Zomato Restaurants Ratings Analysis

Problem Statement

Company: Zomato, a leading food delivery and restaurant discovery platform.

Background: Zomato is planning to expand its business on a global scale and aims to provide a seamless dining experience to its users. To achieve this, they need to gain a comprehensive understanding of the restaurant landscape and customer preferences worldwide. Specifically, they want to analyze their ratings data to uncover valuable insights about restaurants, services, cuisines, and customer sentiments in various countries and cities.

Objective: As a data scientist, your task is to analyze the Zomato ratings data and provide insights into the following key aspects:

1. **Country-Level Analysis:**
   - Identify the top-performing countries in terms of restaurant ratings.
   - Analyze the overall customer satisfaction levels by country.
2. **City-Level Analysis:**
   - Explore customer ratings and reviews across different cities.
   - Determine the most popular and highly-rated cities for dining.
3. **Rating Distributions:**
   - Examine the distribution of customer ratings and identify any trends or patterns.
4. **Restaurant Insights:**
   - Uncover insights about the highest and lowest-rated restaurants.
   - Evaluate the correlation between restaurant attributes and ratings.
5. **Service Quality:**
   - Investigate customer feedback on restaurant services and identify areas for improvement.
6. **Cuisine Analysis:**
   - Determine the popularity of different cuisines in various regions.
   - Analyze the relationship between cuisines and customer ratings.

Deliverables: Your findings and insights will be presented in a comprehensive report, which will guide Zomato in making data-driven decisions to enhance their global business expansion strategy.

Success Criteria: Successful completion of this project will provide Zomato with actionable insights to make informed decisions for their global expansion, improve customer satisfaction, and optimize their platform's restaurant offerings.

Dataset Description :

Columns

- **Restaurant Id**: Unique identifier for each restaurant.
- **Restaurant Name**: The name of the restaurant.
- **Country Code**: The country in which the restaurant is located.
- **City**: The city in which the restaurant is located.
- **Address**: The street address of the restaurant.
- **Locality**: The specific location within the city.
- **Locality Verbose**: A detailed description of the locality.
- **Longitude**: The longitude coordinate of the restaurant's location.
- **Latitude**: The latitude coordinate of the restaurant's location.
- **Cuisines**: The types of cuisines offered by the restaurant.
- **Average Cost for Two**: The cost for two people in different currencies (local currency).
- **Currency**: The currency used in the country.
- **Has Table Booking**: Whether the restaurant offers table booking (yes/no).
- **Has Online Delivery**: Whether the restaurant offers online delivery (yes/no).
- **Is Delivering**: Whether the restaurant is currently delivering (yes/no).
- **Switch to Order Menu**: Whether the restaurant allows switching to an order menu (yes/no).
- **Price Range**: The range of prices for food.
- **Aggregate Rating**: The average rating out of 5.
- **Rating Color**: The color representation of the average rating.
- **Rating Text**: Textual representation of the rating.
- **Votes**: The number of ratings given by customers.

Data Source

[https://www.kaggle.com/datasets/shrutimehta/zomato-restaurants-data?select=zomato.csv] (https://www.kaggle.com/datasets/shrutimehta/zomato-restaurants-data?select=zomato.csv%5D)

Importing the necessary libraries (pandas, numpy, matplotlib, seaborn, warnings)

In [921]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
from pandas import Series
%matplotlib inline
warnings.filterwarnings('ignore')
```

Reading the dataset

In [689]:
```
data = pd.read_csv("Data/zomato.csv", encoding = "latin-1")
```

In [690]:
```
original_data = data.copy()
```

Viewing the top five records

In [691]:
```
data.head()
```

Out[691]:

	Restaurant ID	Restaurant Name	Country Code	City	Address	Locality	Locality Verbose	L
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...	12
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...	12
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...	12
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...	12
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...	12

5 rows × 21 columns

viewing the last five records

In [692]:
```
data.tail()
```

Out[692]:

	Restaurant ID	Restaurant Name	Country Code	City	Address	Locality	Locality Verbose
9546	5915730	NamlÛ± Gurme	208	ÛÁstanbul	Kemanke□ô Karamustafa Pa□ôa Mahallesi, RÛ±htÛ±...	Karakí_y	Karakí_y, ÛÁstanbul
9547	5908749	Ceviz AÛôacÛ±	208	ÛÁstanbul	Ko□ôuyolu Mahallesi, Muhittin íistí_ndaÛô Cadd...	Ko□ôuyolu	Ko□ôuyolu, ÛÁstanbul
9548	5915807	Huqqa	208	ÛÁstanbul	Kuruí_e□ôme Mahallesi, Muallim Naci Caddesi, N...	Kuruí_e□ôme	Kuruí_e□ôme, ÛÁstanbul
9549	5916112	A□ô□ôk Kahve	208	ÛÁstanbul	Kuruí_e□ôme Mahallesi, Muallim Naci Caddesi, N...	Kuruí_e□ôme	Kuruí_e□ôme, ÛÁstanbul
9550	5927402	Walter's Coffee Roastery	208	ÛÁstanbul	CafeaÛôa Mahallesi, BademaltÛ± Sokak, No 21/B,...	Moda	Moda, ÛÁstanbul

5 rows × 21 columns

Exploring the features in a dataset

In [693]:
```
data.columns
```

Out[693]:
```
Index(['Restaurant ID', 'Restaurant Name', 'Country Code', 'City',
'Address',
       'Locality', 'Locality Verbose', 'Longitude', 'Latitude', 'C
uisines',
       'Average Cost for two', 'Currency', 'Has Table booking',
       'Has Online delivery', 'Is delivering now', 'Switch to orde
r menu',
       'Price range', 'Aggregate rating', 'Rating color', 'Rating
text',
       'Votes'],
      dtype='object')
```

Exploring the shape of the dataset

In [694]:
```
data.shape
```

Out[694]:
```
(9551, 21)
```

Checking for null values

In [695]:
```
data.isna().sum()
```

Out[695]:
```
Restaurant ID           0
Restaurant Name         0
Country Code            0
City                    0
Address                 0
Locality                0
Locality Verbose        0
Longitude               0
Latitude                0
Cuisines                9
Average Cost for two    0
Currency                0
Has Table booking       0
Has Online delivery     0
Is delivering now       0
Switch to order menu    0
Price range             0
Aggregate rating        0
Rating color            0
Rating text             0
Votes                   0
dtype: int64
```

Insights :

- Out of **21** features **1** feature named **(Cuisines)** has a missing values of **(9)**

Checking for datatypes

In [696]:
```
data.dtypes
```

Out[696]:
```
Restaurant ID              int64
Restaurant Name           object
Country Code               int64
City                      object
Address                   object
Locality                  object
Locality Verbose          object
Longitude                float64
Latitude                 float64
Cuisines                  object
Average Cost for two       int64
Currency                  object
Has Table booking         object
Has Online delivery       object
Is delivering now         object
Switch to order menu      object
Price range                int64
Aggregate rating         float64
Rating color              object
Rating text               object
Votes                      int64
dtype: object
```

Exploring the unique features

In [697]:
```
data.nunique()
```

Out[697]:
```
Restaurant ID           9551
Restaurant Name         7446
Country Code              15
City                     141
Address                 8918
Locality                1208
Locality Verbose        1265
Longitude               8120
Latitude                8677
Cuisines                1825
Average Cost for two     140
Currency                  12
Has Table booking          2
Has Online delivery        2
Is delivering now          2
Switch to order menu       1
Price range                4
Aggregate rating          33
Rating color               6
Rating text                6
Votes                   1012
dtype: int64
```

Checking for duplicated values in the features

In [698]:
```
data.duplicated().sum()
```
Out[698]:
```
0
```

Insights:

- There are no duplicated features **(0)**

Adding new feature (Country)

In [699]:
```
country_map = {
    1: "India",
    14: "Australia",
    30: "Brazil",
    37: "Canada",
    94: "Indonesia",
    148: "NewZealand",
    162: "Philippines",
    166: "Quatar",
    184: "Singapore",
    189: "South Africa",
    191: "Sri Lanka",
    208: "Turkey",
    214: "Abu Dhabi",
    215: "UK",
    216: "USA"

}
```

In [700]:
```
data["Country"] = data["Country Code"].map(country_map)
```

In [701]:
```
data.head()
```

Out[701]:

	Restaurant ID	Restaurant Name	Country Code	City	Address	Locality	Locality Verbose	L
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...	12
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...	12
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...	12
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...	12
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...	12

5 rows × 22 columns

Checking for duplicated records after adding Country feature

In [702]:
```
data.duplicated().sum()
```

Out[702]: 0

dropping the unwanted features

In [703]:
```
features_to_drop = ["Restaurant ID","Longitude","Latitude","Localit
data.drop(columns=features_to_drop, axis=1, inplace=True)
```

In [704]:
```python
data.drop(columns=["Address"], axis=1, inplace=True)
```

In [705]:
```python
data.drop(columns=["Country Code"], axis=1, inplace=True)
```

In [706]:
```python
data.head()
```

Out[706]:

	Restaurant Name	City	Cuisines	Average Cost for two	Currency	Has Table booking	Has Online delivery	Is delivering now	S
0	Le Petit Souffle	Makati City	French, Japanese, Desserts	1100	Botswana Pula(P)	Yes	No	No	
1	Izakaya Kikufuji	Makati City	Japanese	1200	Botswana Pula(P)	Yes	No	No	
2	Heat - Edsa Shangri-La	Mandaluyong City	Seafood, Asian, Filipino, Indian	4000	Botswana Pula(P)	Yes	No	No	
3	Ooma	Mandaluyong City	Japanese, Sushi	1500	Botswana Pula(P)	No	No	No	
4	Sambo Kojin	Mandaluyong City	Japanese, Korean	1500	Botswana Pula(P)	Yes	No	No	

In [707]:
```python
data.shape
```

Out[707]:
```
(9551, 15)
```

Checking for duplicated records after dropping unwanted features from the dataset

In [708]:
```python
res = data.duplicated().reset_index()
```

In [709]:
```python
res[res[0] == True].sum()
```

Out[709]:
```
index    289479
0            43
dtype: int64
```

Insights :

- There are **43** duplicated records

Dropping duplicated records

In [710]:
```python
data = data.drop_duplicates()
```

In [711]:
```
data.duplicated().sum()
```
Out[711]:
```
0
```

Exploring numerical and categorical features

In [712]:
```
numerical_features = [feature for feature in data.columns if data[f
```
In [713]:
```
numerical_features
```
Out[713]:
```
['Average Cost for two', 'Price range', 'Aggregate rating', 'Vote
s']
```
In [714]:
```
categorical_features = [feature for feature in data.columns if data
```
In [715]:
```
categorical_features
```
Out[715]:
```
['Restaurant Name',
 'City',
 'Cuisines',
 'Currency',
 'Has Table booking',
 'Has Online delivery',
 'Is delivering now',
 'Switch to order menu',
 'Rating color',
 'Rating text',
 'Country']
```
In [716]:
```
print(f"Number of numerical features : {len(numerical_features)}")
print(f"Number of categorical features : {len(categorical_features)
```
```
Number of numerical features : 4
Number of categorical features : 11
```

Insights :

- Out of **15** features - **4** features are **numerical features** and **11** features are **categorical features**

Removing Missing values using mode

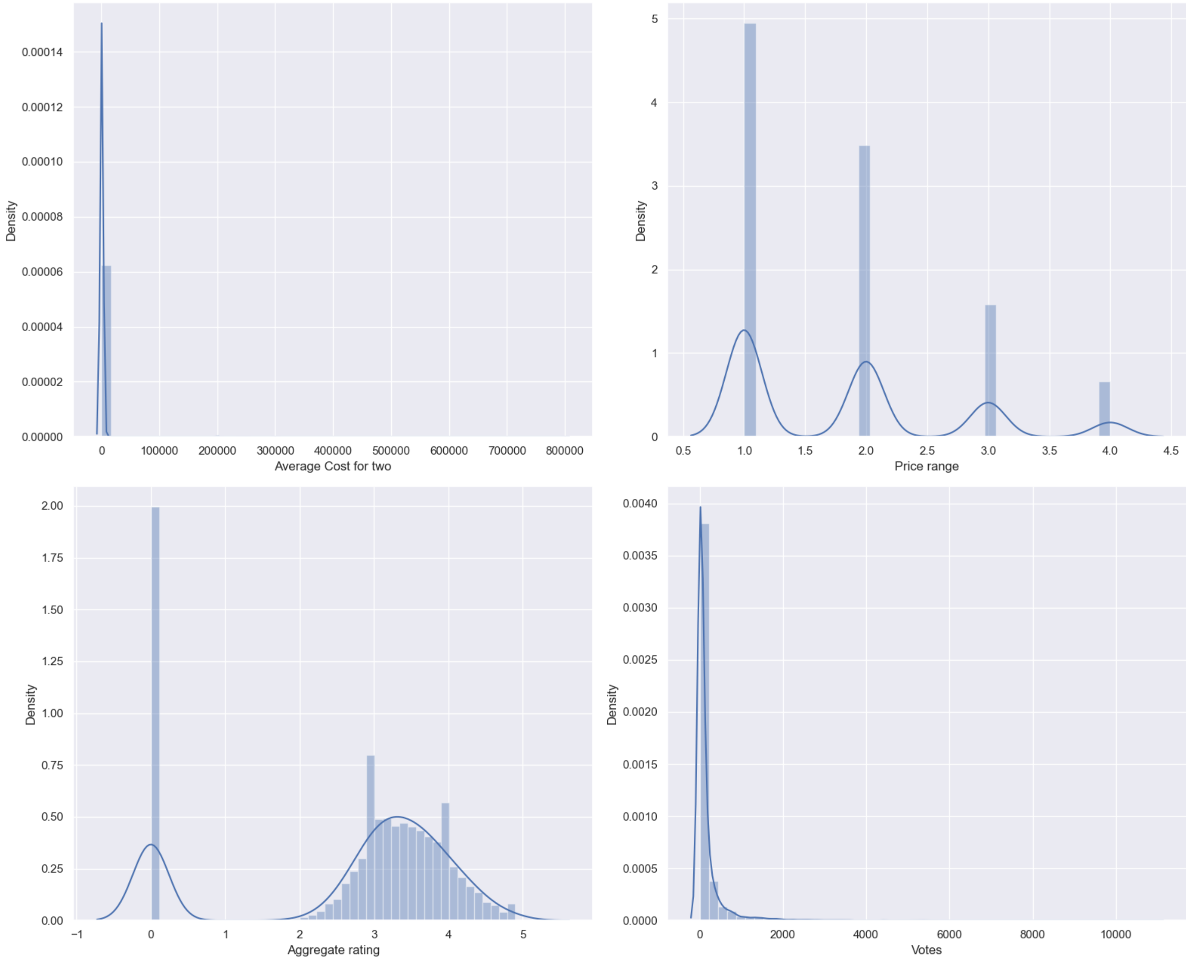
In [717]:
```
mode_value = data["Cuisines"].mode()[0]
data["Cuisines"].fillna(mode_value,inplace=True)
```

In [718]:
```
data.isnull().sum()
```

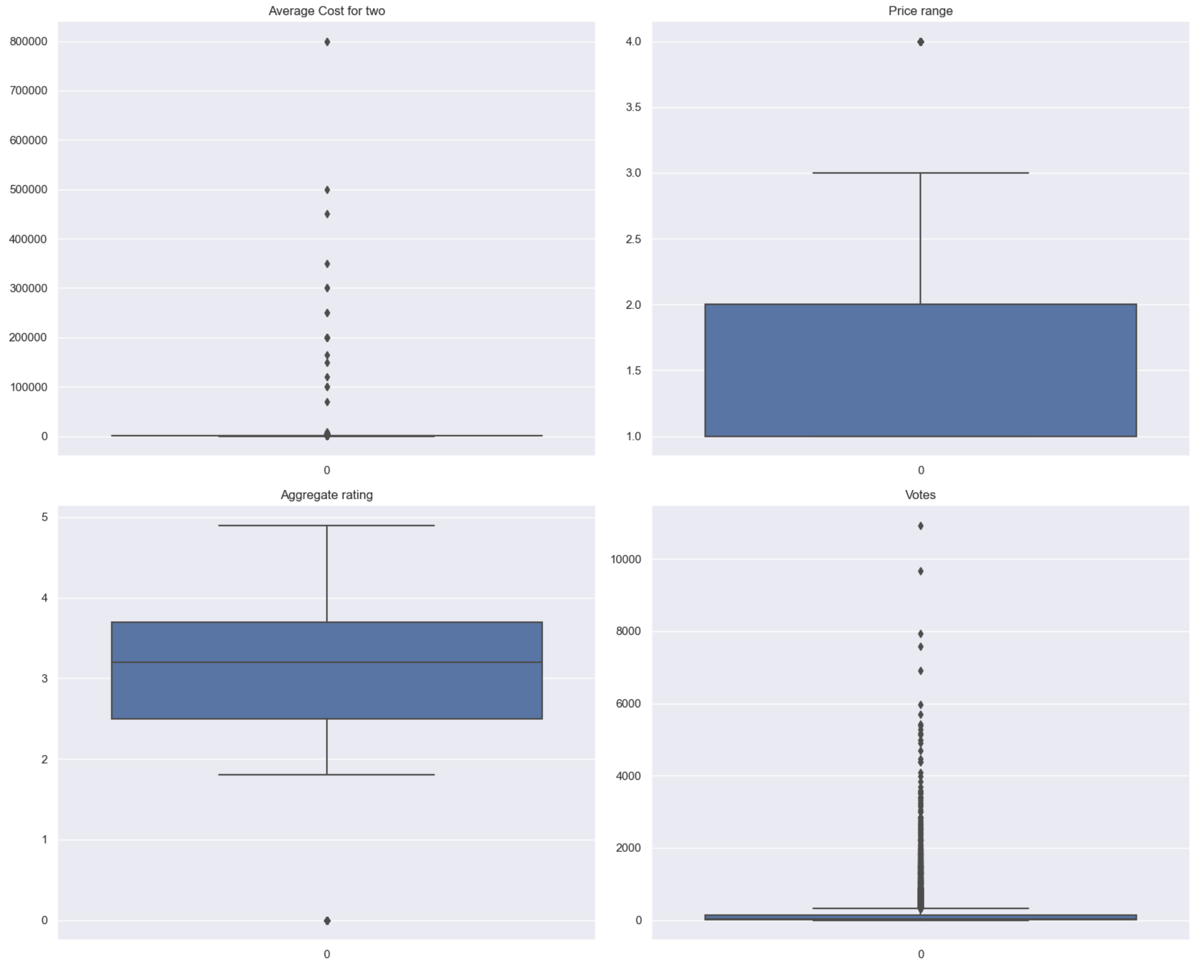
Out[718]:
```
Restaurant Name         0
City                    0
Cuisines                0
Average Cost for two    0
Currency                0
Has Table booking       0
Has Online delivery     0
Is delivering now       0
Switch to order menu    0
Price range             0
Aggregate rating        0
Rating color            0
Rating text             0
Votes                   0
Country                 0
dtype: int64
```

Outlier Detection

In [719]:
```
count = 1
plt.figure(figsize=(16,13))
for feature in numerical_features:
    plt.subplot(2,2,count)
    sns.distplot(data[feature])
    count+=1
plt.tight_layout()
plt.show()
```



In [720]:
```
count = 1
plt.figure(figsize=(16,13))
for feature in numerical_features:
    plt.subplot(2,2,count)
    sns.boxplot(data[feature])
    plt.title(f"{feature}")
    count+=1
    plt.title(f'{feature}')
plt.tight_layout()
plt.show()
```



EDA(Exploratory Data Analysis)

Distribution of restaurants across countries and cities

In [721]:
```
country_rank_list = data.groupby("Country")["Votes"].sum().reset_in
```

In [722]:
```
country_rank_list
```

Out[722]:

	Country	Votes
0	Abu Dhabi	29611
1	Australia	2674
2	Brazil	1177
3	Canada	412
4	India	1187070
5	Indonesia	16214
6	NewZealand	9721
7	Philippines	8963
8	Quatar	3276
9	Singapore	638
10	South Africa	18910
11	Sri Lanka	2929
12	Turkey	14670
13	UK	16439
14	USA	185848

In [723]:
```
country_rank_list = country_rank_list.sort_values(
    by="Votes",
    ascending=False
)
```

In [724]:
```
country_rank_list
```
Out[724]:

	Country	Votes
4	India	1187070
14	USA	185848
0	Abu Dhabi	29611
10	South Africa	18910
13	UK	16439
5	Indonesia	16214
12	Turkey	14670
6	NewZealand	9721
7	Philippines	8963
8	Quatar	3276
11	Sri Lanka	2929
1	Australia	2674
2	Brazil	1177
9	Singapore	638
3	Canada	412

In [725]:
```
top_five_voted_countries = country_rank_list.head(5)
```
In [726]:
```
top_five_voted_countries
```
Out[726]:

	Country	Votes
4	India	1187070
14	USA	185848
0	Abu Dhabi	29611
10	South Africa	18910
13	UK	16439

In [727]:
```
least_five_voted_countries = country_rank_list.tail(5)
```

In [728]:
```
least_five_voted_countries
```

Out[728]:

	Country	Votes
11	Sri Lanka	2929
1	Australia	2674
2	Brazil	1177
9	Singapore	638
3	Canada	412

In [729]:
```
maximum_voted_country = country_rank_list.head(1)
```

In [730]:
```
maximum_voted_country
```

Out[730]:

	Country	Votes
4	India	1187070

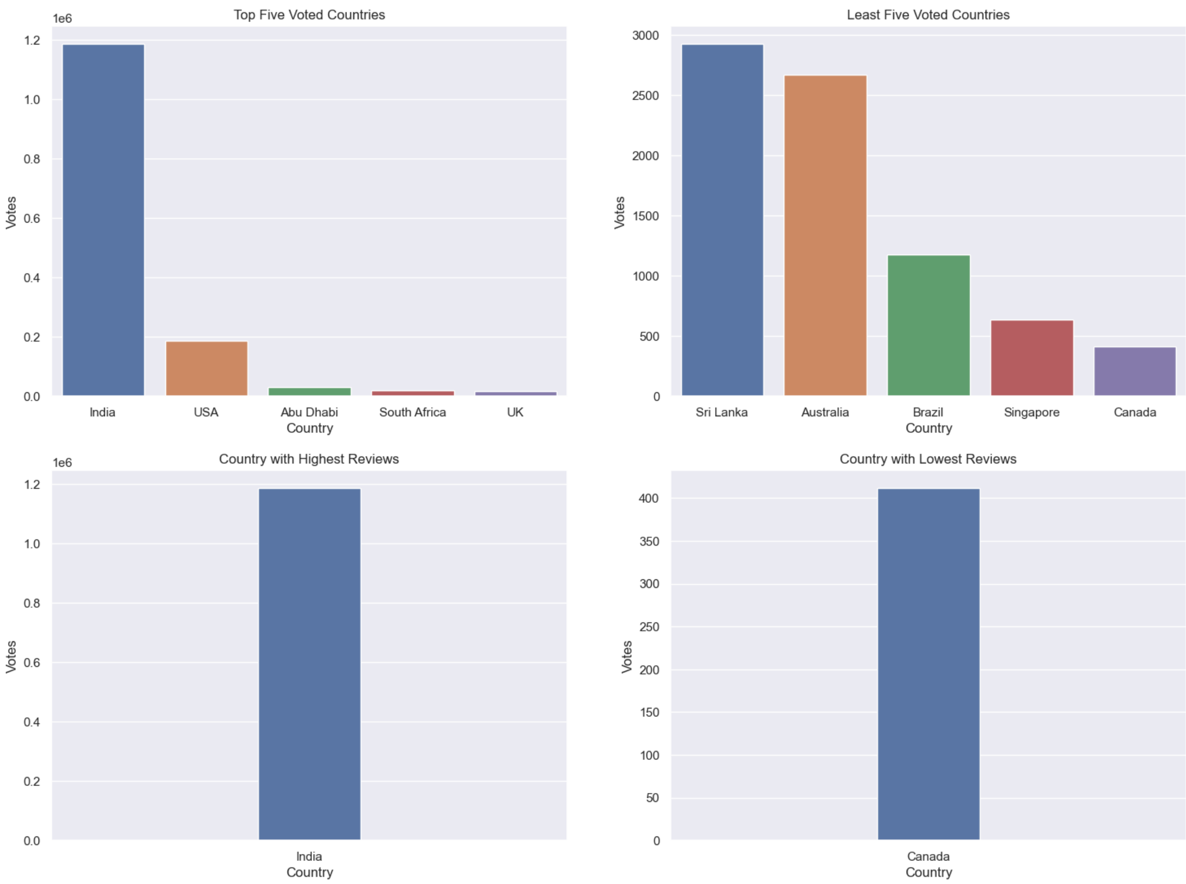
In [731]:
```
minimum_voted_country  = country_rank_list.sort_values(by="Votes").
```

In [732]:
```
minimum_voted_country
```

Out[732]:

	Country	Votes
3	Canada	412

```
In [733]: sns.set(style="darkgrid")
plt.figure(figsize=(18,13))
plt.subplot(2,2,1)
axis_1 = sns.barplot(x='Country', y='Votes', data=top_five_voted_co
plt.title("Top Five Voted Countries")
plt.subplot(2,2,2)
axis_2 = sns.barplot(x='Country', y='Votes', data=least_five_voted_
plt.title("Least Five Voted Countries")
plt.subplot(2,2,3)
axis_3 = sns.barplot(x='Country', y='Votes', data=maximum_voted_cou
plt.title("Country with Highest Reviews")
plt.subplot(2,2,4)
axis_4 = sns.barplot(x='Country', y='Votes', data=minimum_voted_cou
plt.title("Country with Lowest Reviews")
plt.show()
```



Observation :

- From the above observations, **India** and the **USA** have the **maximum** number of customers across the globe.
- The country with the maximum number of customers is **India**.
- The country with the minimum number of customers is **Canada**.

Cities with maximum reviews from India

```
In [734]: grouped_data = data.groupby("Country")
```

In [735]:
```
reviews_from_india = grouped_data.get_group("India")
```

In [736]:
```
reviews_from_india
```

Out[736]:

	Restaurant Name	City	Cuisines	Average Cost for two	Currency	Has Table booking	Has Online delivery	Is delivering now	S
624	Jahanpanah	Agra	North Indian, Mughlai	850	Indian Rupees(Rs.)	No	No	No	
625	Rangrezz Restaurant	Agra	North Indian, Mughlai	700	Indian Rupees(Rs.)	No	No	No	
626	Time2Eat - Mama Chicken	Agra	North Indian	500	Indian Rupees(Rs.)	No	No	No	
627	Chokho Jeeman Marwari Jain Bhojanalya	Agra	Rajasthani	400	Indian Rupees(Rs.)	No	No	No	
628	Pinch Of Spice	Agra	North Indian, Chinese, Mughlai	1000	Indian Rupees(Rs.)	No	No	No	
...	...	...	...	...	...	...	...	...	
9271	D Cabana	Vizag	Continental, Seafood, Chinese, North Indian, B...	600	Indian Rupees(Rs.)	No	No	No	
9272	Kaloreez	Vizag	Cafe, North Indian, Chinese	400	Indian Rupees(Rs.)	No	No	No	
9273	Plot 17	Vizag	Burger, Pizza, Biryani	600	Indian Rupees(Rs.)	No	No	No	
9274	Vista - The Park	Vizag	American, North Indian, Thai, Continental	1500	Indian Rupees(Rs.)	No	No	No	
9275	Flying Spaghetti Monster	Vizag	Italian	1400	Indian Rupees(Rs.)	No	No	No	

8609 rows × 15 columns

In [737]:
```
reviews_from_indian_cities = reviews_from_india.groupby("City")
```

In [738]:
```
votings_from_indian_cities = reviews_from_indian_cities["Votes"].su
```

In [739]:
```
votings_from_indian_cities = votings_from_indian_cities.reset_index
```

In [740]:
```
votings_from_indian_cities.sort_values(
    by="Votes",
    ascending = False,
    inplace = True

)
```

In [741]:
```
votings_from_indian_cities.head()
```

Out[741]:

	City	Votes
31	New Delhi	628262
15	Gurgaon	132155
32	Noida	73478
5	Bangalore	56115
22	Kolkata	44593

In [742]:
```
top_five_voted_indian_cities = votings_from_indian_cities.head(5)
```

In [743]:
```
top_five_voted_indian_cities
```

Out[743]:

	City	Votes
31	New Delhi	628262
15	Gurgaon	132155
32	Noida	73478
5	Bangalore	56115
22	Kolkata	44593

In [744]:
```
reviews_from_india
```

Out[744]:

	Restaurant Name	City	Cuisines	Average Cost for two	Currency	Has Table booking	Has Online delivery	Is delivering now	S
624	Jahanpanah	Agra	North Indian, Mughlai	850	Indian Rupees(Rs.)	No	No	No	
625	Rangrezz Restaurant	Agra	North Indian, Mughlai	700	Indian Rupees(Rs.)	No	No	No	
626	Time2Eat - Mama Chicken	Agra	North Indian	500	Indian Rupees(Rs.)	No	No	No	
627	Chokho Jeeman Marwari Jain Bhojanalya	Agra	Rajasthani	400	Indian Rupees(Rs.)	No	No	No	
628	Pinch Of Spice	Agra	North Indian, Chinese, Mughlai	1000	Indian Rupees(Rs.)	No	No	No	
...	...	...	...	...	...	...	...	...	
9271	D Cabana	Vizag	Continental, Seafood, Chinese, North Indian, B...	600	Indian Rupees(Rs.)	No	No	No	
9272	Kaloreez	Vizag	Cafe, North Indian, Chinese	400	Indian Rupees(Rs.)	No	No	No	
9273	Plot 17	Vizag	Burger, Pizza, Biryani	600	Indian Rupees(Rs.)	No	No	No	
9274	Vista - The Park	Vizag	American, North Indian, Thai, Continental	1500	Indian Rupees(Rs.)	No	No	No	
9275	Flying Spaghetti Monster	Vizag	Italian	1400	Indian Rupees(Rs.)	No	No	No	

8609 rows × 15 columns

Overall Rating Categories in india

In [745]:
```
reviews_from_india.head()
```

Out[745]:

	Restaurant Name	City	Cuisines	Average Cost for two	Currency	Has Table booking	Has Online delivery	Is delivering now	Swit ord mer
624	Jahanpanah	Agra	North Indian, Mughlai	850	Indian Rupees(Rs.)	No	No	No	N
625	Rangrezz Restaurant	Agra	North Indian, Mughlai	700	Indian Rupees(Rs.)	No	No	No	N
626	Time2Eat - Mama Chicken	Agra	North Indian	500	Indian Rupees(Rs.)	No	No	No	N
627	Chokho Jeeman Marwari Jain Bhojanalya	Agra	Rajasthani	400	Indian Rupees(Rs.)	No	No	No	N
628	Pinch Of Spice	Agra	North Indian, Chinese, Mughlai	1000	Indian Rupees(Rs.)	No	No	No	N

In [746]:
```
rating = reviews_from_india[["City","Aggregate rating","Rating colo
```

Cities With Zero Poor Ratings,Zero Not Rated Ratings And Excellent Ratings

In [747]:
```
excellent_rating = rating[(rating["Aggregate rating"]>=4.7) & (rati
```

In [748]:
```
excellent_rating = rating[(rating["Rating text"] != "Poor") & (rati
```

In [749]:
```
excellent_rating.head()
```

Out[749]:

	City	Aggregate rating	Rating color	Rating text
637	Agra	4.9	Dark Green	Excellent
646	Ahmedabad	4.5	Dark Green	Excellent
653	Ahmedabad	4.6	Dark Green	Excellent
660	Ahmedabad	4.5	Dark Green	Excellent
727	Bangalore	4.7	Dark Green	Excellent

In [750]:
```
excellent_rating_cities = excellent_rating[["City","Rating text"]]
```

In [751]:
```python
excellent_rating_cities
```

Out[751]:

	City	Rating text
637	Agra	Excellent
646	Ahmedabad	Excellent
653	Ahmedabad	Excellent
660	Ahmedabad	Excellent
727	Bangalore	Excellent
...	...	...
9195	Secunderabad	Excellent
9216	Vadodara	Excellent
9228	Vadodara	Excellent
9256	Vizag	Excellent
9262	Vizag	Excellent

116 rows × 2 columns

In [752]:
```python
excellent_rating_cities = excellent_rating_cities.groupby("City").c
```

In [753]:
```python
excellent_rating_cities = excellent_rating_cities.sort_values(
    by="Rating text",
    ascending = False
)
```

Top Cities With Excellent Ratings

In [754]:
```python
top_five_excellent_rating_cities = excellent_rating_cities.head(5)
```

In [755]:
```python
top_five_excellent_rating_cities = top_five_excellent_rating_cities
```

In [756]:
```python
top_five_excellent_rating_cities
```

Out[756]:

	City	Rating text
0	New Delhi	28
1	Gurgaon	12
2	Bangalore	9
3	Chennai	6
4	Goa	6

Cities With Zero Poor Ratings,Zero Not Rated Very And Good Ratings

In [757]:
```
very_good_rating = rating[(rating["Rating text"] != "Poor") & (rati
```

In [758]:
```
very_good_rating.reset_index(inplace=True)
very_good_rating = very_good_rating[["City","Rating text"]]
```

In [759]:
```
very_good_rating = very_good_rating.groupby("City").count()
```

In [760]:
```
very_good_rating = very_good_rating.sort_values(
    by="Rating text",
    ascending = False
)
```

Top Cities With Very Good Rating

In [761]:
```
very_good_rating = very_good_rating.head()
```

In [762]:
```
very_good_rating = very_good_rating.reset_index()
```

In [763]:
```
very_good_rating
```

Out[763]:

	City	Rating text
0	New Delhi	300
1	Gurgaon	83
2	Noida	27
3	Guwahati	15
4	Pune	14

Cities With Zero Poor Ratings,Zero Not Rated And Good Ratings

In [764]:
```
good_rating = rating[(rating["Rating text"] != "Poor") & (rating["R
```

In [765]:
```
good_rating.reset_index(inplace=True)
good_rating = good_rating[["City","Rating text"]]
```

In [766]:
```
good_rating = good_rating.groupby("City").count()
```

In [767]:
```
good_rating = good_rating.sort_values(
    by="Rating text",
    ascending = False
)
```

Top Cities With Good Ratings

In [768]:
```
good_rating = good_rating.head()
```

In [769]:
```
good_rating = good_rating.reset_index()
```

In [770]:
```
good_rating
```

Out[770]:

	City	Rating text
0	New Delhi	1128
1	Gurgaon	257
2	Noida	173
3	Faridabad	22
4	Mangalore	18

Cities With Zero Poor Ratings,Zero Not Rated And Average Ratings

In [771]:
```
average_rating = rating[(rating["Rating text"] != "Poor") & (rating
```

In [772]:
```
average_rating.reset_index(inplace=True)
average_rating = average_rating[["City","Rating text"]]
```

In [773]:
```
average_rating = average_rating.groupby("City").count()
```

In [774]:
```
average_rating = average_rating.sort_values(
    by="Rating text",
    ascending = False
)
```

Top Cities With Average Ratings

In [775]:
```
average_rating = average_rating.head()
```

In [776]:
```
average_rating = average_rating.reset_index()
```

In [777]:
```
average_rating
```
Out[777]:

	City	Rating text
0	New Delhi	2489
1	Gurgaon	504
2	Noida	448
3	Faridabad	123
4	Ghaziabad	18

Cities With Poor Ratings

In [778]:
```
poor_rating = rating[(rating["Rating text"] == "Poor")]
```
In [779]:
```
poor_rating.reset_index(inplace=True)
poor_rating = poor_rating[["City","Rating text"]]
```
In [780]:
```
poor_rating = poor_rating.groupby("City").count()
```
In [781]:
```
poor_rating = poor_rating.sort_values(
    by="Rating text",
    ascending = False
)
```

Top Cities With Poor Ratings

In [782]:
```
poor_rating = poor_rating.head()
```
In [783]:
```
poor_rating = poor_rating.reset_index()
```
In [784]:
```
poor_rating
```
Out[784]:

	City	Rating text
0	New Delhi	97
1	Noida	45
2	Gurgaon	34
3	Faridabad	2
4	Ghaziabad	1

Cities With Not Ratings

In [785]:
```
not_rated_rating = rating[(rating["Rating text"] == "Not rated")]
```

In [786]:
```
not_rated_rating.reset_index(inplace=True)
not_rated_rating = not_rated_rating[["City","Rating text"]]
```

In [787]:
```
not_rated_rating = not_rated_rating.groupby("City").count()
```

In [788]:
```
not_rated_rating = not_rated_rating.sort_values(
    by="Rating text",
    ascending = False
)
```

Top Not Rated Cities

In [789]:
```
not_rated_rating = not_rated_rating.head()
```

In [790]:
```
not_rated_rating = not_rated_rating.reset_index()
```

In [791]:
```
not_rated_rating
```

Out[791]:

	City	Rating text
0	New Delhi	1408
1	Noida	370
2	Gurgaon	224
3	Faridabad	99
4	Ghaziabad	2

Ratings Categories

In [792]:
```
category = {
    "Dark Green":"Excellent",
    "Green":"Very Good",
    "Yellow":"Good",
    "Orange":"Average",
    "Red":"Poor",
    "White":"Not rated"
}
```

In [793]:
```
rating_group = rating.groupby("Rating color").count()
```

In [794]:
```
rating_group = rating_group.reset_index()
```

In [795]:
```
rating_group["Rating category"] = rating_group["Rating color"].map(
```

In [796]:
```
overall_ratings = rating_group
```

In [797]:
```
overall_ratings
```

Out[797]:

	Rating color	City	Aggregate rating	Rating text	Rating category
0	Dark Green	116	116	116	Excellent
1	Green	692	692	692	Very Good
2	Orange	3671	3671	3671	Average
3	Red	180	180	180	Poor
4	White	2103	2103	2103	Not rated
5	Yellow	1847	1847	1847	Good

In [798]:
```
overall_rating = overall_ratings[["Rating category","Aggregate rati
```

In [799]:
```
overall_rating
```

Out[799]:

	Rating category	Aggregate rating
0	Excellent	116
1	Very Good	692
2	Average	3671
3	Poor	180
4	Not rated	2103
5	Good	1847

In [ ]:
```

```

Ranking Cities Across India with Rating Categories

In [800]:
```
rating_categories_by_cities = reviews_from_india.groupby(["City","R
```

In [801]:
```
rating_categories_by_cities.reset_index(inplace=True)
```

In [802]:
```
ranked_cities_by_ratings = rating_categories_by_cities.sort_values(
    by = ["Excellent","Very Good","Good","Average","Poor"],
    ascending = [False,False,False,False,False]
)
```

Cities With Rankings

In [803]:
```
ranked_cities_by_ratings
```

Out[803]:

Rating text	City	Average	Excellent	Good	Not rated	Poor	Very Good
31	New Delhi	2489	28	1128	1408	97	300
15	Gurgaon	504	12	257	224	34	83
5	Bangalore	0	9	2	0	0	9
9	Chennai	0	6	2	0	0	12
17	Hyderabad	0	6	2	0	0	10
14	Goa	0	6	5	0	0	9
22	Kolkata	0	5	4	0	0	11
36	Pune	0	4	2	0	0	14
23	Lucknow	0	4	4	0	0	13
27	Mumbai	1	4	7	0	0	8
16	Guwahati	0	3	3	0	0	15
1	Ahmedabad	0	3	5	0	0	13
19	Jaipur	1	3	4	0	0	12
10	Coimbatore	0	3	6	0	0	11
32	Noida	448	2	173	370	45	27
21	Kochi	0	2	6	0	0	12
8	Chandigarh	2	2	2	0	0	12
29	Nagpur	1	2	6	0	1	10
7	Bhubaneshwar	0	2	10	0	0	9
40	Vadodara	0	2	9	0	0	9
42	Vizag	0	2	10	0	0	8
11	Dehradun	0	1	9	0	0	10
0	Agra	1	1	9	0	0	9
6	Bhopal	2	1	8	0	0	9
24	Ludhiana	0	1	11	0	0	8
12	Faridabad	123	1	22	99	2	3
38	Secunderabad	0	1	0	0	0	1
18	Indore	1	0	7	0	0	12
39	Surat	1	0	9	0	0	10
20	Kanpur	3	0	10	0	0	7
3	Amritsar	5	0	12	0	0	4
28	Mysore	2	0	15	0	0	3
35	Puducherry	2	0	15	0	0	3
25	Mangalore	0	0	18	0	0	2
41	Varanasi	8	0	11	0	0	1

37	Ranchi	11	0	8	0	0	1
26	Mohali	0	0	0	0	0	1
33	Panchkula	0	0	0	0	0	1
30	Nashik	8	0	12	0	0	0
34	Patna	9	0	11	0	0	0
2	Allahabad	15	0	5	0	0	0
13	Ghaziabad	18	0	4	2	1	0
4	Aurangabad	16	0	4	0	0	0

Cities With above average ratings

In [804]:
```
cities_with_good_ratings = ranked_cities_by_ratings[["City","Good",
```

Top Five Cities With Above Average Ratings

In [805]:
```
cities_with_good_ratings["Total"] = cities_with_good_ratings.sum(ax
```

In [806]:
```
cities_with_good_ratings = cities_with_good_ratings.head(5)
cities_with_good_ratings["Total"] = cities_with_good_ratings.sum(ax
cities_with_good_ratings["Average"] = cities_with_good_ratings.medi
```

In [807]:
```
cities_with_good_ratings = cities_with_good_ratings.head()
```

In [808]:
```
cities_with_good_ratings
```

Out[808]:

Rating text	City	Good	Very Good	Excellent	Total	Average
31	New Delhi	1128	300	28	2912	714.0
15	Gurgaon	257	83	12	704	170.0
5	Bangalore	2	9	9	40	9.0
9	Chennai	2	12	6	40	9.0
17	Hyderabad	2	10	6	36	8.0

Cities with Zero Poor Ratings

In [809]:
```
cities_with_poor_ratings = ranked_cities_by_ratings[["City","Poor"]
```

In [810]:
```
cities_with_poor_ratings
```

Out[810]:

Rating text	City	Poor
31	New Delhi	97

15	Gurgaon	34
5	Bangalore	0
9	Chennai	0
17	Hyderabad	0
14	Goa	0
22	Kolkata	0
36	Pune	0
23	Lucknow	0
27	Mumbai	0
16	Guwahati	0
1	Ahmedabad	0
19	Jaipur	0
10	Coimbatore	0
32	Noida	45
21	Kochi	0
8	Chandigarh	0
29	Nagpur	1
7	Bhubaneshwar	0
40	Vadodara	0
42	Vizag	0
11	Dehradun	0
0	Agra	0
6	Bhopal	0
24	Ludhiana	0
12	Faridabad	2
38	Secunderabad	0
18	Indore	0
39	Surat	0
20	Kanpur	0
3	Amritsar	0
28	Mysore	0
35	Puducherry	0
25	Mangalore	0
41	Varanasi	0
37	Ranchi	0
26	Mohali	0

33	Panchkula	0
30	Nashik	0
34	Patna	0
2	Allahabad	0
13	Ghaziabad	1
4	Aurangabad	0

In [811]:
```
zero_ratings = cities_with_poor_ratings[cities_with_poor_ratings["P
```

In [812]:
```
zero_ratings = zero_ratings.sort_values(
    by="Poor",
    ascending = True
)
```

In [813]:
```
zero_ratings = zero_ratings.groupby("City").count().reset_index()
```

In [911]:
```
zero_ratings = zero_ratings.head(5)
```

Top Not Rated Cities

In [815]:
```
cities_with_not_rated_ratings = ranked_cities_by_ratings[["City","N
```

In [816]:
```
cities_with_not_rated_ratings
```

Out[816]:

Rating text	City	Not rated
31	New Delhi	1408
15	Gurgaon	224
5	Bangalore	0
9	Chennai	0
17	Hyderabad	0
14	Goa	0
22	Kolkata	0
36	Pune	0
23	Lucknow	0
27	Mumbai	0
16	Guwahati	0
1	Ahmedabad	0
19	Jaipur	0
10	Coimbatore	0

32	Noida	370
21	Kochi	0
8	Chandigarh	0
29	Nagpur	0
7	Bhubaneshwar	0
40	Vadodara	0
42	Vizag	0
11	Dehradun	0
0	Agra	0
6	Bhopal	0
24	Ludhiana	0
12	Faridabad	99
38	Secunderabad	0
18	Indore	0
39	Surat	0
20	Kanpur	0
3	Amritsar	0
28	Mysore	0
35	Puducherry	0
25	Mangalore	0
41	Varanasi	0
37	Ranchi	0
26	Mohali	0
33	Panchkula	0
30	Nashik	0
34	Patna	0
2	Allahabad	0
13	Ghaziabad	2
4	Aurangabad	0

In [817]:

```
not_rated_cities = cities_with_not_rated_ratings.sort_values(
    by="Not rated",
    ascending = False
)
```

In [818]:
```
not_rated_cities.head()
```

Out[818]:

Rating text	City	Not rated
31	New Delhi	1408
32	Noida	370
15	Gurgaon	224
12	Faridabad	99
13	Ghaziabad	2

In [819]:
```
not_rated_cities = not_rated_cities.head(5)
```

In [820]:
```
not_rated_cities
```

Out[820]:

Rating text	City	Not rated
31	New Delhi	1408
32	Noida	370
15	Gurgaon	224
12	Faridabad	99
13	Ghaziabad	2

Cities With Zero Not Rated Cities

In [821]:
```
zero_not_rated = ranked_cities_by_ratings[["City","Not rated"]]
```

In [822]:
```
zero_not_rated = zero_not_rated[zero_not_rated["Not rated"] == 0]
```

In [823]:
```
zero_not_rated
```

Out[823]:

Rating text	City	Not rated
5	Bangalore	0
9	Chennai	0
17	Hyderabad	0
14	Goa	0
22	Kolkata	0
36	Pune	0
23	Lucknow	0
27	Mumbai	0
16	Guwahati	0

1	Ahmedabad	0
19	Jaipur	0
10	Coimbatore	0
21	Kochi	0
8	Chandigarh	0
29	Nagpur	0
7	Bhubaneshwar	0
40	Vadodara	0
42	Vizag	0
11	Dehradun	0
0	Agra	0
6	Bhopal	0
24	Ludhiana	0
38	Secunderabad	0
18	Indore	0
39	Surat	0
20	Kanpur	0
3	Amritsar	0
28	Mysore	0
35	Puducherry	0
25	Mangalore	0
41	Varanasi	0
37	Ranchi	0
26	Mohali	0
33	Panchkula	0
30	Nashik	0
34	Patna	0
2	Allahabad	0
4	Aurangabad	0

In [824]:
```
top_zero_not_rated_cities = zero_not_rated.head(5)
```

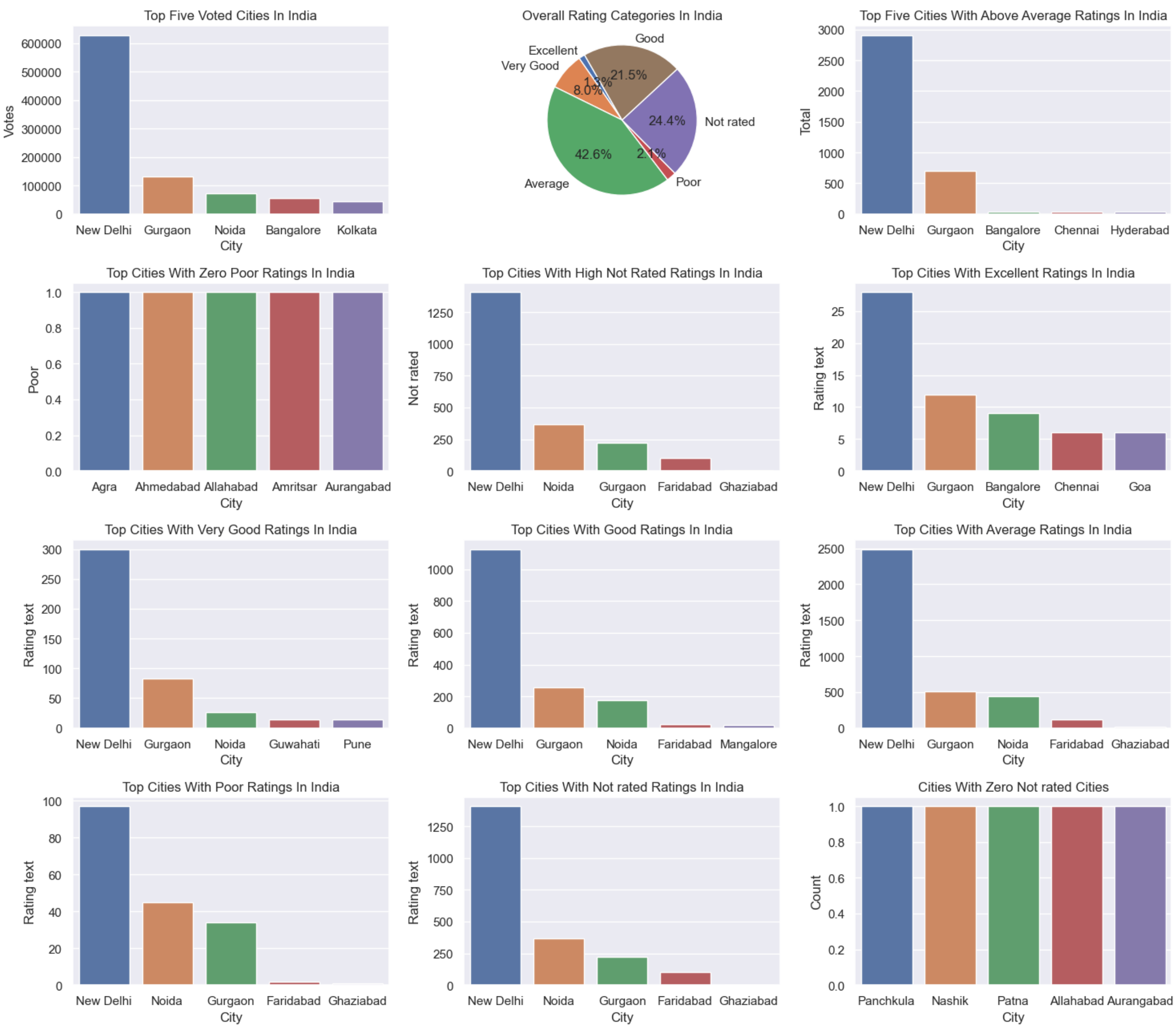
In [825]:
```
top_zero_not_rated_cities
```

Out[825]:

Rating text	City	Not rated
5	Bangalore	0
9	Chennai	0
17	Hyderabad	0
14	Goa	0
22	Kolkata	0

In [912]:
```
sns.set(style="darkgrid")
plt.figure(figsize=(15,13))
plt.subplot(4,3,1)
sns.barplot(x="City", y="Votes", data=top_five_voted_indian_cities)
plt.title("Top Five Voted Cities In India")
plt.subplot(4,3,2)
plt.pie(
    overall_rating['Aggregate rating'],
    labels=overall_rating['Rating category'],
    autopct='%1.1f%%',
    startangle=120
)
plt.title("Overall Rating Categories In India")
plt.subplot(4,3,3)
sns.barplot(x="City", y="Total", data=cities_with_good_ratings)
plt.title("Top Five Cities With Above Average Ratings In India")
plt.subplot(4,3,4)
sns.barplot(x="City", y="Poor", data=zero_ratings)
plt.title("Top Cities With Zero Poor Ratings In India")
plt.subplot(4,3,5)
sns.barplot(x="City", y="Not rated", data=not_rated_cities)
plt.title("Top Cities With High Not Rated Ratings In India")
plt.subplot(4,3,6)
sns.barplot(x="City", y="Rating text", data=top_five_excellent_rati
plt.title("Top Cities With Excellent Ratings In India")
plt.subplot(4,3,7)
sns.barplot(x="City", y="Rating text", data=very_good_rating)
plt.title("Top Cities With Very Good Ratings In India")
plt.subplot(4,3,8)
sns.barplot(x="City", y="Rating text", data=good_rating)
plt.title("Top Cities With Good Ratings In India")
plt.subplot(4,3,9)
sns.barplot(x="City", y="Rating text", data=average_rating)
plt.title("Top Cities With Average Ratings In India")
plt.subplot(4,3,10)
sns.barplot(x="City", y="Rating text", data=poor_rating)
plt.title("Top Cities With Poor Ratings In India")
plt.subplot(4,3,11)
sns.barplot(x="City", y="Rating text", data=not_rated_rating)
plt.title("Top Cities With Not rated Ratings In India")
plt.subplot(4,3,12)
sns.countplot(x='City', data=zero_not_rated.tail())
plt.xlabel("City")
```

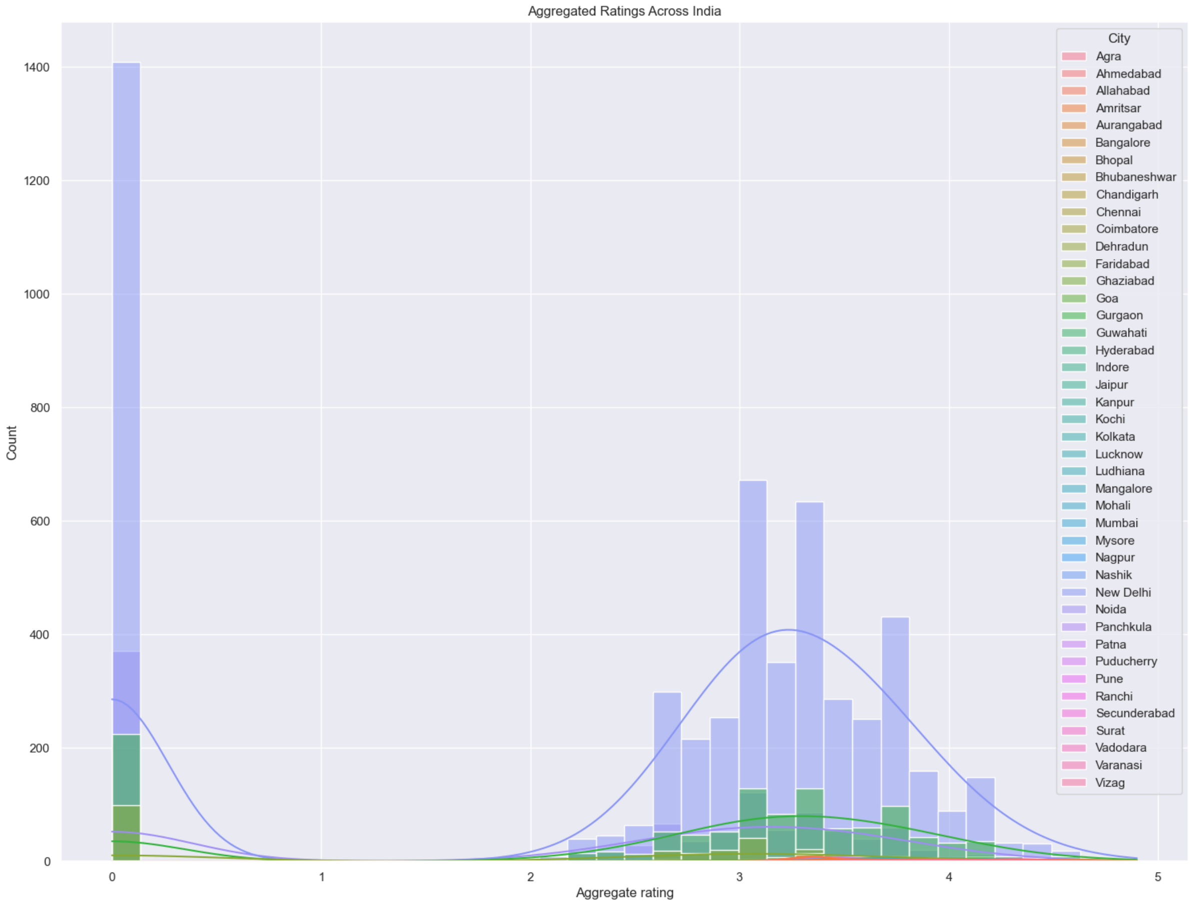
```python
plt.ylabel("Count")
plt.title("Cities With Zero Not rated Cities")
plt.tight_layout()
plt.show()
```



Observation

- The Overall performance of indian restaurants are Average **(42%)**
- Top five cities with highest votings from the customers are **New Delhi**,**Gurgaon**,**Noida**,**Guwahati**,**Pune**
- Top five above average rating cities are **New Delhi**,**Gurgoan**,**Banglore**,**Chennai**,**Hyderabad**
- Top five excellent rating cities are **New Delhi**,**Gurgoan**,**Banglore**,**Chennai**,**Goa**
- Top five very good rating cities are **New Delhi**,**Gurgoan**,**Noida**,**Guwahati**,**Pune**
- Top five good rating cities are **New Delhi**,**Gurgoan**,**Noida**,**Faridabad**,**Manglore**
- Top five average rating cities are **New Delhi**,**Gurgoan**,**Noida**,**Faridabad**,**Ghaziabad**
- Top five poor rated cities are **New Delhi**,**Noida**,**Gurgoan**,**Faridabad**,**Ghaziabad**
- Top five not rated cities are **New Delhi**,**Noida**,**Gurgoan**,**Faridabad**,**Ghaziabad**
- Top five cities with zero poor ratings are **Banglore**,**Indore**,**Kanpur**,**Secunderabad**,**Surat**
- Top five cities with zero not rated are **Panchkula**,**Nashik**,**Patna**,**Allahabad**,**Aurangabad**

In [827]:
```
sns.set(style="darkgrid")
plt.figure(figsize=(40,30))
plt.subplot(2,2,1)
sns.histplot(
    x="Aggregate rating",
    data=reviews_from_india,
    hue="City",
    kde=True
)
plt.title("Aggregated Ratings Across India")
plt.show()
```



Ratings Across Restaurants,Cusines and cities

In [828]:
```
reviews_from_india.head()
```

Out[828]:

	Restaurant Name	City	Cuisines	Average Cost for two	Currency	Has Table booking	Has Online delivery	Is delivering now	Swit ord me
624	Jahanpanah	Agra	North Indian, Mughlai	850	Indian Rupees(Rs.)	No	No	No	N
625	Rangrezz Restaurant	Agra	North Indian, Mughlai	700	Indian Rupees(Rs.)	No	No	No	N
626	Time2Eat - Mama Chicken	Agra	North Indian	500	Indian Rupees(Rs.)	No	No	No	N
627	Chokho Jeeman Marwari Jain Bhojanalya	Agra	Rajasthani	400	Indian Rupees(Rs.)	No	No	No	N
628	Pinch Of Spice	Agra	North Indian, Chinese, Mughlai	1000	Indian Rupees(Rs.)	No	No	No	N

In [829]:
```
indian_restaurants_data = reviews_from_india[["Restaurant Name","Ci
```

In [830]:
```
indian_restaurants_data.head()
```

Out[830]:

	Restaurant Name	City	Cuisines	Average Cost for two	Price range	Has Table booking	Has Online delivery	Is delivering now	Switch to order menu	R
624	Jahanpanah	Agra	North Indian, Mughlai	850	3	No	No	No	No	
625	Rangrezz Restaurant	Agra	North Indian, Mughlai	700	2	No	No	No	No	
626	Time2Eat - Mama Chicken	Agra	North Indian	500	2	No	No	No	No	
627	Chokho Jeeman Marwari Jain Bhojanalya	Agra	Rajasthani	400	2	No	No	No	No	
628	Pinch Of Spice	Agra	North Indian, Chinese, Mughlai	1000	3	No	No	No	No	

How many Restaurants has Online Delivery in India ?

In [831]:
```
online_delivery = indian_restaurants_data["Has Online delivery"].va
```

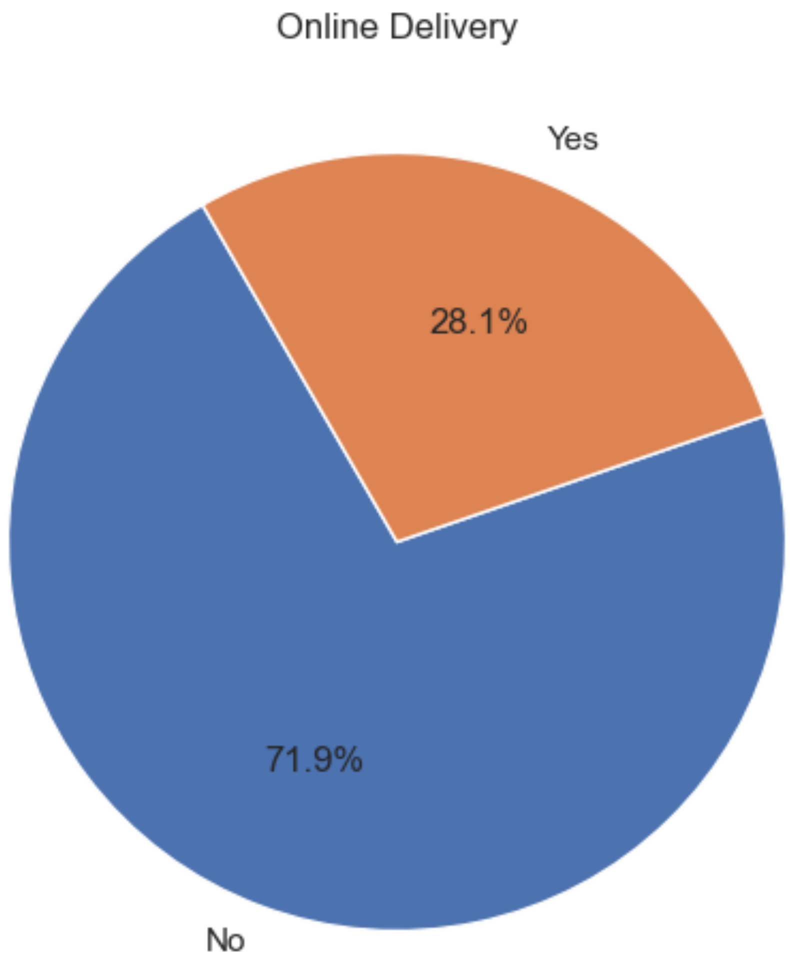
In [832]:
```
online_delivery = online_delivery.rename(
    columns = {"index":"Online Delivery","Has Online delivery":"Cou
)
```

In [833]:
```
online_delivery
```

Out[833]:

	Online Delivery	Counts
0	No	6192
1	Yes	2417

In [834]:
```
plt.figure(figsize=(16,13))
plt.subplot(2,2,1)
plt.pie(
    online_delivery["Counts"],
    labels = online_delivery["Online Delivery"],
    autopct='%1.1f%%',
    startangle=120
)
plt.title("Online Delivery")
plt.show()
```



Observation

- In **India**, only **28.1%** of the restaurants offer **online delivery**, while the remaining **71.9%** do not provide this service.

How many Restaurants has table booking in india ?

In [835]:
```
table_booking = indian_restaurants_data["Has Table booking"].value_
```

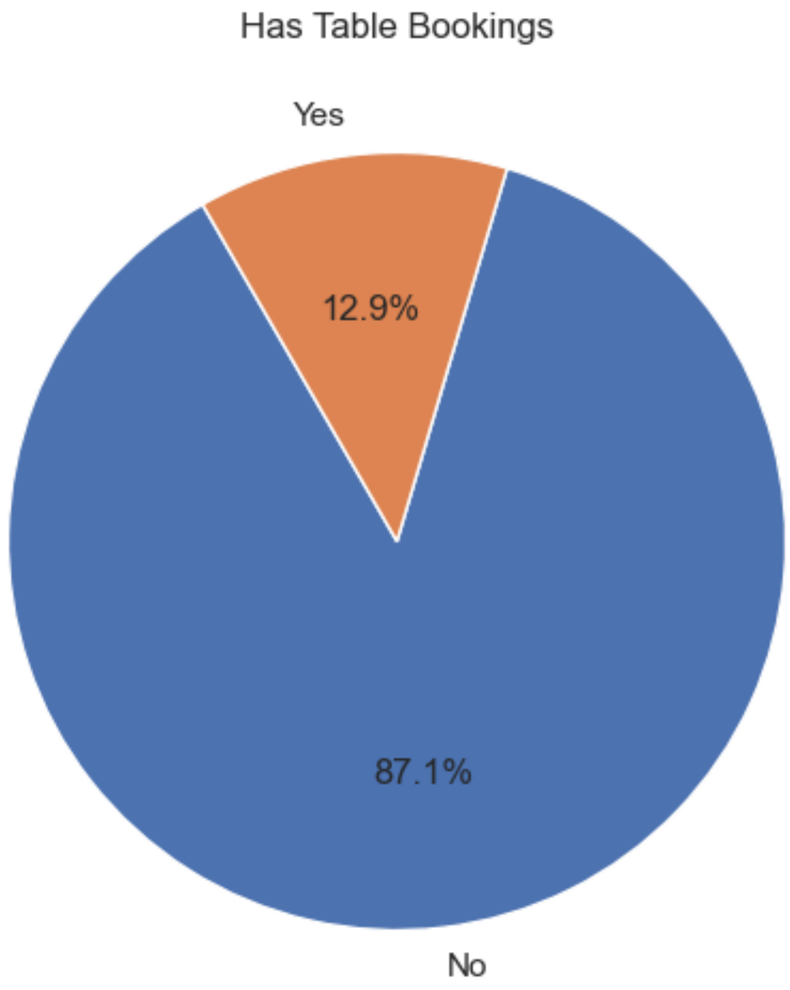
In [836]:
```
table_booking = table_booking.rename(
    columns = {"index":"Table Booking","Has Table booking":"Counts"
)
```

In [837]:
```
table_booking
```

Out[837]:

	Table Booking	Counts
0	No	7498
1	Yes	1111

In [838]:
```
plt.figure(figsize=(16,13))
plt.subplot(2,2,1)
plt.pie(
    table_booking["Counts"],
    labels = table_booking["Table Booking"],
    autopct='%1.1f%%',
    startangle=120
)
plt.title("Has Table Bookings")
plt.show()
```



Observation :

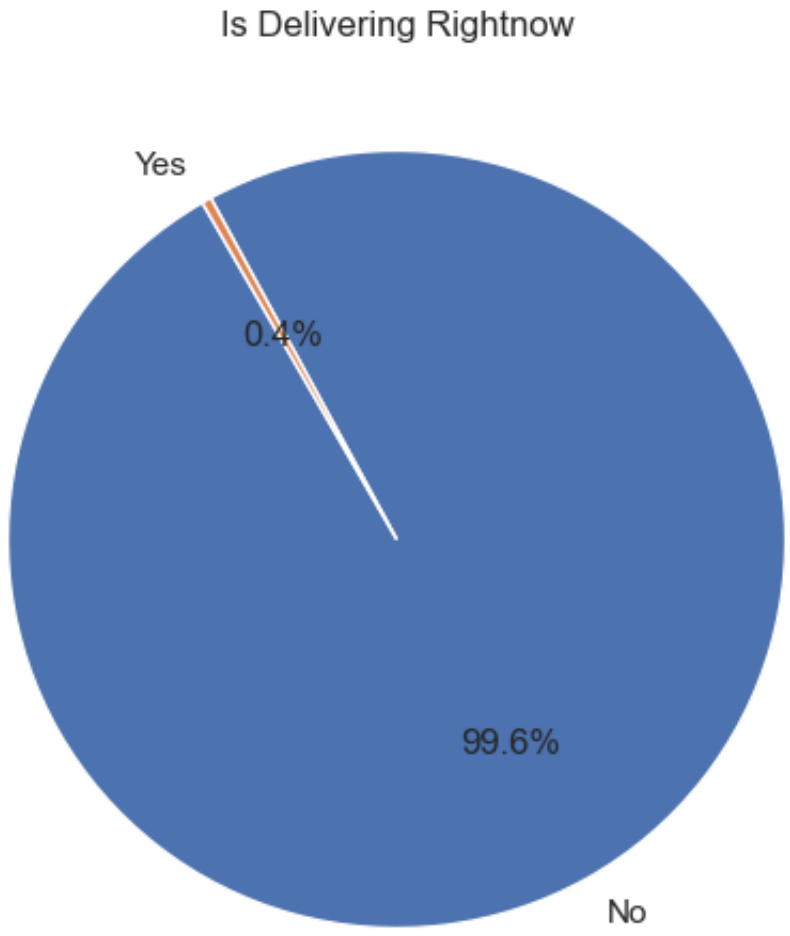
- In India, only **12.9%** of the restaurants has a table bookings, while the remaining 87.1% do not provide this service.

How many restaurants are delivering right now

In [839]:
```
delivering = indian_restaurants_data["Is delivering now"].value_cou
```

In [840]:
```
delivering = delivering.rename(
    columns = {"index":"Delivering","Is delivering now":"Counts"}
)
```

In [841]:
```
plt.figure(figsize=(16,13))
plt.subplot(2,2,1)
plt.pie(
    delivering["Counts"],
    labels = delivering["Delivering"],
    autopct='%1.1f%%',
    startangle=120
)
plt.title("Is Delivering Rightnow")
plt.show()
```



Observation:

- In india, only **0.4%** of the restaurants are delivering right now, while the rest **99.6%** of the restaurants are not delivering right now

Cuisines Ratings In India

In [842]:
```
cuisines_count_india = indian_restaurants_data["Cuisines"].value_co
```

Top Rated Cuisines Distribution

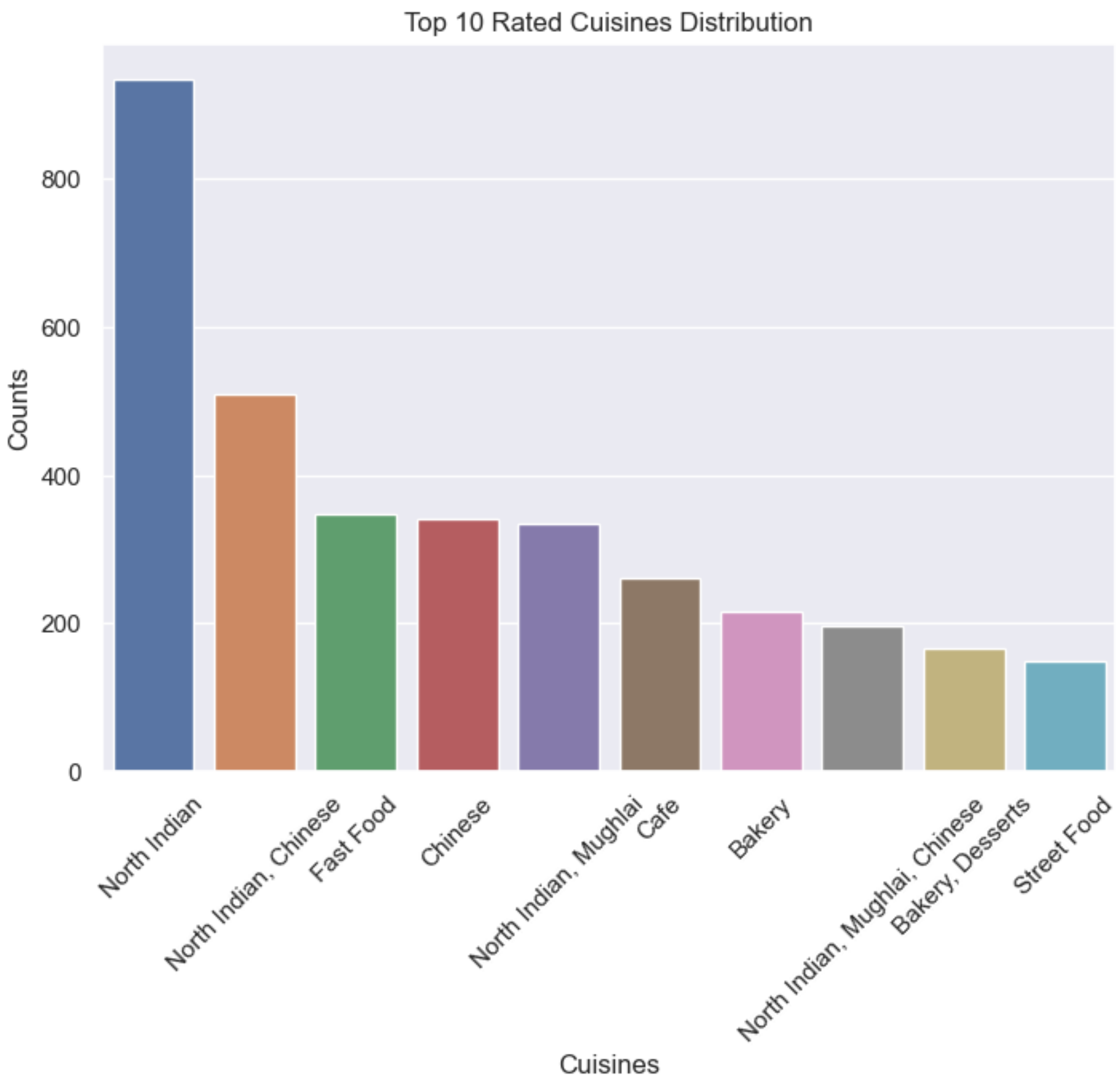
In [843]:
```
top_10_rated_cuisines_in_india = cuisines_count_india.head(10)
```

In [844]:
```
top_10_rated_cuisines_in_india
```

Out[844]:

	index	Cuisines
0	North Indian	935
1	North Indian, Chinese	510
2	Fast Food	347
3	Chinese	340
4	North Indian, Mughlai	334
5	Cafe	262
6	Bakery	216
7	North Indian, Mughlai, Chinese	197
8	Bakery, Desserts	167
9	Street Food	149

```
In [845]: plt.figure(figsize=(18,13))
plt.subplot(2,2,1)
cs = sns.barplot(
    x = top_10_rated_cuisines_in_india["index"],
    y = top_10_rated_cuisines_in_india["Cuisines"]
)
cs.set_xticklabels(cs.get_xticklabels(),rotation=45);
plt.xlabel("Cuisines")
plt.ylabel("Counts")
plt.title("Top 10 Rated Cuisines Distribution")
plt.show()
```



Excellent Rating Restaurants

```
In [846]: restaurants_with_excellent_ratings = indian_restaurants_data[indian
```

In [847]:
```python
restaurants_with_excellent_ratings
```

Out[847]:

	Restaurant Name	City	Cuisines	Average Cost for two	Price range	Has Table booking	Has Online delivery	delive
637	Sheroes Hangout	Agra	Cafe, North Indian, Chinese	0	1	No	No	
646	Huber & Holly	Ahmedabad	Ice Cream, Desserts, Continental	300	1	No	Yes	
653	Cryo Lab	Ahmedabad	Desserts, Ice Cream	350	2	No	No	
660	Nini's Kitchen	Ahmedabad	North Indian, Continental, Beverages, Italian,...	950	3	No	Yes	
727	The Fatty Bao - Asian Gastro Bar	Bangalore	Asian	2400	4	Yes	Yes	
...	...	...	...	...	...	...	...	
9195	Coffee Cup	Secunderabad	Cafe, Continental	800	2	Yes	No	
9216	22nd Parallel	Vadodara	South Indian	400	2	No	No	
9228	La Quello - Mediterranean Kitchen	Vadodara	Mediterranean, Italian	1300	3	No	No	
9256	Pizza Hut	Vizag	Pizza, Fast Food	600	2	No	No	
9262	Barbeque Nation	Vizag	North Indian, Chinese, Mediterranean	1600	4	No	No	

116 rows × 10 columns

In [848]:
```python
excellent_rating_restaurants = restaurants_with_excellent_ratings.g
```

In [849]:
```python
excellent_rating_restaurants = excellent_rating_restaurants.reset_i
```

In [850]:
```python
excellent_rating_restaurants = excellent_rating_restaurants.sort_va
    by="Rating text",
    ascending = False
)
```

Top Excellent Rating Restaurants

In [851]:
```python
top_excellent_rating_restaurants = excellent_rating_restaurants.hea
```

In [852]:
```
top_excellent_rating_restaurants
```
Out[852]:

	Restaurant Name	Rating text
6	Barbeque Nation	10
1	AB's - Absolute Barbecues	4
20	Chili's	4
88	Twigly	2
61	Onesta	2

Top Excellent Rating Cuisines

In [853]:
```
excellent_rating_cuisines = restaurants_with_excellent_ratings["Cui
```
In [854]:
```
excellent_rating_cuisines = excellent_rating_cuisines.head(10)
```
In [855]:
```
excellent_rating_cuisines = excellent_rating_cuisines.rename(
    columns = {"index":"Cuisines","Cuisines":"Counts"}
)
```
In [856]:
```
excellent_rating_cuisines
```
Out[856]:

	Cuisines	Counts
0	North Indian	7
1	Ice Cream	5
2	North Indian, Mughlai	4
3	Modern Indian	3
4	Bakery, Desserts	3
5	Cafe	3
6	Mexican, American, Tex-Mex, Burger	3
7	European, Mediterranean, North Indian	3
8	North Indian, Chinese	3
9	Fast Food	2

Very Good Rating Restaurants

In [857]:
```
restaurants_with_very_good_ratings = indian_restaurants_data[indian
```
In [858]:
```
very_good_rating_restaurants = restaurants_with_very_good_ratings.g
```

In [859]:
```python
very_good_rating_restaurants = very_good_rating_restaurants.reset_i
```

In [860]:
```python
very_good_rating_restaurants = very_good_rating_restaurants.sort_va
    by="Rating text",
    ascending = False
)
```

Top Very Good Ratings

In [861]:
```python
top_very_good_rating_restaurants = very_good_rating_restaurants.hea
```

In [862]:
```python
top_very_good_rating_restaurants
```

Out[862]:

	Restaurant Name	Rating text
41	Barbeque Nation	12
409	Pizza Hut	6
201	Farzi Cafe	5
340	Mocha	4
414	Punjab Grill	4

Top Very Good Rating Cuisines

In [863]:
```python
very_good_rating_cuisines = restaurants_with_very_good_ratings["Cui
```

In [864]:
```python
very_good_rating_cuisines = very_good_rating_cuisines.head()
```

In [865]:
```python
very_good_rating_cuisines
```

Out[865]:

	index	Cuisines
0	North Indian, Mughlai	25
1	North Indian, Chinese	19
2	Cafe	18
3	North Indian	16
4	Chinese	14

In [866]:
```python
very_good_rating_cuisines = very_good_rating_cuisines.rename(
    columns = {"index":"Cuisines","Cuisines":"Counts"}
)
```

In [867]:
```
very_good_rating_cuisines
```
Out[867]:

	Cuisines	Counts
0	North Indian, Mughlai	25
1	North Indian, Chinese	19
2	Cafe	18
3	North Indian	16
4	Chinese	14

In [ ]:

In [ ]:

Good Rating Restaurants

In [868]:
```
restaurants_with_good_ratings = indian_restaurants_data[indian_rest
```
In [869]:
```
good_restaurant_count = restaurants_with_good_ratings.shape[0]
```
In [870]:
```
good_restaurant_count
```
Out[870]: 1847

In [871]:
```
good_rating_restaurants = restaurants_with_good_ratings.groupby("Re
```
In [872]:
```
good_rating_restaurants = good_rating_restaurants.reset_index()
```
In [873]:
```
good_rating_restaurants = good_rating_restaurants.sort_values(
    by="Rating text",
    ascending = False
)
```

Top Good Ratings

In [874]:
```
top_good_rating_restaurants = good_rating_restaurants.head()
```

In [875]:
```
top_good_rating_restaurants
```

Out[875]:

	Restaurant Name	Rating text
806	McDonald's	20
433	Dunkin' Donuts	16
556	Haldiram's	15
1184	Starbucks	15
682	Keventers	13

Top Good Rating Cuisines

In [876]:
```
good_rating_cuisines = restaurants_with_good_ratings["Cuisines"].va
```

In [877]:
```
good_rating_cuisines = good_rating_cuisines.head(10)
```

In [878]:
```
good_rating_cuisines = good_rating_cuisines.rename(
    columns = {"index":"Cuisines","Cuisines":"Counts"}
)
```

In [879]:
```
good_rating_cuisines
```

Out[879]:

	Cuisines	Counts
0	North Indian	92
1	Cafe	75
2	North Indian, Mughlai	71
3	Fast Food	56
4	North Indian, Chinese	55
5	Street Food	39
6	Chinese	38
7	Bakery, Desserts	31
8	Bakery, Desserts, Fast Food	31
9	North Indian, Mughlai, Chinese	29

Average Rating Restaurants

In [880]:
```
restaurants_with_average_ratings = indian_restaurants_data[indian_r
```

In [881]:
```
average_restaurant_count = restaurants_with_average_ratings.shape[0
```

In [882]:
```
average_restaurant_count
```
Out[882]:
```
3671
```

In [883]:
```
average_rating_restaurants = restaurants_with_average_ratings.group
```

In [884]:
```
average_rating_restaurants = average_rating_restaurants.reset_index
```

In [885]:
```
average_rating_restaurants = average_rating_restaurants.sort_values
    by="Rating text",
    ascending = False
)
```

Top Average Ratings Restaurants

In [886]:
```
top_average_rating_restaurants = average_rating_restaurants.head()
```

In [887]:
```
top_average_rating_restaurants
```
Out[887]:

	Restaurant Name	Rating text
430	Cafe Coffee Day	56
816	Domino's Pizza	53
2399	Subway	43
1030	Green Chick Chop	38
1543	McDonald's	27

Top Average Rating Cuisines

In [888]:
```
average_rating_cuisines = restaurants_with_average_ratings["Cuisine
```

In [889]:
```
average_rating_cuisines = average_rating_cuisines.head(10)
```

In [890]:
```
average_rating_cuisines = average_rating_cuisines.rename(
    columns = {"index":"Cuisines","Cuisines":"Counts"}
)
```

In [891]:
```
average_rating_cuisines
```
Out[891]:

	Cuisines	Counts
0	North Indian	366
1	North Indian, Chinese	303
2	North Indian, Mughlai	185
3	Chinese	150
4	Fast Food	150
5	Cafe	129
6	North Indian, Mughlai, Chinese	122
7	Bakery	96
8	Pizza, Fast Food	77
9	Bakery, Desserts	72

Poor Rating Restaurants

In [892]:
```
restaurants_with_poor_ratings = indian_restaurants_data[indian_rest
```
In [893]:
```
poor_restaurant_count = restaurants_with_poor_ratings.shape[0]
```
In [894]:
```
poor_restaurant_count
```
Out[894]:
```
180
```
In [895]:
```
poor_rating_restaurants = restaurants_with_poor_ratings.groupby("Re
```
In [896]:
```
poor_rating_restaurants = poor_rating_restaurants.reset_index()
```
In [897]:
```
poor_rating_restaurants = poor_rating_restaurants.sort_values(
    by="Rating text",
    ascending = False
)
```

Top Poor Rating Restaurants

In [898]:
```
top_poor_rating_restaurants = poor_rating_restaurants.head()
```

In [899]:
```
top_poor_rating_restaurants
```
Out[899]:

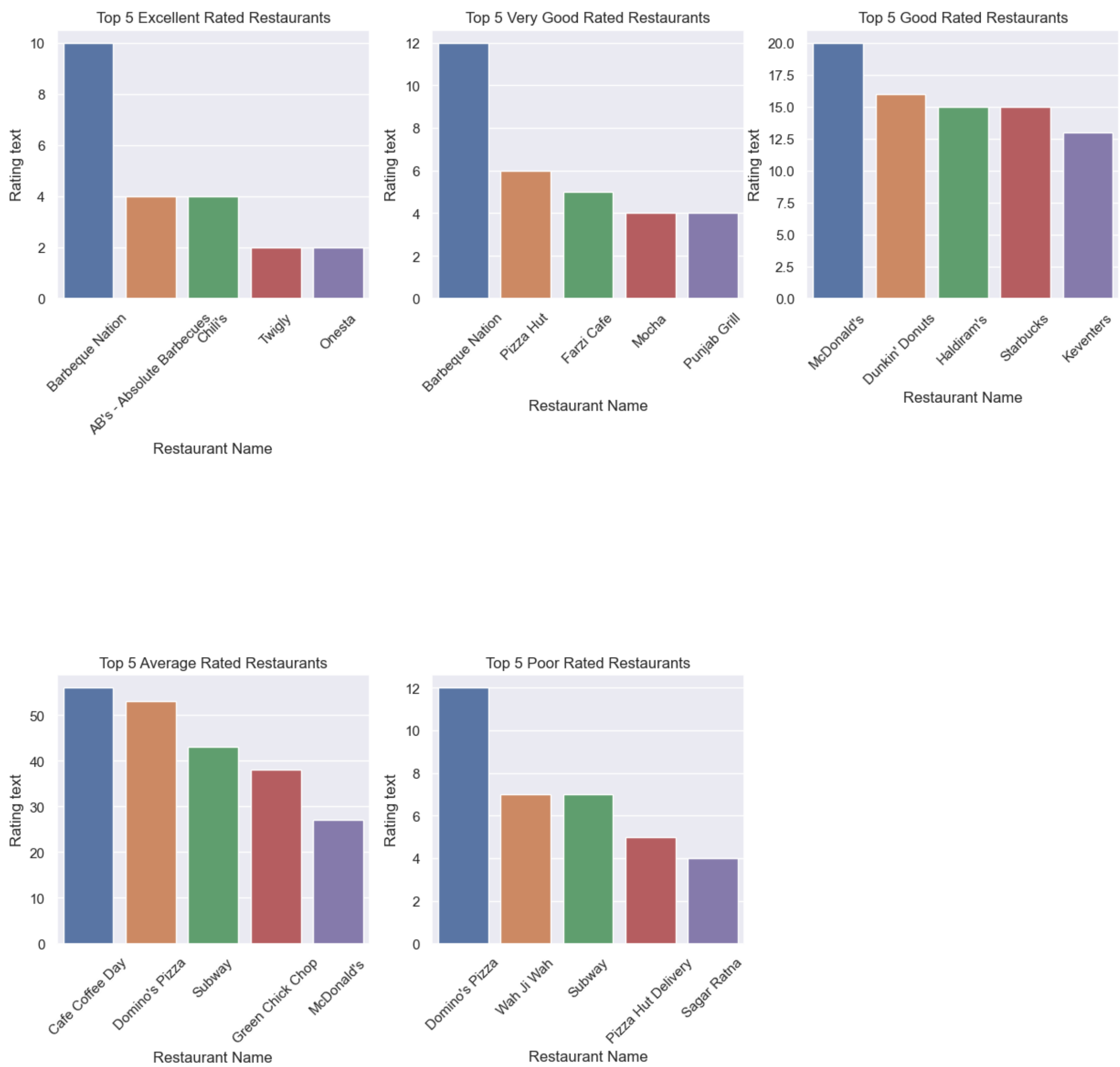
	Restaurant Name	Rating text
39	Domino's Pizza	12
121	Wah Ji Wah	7
102	Subway	7
86	Pizza Hut Delivery	5
94	Sagar Ratna	4

Top Poor Rating Cuisines

In [900]:
```
poor_rating_cuisines = restaurants_with_poor_ratings["Cuisines"].va
```
In [901]:
```
poor_rating_cuisines = poor_rating_cuisines.head(10)
```
In [902]:
```
poor_rating_cuisines = poor_rating_cuisines.rename(
    columns = {"index":"Cuisines","Cuisines":"Counts"}
)
```
In [903]:
```
poor_rating_cuisines
```
Out[903]:

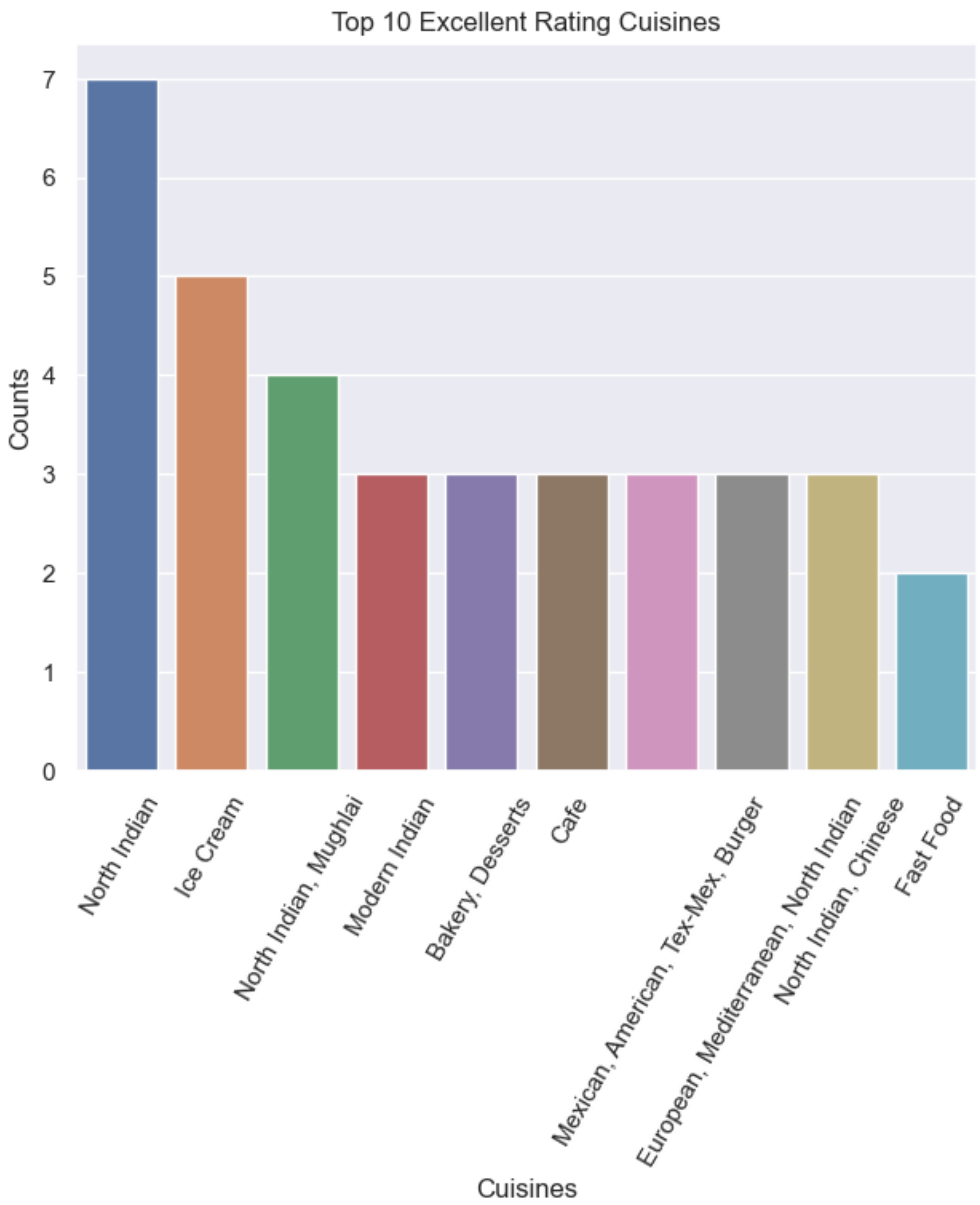
	Cuisines	Counts
0	North Indian, Chinese	17
1	Pizza, Fast Food	13
2	North Indian	11
3	North Indian, Mughlai, Chinese	9
4	North Indian, Mughlai	9
5	Chinese	8
6	American, Fast Food, Salad, Healthy Food	7
7	South Indian, North Indian, Chinese	5
8	Chinese, Fast Food	5
9	Biryani, North Indian	4

In [904]:
```
sns.set_style("darkgrid")
plt.figure(figsize=(15,13))
plt.subplot(3,3,1)
er = sns.barplot(x="Restaurant Name",y="Rating text",data=top_excel
plt.title("Top 5 Excellent Rated Restaurants")
er.set_xticklabels(er.get_xticklabels(),rotation=45);
plt.subplot(3,3,2)
vg = sns.barplot(x="Restaurant Name",y="Rating text",data=top_very_
plt.title("Top 5 Very Good Rated Restaurants")
vg.set_xticklabels(vg.get_xticklabels(),rotation=45);
plt.subplot(3,3,3)
g = sns.barplot(x="Restaurant Name",y="Rating text",data=top_good_r
plt.title("Top 5 Good Rated Restaurants")
g.set_xticklabels(g.get_xticklabels(),rotation=45);
plt.subplot(3,3,7)
av = sns.barplot(x="Restaurant Name",y="Rating text",data=top_avera
plt.title("Top 5 Average Rated Restaurants")
av.set_xticklabels(av.get_xticklabels(),rotation=45);
plt.subplot(3,3,8)
av = sns.barplot(x="Restaurant Name",y="Rating text",data=top_poor_
plt.title("Top 5 Poor Rated Restaurants")
av.set_xticklabels(av.get_xticklabels(),rotation=45);
plt.show()
```



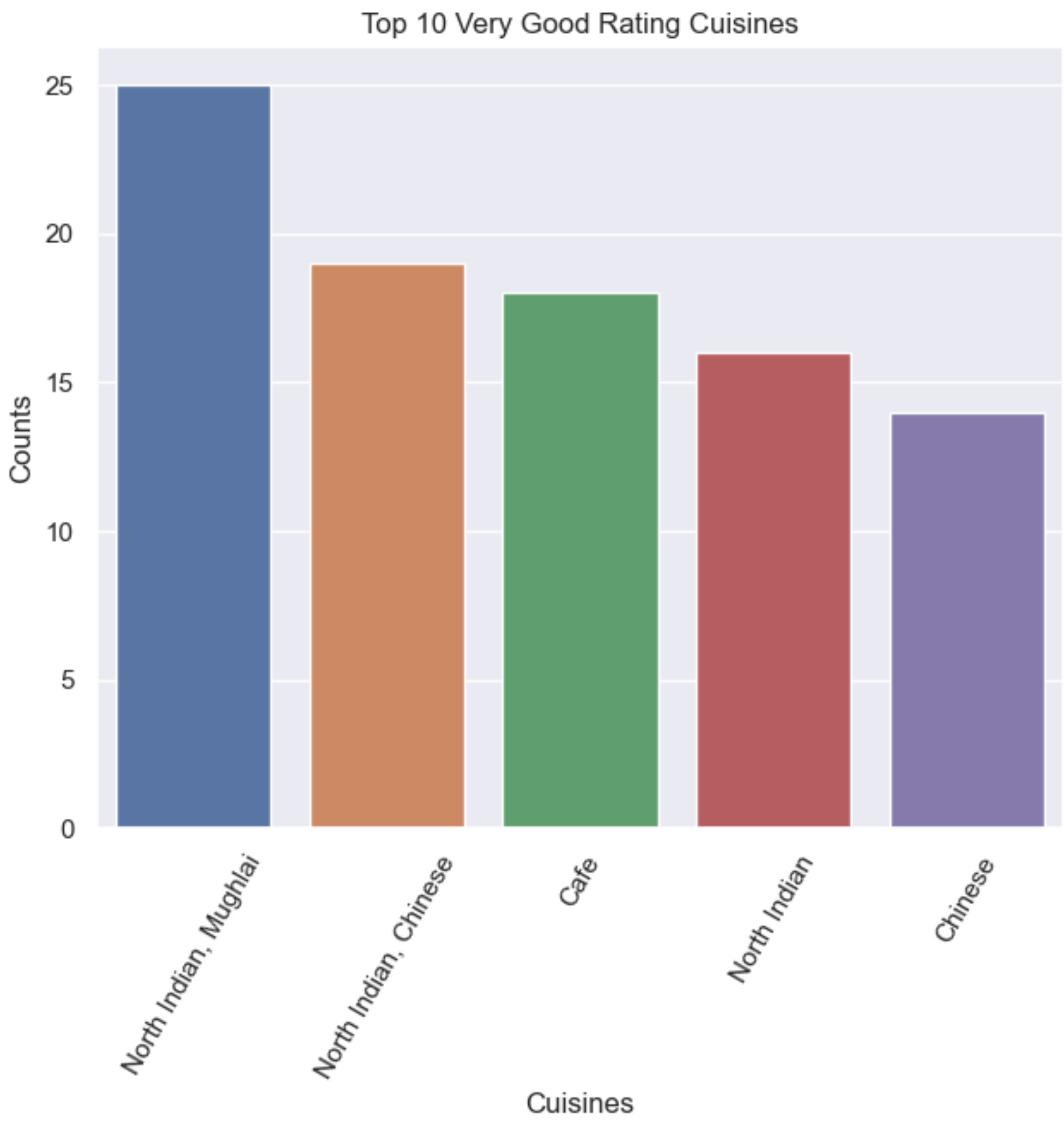
In [905]:
```
sns.set_style("darkgrid")
plt.figure(figsize=(16,13))
plt.subplot(2,2,1)
exc = sns.barplot(
    x = "Cuisines",
    y = "Counts",
    data = excellent_rating_cuisines
)
exc.set_xticklabels(exc.get_xticklabels(),rotation=60);
plt.title("Top 10 Excellent Rating Cuisines")
```

Out[905]:
```
Text(0.5, 1.0, 'Top 10 Excellent Rating Cuisines')
```



In [906]:
```
sns.set_style("darkgrid")
plt.figure(figsize=(16,13))
plt.subplot(2,2,1)
vgc = sns.barplot(
    x = "Cuisines",
    y = "Counts",
    data = very_good_rating_cuisines
)
vgc.set_xticklabels(vgc.get_xticklabels(),rotation=60);
plt.title("Top 10 Very Good Rating Cuisines")
```

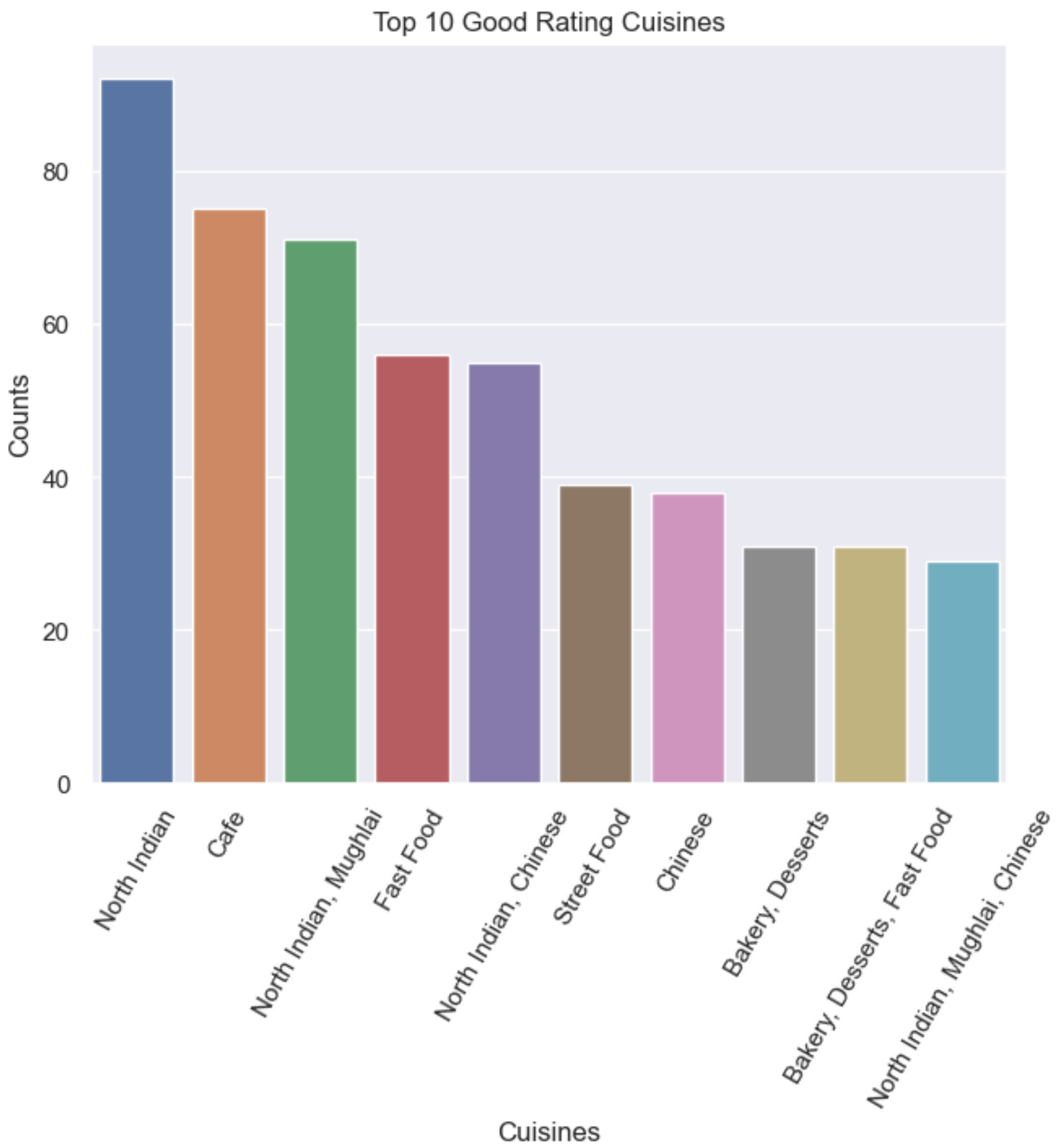
Out[906]:
```
Text(0.5, 1.0, 'Top 10 Very Good Rating Cuisines')
```



In [907]:

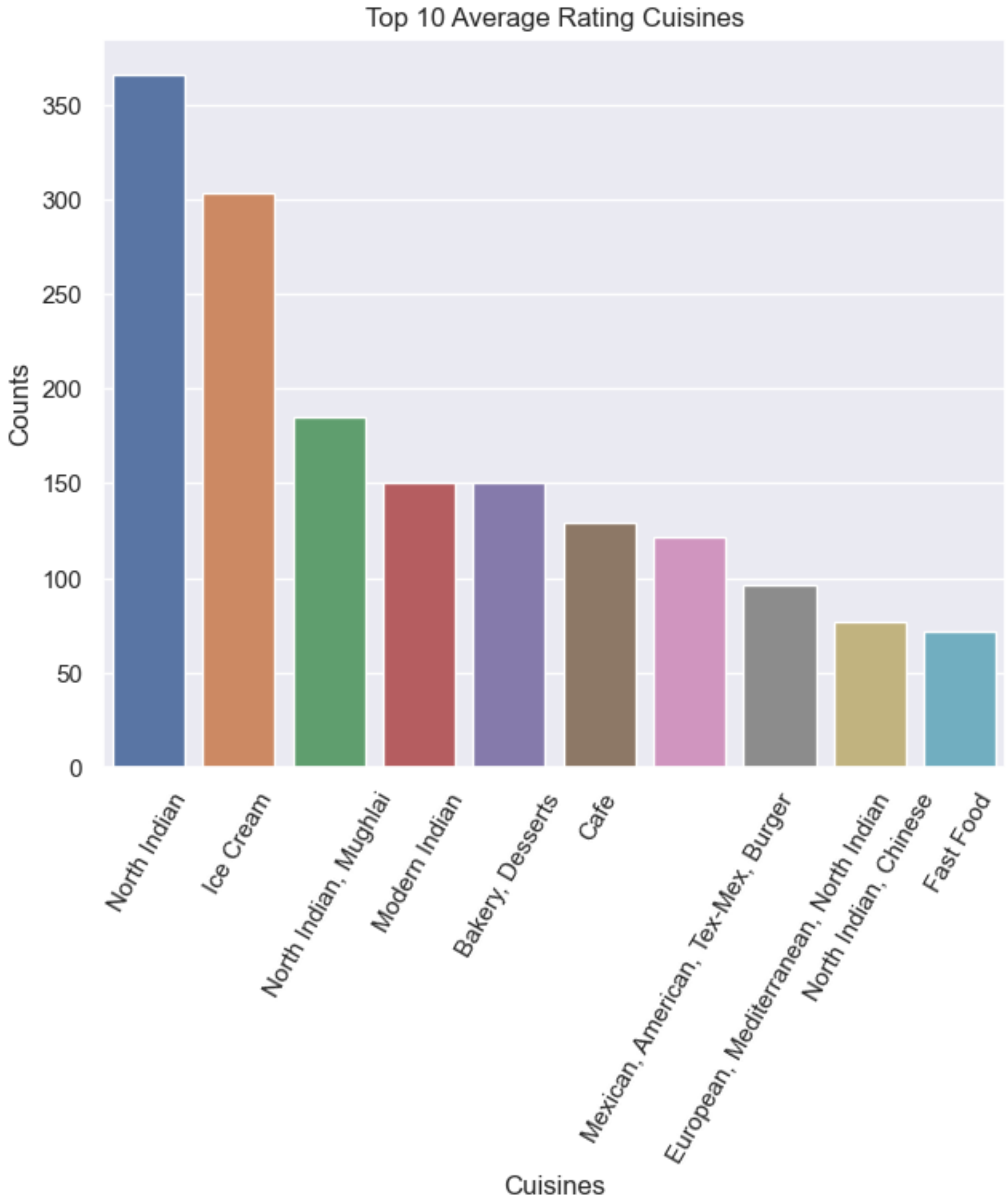
```python
sns.set_style("darkgrid")
plt.figure(figsize=(16,13))
plt.subplot(2,2,1)
gc = sns.barplot(
    x = "Cuisines",
    y = "Counts",
    data = good_rating_cuisines
)
gc.set_xticklabels(gc.get_xticklabels(),rotation=60);
plt.title("Top 10 Good Rating Cuisines")
```

Out[907]: Text(0.5, 1.0, 'Top 10 Good Rating Cuisines')



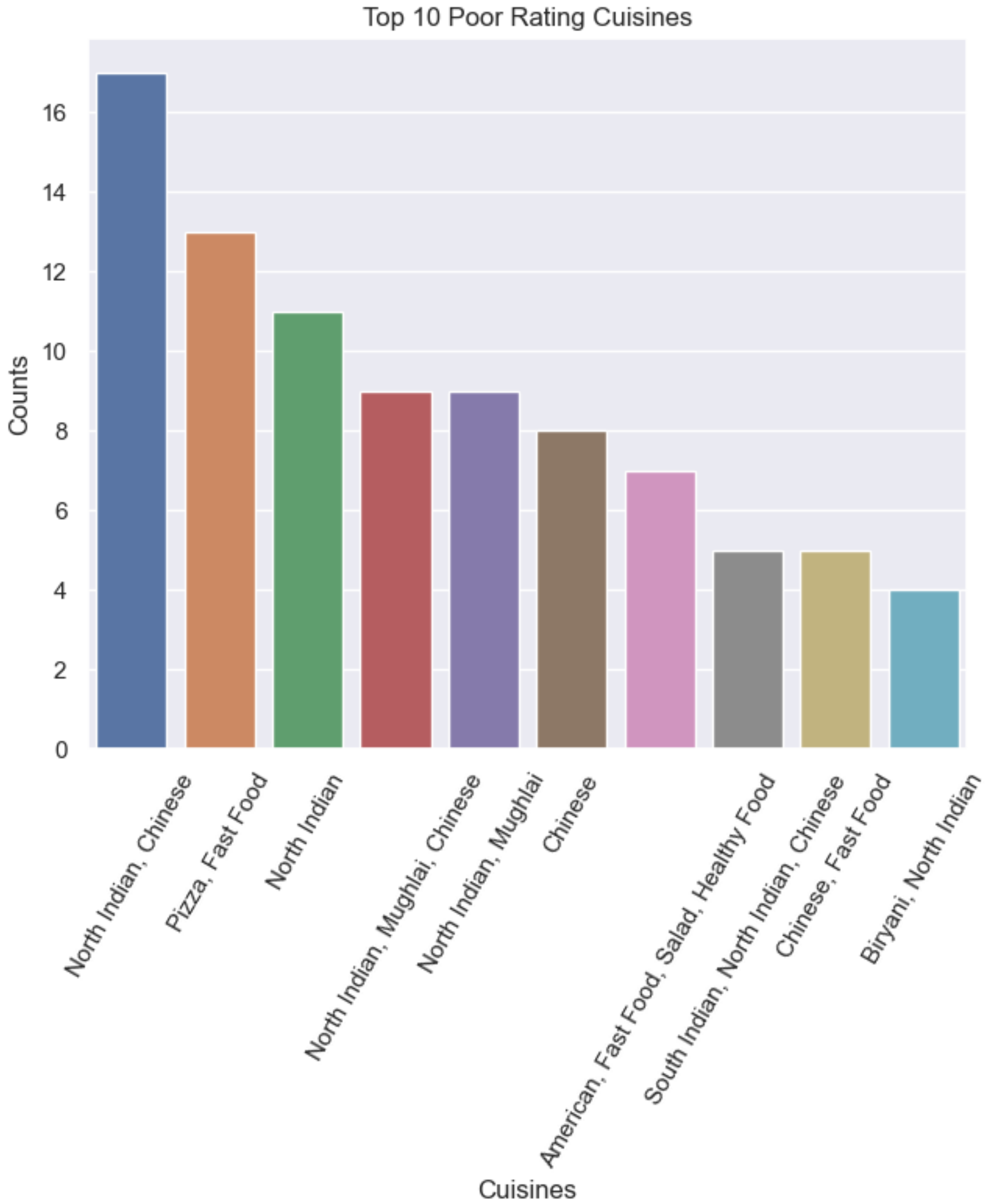
```
In [908]: sns.set_style("darkgrid")
plt.figure(figsize=(16,13))
plt.subplot(2,2,1)
avc = sns.barplot(
    x = "Cuisines",
    y = "Counts",
    data = average_rating_cuisines
)
avc.set_xticklabels(exc.get_xticklabels(),rotation=60);
plt.title("Top 10 Average Rating Cuisines")
```

```
Out[908]: Text(0.5, 1.0, 'Top 10 Average Rating Cuisines')
```



In [909]:
```
sns.set_style("darkgrid")
plt.figure(figsize=(16,13))
plt.subplot(2,2,1)
pc = sns.barplot(
    x = "Cuisines",
    y = "Counts",
    data = poor_rating_cuisines
)
pc.set_xticklabels(pc.get_xticklabels(),rotation=60);
plt.title("Top 10 Poor Rating Cuisines")
```

Out[909]:
```
Text(0.5, 1.0, 'Top 10 Poor Rating Cuisines')
```



Average Cost for two people in India

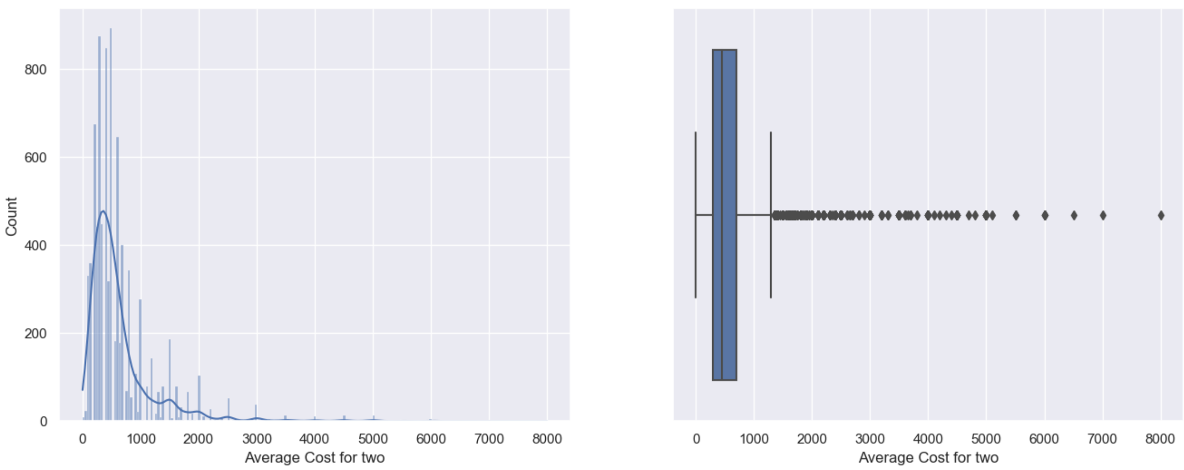
In [913]:
```python
cost_for_two = reviews_from_india
```

In [914]:
```python
cost_for_two.head()
```

Out[914]:

	Restaurant Name	City	Cuisines	Average Cost for two	Currency	Has Table booking	Has Online delivery	Is delivering now	Swit ord me
624	Jahanpanah	Agra	North Indian, Mughlai	850	Indian Rupees(Rs.)	No	No	No	N
625	Rangrezz Restaurant	Agra	North Indian, Mughlai	700	Indian Rupees(Rs.)	No	No	No	N
626	Time2Eat - Mama Chicken	Agra	North Indian	500	Indian Rupees(Rs.)	No	No	No	N
627	Chokho Jeeman Marwari Jain Bhojanalya	Agra	Rajasthani	400	Indian Rupees(Rs.)	No	No	No	N
628	Pinch Of Spice	Agra	North Indian, Chinese, Mughlai	1000	Indian Rupees(Rs.)	No	No	No	N

In [916]:
```python
plt.figure(figsize=(16,13))
plt.subplot(2,2,1)
sns.histplot(
    x=cost_for_two["Average Cost for two"],
    kde = True
)
plt.subplot(2,2,2)
sns.boxplot(
    x=cost_for_two["Average Cost for two"]
)
plt.show()
```



Observation

- From the above observation it is clearly visible that the distribution of the **Average Cost for two people** is right skewed so to estimate the centeral value median is the best measure to go with beacause **mean** is more sensitive to outliers, so it is not the best measure to go with
- **Median** and **Mode** are the best measures to go with in estimating centeral values in case of outliers.

In [940]:
```
def percentile_analysis(data : Series):
    columns = ["Median_Cost","25_percentile","50_percentile","75_pe
    P25 = data.quantile(0.25)
    P50 = data.quantile(0.50)
    P75 = data.quantile(0.75)
    P100 = data.quantile(1.0)
    values = [P50,P25,P50,P75,P100]
    stats_data = {
        "statistical_Features":columns,
        "Values":values
    }
    stats_data = pd.DataFrame(stats_data)
    return stats_data
```

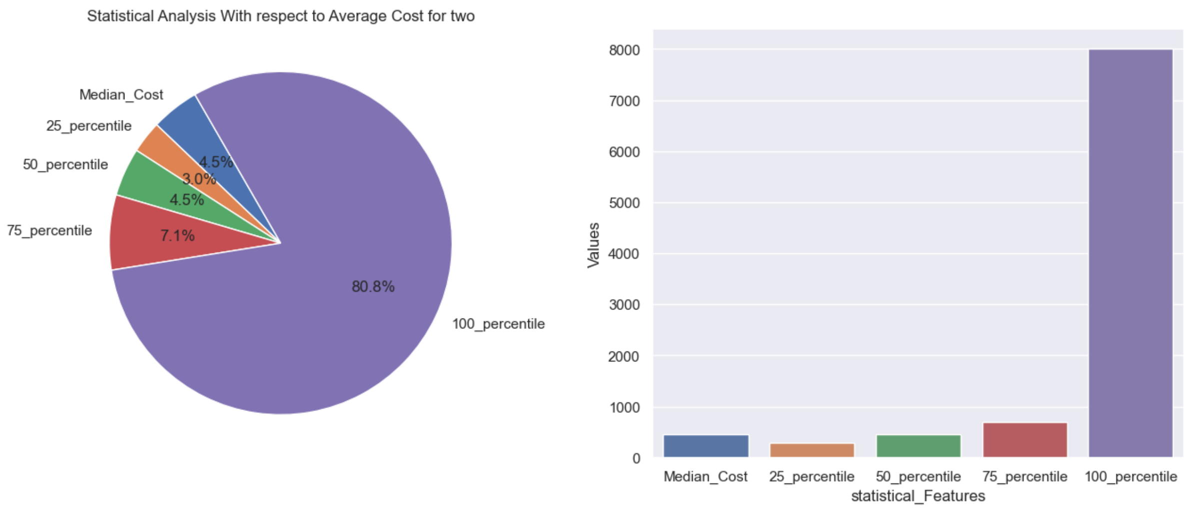
In [943]:
```
result = percentile_analysis(cost_for_two["Average Cost for two"])
result.head()
```

Out[943]:

	statistical_Features	Values
0	Median_Cost	450.0
1	25_percentile	300.0
2	50_percentile	450.0
3	75_percentile	700.0
4	100_percentile	8000.0

In [945]:

```
plt.figure(figsize=(16,13))
plt.subplot(2,2,1)
plt.pie(
    result["Values"],
    labels = result["statistical_Features"],
    autopct='%1.1f%%',
    startangle=120
)
plt.title("Statistical Analysis With respect to Average Cost for tw
plt.subplot(2,2,2)
sns.barplot(
    x = result["statistical_Features"],
    y = result["Values"]
)
plt.show()
```



Observation

- Median value with respect to Average Cost for two people is **Rs.450**

In [ ]: