shop', 'Bakery, Quick Bites',
                  'Sweet Shop, Quick Bites', 'Kiosk', 'Food Truck',
                  'Quick Bites, Dessert Parlor', 'Beverage Shop, Quick Bites',
                  'Beverage Shop, Dessert Parlor', 'Takeaway', 'Pub, Casual Dining',
                  'Casual Dining, Bar', 'Dessert Parlor, Beverage Shop',
                  'Quick Bites, Bakery', 'Microbrewery, Casual Dining', 'Lounge',
                  'Bar, Casual Dining', 'Food Court', 'Cafe, Bakery', 'Dhaba',
                  'Quick Bites, Sweet Shop', 'Microbrewery',
                  'Food Court, Quick Bites', 'Quick Bites, Beverage Shop',
                  'Pub, Bar', 'Casual Dining, Pub', 'Lounge, Bar',
                  'Dessert Parlor, Quick Bites', 'Food Court, Dessert Parlor',
                  'Casual Dining, Sweet Shop', 'Food Court, Casual Dining',
                  'Casual Dining, Microbrewery', 'Lounge, Casual Dining',
                  'Cafe, Food Court', 'Beverage Shop, Cafe', 'Cafe, Dessert Parlor',
                  'Dessert Parlor, Cafe', 'Dessert Parlor, Bakery',
                  'Microbrewery, Pub', 'Bakery, Food Court', 'Club',
                  'Quick Bites, Food Court', 'Bakery, Cafe', 'Pub, Cafe',
                  'Casual Dining, Irani Cafee', 'Fine Dining, Lounge',
                  'Bar, Quick Bites', 'Confectionery', 'Pub, Microbrewery',
                  'Microbrewery, Lounge', 'Fine Dining, Microbrewery',
                  'Fine Dining, Bar', 'Dessert Parlor, Kiosk', 'Bhojanalya',
                  'Casual Dining, Quick Bites', 'Cafe, Bar', 'Casual Dining, Lounge',
                  'Bakery, Beverage Shop', 'Microbrewery, Bar', 'Cafe, Lounge',
                  'Bar, Pub', 'Lounge, Cafe', 'Club, Casual Dining',
                  'Quick Bites, Mess', 'Quick Bites, Meat Shop',
                  'Quick Bites, Kiosk', 'Lounge, Microbrewery',
                  'Food Court, Beverage Shop', 'Dessert Parlor, Food Court',
                  'Bar, Lounge'], dtype=object))
```

Check column *cuisines*

```
In [22]: zomato['cuisines'].isnull().value_counts(), zomato['cuisines'].unique()
```

```
Out[22]: (False    43533
          Name: cuisines, dtype: int64,
          array(['North Indian, Mughlai, Chinese', 'Chinese, North Indian, Thai',
                  'Cafe, Mexican, Italian', ..., 'Tibetan, Nepalese',
                  'North Indian, Street Food, Biryani',
                  'North Indian, Chinese, Arabian, Momos'], dtype=object))
```

Check column *cost* : column needs to be cleaned for comma and then data type to be converted into int

```
In [23]: zomato['cost'].isnull().value_counts(), zomato['cost'].unique()
```

```
Out[23]: (False      43533
          Name: cost, dtype: int64,
          array(['800', '300', '600', '700', '550', '500', '450', '650', '400',
                 '900', '200', '750', '150', '850', '100', '1,200', '350', '250',
                 '950', '1,000', '1,500', '1,300', '199', '80', '1,100', '160',
                 '1,600', '230', '130', '1,700', '1,400', '1,350', '2,200', '2,000',
                 '1,800', '1,900', '180', '330', '2,500', '2,100', '3,000', '2,800',
                 '3,400', '50', '40', '1,250', '3,500', '4,000', '2,400', '2,600',
                 '1,450', '70', '3,200', '560', '240', '360', '6,000', '1,050',
                 '2,300', '4,100', '120', '5,000', '3,700', '1,650', '2,700',
                 '4,500'], dtype=object))
```

```
In [24]: zomato['cost'] = zomato['cost'].apply(lambda x:x.replace(',','')).astype(int)
```

```
In [25]: zomato['cost'].isnull().value_counts(), zomato['cost'].unique()
```

```
Out[25]: (False      43533
          Name: cost, dtype: int64,
          array([ 800,  300,  600,  700,  550,  500,  450,  650,  400,  900,  200,
                 750,  150,  850,  100, 1200,  350,  250,  950, 1000, 1500, 1300,
                 199,   80, 1100, 160, 1600, 230, 130, 1700, 1400, 1350, 2200,
                 2000, 1800, 1900, 180, 330, 2500, 2100, 3000, 2800, 3400,   50,
                 40, 1250, 3500, 4000, 2400, 2600, 1450,    70, 3200,  560,  240,
                 360, 6000, 1050, 2300, 4100, 120, 5000, 3700, 1650, 2700, 4500]))
```

Check column type

```
In [26]: zomato['type'].unique()
```

```
Out[26]: array(['Buffet', 'Cafes', 'Delivery', 'Desserts', 'Dine-out',
                 'Drinks & nightlife', 'Pubs and bars'], dtype=object)
```

Check column city

```
In [27]: zomato['city'].unique()
```

```
Out[27]: array(['Banashankari', 'Bannerghatta Road', 'Basavanagudi', 'Bellandur',
                 'Brigade Road', 'Brookefield', 'BTM', 'Church Street',
                 'Electronic City', 'Frazer Town', 'HSR', 'Indiranagar',
                 'Jayanagar', 'JP Nagar', 'Kalyan Nagar', 'Kammanahalli',
                 'Koramangala 4th Block', 'Koramangala 5th Block',
                 'Koramangala 6th Block', 'Koramangala 7th Block', 'Lavelle Road',
                 'Malleshwaram', 'Marathahalli', 'MG Road', 'New BEL Road',
                 'Old Airport Road', 'Rajajinagar', 'Residency Road',
                 'Sarjapur Road', 'Whitefield'], dtype=object)
```

## Dropping duplicate values

```
In [28]: zomato.duplicated().value_counts()
```

```
Out[28]: False      43453
          True       80
          dtype: int64
```

We can see there are 80 duplicate values. We will delete the duplicates and keep the latest entries

```
In [29]: zomato.drop_duplicates(keep='last', inplace=True)
```

---

```
In [30]: zomato.duplicated().value_counts()
```

```
Out[30]: False    43453  
dtype: int64
```

```
In [31]: zomato
```

Out[31]:

	restaurant	online_order	book_table	rating	votes	location	rest_type	cuisines	cos
0	Jalsa	Yes	Yes	4.1	775	Banashankari	Casual Dining	North Indian, Mughlai, Chinese	80
1	Spice Elephant	Yes	No	4.1	787	Banashankari	Casual Dining	Chinese, North Indian, Thai	80
2	San Churro Cafe	Yes	No	3.8	918	Banashankari	Cafe, Casual Dining	Cafe, Mexican, Italian	80
3	Addhuri Udupi Bhojana	No	No	3.7	88	Banashankari	Quick Bites	South Indian, North Indian	30
4	Grand Village	No	No	3.8	166	Basavanagudi	Casual Dining	North Indian, Rajasthani	60
...	...	...	...	...	...	...	...	...	...
51709	The Farm House Bar n Grill	No	No	3.7	34	Whitefield	Casual Dining, Bar	North Indian, Continental	80
51711	Bhagini	No	No	2.5	81	Whitefield	Casual Dining, Bar	Andhra, South Indian, Chinese, North Indian	80
51712	Best Brews - Four Points by Sheraton Bengaluru...	No	No	3.6	27	Whitefield	Bar	Continental	150
51715	Chime - Sheraton Grand Bengaluru Whitefield Ho...	No	Yes	4.3	236	ITPL Main Road, Whitefield	Bar	Finger Food	250
51716	The Nest - The Den Bengaluru	No	No	3.4	13	ITPL Main Road, Whitefield	Bar, Casual Dining	Finger Food, North Indian, Continental	150

43453 rows × 11 columns



In [32]: zomato.info()

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 43453 entries, 0 to 51716
Data columns (total 11 columns):
 #   Column      Non-Null Count  Dtype  
--- 
 0   restaurant    43453 non-null   object  
 1   online_order   43453 non-null   object  
 2   book_table     43453 non-null   object  
 3   rating         43453 non-null   float64 
 4   votes          43453 non-null   int64   
 5   location        43453 non-null   object  
 6   rest_type       43453 non-null   object  
 7   cuisines        43453 non-null   object  
 8   cost            43453 non-null   int32  
 9   type            43453 non-null   object  
 10  city            43453 non-null   object  
dtypes: float64(1), int32(1), int64(1), object(8)
memory usage: 3.8+ MB
```

```
In [33]: zomato.to_csv(r'C:\Users\patil\Desktop\Project 1\zomato_cleaned.csv')
```

```
In [34]: zomato
```

Out[34]:

	restaurant	online_order	book_table	rating	votes	location	rest_type	cuisines	cos
0	Jalsa	Yes	Yes	4.1	775	Banashankari	Casual Dining	North Indian, Mughlai, Chinese	80
1	Spice Elephant	Yes	No	4.1	787	Banashankari	Casual Dining	Chinese, North Indian, Thai	80
2	San Churro Cafe	Yes	No	3.8	918	Banashankari	Cafe, Casual Dining	Cafe, Mexican, Italian	80
3	Addhuri Udupi Bhojana	No	No	3.7	88	Banashankari	Quick Bites	South Indian, North Indian	30
4	Grand Village	No	No	3.8	166	Basavanagudi	Casual Dining	North Indian, Rajasthani	60
...	...	...	...	...	...	...	...	...	...
51709	The Farm House Bar n Grill	No	No	3.7	34	Whitefield	Casual Dining, Bar	North Indian, Continental	80
51711	Bhagini	No	No	2.5	81	Whitefield	Casual Dining, Bar	Andhra, South Indian, Chinese, North Indian	80
51712	Best Brews - Four Points by Sheraton Bengaluru...	No	No	3.6	27	Whitefield	Bar	Continental	150
51715	Chime - Sheraton Grand Bengaluru Whitefield Ho...	No	Yes	4.3	236	ITPL Main Road, Whitefield	Bar	Finger Food	250
51716	The Nest - The Den Bengaluru	No	No	3.4	13	ITPL Main Road, Whitefield	Bar, Casual Dining	Finger Food, North Indian, Continental	150

43453 rows × 11 columns

## 2) Data Visualisation

- Table showing number of restaurants who provide service like online order and table booking

```
In [35]: col1 = ['city','online_order']
col2 = ['city','book_table']

tab1 = zomato[col1]
tab2 = zomato[col2]

tab_1 = tab1.groupby('city')['online_order'].value_counts(normalize=True).to_frame(name='online_order_acceptance')
tab_1['online_order_acceptance'] = round(tab_1['online_order_acceptance'],4)*100
tab_1 = tab_1.filter(like='Yes',axis=0)
tab_1.reset_index(inplace=True)
tab_1 = tab_1.drop(['online_order'],axis=1)

tab_2 = tab2.groupby('city')['book_table'].value_counts(normalize=True).to_frame(name='book_table_acceptance')
tab_2['book_table_acceptance'] = round(tab_2['book_table_acceptance'],4)*100
tab_2 = tab_2.filter(like='Yes',axis=0)
tab_2.reset_index(inplace=True)
tab_2 = tab_2.drop(['book_table'],axis=1)
```

```
In [36]: tab_1.set_index('city')
```

Out[36]:

online\_order\_acceptance

city	
BTM	70.87
Banashankari	68.80
Bannerghatta Road	69.92
Basavanagudi	66.58
Bellandur	72.71
Brigade Road	54.49
Brookefield	70.24
Church Street	51.38
Electronic City	55.86
Frazer Town	66.16
HSR	75.35
Indiranagar	60.85
JP Nagar	70.62
Jayanagar	73.11
Kalyan Nagar	67.88
Kammanahalli	67.01
Koramangala 4th Block	70.07
Koramangala 5th Block	68.45
Koramangala 6th Block	68.12
Koramangala 7th Block	66.98
Lavelle Road	47.58
MG Road	52.71
Malleshwaram	64.68
Marathahalli	64.76
New BEL Road	64.38
Old Airport Road	62.61
Rajajinagar	57.21
Residency Road	48.22
Sarjapur Road	68.88
Whitefield	61.89

In [37]:

```
tab_2.set_index('city')
```

Out[37]:

**book\_table\_acceptance**

city	
<b>BTM</b>	13.77
<b>Banashankari</b>	5.48
<b>Bannerghatta Road</b>	8.84
<b>Basavanagudi</b>	10.79
<b>Bellandur</b>	12.96
<b>Brigade Road</b>	22.72
<b>Brookefield</b>	9.92
<b>Church Street</b>	22.99
<b>Electronic City</b>	10.13
<b>Frazer Town</b>	12.99
<b>HSR</b>	12.40
<b>Indiranagar</b>	20.18
<b>JP Nagar</b>	10.78
<b>Jayanagar</b>	11.99
<b>Kalyan Nagar</b>	11.24
<b>Kammanahalli</b>	10.15
<b>Koramangala 4th Block</b>	15.52
<b>Koramangala 5th Block</b>	15.16
<b>Koramangala 6th Block</b>	14.61
<b>Koramangala 7th Block</b>	15.32
<b>Lavelle Road</b>	21.95
<b>MG Road</b>	22.90
<b>Malleshwaram</b>	12.73
<b>Marathahalli</b>	11.30
<b>New BEL Road</b>	8.56
<b>Old Airport Road</b>	18.61
<b>Rajajinagar</b>	10.06
<b>Residency Road</b>	22.37
<b>Sarjapur Road</b>	12.14
<b>Whitefield</b>	15.21

In [38]:

```
tab_comb = tab_1.merge(tab_2, how='outer', on='city')
```

In [39]: tab\_comb

Out[39]:

	city	online_order_acceptance	book_table_acceptance
0	BTM	70.87	13.77
1	Banashankari	68.80	5.48
2	Bannerghatta Road	69.92	8.84
3	Basavanagudi	66.58	10.79
4	Bellandur	72.71	12.96
5	Brigade Road	54.49	22.72
6	Brookefield	70.24	9.92
7	Church Street	51.38	22.99
8	Electronic City	55.86	10.13
9	Frazer Town	66.16	12.99
10	HSR	75.35	12.40
11	Indiranagar	60.85	20.18
12	JP Nagar	70.62	10.78
13	Jayanagar	73.11	11.99
14	Kalyan Nagar	67.88	11.24
15	Kammanahalli	67.01	10.15
16	Koramangala 4th Block	70.07	15.52
17	Koramangala 5th Block	68.45	15.16
18	Koramangala 6th Block	68.12	14.61
19	Koramangala 7th Block	66.98	15.32
20	Lavelle Road	47.58	21.95
21	MG Road	52.71	22.90
22	Malleshwaram	64.68	12.73
23	Marathahalli	64.76	11.30
24	New BEL Road	64.38	8.56
25	Old Airport Road	62.61	18.61
26	Rajajinagar	57.21	10.06
27	Residency Road	48.22	22.37
28	Sarjapur Road	68.88	12.14
29	Whitefield	61.89	15.21

In [40]:

```
x = tab_comb['city']
y = tab_comb['online_order_acceptance']
z = tab_comb['book_table_acceptance']
```

```

length = np.arange(len(x))
width = 0.4

fig,ax = plt.subplots()

online = ax.bar(length-width/2,y,width,label='Online Orders Acceptable',linewidth=20)
table = ax.bar(length+width/2,z,width,label='Table Booking Possible',linewidth=20)

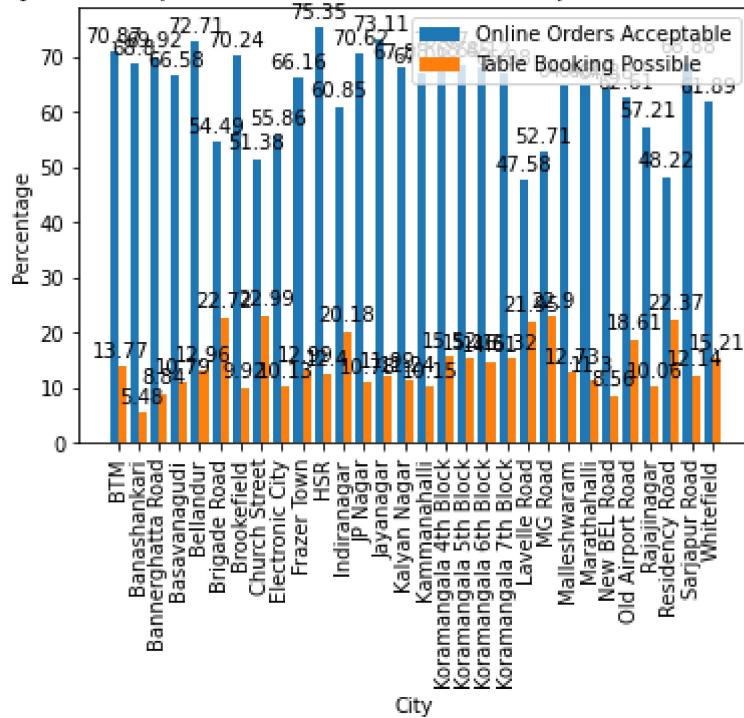
ax.set_ylabel('Percentage')
ax.set_xlabel('City')
ax.set_title('City wise acceptance of online orders and acceptance of table booking')
ax.set_xticks(length,x)
plt.xticks(rotation=90)
ax.legend()

ax.bar_label(online,padding=3)
ax.bar_label(table,padding=3)

fig.tight_layout
plt.show()

```

City wise acceptance of online orders and acceptance of table booking



- Best location based on rating and votes

```

In [41]: columns2 = ['location', 'votes', 'rating']
bestloc = zomato[columns2]
bestloc = bestloc.groupby('location').agg({'votes':'sum', 'rating':'mean'})
bestloc.reset_index(inplace=True)

```

```

In [42]: x = bestloc['location']
y = bestloc['votes']
z = bestloc['rating']

plt.rcParams["figure.figsize"] = [40,20]

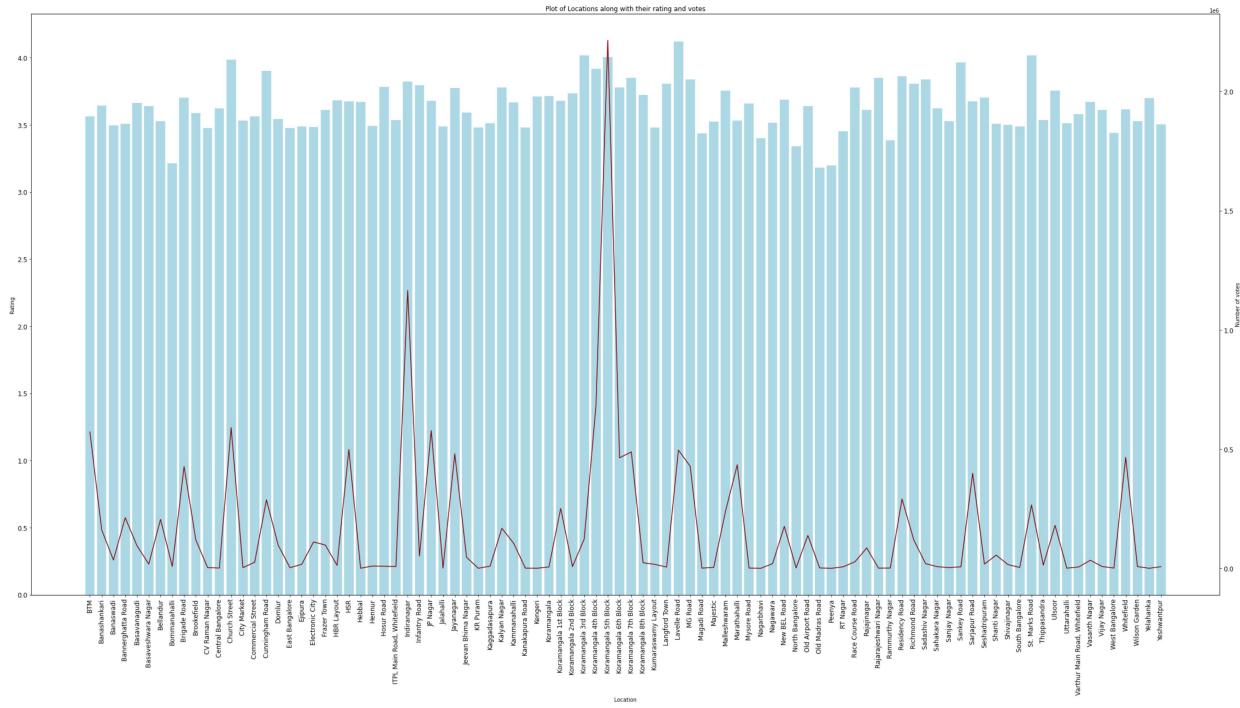
```

```

fig,ax1 = plt.subplots()
ax1.bar(x,z,color="#ADD8E6")
ax1.set_xlabel('Location',loc='center')
ax1.set_ylabel('Rating',loc='center')
ax1.set_title('Plot of Locations along with their rating and votes')
ax1.tick_params(labelsize=12)
plt.xticks(rotation=90)
ax2=ax1.twinx()
ax2.plot(x,y,color="#800000")
ax2.set_ylabel('Number of votes',loc='center')
ax2.tick_params(labelsize=12)

plt.show()

```



- Type of restaurants and their counts

```

In [43]: column3 = ['type']
typeofres = zomato[column3]
typeofres = typeofres.groupby('type').value_counts().to_frame(name='total_restaurants')
typeofres.reset_index(inplace=True)

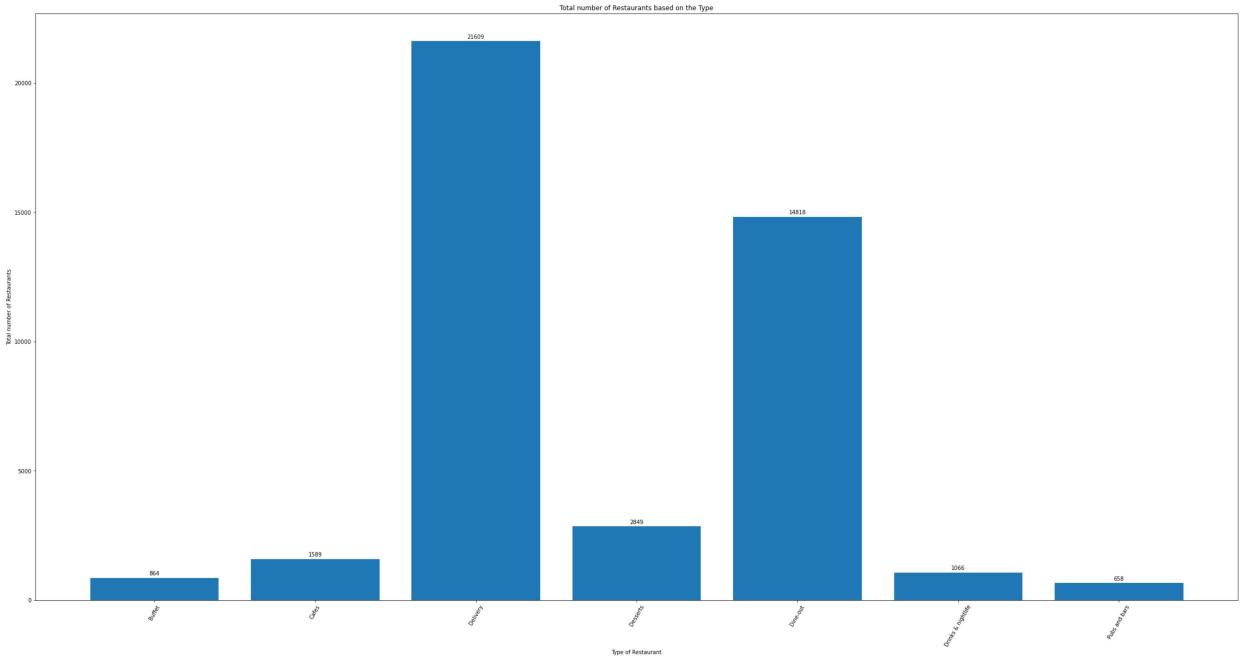
```

```

In [44]: x = typeofres['type']
y = typeofres['total_restaurants']

fig,ax = plt.subplots()
plt.rcParams["figure.figsize"]=[9,6]
aa = ax.bar(x,y)
ax.set_xlabel('Type of Restaurant')
ax.set_ylabel('Total number of Restaurants')
ax.set_title('Total number of Restaurants based on the Type')
ax.bar_label(aa,padding=3)
plt.xticks(rotation=60)
plt.show()

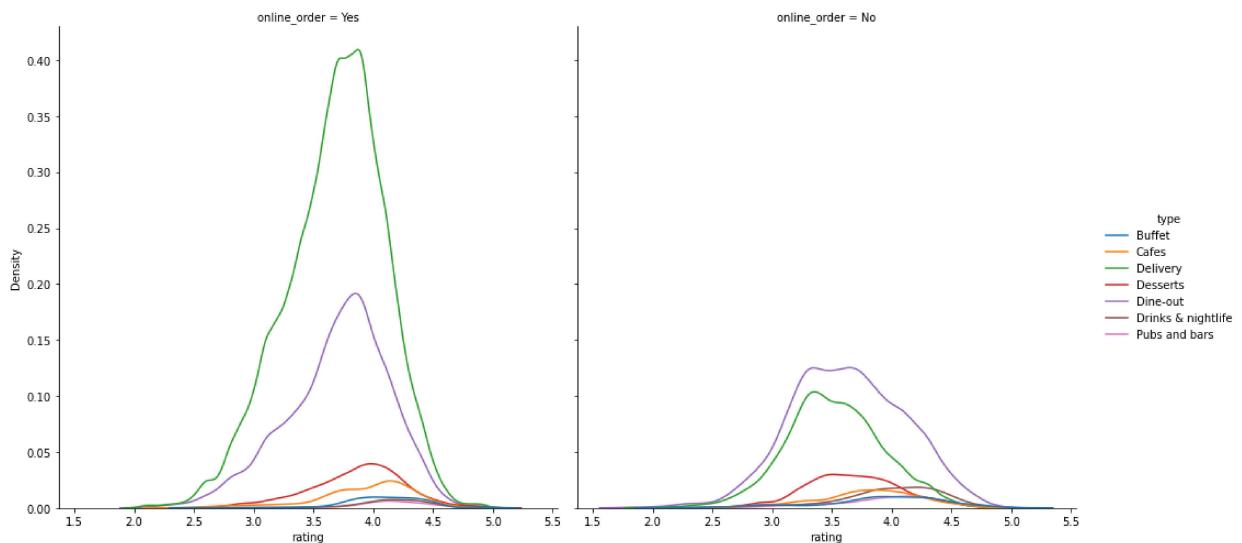
```



- Gaussian / Normal Distribution of Rating

```
In [45]: sns.displot(data=zomato,x='rating',kind='kde',hue='type',col='online_order',height=7)
```

```
Out[45]: <seaborn.axisgrid.FacetGrid at 0x1642556ef10>
```



- Restaurant chains and their total restaurants

```
In [46]: column3 = ['restaurant']
chains = zomato[column3]
chains = chains.groupby('restaurant').value_counts().to_frame(name='number_of_outlets')
chains.reset_index(inplace=True)
```

```
In [47]: chain = chains.sort_values(by='number_of_outlets',ascending=False).head(20)
```

```
In [48]: x = chain['restaurant']
y = chain['number_of_outlets']
```

```

plt.figure(figsize=(12,8))
aaaa = sns.barplot(y=x,x=y,orient='h')
plt.title('Famous Restaurant Chain having maximum number outlets')
plt.show()

```

