In [43]:
```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import matplotlib.colors as mcolors
```

Zomato is an Indian multinational restaurant aggregator and food delivery company, founded by Deepinder Goyal and Pankaj Chaddah in 2008.

Zomato is one of the most comprehensive and user-friendly apps for finding nearby restaurants and cafés to dine in or to order food online. It also gives menus, reviews, and ratings to acquire factual information on eateries.

Zomato Limited is an online restaurant guide and food ordering platform.

It offered comprehensive details on more than 1.4 million establishments in 23 countries. There were restaurant names, menu items, pricing, reviews, and other information. It has evolved into an internet platform for meal delivery over the years

Zomato's tagline – “Never have a bad meal”. It serves as a comprehensive encyclopedia of restaurants, replete with ratings, average pricing, menus, and reviews. It is currently the only app of its kind in India.

In [44]:
```
df=pd.read_csv(r'C:\Users\LENOVO\OneDrive\Desktop\New folder (2)\zomato.csv',encoding='latin')
df
```

Out[44]:

| | Restaurant ID | Restaurant Name | Country Code | City | Address | Locality | Locality Verbose | Longitude | Latitude | Cuisines | ... | Currency | Has Table booking | Has Online delivery | Is delivering now | Sv c n |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 6317637 | Le Petit Souffle | 162 | Makati City | Third Floor, Century City Mall, Kalayaan Avenu... | Century City Mall, Poblacion, Makati City | Century City Mall, Poblacion, Makati City, Mak... | 121.027535 | 14.565443 | French, Japanese, Desserts | ... | Botswana Pula(P) | Yes | No | No | |
| **1** | 6304287 | Izakaya Kikufuji | 162 | Makati City | Little Tokyo, 2277 Chino Roces Avenue, Legaspi... | Little Tokyo, Legaspi Village, Makati City | Little Tokyo, Legaspi Village, Makati City, Ma... | 121.014101 | 14.553708 | Japanese | ... | Botswana Pula(P) | Yes | No | No | |
| **2** | 6300002 | Heat - Edsa Shangri-La | 162 | Mandaluyong City | Edsa Shangri-La, 1 Garden Way, Ortigas, Mandal... | Edsa Shangri-La, Ortigas, Mandaluyong City | Edsa Shangri-La, Ortigas, Mandaluyong City, Ma... | 121.056831 | 14.581404 | Seafood, Asian, Filipino, Indian | ... | Botswana Pula(P) | Yes | No | No | |
| **3** | 6318506 | Ooma | 162 | Mandaluyong City | Third Floor, Mega Fashion Hall, SM Megamall, O... | SM Megamall, Ortigas, Mandaluyong City | SM Megamall, Ortigas, Mandaluyong City, Mandal... | 121.056475 | 14.585318 | Japanese, Sushi | ... | Botswana Pula(P) | No | No | No | |
| **4** | 6314302 | Sambo Kojin | 162 | Mandaluyong City | Third Floor, Mega Atrium, SM Megamall, Ortigas... | SM Megamall, Ortigas, Mandaluyong City | SM Megamall, Ortigas, Mandaluyong City, Mandal... | 121.057508 | 14.584450 | Japanese, Korean | ... | Botswana Pula(P) | Yes | No | No | |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | |
| **9546** | 5915730 | NamlÛ± Gurme | 208 | ÛÁstanbul | Kemanke□ô Karamustafa Pa□ôa Mahallesi, RÛ±htÛ±... | Karakí_y | Karakí_y, ÛÁstanbul | 28.977392 | 41.022793 | Turkish | ... | Turkish Lira(TL) | No | No | No | |
| **9547** | 5908749 | Ceviz AÛôacÛ± | 208 | ÛÁstanbul | Ko□ôuyolu Mahallesi, Muhittin íìstí_ndaÛô Cadd... | Ko□ôuyolu | Ko□ôuyolu, ÛÁstanbul | 29.041297 | 41.009847 | World Cuisine, Patisserie, Cafe | ... | Turkish Lira(TL) | No | No | No | |
| **9548** | 5915807 | Huqqa | 208 | ÛÁstanbul | Kuruí_e□ôme Mahallesi, Muallim Naci Caddesi, N... | Kuruí_e□ôme | Kuruí_e□ôme, ÛÁstanbul | 29.034640 | 41.055817 | Italian, World Cuisine | ... | Turkish Lira(TL) | No | No | No | |
| **9549** | 5916112 | A□ô□ôk Kahve | 208 | ÛÁstanbul | Kuruí_e□ôme Mahallesi, Muallim Naci Caddesi, N... | Kuruí_e□ôme | Kuruí_e□ôme, ÛÁstanbul | 29.036019 | 41.057979 | Restaurant Cafe | ... | Turkish Lira(TL) | No | No | No | |
| **9550** | 5927402 | Walter's Coffee Roastery | 208 | ÛÁstanbul | CafeaÛôa Mahallesi, BademaltÛ± Sokak, No 21/B,... | Moda | Moda, ÛÁstanbul | 29.026016 | 40.984776 | Cafe | ... | Turkish Lira(TL) | No | No | No | |

9551 rows × 21 columns

In [3]:
```
df.head()
```

Out[3]:

| | Restaurant ID | Restaurant Name | Country Code | City | Address | Locality | Locality Verbose | Longitude | Latitude | Cuisines | ... | Currency | Has Table booking | Has Online delivery | Is delivering now | Switch to order menu | P ra |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 6317637 | Le Petit Souffle | 162 | Makati City | Third Floor, Century City Mall, Kalayaan Avenu... | Century City Mall, Poblacion, Makati City | Century City Mall, Poblacion, Makati City, Mak... | 121.027535 | 14.565443 | French, Japanese, Desserts | ... | Botswana Pula(P) | Yes | No | No | No | |
| **1** | 6304287 | Izakaya Kikufuji | 162 | Makati City | Little Tokyo, 2277 Chino Roces Avenue, Legaspi... | Little Tokyo, Legaspi Village, Makati City | Little Tokyo, Legaspi Village, Makati City, Ma... | 121.014101 | 14.553708 | Japanese | ... | Botswana Pula(P) | Yes | No | No | No | |
| **2** | 6300002 | Heat - Edsa Shangri-La | 162 | Mandaluyong City | Edsa Shangri-La, 1 Garden Way, Ortigas, Mandal... | Edsa Shangri-La, Ortigas, Mandaluyong City | Edsa Shangri-La, Ortigas, Mandaluyong City, Ma... | 121.056831 | 14.581404 | Seafood, Asian, Filipino, Indian | ... | Botswana Pula(P) | Yes | No | No | No | |
| **3** | 6318506 | Ooma | 162 | Mandaluyong City | Third Floor, Mega Fashion Hall, SM Megamall, O... | SM Megamall, Ortigas, Mandaluyong City | SM Megamall, Ortigas, Mandaluyong City, Mandal... | 121.056475 | 14.585318 | Japanese, Sushi | ... | Botswana Pula(P) | No | No | No | No | |
| **4** | 6314302 | Sambo Kojin | 162 | Mandaluyong City | Third Floor, Mega Atrium, SM Megamall, Ortigas... | SM Megamall, Ortigas, Mandaluyong City | SM Megamall, Ortigas, Mandaluyong City, Mandal... | 121.057508 | 14.584450 | Japanese, Korean | ... | Botswana Pula(P) | Yes | No | No | No | |

5 rows × 21 columns

In [4]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9551 entries, 0 to 9550
Data columns (total 21 columns):
 #   Column                Non-Null Count  Dtype  
---  ------                --------------  -----  
 0   Restaurant ID         9551 non-null   int64  
 1   Restaurant Name       9551 non-null   object 
 2   Country Code          9551 non-null   int64  
 3   City                  9551 non-null   object 
 4   Address               9551 non-null   object 
 5   Locality              9551 non-null   object 
 6   Locality Verbose      9551 non-null   object 
 7   Longitude             9551 non-null   float64
 8   Latitude              9551 non-null   float64
 9   Cuisines              9542 non-null   object 
 10  Average Cost for two  9551 non-null   int64  
 11  Currency              9551 non-null   object 
 12  Has Table booking     9551 non-null   object 
 13  Has Online delivery   9551 non-null   object 
 14  Is delivering now     9551 non-null   object 
 15  Switch to order menu  9551 non-null   object 
 16  Price range           9551 non-null   int64  
 17  Aggregate rating      9551 non-null   float64
 18  Rating color          9551 non-null   object 
 19  Rating text           9551 non-null   object 
 20  Votes                 9551 non-null   int64  
dtypes: float64(3), int64(5), object(13)
memory usage: 1.5+ MB
```

In [25]:
```
df.describe()
```

Out[25]:

| | Restaurant ID | Country Code | Longitude | Latitude | Average Cost for two | Price range | Aggregate rating | Votes |
|---|---|---|---|---|---|---|---|---|
| **count** | 9.551000e+03 | 9551.000000 | 9551.000000 | 9551.000000 | 9551.000000 | 9551.000000 | 9551.000000 | 9551.000000 |
| **mean** | 9.051128e+06 | 18.365616 | 64.126574 | 25.854381 | 1199.210763 | 1.804837 | 2.666370 | 156.909748 |
| **std** | 8.791521e+06 | 56.750546 | 41.467058 | 11.007935 | 16121.183073 | 0.905609 | 1.516378 | 430.169145 |
| **min** | 5.300000e+01 | 1.000000 | -157.948486 | -41.330428 | 0.000000 | 1.000000 | 0.000000 | 0.000000 |
| **25%** | 3.019625e+05 | 1.000000 | 77.081343 | 28.478713 | 250.000000 | 1.000000 | 2.500000 | 5.000000 |
| **50%** | 6.004089e+06 | 1.000000 | 77.191964 | 28.570469 | 400.000000 | 2.000000 | 3.200000 | 31.000000 |
| **75%** | 1.835229e+07 | 1.000000 | 77.282006 | 28.642758 | 700.000000 | 2.000000 | 3.700000 | 131.000000 |
| **max** | 1.850065e+07 | 216.000000 | 174.832089 | 55.976980 | 800000.000000 | 4.000000 | 4.900000 | 10934.000000 |

In [26]:
```
df['Restaurant Name'].value_counts()
```

Out[26]:
```
Cafe Coffee Day            83
Domino's Pizza             79
Subway                     63
Green Chick Chop           51
McDonald's                 48
                           ..
Odeon Social                1
Johnny Rockets              1
House of Commons            1
HotMess                     1
Walter's Coffee Roastery    1
Name: Restaurant Name, Length: 7446, dtype: int64
```

In [27]:
```
df['City'].value_counts()
```

Out[27]:
```
New Delhi           5473
Gurgaon             1118
Noida               1080
Faridabad            251
Ghaziabad             25
                    ...
Panchkula              1
Mc Millan              1
Mayfield               1
Macedon                1
Vineland Station       1
Name: City, Length: 141, dtype: int64
```

In [28]:
```
df['Address'].value_counts()
```

Out[28]:
```
Dilli Haat, INA, New Delhi                                                        11
Sector 41, Noida                                                                  11
Greater Kailash (GK) 1, New Delhi                                                 10
The Imperial, Janpath, New Delhi                                                   9
Cyber Hub, DLF Cyber City, Gurgaon                                                 8
                                                                                  ..
23-24, Defence Colony Market, Defence Colony, New Delhi                            1
28, Main Market, Defence Colony, New Delhi                                         1
Daryaganj, New Delhi                                                               1
Ground Floor, E-23, Netaji Subhash Marg, Opposite Golcha Cinema, Daryaganj, New Delhi      1
CafeaÛôa Mahallesi, BademaltÛ± Sokak, No 21/B, KadÛ±kí_y, ÛÁstanbul              1
Name: Address, Length: 8918, dtype: int64
```

In [29]:
```
df['Locality'].value_counts()
```

Out[29]:
```
Connaught Place                             122
Rajouri Garden                               99
Shahdara                                     87
Defence Colony                               86
Malviya Nagar                                85
                                           ...
Lemon Tree Premier, Sector 29                 1
Omaxe Celebration Mall, Sohna Road, Gurgaon   1
Park Inn, Sector 15, Gurgaon                  1
Plaza Mall, MG Road                           1
Moda                                          1
Name: Locality, Length: 1208, dtype: int64
```

In [30]:
```
df['Locality Verbose'].value_counts()
```

Out[30]:
```
Connaught Place, New Delhi                 122
Rajouri Garden, New Delhi                   99
Shahdara, New Delhi                         87
Defence Colony, New Delhi                   86
Pitampura, New Delhi                        85
                                          ...
Ramada Gurgaon Central, Sector 44, Gurgaon   1
Sector 5, Gurgaon                            1
Sector 53, Gurgaon                           1
The Claremont, MG Road, Gurgaon              1
Moda, ÛÁstanbul                              1
Name: Locality Verbose, Length: 1265, dtype: int64
```

In [31]:
```
df['Cuisines'].value_counts()
```

Out[31]:
```
North Indian                                        936
North Indian, Chinese                               511
Chinese                                             354
Fast Food                                           354
North Indian, Mughlai                               334
                                                   ...
Bengali, Fast Food                                    1
North Indian, Rajasthani, Asian                       1
Chinese, Thai, Malaysian, Indonesian                  1
Bakery, Desserts, North Indian, Bengali, South Indian      1
Italian, World Cuisine                                1
Name: Cuisines, Length: 1825, dtype: int64
```

In [32]:
```
df['Currency'].value_counts()
```

Out[32]:
```
Indian Rupees(Rs.)        8652
Dollar($)                  482
Pounds(￡)                   80
Brazilian Real(R$)          60
Emirati Diram(AED)          60
Rand(R)                     60
NewZealand($)               40
Turkish Lira(TL)            34
Botswana Pula(P)            22
Indonesian Rupiah(IDR)      21
Qatari Rial(QR)             20
Sri Lankan Rupee(LKR)       20
Name: Currency, dtype: int64
```

In [33]:
```
df['Has Table booking'].value_counts()
```

Out[33]:
```
No     8393
Yes    1158
Name: Has Table booking, dtype: int64
```

In [34]:
```
df['Has Online delivery'].value_counts()
```

Out[34]:
```
No     7100
Yes    2451
Name: Has Online delivery, dtype: int64
```

In [35]:
```
df['Is delivering now'].value_counts()
```

Out[35]:
```
No     9517
Yes      34
Name: Is delivering now, dtype: int64
```

In [36]:
```
df['Switch to order menu'].value_counts()
```

Out[36]:
```
No    9551
Name: Switch to order menu, dtype: int64
```

In [37]:
```
df['Price range'].value_counts()
```

Out[37]:
```
1    4444
2    3113
3    1408
4     586
Name: Price range, dtype: int64
```

In [38]:
```
df['Rating color'].value_counts()
```

Out[38]:
```
Orange        3737
White         2148
Yellow        2100
Green         1079
Dark Green     301
Red            186
Name: Rating color, dtype: int64
```

In [39]:
```
df['Rating text'].value_counts()
```

Out[39]:
```
Average      3737
Not rated    2148
Good         2100
Very Good    1079
Excellent     301
Poor          186
Name: Rating text, dtype: int64
```

In [40]:
```
df['Votes'].value_counts()
```

Out[40]:
```
0      1094
1       483
2       327
3       244
4       207
       ...
615       1
538       1
823       1
650       1
901       1
Name: Votes, Length: 1012, dtype: int64
```

In [ ]:

## Checking if dataset contains any null

In [41]:
```
nan_values = df.isna()
nan_columns = nan_values.any()

columns_with_nan = df.columns[nan_columns].tolist()
print(columns_with_nan)
```

```
['Cuisines']
```

df.isna() nan_values.any() print(columns_with_nan)

In [42]:
```
df.isna()
nan_values.any()
```

Out[42]:
```
Restaurant ID           False
Restaurant Name         False
Country Code            False
City                    False
Address                 False
Locality                False
Locality Verbose        False
Longitude               False
Latitude                False
Cuisines                 True
Average Cost for two    False
Currency                False
Has Table booking       False
Has Online delivery     False
Is delivering now       False
Switch to order menu    False
Price range             False
Aggregate rating        False
Rating color            False
Rating text             False
Votes                   False
dtype: bool
```

```
Cuisines seems to contain null values. Hence any further analysis involving Cuisines the NaN values has to be considered.
There is an other file which is also available along with this dataset
```

In [45]:
```
df1 = pd.read_excel(r'C:\Users\LENOVO\OneDrive\Desktop\New folder (2)\Country-Code.xlsx')
df1
```

Out[45]:

| | Country Code | Country |
|---|---|---|
| 0 | 1 | India |
| 1 | 14 | Australia |
| 2 | 30 | Brazil |
| 3 | 37 | Canada |
| 4 | 94 | Indonesia |
| 5 | 148 | New Zealand |
| 6 | 162 | Phillipines |
| 7 | 166 | Qatar |
| 8 | 184 | Singapore |
| 9 | 189 | South Africa |
| 10 | 191 | Sri Lanka |
| 11 | 208 | Turkey |
| 12 | 214 | UAE |
| 13 | 215 | United Kingdom |
| 14 | 216 | United States |

In [8]:
```
df1.head()
```

Out[8]:

| | Country Code | Country |
|---|---|---|
| 0 | 1 | India |
| 1 | 14 | Australia |
| 2 | 30 | Brazil |
| 3 | 37 | Canada |
| 4 | 94 | Indonesia |

**Let merge the both dataset this will help us to understand the data set countrywise.**

In [46]:
```
df2=pd.merge(df,df1,on='Country Code',how='left')
df2.head()
```

Out[46]:

| | Restaurant ID | Restaurant Name | Country Code | City | Address | Locality | Locality Verbose | Longitude | Latitude | Cuisines | ... | Has Table booking | Has Online delivery | Is delivering now | Switch to order menu | Price range | Aggre r |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 6317637 | Le Petit Souffle | 162 | Makati City | Third Floor, Century City Mall, Kalayaan Avenu... | Century City Mall, Poblacion, Makati City | Century City Mall, Poblacion, Makati City, Mak... | 121.027535 | 14.565443 | French, Japanese, Desserts | ... | Yes | No | No | No | 3 | |
| 1 | 6304287 | Izakaya Kikufuji | 162 | Makati City | Little Tokyo, 2277 Chino Roces Avenue, Legaspi... | Little Tokyo, Legaspi Village, Makati City | Little Tokyo, Legaspi Village, Makati City, Ma... | 121.014101 | 14.553708 | Japanese | ... | Yes | No | No | No | 3 | |
| 2 | 6300002 | Heat - Edsa Shangri-La | 162 | Mandaluyong City | Edsa Shangri-La, 1 Garden Way, Ortigas, Mandal... | Edsa Shangri-La, Ortigas, Mandaluyong City | Edsa Shangri-La, Ortigas, Mandaluyong City, Ma... | 121.056831 | 14.581404 | Seafood, Asian, Filipino, Indian | ... | Yes | No | No | No | 4 | |
| 3 | 6318506 | Ooma | 162 | Mandaluyong City | Third Floor, Mega Fashion Hall, SM Megamall, O... | SM Megamall, Ortigas, Mandaluyong City | SM Megamall, Ortigas, Mandaluyong City, Mandal... | 121.056475 | 14.585318 | Japanese, Sushi | ... | No | No | No | No | 4 | |
| 4 | 6314302 | Sambo Kojin | 162 | Mandaluyong City | Third Floor, Mega Atrium, SM Megamall, Ortigas... | SM Megamall, Ortigas, Mandaluyong City | SM Megamall, Ortigas, Mandaluyong City, Mandal... | 121.057508 | 14.584450 | Japanese, Korean | ... | Yes | No | No | No | 4 | |

5 rows × 22 columns

In [10]:
```
df2['Country Code'].value_counts()
```

Out[10]:
```
1      8652
216     434
215      80
30       60
214      60
189      60
148      40
208      34
14       24
162      22
94       21
184      20
166      20
191      20
37        4
Name: Country Code, dtype: int64
```

In [11]:
```
df2.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 9551 entries, 0 to 9550
Data columns (total 22 columns):
 #   Column                Non-Null Count  Dtype  
---  ------                --------------  -----  
 0   Restaurant ID         9551 non-null   int64  
 1   Restaurant Name       9551 non-null   object 
 2   Country Code          9551 non-null   int64  
 3   City                  9551 non-null   object 
 4   Address               9551 non-null   object 
 5   Locality              9551 non-null   object 
 6   Locality Verbose      9551 non-null   object 
 7   Longitude             9551 non-null   float64
 8   Latitude              9551 non-null   float64
 9   Cuisines              9542 non-null   object 
 10  Average Cost for two  9551 non-null   int64  
 11  Currency              9551 non-null   object 
 12  Has Table booking     9551 non-null   object 
 13  Has Online delivery   9551 non-null   object 
 14  Is delivering now     9551 non-null   object 
 15  Switch to order menu  9551 non-null   object 
 16  Price range           9551 non-null   int64  
 17  Aggregate rating      9551 non-null   float64
 18  Rating color          9551 non-null   object 
 19  Rating text           9551 non-null   object 
 20  Votes                 9551 non-null   int64  
 21  Country               9551 non-null   object 
dtypes: float64(3), int64(5), object(14)
memory usage: 1.7+ MB
```

#### Brief Analysis and Data Visualization of Zomato Data

```
First up all, we have to get the whole restaurants data which is geographical spread,which help us to understand the rating,
Currency,Online Delivery,City Coverage ....and many more...
```

## List of countries the survey is spread across

In [48]:
```
print('List of counteris the survey is spread accross - ')
for x in pd.unique(df2.Country): print(x)
print('Total number to country', len(pd.unique(df2.Country)))
```

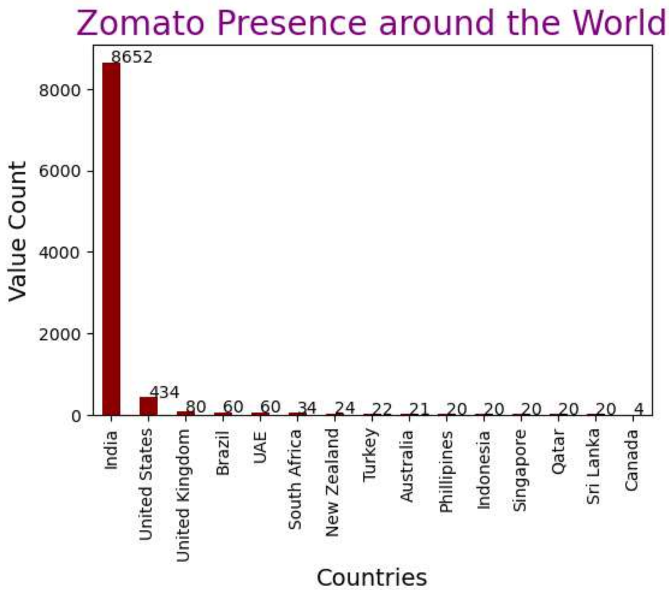
```
List of counteris the survey is spread accross -
Phillipines
Brazil
United States
Australia
Canada
Singapore
UAE
India
Indonesia
New Zealand
United Kingdom
Qatar
South Africa
Sri Lanka
Turkey
Total number to country 15
```

The survey seems to have spread across 15 countries all over world. This shows that Zomato is a multinational company having actives business in all those countries.

In [49]:
```
df2['Country'].value_counts()
```

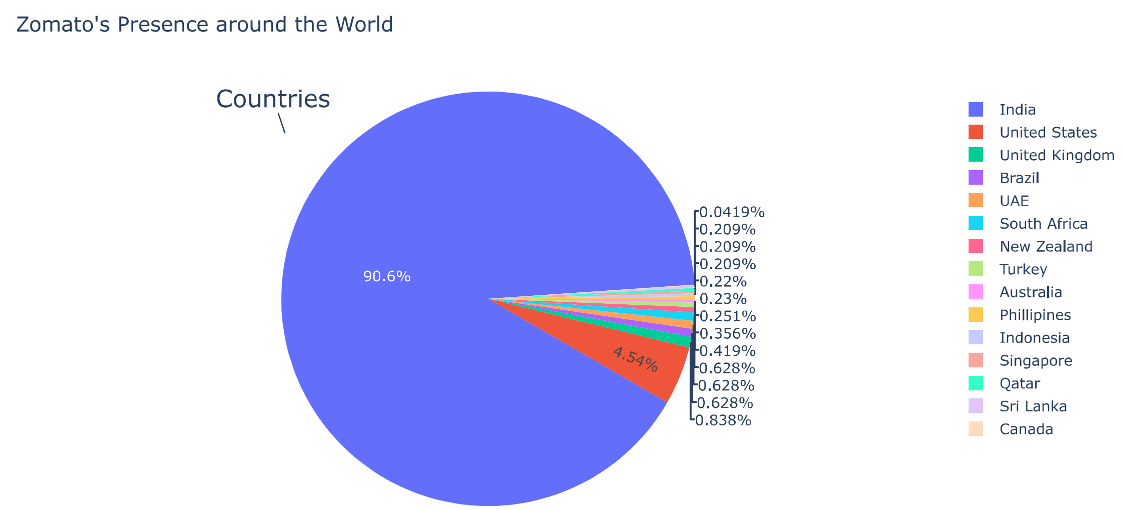
Out[49]:
```
India             8652
United States      434
United Kingdom      80
Brazil              60
UAE                 60
South Africa        60
New Zealand         40
Turkey              34
Australia           24
Phillipines         22
Indonesia           21
Singapore           20
Qatar               20
Sri Lanka           20
Canada               4
Name: Country, dtype: int64
```

In [47]:
```
plt.figure(figsize=(6,4))
df2['Country'].value_counts().plot(kind='bar',color='darkred')
plt.title('Zomato Presence around the World',color='purple',fontsize='20')
plt.xlabel('Countries',color='k',fontsize='14')
plt.ylabel('Value Count',color='k',fontsize='14')
plt.text(0,8653,8652,fontsize=10)
plt.text(1,435,434,fontsize=10)
plt.text(2,81,80,fontsize=10)
plt.text(3,61,60,fontsize=10)
plt.text(4,61,60,fontsize=10)
plt.text(5,39,34,fontsize=10)
plt.text(6,25,24,fontsize=10)
plt.text(7,23,22,fontsize=10)
plt.text(8,22,21,fontsize=10)
plt.text(9,21,20,fontsize=10)
plt.text(10,21,20,fontsize=10)
plt.text(11,23,20,fontsize=10)
plt.text(12,21,20,fontsize=10)
plt.text(13,21,20,fontsize=10)
plt.text(14,5,4,fontsize=10)
plt.show()
```



In [48]:
```
from plotly.offline import init_notebook_mode, plot, iplot
plt.figure(figsize=(6,4))
labels = list(df2.Country.value_counts().index)
values = list(df2.Country.value_counts().values)

fig = {"data":[{"labels" : labels,"values" : values,"hoverinfo" :'label+percent',"domain": {"x": [0, .9]},"type" : "pie",
            "rotation":120,},],"layout": {"title" : "Zomato's Presence around the World",
       "annotations": [{"font": {"size":20},"showarrow": True,"text": "Countries", "x":0.2,"y":0.9,},]}}
iplot(fig)
```




```
<Figure size 600x400 with 0 Axes>
```

## As Zomato is a startup from India hence it makes sense that it has maximum business spread across restaurants in India

### Understanding the Rating aggregate, color and text

In [6]:
```
df3 = df2.groupby(['Aggregate rating','Rating color', 'Rating text']).size().reset_index().rename(columns={0:'Rating Count'})
df3
```

Out[6]:

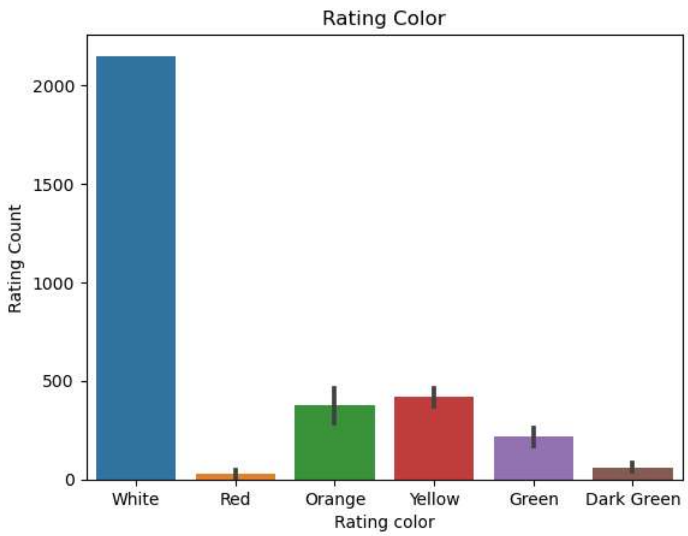
| | Aggregate rating | Rating color | Rating text | Rating Count |
|---|---|---|---|---|
| 0 | 0.0 | White | Not rated | 2148 |
| 1 | 1.8 | Red | Poor | 1 |
| 2 | 1.9 | Red | Poor | 2 |
| 3 | 2.0 | Red | Poor | 7 |
| 4 | 2.1 | Red | Poor | 15 |
| 5 | 2.2 | Red | Poor | 27 |
| 6 | 2.3 | Red | Poor | 47 |
| 7 | 2.4 | Red | Poor | 87 |
| 8 | 2.5 | Orange | Average | 110 |
| 9 | 2.6 | Orange | Average | 191 |
| 10 | 2.7 | Orange | Average | 250 |
| 11 | 2.8 | Orange | Average | 315 |
| 12 | 2.9 | Orange | Average | 381 |
| 13 | 3.0 | Orange | Average | 468 |
| 14 | 3.1 | Orange | Average | 519 |
| 15 | 3.2 | Orange | Average | 522 |
| 16 | 3.3 | Orange | Average | 483 |
| 17 | 3.4 | Orange | Average | 498 |
| 18 | 3.5 | Yellow | Good | 480 |
| 19 | 3.6 | Yellow | Good | 458 |
| 20 | 3.7 | Yellow | Good | 427 |
| 21 | 3.8 | Yellow | Good | 400 |
| 22 | 3.9 | Yellow | Good | 335 |
| 23 | 4.0 | Green | Very Good | 266 |
| 24 | 4.1 | Green | Very Good | 274 |
| 25 | 4.2 | Green | Very Good | 221 |
| 26 | 4.3 | Green | Very Good | 174 |
| 27 | 4.4 | Green | Very Good | 144 |
| 28 | 4.5 | Dark Green | Excellent | 95 |
| 29 | 4.6 | Dark Green | Excellent | 78 |
| 30 | 4.7 | Dark Green | Excellent | 42 |
| 31 | 4.8 | Dark Green | Excellent | 25 |
| 32 | 4.9 | Dark Green | Excellent | 61 |

```
The above information helps us to understand the realation between Aggregate rating,
color and text. We conclude the following color assigned to the ratings:
```

```
Rating 0 — White — Not rated
Rating 1.8 to 2.4 — Red — Poor
Rating 2.5 to 3.4 — Orange — Average
Rating 3.5 to 3.9 — Yellow — Good
Rating 4.0 to 4.4 — Green — Very Good
Rating 4.5 to 4.9 — Dark Green — Excellent
```

```
Let us try to understand the spread of rating across restaurants
```

In [8]:
```
sns.barplot(x=df3['Rating color'], y=df3['Rating Count'])
plt.title('Rating Color',color='k')
plt.show()
```




```
Interesting, Maximum restaurants seems to have gone No ratings.
Let us check if these restaurants belong to some specific country.
```

In [54]:
```
No_rating = df2[df2['Rating color']=='White'].groupby('Country').size().reset_index().rename(columns={0:'Rating Count'})
No_rating
```

Out[54]:

| | Country | Rating Count |
|---|---|---|
| 0 | Brazil | 5 |
| 1 | India | 2139 |
| 2 | United Kingdom | 1 |
| 3 | United States | 3 |

```
ndia seems to have maximum unrated restaurants. In India the culture of ordering online food is still gaining momentum hence most of the
restaurants are still unrated on Zomato as people might be preferring to visiting the restaurant for a meal.
```

### Country and Currency

In [55]:
```
d1=df2[['Country','Currency']].groupby(['Country','Currency']).size().reset_index(name='Count')
d1
```

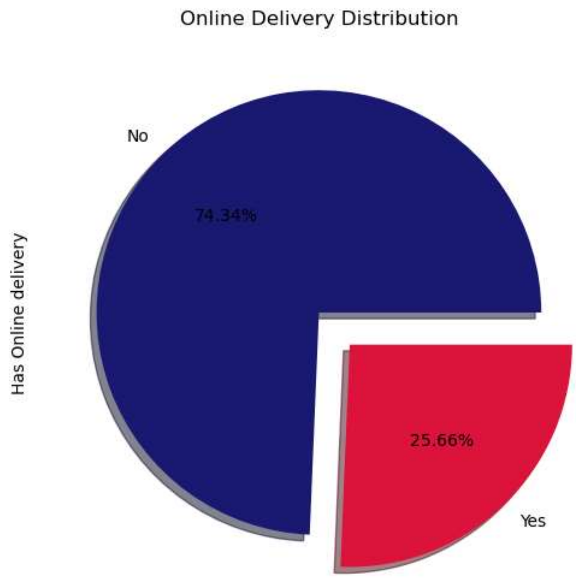
Out[55]:

| | Country | Currency | Count |
|---|---|---|---|
| 0 | Australia | Dollar($) | 24 |
| 1 | Brazil | Brazilian Real(R$) | 60 |
| 2 | Canada | Dollar($) | 4 |
| 3 | India | Indian Rupees(Rs.) | 8652 |
| 4 | Indonesia | Indonesian Rupiah(IDR) | 21 |
| 5 | New Zealand | NewZealand($) | 40 |
| 6 | Phillipines | Botswana Pula(P) | 22 |
| 7 | Qatar | Qatari Rial(QR) | 20 |
| 8 | Singapore | Dollar($) | 20 |
| 9 | South Africa | Rand(R) | 60 |
| 10 | Sri Lanka | Sri Lankan Rupee(LKR) | 20 |
| 11 | Turkey | Turkish Lira(TL) | 34 |
| 12 | UAE | Emirati Diram(AED) | 60 |
| 13 | United Kingdom | Pounds(□£) | 80 |
| 14 | United States | Dollar($) | 434 |

```
Above table display country and the currency they accept.
Large no. of count are from India as we discuss before.
Interestingly four countries seems to be accepting currency in dollars.
```

### Online delivery distribution

In [6]:
```
plt.figure(figsize=(8,6))
plt.title('Online Delivery Distribution')
df2['Has Online delivery'].value_counts().plot(kind='pie',autopct='%0.2f%%',colors=['midnightblue','crimson'],shadow=True,
                                             explode=[0,0.2])
plt.show()
```



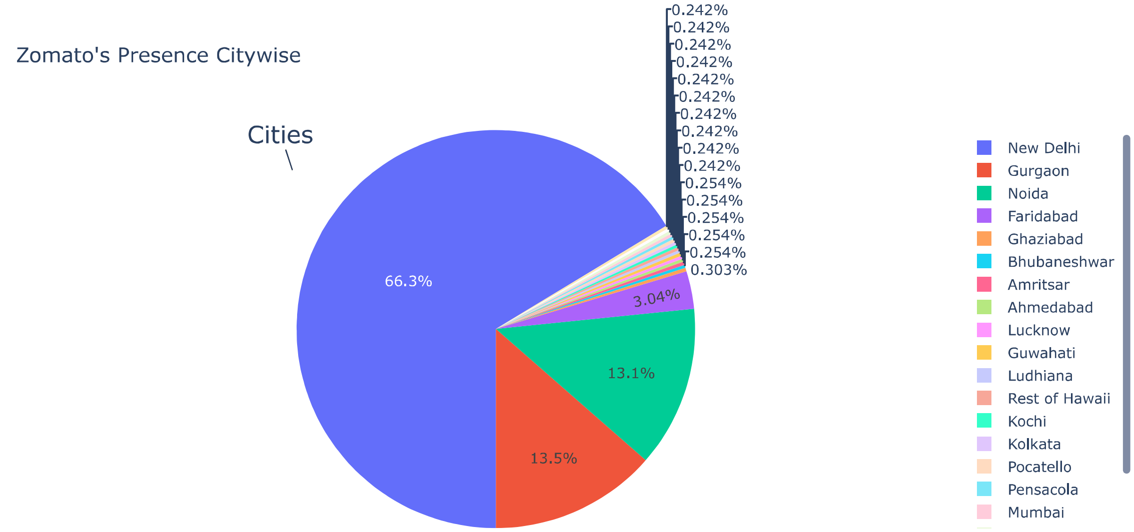

```
Only 25% of restaurants accepts online delivery. This data might be biased as we have maximum restaurants listed here are from India. Maybe
analysis over city wise would be more helpful.
```

### Let us try to understand the coverage of city

In [57]:
```
df4=df2['City'].value_counts()
df4
```

Out[57]:
```
New Delhi          5473
Gurgaon            1118
Noida              1080
Faridabad           251
Ghaziabad            25
                   ...
Panchkula             1
Mc Millan             1
Mayfield              1
Macedon               1
Vineland Station      1
Name: City, Length: 141, dtype: int64
```

In [49]:
```
from plotly.offline import init_notebook_mode, plot, iplot
import plotly.graph_objs as go
plt.figure(figsize=(12,6))
# import plotly.plotly as py
labels = list(df2.City.value_counts().head(20).index)
values = list(df2.City.value_counts().head(20).values)
df7 = {"data":[{"labels" :labels,"values" : values,"hoverinfo" : 'label+percent',"domain": {"x": [0, .9]},"type" : "pie",
             "rotation":180,},],"layout": {"title" : "Zomato's Presence Citywise","annotations": [{"font": {"size":20},
             "showarrow": True,"text": "Cities","x":0.2,"y":0.9,},]}}
iplot(df7);
plt.show()
```




```
<Figure size 1200x600 with 0 Axes>
```

The data seems to be skewed towards New Delhi, Gurgaon and Noida. I see minimal data for other cities.
Hence I would do my analysis predominantly on New Delhi.

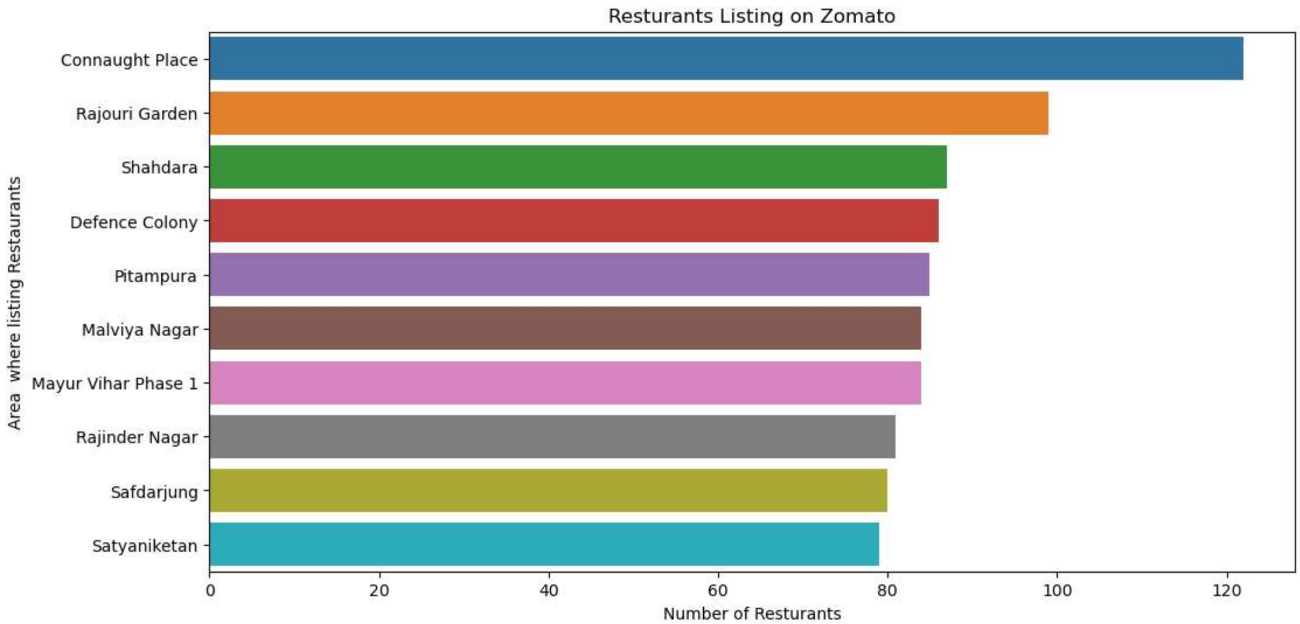
We've already gained several insights about the restaurants present in the survey.
Let us try some visualization and operation using ths data....to get more imformation...about the restaurants...

**Lets us try some experiment how the Zomato is being spread throughout the different section.....**

## Locality having maximum hotels are listed in Zomato

In [10]:
```
plt.figure(figsize=(6,4))
Delhi = df2[(df2.City == 'New Delhi')]
plt.figure(figsize=(12,6))
sns.barplot(x=Delhi.Locality.value_counts().head(10), y=Delhi.Locality.value_counts().head(10).index)
plt.ylabel('Area  where listing Restaurants');
plt.xlabel('Number of Resturants')
plt.title('Resturants Listing on Zomato');
```

```
<Figure size 600x400 with 0 Axes>
```




```
Connaught place seems to have high no of restaurants registered with Zomato, Let us understand the cuisines the top rated restaurants has to offer
```

**From New Delhi which type of Cuisines is famous from the highly rates restaurants....**

```
To get the idea about the restaurant having Excellent and Very Good Rating....
we exctract all the cuisiens in a single list
```

In [61]:
```
df4=df2['Cuisines'].value_counts()
df4
```

Out[61]:
```
North Indian                                      936
North Indian, Chinese                             511
Chinese                                           354
Fast Food                                         354
North Indian, Mughlai                             334
                                                 ...
Bengali, Fast Food                                  1
North Indian, Rajasthani, Asian                     1
Chinese, Thai, Malaysian, Indonesian                1
Bakery, Desserts, North Indian, Bengali, South Indian    1
Italian, World Cuisine                              1
Name: Cuisines, Length: 1825, dtype: int64
```

In [62]:
```
## Fetching the resturants having 'Excellent' and 'Very Good' rating
ConnaughtPlace = Delhi[(Delhi.Locality.isin(['Connaught Place'])) & (Delhi['Rating text'].isin(['Excellent','Very Good']))]
ConnaughtPlace = ConnaughtPlace.Cuisines.value_counts().reset_index()
## Extracing all the cuisens in a single list
cuisien = []
for x in ConnaughtPlace['index']:
  cuisien.append(x)
cuisien = '[%s]'%', '.join(map(str, cuisien))
cuisien
```

Out[62]:
```
'[North Indian, Chinese, Italian, Continental, Continental, Italian, Asian, Indian, Continental, Mediterranean, Italian, North Indian, Bakery, Desserts, Fast Food, North Indian, Continental, North Indian, European, Asian, Mediterranean, Continental, North Indian, Italian, Asian, North Indian, Afghani, Mughlai, North Indian, European, Cafe, Continental, Mexican, Burger, American, Pizza, Tex-Mex, South Indian, Asian, North Indian, Italian, Mexican, Continental, North Indian, Finger Food, Continental, American, Asian, North Indian, Fast Food, American, Burger, North Indian, Mediterranean, Asian, Fast Food, Ice Cream, Healthy Food, Continental, Italian, Japanese, Modern Indian, Chinese, Continental, North Indian, Chinese, Mediterranean, North Indian, Chinese, Italian, American, Middle Eastern, Biryani, Hyderabadi, Biryani, North Indian, Hyderabadi, North Indian, Chinese, North Indian, Chinese, Continental, Italian, North Indian, Italian, Asian, American, North Indian, Bakery, Fast Food, Desserts]'
```

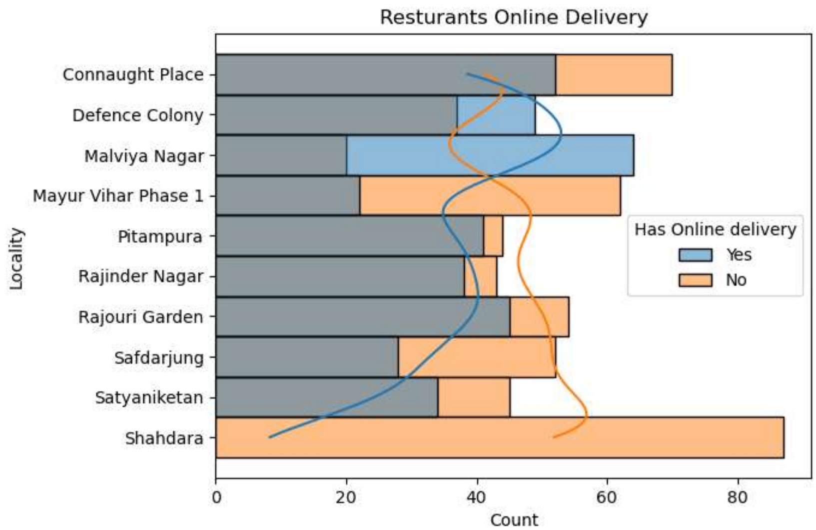
```
Top rated restaurants seems to be doing well in the following cuisine

North Indian
```

```
Chinese
Italian
American
```

```
Lets us search in Delhi which having high rated restaurant except online delivery or not..
```

In [17]:
```
top_locality = Delhi.Locality.value_counts().head(10)
ax=sns.histplot(data=Delhi[Delhi.Locality.isin(top_locality.index)],y='Locality',hue='Has Online delivery',kde=True)
plt.title('Resturants Online Delivery');
```



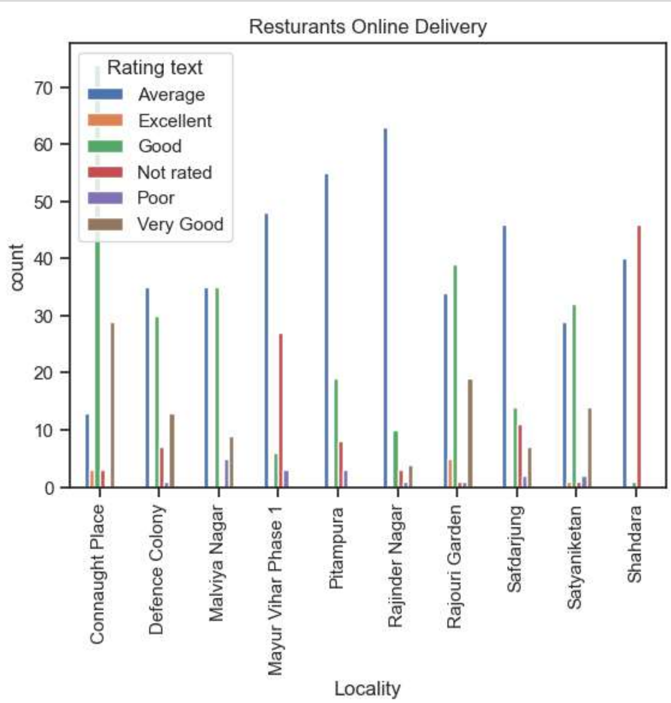

```
Apart from Shahdara locality, restaurants in other locality accepts online delivery.
Online Delivery seems to be on higher side in Defence colony and Malviya Nagar
```

**Lets us understand the Restaurants Rating localities..**

```
Apart from Malviya nagar, Defence colony in rest of the locality people seems to prefer visiting the restaurants rather ordering food online.
I would now like to understand the rating of these restaurants that are providing online delivery in Malviya nagar, Defence colony.
```

In [65]:
```
t1=Delhi[Delhi.Locality.isin(top_locality.index)]
```

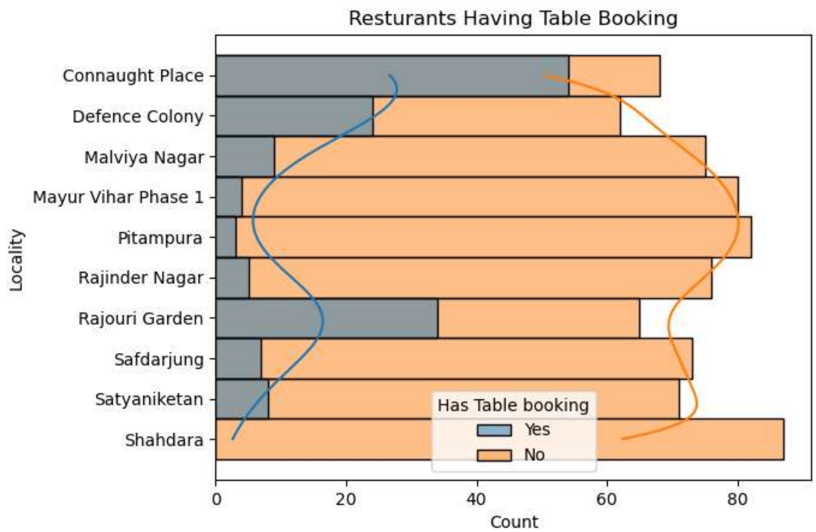
In [130]:
```
pd.crosstab(t1['Locality'],t1['Rating text']).plot(kind='bar')
plt.title('Resturants Online Delivery')
plt.ylabel('count')
plt.show()
```



```
Defence colony seems to have high no of highly rated restaurants but Malviya Nagar seems to done better in terms of Good and Average restaurants.
As restaurants with 'Poor' and 'Not Rated' is far lesser that 'Good', 'Very Good' and 'Excellent' restaurants. Hence people in these localities prefer online ordering
```

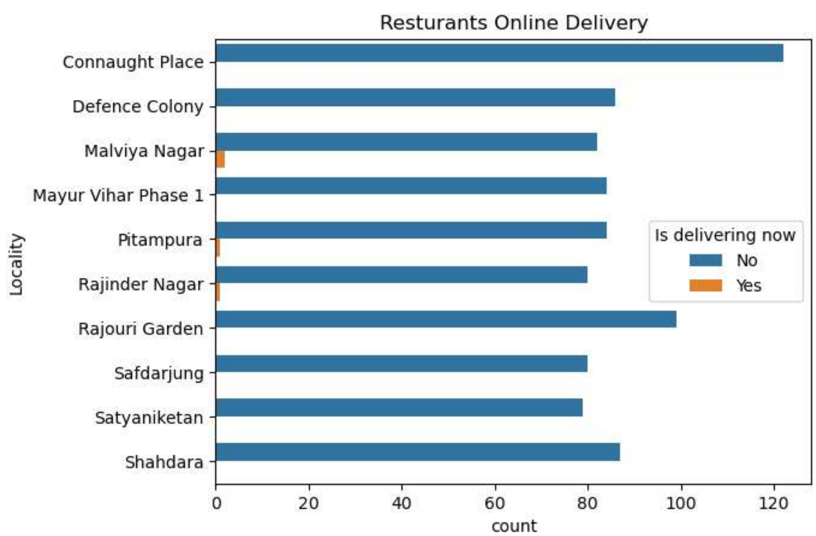
**Restaurant Having Table Booking or Not**

In [18]:
```
ax=sns.histplot(data=Delhi[Delhi.Locality.isin(top_locality.index)],y='Locality',hue='Has Table booking',kde=True)
plt.title('Resturants Having Table Booking');
```




**Restaurant Is Delivering Now**

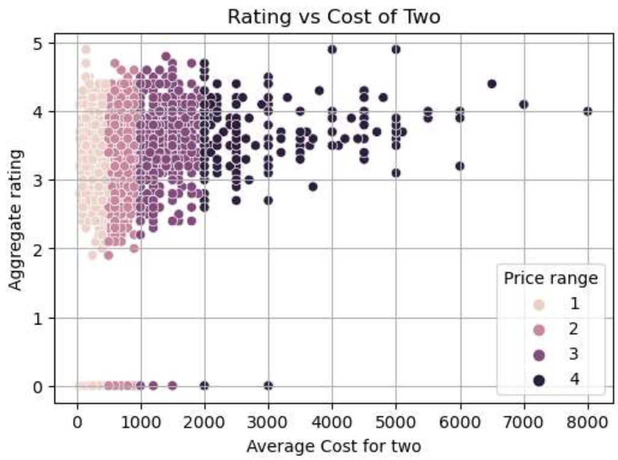
In [25]:
```
ax=sns.countplot(data=Delhi[Delhi.Locality.isin(top_locality.index)],y='Locality', hue='Is delivering now')
plt.title('Resturants Online Delivery');
```




### Let us understand the cost of Dinning vs their Aggregate Rating

```
That would help people to understand which restaurant is good in ratings according to reviews and cost....
That it is quite expensive or not and distribute them wr to price range....
```

In [26]:
```
plt.figure(figsize=(6,4))
sns.scatterplot(x="Average Cost for two", y="Aggregate rating", hue='Price range', data=Delhi)
plt.xlabel("Average Cost for two")
plt.ylabel("Aggregate rating")
plt.title('Rating vs Cost of Two')
plt.grid();
```




```
I observe there is no linear relation between price and rating. For instance, Restaurants with good rating (like 4-5) have restaurants with all the price range and spread across the entire X axis
```

**Location of Highly rated restaurants across New Delhi**

In [27]:
```
Delhi['Rating text'].value_counts()
```

Out[27]:
```
Average      2495
Not rated    1425
Good         1128
Very Good     300
Poor           97
Excellent      28
Name: Rating text, dtype: int64
```

In [28]:
```
Highly_rated = Delhi[Delhi['Rating text'].isin(['Excellent'])]
```

In [29]:
```
x=sns.countplot(data=Highly_rated,y='Locality')
plt.title('Excellent Rated Restaurant Locality');
```


```
The aforementioned four cities represent nearly 65% of the total data available in the dataset. Apart from the higly rated local restaurants, it’d be intersting to know where the known-eateries that are commonplace. The verticles across which these can be located are -

Breakfast
American Fast Food
Ice Creams, Shakes & Desserts
```

## Common Eateries

### Breakfast and Coffee locations

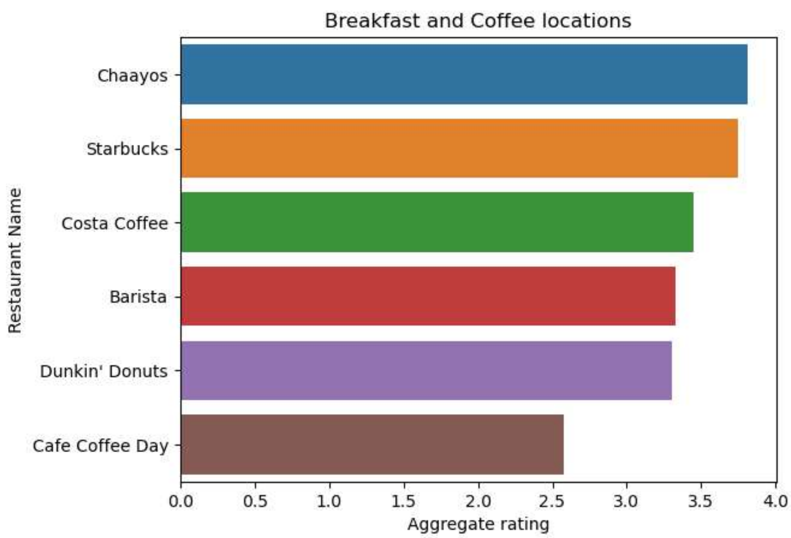
In [30]:
```
types = {"Breakfast and Coffee" : ["Cafe Coffee Day", "Starbucks", "Barista", "Costa Coffee", "Chaayos", "Dunkin' Donuts"],
    "American": ["Domino's Pizza", "McDonald's", "Burger King", "Subway", "Dunkin' Donuts", "Pizza Hut"],
    "Ice Creams and Shakes": ["Keventers", "Giani", "Giani's", "Starbucks", "Baskin Robbins", "Nirula's Ice Cream"]}
breakfast = Delhi[Delhi['Restaurant Name'].isin(types['Breakfast and Coffee'])]
american = Delhi[Delhi['Restaurant Name'].isin(types['American'])]
ice_cream = Delhi[Delhi['Restaurant Name'].isin(types['Ice Creams and Shakes'])]
```

In [31]:
```
breakfast = breakfast[['Restaurant Name','Aggregate rating']].groupby('Restaurant Name').mean().reset_index().sort_values('Aggregate rating',as
breakfast.drop("index",axis=1)
```

Out[31]:

| | Restaurant Name | Aggregate rating |
|---|---|---|
| 0 | Chaayos | 3.812500 |
| 1 | Starbucks | 3.750000 |
| 2 | Costa Coffee | 3.450000 |
| 3 | Barista | 3.325000 |
| 4 | Dunkin' Donuts | 3.300000 |
| 5 | Cafe Coffee Day | 2.573684 |

In [35]:
```
df= breakfast
d2=sns.barplot(data=df,y='Restaurant Name',x='Aggregate rating')
plt.title('Breakfast and Coffee locations')
plt.show()
```



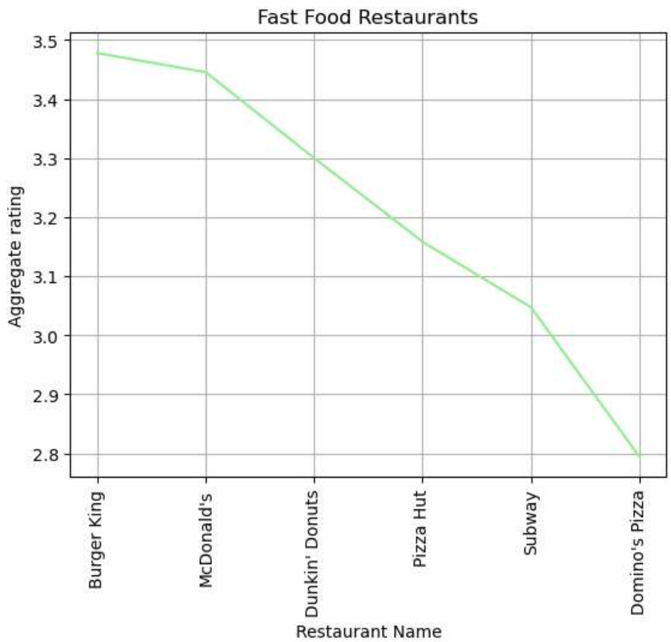

### Fast Food Restaurants

In [36]:
```
american = american[['Restaurant Name','Aggregate rating']].groupby('Restaurant Name').mean().reset_index().sort_values('Aggregate rating',asce
american
```

Out[36]:

| | Restaurant Name | Aggregate rating |
|---|---|---|
| 0 | Burger King | 3.477778 |
| 3 | McDonald's | 3.445455 |
| 2 | Dunkin' Donuts | 3.300000 |
| 4 | Pizza Hut | 3.158333 |
| 5 | Subway | 3.047368 |
| 1 | Domino's Pizza | 2.794545 |

In [42]:
```
df2= american
d2=sns.lineplot(data=df2,x='Restaurant Name',y='Aggregate rating',color='lightgreen')
plt.title('Fast Food Restaurants')
plt.xticks(rotation=90)
plt.grid()
plt.show()
```



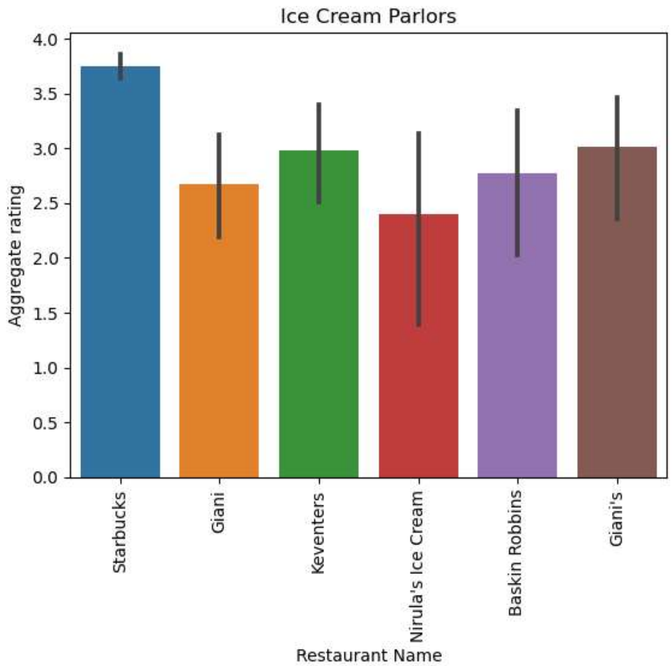

### Ice Cream Parlors

In [80]:
```
ice_cream =ice_cream[['Restaurant Name','Aggregate rating']].groupby('Restaurant Name').mean().reset_index().sort_values('Aggregate rating',asc
ice_cream
```

Out[80]:

| | Restaurant Name | Aggregate rating |
|---|---|---|
| 5 | Starbucks | 3.750000 |
| 2 | Giani's | 3.011765 |
| 3 | Keventers | 2.983333 |
| 0 | Baskin Robbins | 2.769231 |
| 1 | Giani | 2.675000 |
| 4 | Nirula's Ice Cream | 2.400000 |

In [41]:
```
df2= ice_cream
d2=sns.barplot(data=df2,x='Restaurant Name',y='Aggregate rating')
plt.xticks(rotation=90)
plt.title('Ice Cream Parlors')
plt.show()
```




```
Foreign brands seems to be doing better than the local brands
```

## Inferences and Conclusions

```
We've drawn many inferences from the survey. Here's a summary of a few of them:

The dataset is skewed towards India and doesn't represent the complete data of restaurants worldwide.
Restaurants rating is categorized in categories
Not Rated
Average
Good
Very Good
Excellent
Connaught Palace have maximum restaurants listed on Zomato but in terms of online delivery acceptance Defence colony and Malviya nagar seems to be doing better.
The top rated restaurants seems to be getting better rating on the following cuisine
North Indian
Chinese
American
Italian
There is no relation between cost and rating. Some of the best rated restaurants are low on cost and vice versa.
On common Eateries, For Breakfast and Coffee location Indian restaurants seems to be better rated but for Fast food chain and Ice cream parlors American restaurants seems to be doing better.
```

```
Future of Zomato in India......
Here's why Zomato works well in these countries and it's hard for someone like Yelp to beat it: Cheap manual labor: The Zomato model needs a lot of paid manual labor. Countries with cheaper labor costs like India are the ones where Zomato has succeeded till now.
```

"Food has no religion" basic moto of Zomato

```
First mover advantage – One of the best competitive advantages of Zomato is that it is the first mover in many of the nations where it is establishing itself. Directories and other forms of restaurant ratings might exist.
```

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: