In [1]:

```
# Importing necessary libraries
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

In [2]:

```
data = pd.read_excel('CapstoneData.xlsx')
data.shape
```

Out[2]:

```
(9551, 19)
```

In [3]:

```
df=data.copy()
df.head()
```

Out[3]:

	Restaurant ID	Restaurant Name	Country Code	City	Address	Locality	Locality Verbose	Longitude
0	7402935	Skye	94	Jakarta	Menara BCA, Lantai 56, Jl. MH. Thamrin, Thamri...	Grand Indonesia Mall, Thamrin	Grand Indonesia Mall, Thamrin, Jakarta	106.821999
1	7410290	Satoo - Hotel Shangri-La	94	Jakarta	Hotel Shangri-La, Jl. Jend. Sudirman	Hotel Shangri-La, Sudirman	Hotel Shangri-La, Sudirman, Jakarta	106.818961
2	7420899	Sushi Masa	94	Jakarta	Jl. Tuna Raya No. 5, Penjaringan	Penjaringan	Penjaringan, Jakarta	106.800144
3	7421967	3 Wise Monkeys	94	Jakarta	Jl. Suryo No. 26, Senopati, Jakarta	Senopati	Senopati, Jakarta	106.813400
4	7422489	Avec Moi Restaurant and Bar	94	Jakarta	Gedung PIC, Jl. Teluk Betung 43, Thamrin, Jakarta	Thamrin	Thamrin, Jakarta	106.821023

In [4]:

```
# Uploading the secondary data & merging into primary:
sec = pd.read_excel('Country-Code.xlsx')
df=pd.merge(df,sec,on='Country Code',how='left')
df.head()
```

Out[4]:

	Restaurant ID	Restaurant Name	Country Code	City	Address	Locality	Locality Verbose	Longitude
0	7402935	Skye	94	Jakarta	Menara BCA, Lantai 56, Jl. MH. Thamrin, Thamri...	Grand Indonesia Mall, Thamrin	Grand Indonesia Mall, Thamrin, Jakarta	106.821999
1	7410290	Satoo - Hotel Shangri-La	94	Jakarta	Hotel Shangri-La, Jl. Jend. Sudirman	Hotel Shangri-La, Sudirman	Hotel Shangri-La, Sudirman, Jakarta	106.818961
2	7420899	Sushi Masa	94	Jakarta	Jl. Tuna Raya No. 5, Penjaringan	Penjaringan	Penjaringan, Jakarta	106.800144
3	7421967	3 Wise Monkeys	94	Jakarta	Jl. Suryo No. 26, Senopati, Jakarta	Senopati	Senopati, Jakarta	106.813400
4	7422489	Avec Moi Restaurant and Bar	94	Jakarta	Gedung PIC, Jl. Teluk Betung 43, Thamrin, Jakarta	Thamrin	Thamrin, Jakarta	106.821023

```
Importing, Understanding, and Inspecting Data:

Perform preliminary data inspection, findings the structure of the data, missing values,
duplicates, etc.
Identify duplicates and remove them.
```

In [5]:

```
df.columns = df.columns.str.replace(' ','_')
df.columns
```

Out[5]:

```
Index(['Restaurant_ID', 'Restaurant_Name', 'Country_Code', 'City', 'Addres
s',
       'Locality', 'Locality_Verbose', 'Longitude', 'Latitude', 'Cuisines',
       'Average_Cost_for_two', 'Currency', 'Has_Table_booking',
       'Has_Online_delivery', 'Price_range', 'Aggregate_rating',
       'Rating_color', 'Rating_text', 'Votes', 'Country'],
      dtype='object')
```

In [6]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 9551 entries, 0 to 9550
Data columns (total 20 columns):
Restaurant_ID          9551 non-null int64
Restaurant_Name        9550 non-null object
Country_Code           9551 non-null int64
City                   9551 non-null object
Address                9551 non-null object
Locality               9551 non-null object
Locality_Verbose       9551 non-null object
Longitude              9551 non-null float64
Latitude               9551 non-null float64
Cuisines               9542 non-null object
Average_Cost_for_two   9551 non-null int64
Currency               9551 non-null object
Has_Table_booking      9551 non-null object
Has_Online_delivery    9551 non-null object
Price_range            9551 non-null int64
Aggregate_rating       9551 non-null float64
Rating_color           9551 non-null object
Rating_text            9551 non-null object
Votes                  9551 non-null int64
Country                9551 non-null object
dtypes: float64(3), int64(5), object(12)
memory usage: 1.5+ MB
```

In [7]:

```
# Identifying null values
df.isna().sum()
```

Out[7]:

```
Restaurant_ID          0
Restaurant_Name        1
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
Price_range            0
Aggregate_rating       0
Rating_color           0
Rating_text            0
Votes                  0
Country                0
dtype: int64
```

In [8]:

```
df[df['Restaurant_Name'].isna()]
```

Out[8]:

	Restaurant_ID	Restaurant_Name	Country_Code	City	Address	Locality	Loca
1646	113702	NaN	1	Ahmedabad	Opposite Sindhu Bhawan, Bodakdev, Ahmedabad	Bodakdev	

In [9]:

```
# Dropping the null value of Restaurant Name, as there is only 1 such record. Also reseting
df.dropna(axis=0,subset=['Restaurant_Name'],inplace=True)
df.reset_index(drop=True,inplace=True)
```

In [10]:

```
df[df['Restaurant_Name'].isna()]
```

Out[10]:

Restaurant_ID	Restaurant_Name	Country_Code	City	Address	Locality	Locality_Verbose	Lo

In [11]:

```
df[df['Cuisines'].isna()]
```

Out[11]:

	Restaurant_ID	Restaurant_Name	Country_Code	City	Address	Locality	Local
9082	17374552	Corkscrew Cafe	216	Gainesville	51 W Main St, Dahlonega, GA 30533	Dahlonega	
9085	17501439	Dovetail	216	Macon	543 Cherry St, Macon, GA 31201	Macon	Ma
9093	17059060	Hillstone	216	Orlando	215 South Orlando Avenue, Winter Park, FL 32789	Winter Park	
9405	17284158	Jimmie's Hot Dogs	216	Albany	204 S Jackson St, Albany, GA 31701	Albany	All
9493	17142698	Leonard's Bakery	216	Rest of Hawaii	933 Kapahulu Ave, Honolulu, HI 96816	Kaimuki	Kair
9503	17616465	Tybee Island Social Club	216	Savannah	1311 Butler Ave, Tybee Island, GA 31328	Tybee Island	T
9532	17284105	Cookie Shoppe	216	Albany	115 N Jackson St, Albany, GA 31701	Albany	All
9534	17284211	Pearly's Famous Country Cookng	216	Albany	814 N Slappey Blvd, Albany, GA 31701	Albany	All
9538	17606621	HI Lite Bar & Lounge	216	Miller	109 N Broadway Ave, Miller, SD 57362	Miller	

In [12]:

```
# We have 9 such records with no Cuinies, hence will replace them with 'Others'
df['Cuisines'].fillna('Others', inplace=True)
```

In [13]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9550 entries, 0 to 9549
Data columns (total 20 columns):
Restaurant_ID           9550 non-null int64
Restaurant_Name         9550 non-null object
Country_Code            9550 non-null int64
City                    9550 non-null object
Address                 9550 non-null object
Locality                9550 non-null object
Locality_Verbose        9550 non-null object
Longitude               9550 non-null float64
Latitude                9550 non-null float64
Cuisines                9550 non-null object
Average_Cost_for_two    9550 non-null int64
Currency                9550 non-null object
Has_Table_booking       9550 non-null object
Has_Online_delivery     9550 non-null object
Price_range             9550 non-null int64
Aggregate_rating        9550 non-null float64
Rating_color            9550 non-null object
Rating_text             9550 non-null object
Votes                   9550 non-null int64
Country                 9550 non-null object
dtypes: float64(3), int64(5), object(12)
memory usage: 1.5+ MB
```

In [14]:

```
df.isna().sum()
```

Out[14]:

```
Restaurant_ID           0
Restaurant_Name         0
Country_Code            0
City                    0
Address                 0
Locality                0
Locality_Verbose        0
Longitude               0
Latitude                0
Cuisines                0
Average_Cost_for_two    0
Currency                0
Has_Table_booking       0
Has_Online_delivery     0
Price_range             0
Aggregate_rating        0
Rating_color            0
Rating_text             0
Votes                   0
Country                 0
dtype: int64
```

In [29]:

```
# Geographical distribution of restaurants - By Countries
cntry_wise_res=df.groupby(['Country_Code','Country']).Restaurant_ID.count()
cntry_wise_res.sort_values(ascending = False)
```

Out[29]:

```
Country_Code  Country
1             India             8651
216           United States      434
215           United Kingdom      80
214           UAE                 60
189           South Africa        60
30            Brazil              60
148           New Zealand         40
208           Turkey              34
14            Australia           24
162           Phillipines         22
94            Indonesia           21
191           Sri Lanka           20
184           Singapore           20
166           Qatar               20
37            Canada               4
Name: Restaurant_ID, dtype: int64
```

With above table we can see that with most numbers of 8651 restaurants India is on top and on 2nd position is USA with 434 restaurants and 3rd being UK with 80 restaurants.

In [28]:

```
# Geographical distribution of restaurants - By Cities
City_wise_res=df.groupby(['Country','City']).Restaurant_ID.count()
City_wise_res.sort_values(ascending = False)
```

Out[28]:

```
Country        City
India          New Delhi           5473
               Gurgaon             1118
               Noida               1080
               Faridabad            251
               Ghaziabad             25
               Lucknow               21
               Amritsar              21
               Guwahati              21
               Bhubaneshwar          21
               Vadodara              20
               Nagpur                20
               Nashik                20
               Patna                 20
               Puducherry            20
               Pune                  20
               Mumbai                20
               Ranchi                20
               Surat                 20
               Varanasi              20
               Vizag                 20
New Zealand    Auckland              20
               Wellington City       20
Brazil         Rio de Janeiro        20
Qatar          Doha                  20
Singapore      Singapore             20
India          Mysore                20
               Kolkata               20
               Mangalore             20
               Chennai               20
Brazil         BrasÌ_lia             20
                                   ... 
Australia      Paynesville            1
               Palm Cove              1
               Montville              1
               Middleton Beach        1
               Macedon                1
United States  Winchester Bay         1
Canada         Chatham-Kent           1
               Consort                1
United States  Vernonia               1
               Princeton              1
               Potrero                1
               Ojo Caliente           1
               Monroe                 1
               Miller                 1
               Mc Millan              1
               Lincoln                1
               Lakeview               1
               Fernley                1
               Cochrane               1
               Clatskanie             1
South Africa   Randburg               1
```

```
Phillipines    Tagaytay City         1
               Quezon City           1
Indonesia      Bandung               1
United States  Weirton               1
India          Panchkula             1
               Mohali                1
Canada         Yorkton               1
               Vineland Station      1
Australia      Armidale              1
Name: Restaurant_ID, Length: 141, dtype: int64
```

In [27]:

```
Most_national_presence = df.groupby(['Restaurant_Name','Country']).Country.count()
Most_national_presence.sort_values(ascending = False)
```

Out[27]:

```
Restaurant_Name                      Country
Cafe Coffee Day                      India            83
Domino's Pizza                       India            79
Subway                               India            63
Green Chick Chop                     India            51
McDonald's                           India            48
Keventers                            India            34
Giani                                India            29
Pizza Hut                            India            29
Baskin Robbins                       India            28
Barbeque Nation                      India            25
Barista                              India            22
Dunkin' Donuts                       India            22
Giani's                              India            22
Pind Balluchi                        India            20
Costa Coffee                         India            20
Sagar Ratna                          India            19
Pizza Hut Delivery                   India            19
Twenty Four Seven                    India            19
Wah Ji Wah                           India            19
Chaayos                              India            18
Republic of Chicken                  India            18
KFC                                  India            18
Starbucks                            India            17
Haldiram's                           India            16
Burger King                          India            16
Shree Rathnam                        India            15
Bikanervala                          India            14
Frontier                             India            14
Aggarwal Sweets                      India            14
Moti Mahal Delux                     India            14
                                                      ..
Peshawar Sweets Shop                 India             1
Peshawari                            India             1
Pawan Foods                          India             1
Pavitra Bakers                       India             1
Pauls Food                           India             1
Paul's Homemade                      India             1
Parashar's                           India             1
Paratha Hi Paratha                   India             1
Pardeep Corner                       India             1
Paribar                              Brazil            1
Parikrama - The Revolving Restaurant India             1
Paris 6 Classique                    Brazil            1
Parkash Dhaba                        India             1
Parker's                             UAE               1
Parrot's                             South Africa      1
Parul's Cooking Hub                  India             1
Pash!                                India             1
Passage 2 India                      United States     1
Pasta La Vista                       India             1
Pasta Pizza & Roll Hut               India             1
Pasta Xpress                         India             1
Pastry Place                         India             1
```

```
Pastry Point                         India          1
Pat 'N' Harry                        India          1
Patang - The Revolving Restaurant    India          1
Patiala                              India          1
Patiala Peg - The Imperial           India          1
Patiala Shahi Soups                  India          1
Patna Roll Center                    India          1
12212                                India          1
Name: Country, Length: 7472, dtype: int64
```

The above table shows that 'Cafe Coffee Day' has the maximum presence with 83 outlets, 'Domino's Pizza' at 2nd with 79 outlets and 'Subway' at 3rd position with 63 outlets. All of these are present only in India.

In [30]:

```
## Ratio between restaurants that allow table booking vs. those that do not allow table boo
Booking_table=df.groupby('Has_Table_booking').Restaurant_ID.count()
Booking_table
```

Out[30]:

```
Has_Table_booking
No     8392
Yes    1158
Name: Restaurant_ID, dtype: int64
```

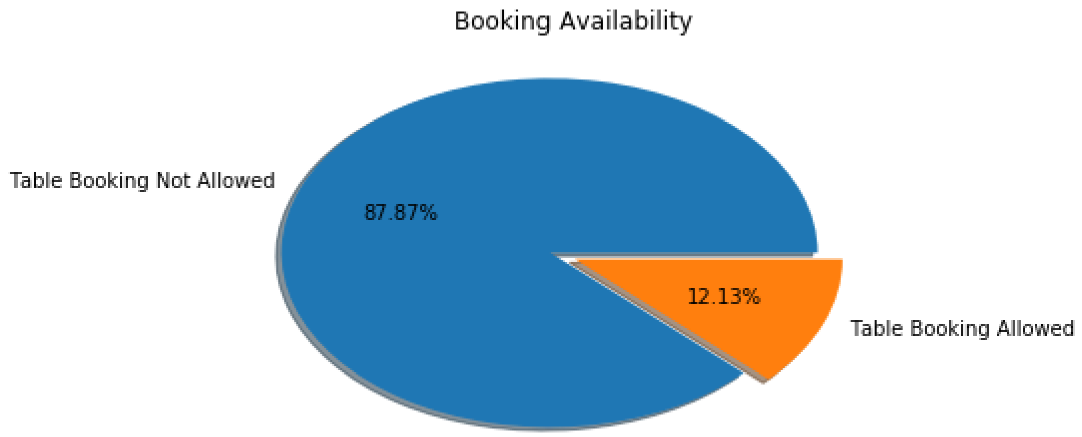
In [31]:

```
print('Ratio between restaurants that allow table booking vs. those that do not allow table
      round((Booking_table.Yes/Booking_table.No),2))
```

```
Ratio between restaurants that allow table booking vs. those that do not all
ow table booking: 0.14
```

In [32]:

```
# Graphical representation of above ration - Pie Chart
plt.title('Booking Availability')
plt.pie(Booking_table,labels=('Table Booking Not Allowed','Table Booking Allowed'),explode=
        autopct='%1.2f%%')
plt.show()
```



Above pie chart clearly shows that only 12.13% restaurants allow table booking and majority of 87.87% don't allow.

In [33]:

```
# Percentage of restaurants providing online delivery
Online_Delivery=df.groupby('Has_Online_delivery').Restaurant_ID.count()
Online_Delivery
```

Out[33]:

```
Has_Online_delivery
No      7099
Yes     2451
Name: Restaurant_ID, dtype: int64
```

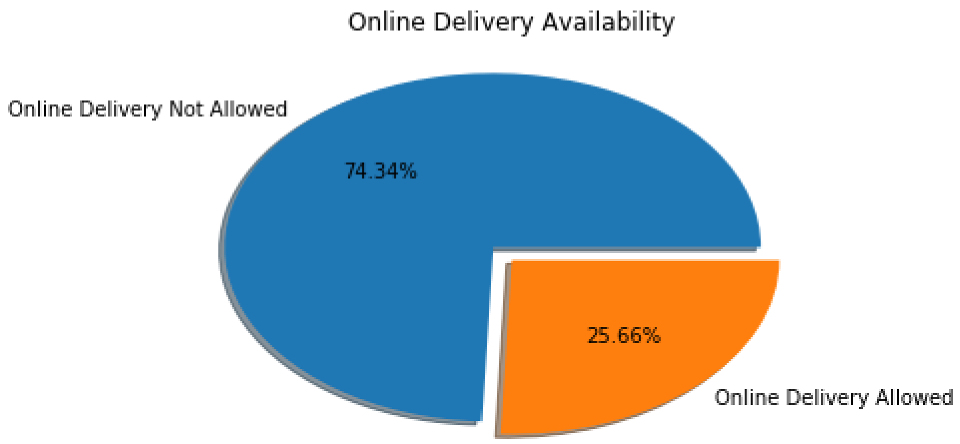
In [34]:

```
print('Percentage of restaurants providing online delivery:', round((Online_Delivery.Yes/On
```

```
Percentage of restaurants providing online delivery: 25.66
```

In [35]:

```
# Graphical representation via Pie Chart
plt.title('Online Delivery Availability')
plt.pie(Online_Delivery,labels=('Online Delivery Not Allowed','Online Delivery Allowed'),ex
        autopct='%1.2f%%')
plt.show()
```




```
The above calculation and the pie chart shows that 25.66% of restaurants provide online
delivery. However, majority of 74.34% do not have this facility
```

In [36]:

```
# Difference in number of votes for the restaurants that deliver and the restaurants that d
Online_Delivery_Votes=df.groupby('Has_Online_delivery').Votes.sum()
Online_Delivery_Votes
```

Out[36]:

```
Has_Online_delivery
No      979962
Yes     517914
Name: Votes, dtype: int64
```

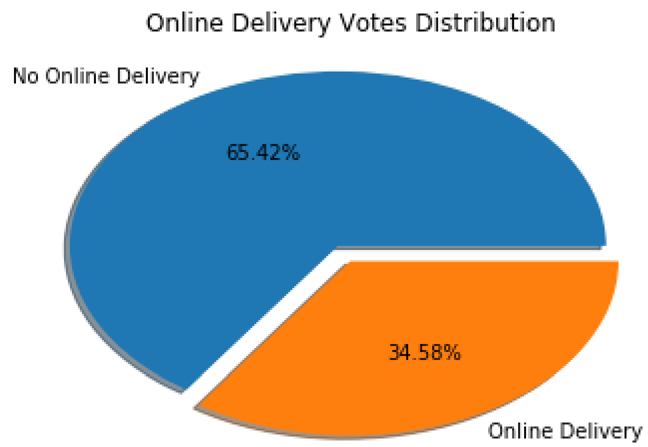
In [37]:

```
Votes_difference = Online_Delivery_Votes.No - Online_Delivery_Votes.Yes
print('Difference in number of votes for the restaurants that deliver and the restaurants t
     Votes_difference)
```

```
Difference in number of votes for the restaurants that deliver and the resta
urants that do not deliver: 462048
```

In [38]:

```
# Graphical representation via Pie Chart
plt.title('Online Delivery Votes Distribution')
plt.pie(Online_Delivery_Votes,labels=('No Online Delivery','Online Delivery'),explode=(0.1,
        autopct='%1.2f%%')
plt.show()
```




```
It means that 65.42% votes have been given to those restaurants who don't have online
delivery and those with online delivery have received voting of 34.58%.
It means that restaurants with no online delivery are getting more votes.
```

In [39]:

```
# Spliting the 'Cuisines' column into multiple columns & renaming the columns
cuisines_split = df['Cuisines'].apply(lambda x: pd.Series(x.split(',')))
cuisines_split.columns=['Cuisine 1','Cuisine 2','Cuisine 3','Cuisine 4','Cuisine 5','Cuisin
cuisines_split.head()
```

Out[39]:

	Cuisine 1	Cuisine 2	Cuisine 3	Cuisine 4	Cuisine 5	Cuisine 6	Cuisine 7	Cuisine 8
0	Italian	Continental	NaN	NaN	NaN	NaN	NaN	NaN
1	Asian	Indonesian	Western	NaN	NaN	NaN	NaN	NaN
2	Sushi	Japanese	NaN	NaN	NaN	NaN	NaN	NaN
3	Japanese	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4	French	Western	NaN	NaN	NaN	NaN	NaN	NaN

In [40]:

```
Cuisine1_count=pd.DataFrame(cuisines_split["Cuisine 1"].value_counts()).reset_index()
Cuisine1_count=Cuisine1_count.rename(columns={"index": "Cuisines", "Cuisine 1": "Counts"})
Cuisine1_count.head(10)
```

Out[40]:

	Cuisines	Counts
0	North Indian	2991
1	Chinese	855
2	Fast Food	672
3	Bakery	621
4	Cafe	617
5	American	278
6	South Indian	262
7	Mithai	246
8	Street Food	236
9	Continental	235

In [41]:

```
Cuisine2_count=pd.DataFrame(cuisines_split["Cuisine 2"].value_counts()).reset_index()
Cuisine2_count=Cuisine2_count.rename(columns={"index": "Cuisines", "Cuisine 2": "Counts"})
Cuisine2_count.head(10)
```

Out[41]:

	Cuisines	Counts
0	Chinese	1156
1	Fast Food	820
2	North Indian	687
3	Mughlai	635
4	Desserts	407
5	Continental	244
6	Italian	240
7	South Indian	219
8	Street Food	216
9	Pizza	110

In [42]:

```
Cuisine3_count=pd.DataFrame(cuisines_split["Cuisine 3"].value_counts()).reset_index()
Cuisine3_count=Cuisine3_count.rename(columns={"index": "Cuisines", "Cuisine 3": "Counts"})
Cuisine3_count.head(10)
```

Out[42]:

	Cuisines	Counts
0	Chinese	594
1	Fast Food	385
2	Continental	193
3	North Indian	191
4	Italian	189
5	Mughlai	124
6	South Indian	123
7	Salad	70
8	Asian	64
9	Thai	50

In [43]:

```
Cuisine4_count=pd.DataFrame(cuisines_split["Cuisine 4"].value_counts()).reset_index()
Cuisine4_count=Cuisine4_count.rename(columns={"index": "Cuisines", "Cuisine 4": "Counts"})
Cuisine4_count.head(10)
```

Out[43]:

	Cuisines	Counts
0	Italian	84
1	Chinese	81
2	Fast Food	77
3	Healthy Food	67
4	North Indian	65
5	Street Food	59
6	Continental	50
7	Asian	31
8	Mithai	26
9	South Indian	25

In [44]:

```
Cuisine5_count=pd.DataFrame(cuisines_split["Cuisine 5"].value_counts()).reset_index()
Cuisine5_count=Cuisine5_count.rename(columns={"index": "Cuisines", "Cuisine 5": "Counts"})
Cuisine5_count.head(10)
```

Out[44]:

	Cuisines	Counts
0	Chinese	41
1	Fast Food	30
2	North Indian	23
3	Thai	20
4	Italian	13
5	Mithai	12
6	Mediterranean	11
7	Continental	11
8	Asian	9
9	Street Food	8

In [45]:

```
Cuisine6_count=pd.DataFrame(cuisines_split["Cuisine 6"].value_counts()).reset_index()
Cuisine6_count=Cuisine6_count.rename(columns={"index": "Cuisines", "Cuisine 6": "Counts"})
Cuisine6_count.head(10)
```

Out[45]:

	Cuisines	Counts
0	Mithai	23
1	Beverages	12
2	Chinese	8
3	Lebanese	8
4	Thai	5
5	Cafe	4
6	Desserts	4
7	Tea	4
8	Japanese	3
9	Finger Food	3

In [48]:

```
Cuisine7_count=pd.DataFrame(cuisines_split["Cuisine 7"].value_counts()).reset_index()
Cuisine7_count=Cuisine7_count.rename(columns={"index": "Cuisines", "Cuisine 7": "Counts"})
Cuisine7_count.head(10)
```

Out[48]:

	Cuisines	Counts
0	Desserts	24
1	Mediterranean	2
2	European	2
3	Mithai	2
4	Ice Cream	1
5	Beverages	1
6	Asian	1
7	North Indian	1
8	Rajasthani	1
9	Bakery	1

In [49]:

```
Cuisine8_count=pd.DataFrame(cuisines_split["Cuisine 8"].value_counts()).reset_index()
Cuisine8_count=Cuisine8_count.rename(columns={"index": "Cuisines", "Cuisine 8": "Counts"})
Cuisine8_count.head(10)
```

Out[49]:

	Cuisines	Counts
0	Mithai	8
1	International	2
2	Finger Food	2
3	Beverages	1
4	Mughlai	1

In [50]:

```
Cuisines_append = Cuisine1_count.append([Cuisine2_count,Cuisine3_count,Cuisine4_count,Cuisi
                                         Cuisine7_count,Cuisine8_count])
Cuisines_append['Cuisines']=Cuisines_append.Cuisines.str.replace(' ','')
Cuisines_append
```

Out[50]:

	Cuisines	Counts
0	NorthIndian	2991
1	Chinese	855
2	FastFood	672
3	Bakery	621
4	Cafe	617
5	American	278
6	SouthIndian	262
7	Mithai	246
8	StreetFood	236
9	Continental	235
10	Italian	234
11	Pizza	232
12	Mughlai	215
13	IceCream	178
14	Desserts	150
15	Burger	116
16	Biryani	112
17	RawMeats	110
18	Beverages	79
19	FingerFood	78
20	Asian	77
21	Japanese	72
22	Mexican	62
23	European	62
24	HealthyFood	56
25	Indian	51
26	Seafood	49
27	Thai	29
28	Mediterranean	28
29	Lebanese	25
...	...	...
27	British	1

	Cuisines	Counts
28	Gujarati	1
29	NorthIndian	1
30	Deli	1
31	Burmese	1
32	Salad	1
33	StreetFood	1
34	Tex-Mex	1
35	HealthyFood	1
0	Desserts	24
1	Mediterranean	2
2	European	2
3	Mithai	2
4	IceCream	1
5	Beverages	1
6	Asian	1
7	NorthIndian	1
8	Rajasthani	1
9	Bakery	1
10	Pizza	1
11	Thai	1
12	Lebanese	1
13	Spanish	1
14	FingerFood	1
15	American	1
0	Mithai	8
1	International	2
2	FingerFood	2
3	Beverages	1
4	Mughlai	1

514 rows × 2 columns

In [51]:

```
Cuisines_group=Cuisines_append.groupby('Cuisines').sum()
Max_Cuisines=Cuisines_group.sort_values(by='Counts', ascending = False)
```

In [52]:

```
Max_Cuisines
```

Out[52]:

	Counts
Cuisines	
NorthIndian	3959
Chinese	2735
FastFood	1986
Mughlai	995
Italian	763
Bakery	745
Continental	735
Cafe	703
Desserts	653
SouthIndian	636
StreetFood	562
American	390
Pizza	381
Mithai	380
Burger	251
Thai	234
Asian	233
Beverages	229
IceCream	226
Mexican	180
Biryani	177
Seafood	174
HealthyFood	150
European	148
Japanese	135
FingerFood	114
RawMeats	114
Mediterranean	112
Salad	93
Sushi	75
...	...
Persian	2
SouthAmerican	2

Cuisines	Counts
Taiwanese	2
Teriyaki	2
AsianFusion	2
DrinksOnly	2
Izgara	2
Argentine	2
Cantonese	2
NewAmerican	2
Oriya	2
Ramen	2
PubFood	2
Cuban	2
Belgian	2
GourmetFastFood	1
Malay	1
Irish	1
FishandChips	1
Durban	1
DÌ_ner	1
BubbleTea	1
Malwani	1
BÌ_rek	1
Peruvian	1
Canadian	1
Mineira	1
SoulFood	1
CuisineVaries	1
Peranakan	1

146 rows × 1 columns

In [53]:

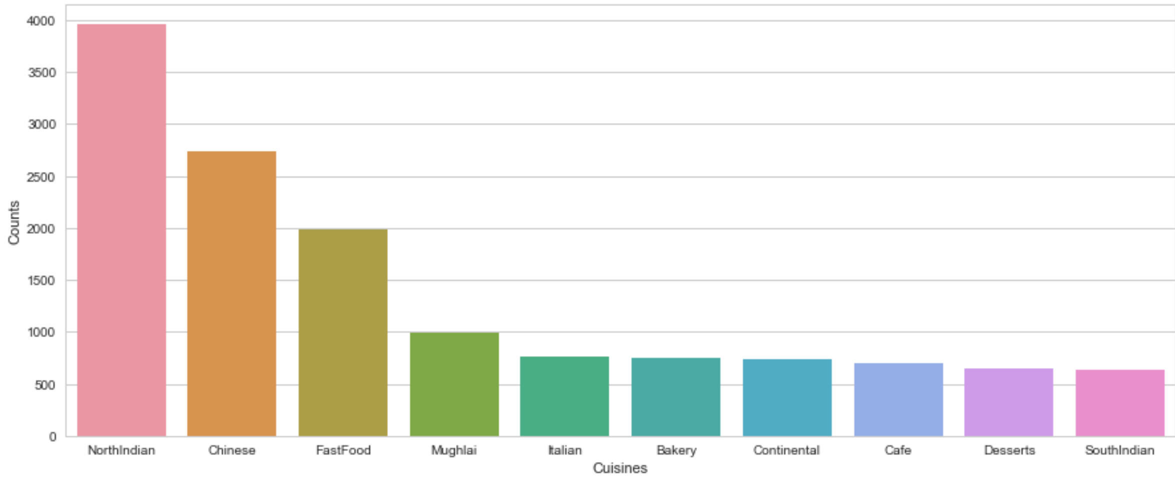
```
# Top 10 cuisines served across cities
Top_10_cuisines=pd.DataFrame(Max_Cuisines.head(10)).reset_index()
Top_10_cuisines
```

Out[53]:

	Cuisines	Counts
0	NorthIndian	3959
1	Chinese	2735
2	FastFood	1986
3	Mughlai	995
4	Italian	763
5	Bakery	745
6	Continental	735
7	Cafe	703
8	Desserts	653
9	SouthIndian	636

In [54]:

```
# Bar plot of Top 10 cuisines
sns.set(style="whitegrid")
f, ax = plt.subplots(figsize=(15, 6))
ax=sns.barplot(x='Cuisines',y='Counts', data=Top_10_cuisines)
```




```
Above are the top 10 cuisines accross cities, where 'North Indian' Cuisine is on the top,
2nd 'Chinese', and 3rd is 'Fast Food'.
Therefore, most served cuisine is 'North Indian'.
```

In [55]:

```
# Maximum and minimum number of cuisines that a restaurant serves
df2 = pd.DataFrame(df, columns=['Restaurant_Name','City','Cuisines'])
df2
```

Out[55]:

	Restaurant_Name	City	Cuisines
0	Skye	Jakarta	Italian, Continental
1	Satoo - Hotel Shangri-La	Jakarta	Asian, Indonesian, Western
2	Sushi Masa	Jakarta	Sushi, Japanese
3	3 Wise Monkeys	Jakarta	Japanese
4	Avec Moi Restaurant and Bar	Jakarta	French, Western
5	Lucky Cat Coffee & Kitchen	Jakarta	Cafe, Western
6	Onokabe	Tangerang	Indonesian
7	Lemongrass	Bogor	Peranakan, Indonesian
8	MONKS	Jakarta	Western, Asian, Cafe
9	Talaga Sampireun	Jakarta	Sunda, Indonesian
10	OJJU	Jakarta	Korean
11	Union Deli	Jakarta	Desserts, Bakery, Western
12	Zenbu	Jakarta	Japanese, Sushi, Ramen
13	Talaga Sampireun	Jakarta	Sunda, Indonesian
14	Talaga Sampireun	Tangerang	Sunda, Indonesian
15	Toodz House	Jakarta	Cafe, Italian, Coffee and Tea, Western, Indone...
16	Noah's Barn Coffeenery	Bandung	Cafe, Coffee and Tea, Western
17	Flip Burger	Jakarta	Burger
18	Fish Streat	Jakarta	Seafood, Western
19	Fish Streat	Jakarta	Seafood, Western
20	Momo Milk	Bogor	Cafe, Desserts, Beverages
21	Orient Express - Taj Palace Hotel	New Delhi	European
22	Tian - Asian Cuisine Studio - ITC Maurya	New Delhi	Asian, Japanese, Korean, Thai, Chinese
23	Bukhara - ITC Maurya	New Delhi	North Indian
24	Spiral - Sofitel Philippine Plaza Manila	Pasay City	European, Asian, Indian
25	Nostalgia at 1911 Brasserie - The Imperial	New Delhi	European, Continental
26	1911 - The Imperial	New Delhi	North Indian, Chinese, South Indian, Italian
27	The Spice Route - The Imperial	New Delhi	Malaysian, Thai, Kerala, Vietnamese, Sri Lankan
28	Wasabi by Morimoto - The Taj Mahal Hotel	New Delhi	Japanese, Sushi
29	MEGU - The Leela Palace	New Delhi	Japanese, Sushi

	Restaurant_Name	City	Cuisines
...	...	...	...
9520	Four Queens Dairy Cream	Waterloo	Desserts
9521	Hong Kong Chinese Restaurant	Waterloo	Chinese
9522	J's Homestyle Cooking	Waterloo	American, Breakfast
9523	Scratch	Waterloo	Coffee and Tea, Desserts, Beverages
9524	Golden China	Waterloo	Chinese
9525	The Screaming Eagle	Waterloo	American, Bar Food
9526	The Thai Bowl	Waterloo	Thai
9527	Pepe's Piri Piri	Birmingham	Fast Food
9528	The Giggling Goat	Dicky Beach	Coffee and Tea, Tea, Modern Australian
9529	La Trattoria of Lavandula	Hepburn Springs	Italian, Fusion, Cafe
9530	Beach Box Cafe	Inverloch	Burger, Coffee and Tea, Modern Australian
9531	Funkey Monkey	Lakes Entrance	Breakfast, Coffee and Tea
9532	Cookie Shoppe	Albany	Others
9533	El Vaquero Mexican Restaurant	Albany	Mexican
9534	Pearly's Famous Country Cookng	Albany	Others
9535	Deorio's	Columbus	Italian, Pizza
9536	Azteca	Davenport	Mexican
9537	Happy Joe's Pizza & Ice Cream	Dubuque	Desserts, Pizza, Ice Cream
9538	HI Lite Bar & Lounge	Miller	Others
9539	Royal Hotel	Pocatello	Pizza, Bar Food
9540	Senor Iguanas	Pocatello	Mexican
9541	The Latitude - Radisson Blu	Agra	North Indian, Chinese, Continental
9542	Sheroes Hangout	Agra	Cafe, North Indian, Chinese
9543	Chapter 1 Cafe	Agra	Cafe, Italian, Mexican, North Indian, Continental
9544	The BrewMaster	Allahabad	North Indian, Chinese, Italian
9545	BMG - All Day Dining	Dehradun	Chinese, North Indian, Fast Food
9546	Atmosphere Grill Cafe Sheesha	Kanpur	Indian, Chinese, Continental
9547	UrbanCrave	Kanpur	Cafe, Continental, Desserts, Ice Cream, Italia...
9548	Deena Chat Bhandar	Varanasi	Street Food
9549	VNS Live Studio	Varanasi	Chinese, North Indian

9550 rows × 3 columns

In [56]:

```
CuisineCount_by_res=pd.DataFrame(df2.groupby('Restaurant_Name').Cuisines.count()).reset_inc
CuisineCount_by_res.sort_values(by = 'Cuisines', ascending = False)
```

Out[56]:

	Restaurant_Name	Cuisines
1098	Cafe Coffee Day	83
2098	Domino's Pizza	79
6105	Subway	63
2716	Green Chick Chop	51
4076	McDonald's	48
3478	Keventers	34
4960	Pizza Hut	30
2619	Giani	29
680	Baskin Robbins	28
663	Barbeque Nation	26
2145	Dunkin' Donuts	22
2620	Giani's	22
670	Barista	22
1758	Costa Coffee	20
4923	Pind Balluchi	20
7224	Wah Ji Wah	19
4961	Pizza Hut Delivery	19
7016	Twenty Four Seven	19
5477	Sagar Ratna	19
6079	Starbucks	18
3312	KFC	18
5316	Republic of Chicken	18
1324	Chaayos	18
1025	Burger King	16
2824	Haldiram's	16
5785	Shree Rathnam	15
821	Bikanervala	14
189	Aggarwal Sweets	14
4300	Moti Mahal Delux	14
2512	Frontier	14
...	...	...
2583	Gardenia - Hotel Grenville	1
2581	Garden Chef	1
2609	Get Lost in Flavours	1

	Restaurant_Name	Cuisines
2610	Get Set Go	1
2611	Getafix Petit	1
2612	Ghalib Kabab Corner	1
2639	Go! Biryani	1
2637	Go Krazy	1
2636	Go Foodie	1
2635	Gluten Free by Deepika	1
2634	Global Local	1
2633	Global Grill	1
2632	Glenz Cafe N Bakers	1
2631	Glen's Bakehouse	1
2630	Giuseppe's Pizza & Italian Specialities	1
2629	Giulios Greek & Italian Restaurant	1
2628	Giri Momos Centre & Chinese Fast Food	1
2627	Giri Manja's	1
2626	Giovanni's Shrimp Truck	1
2625	Ginnis Oven	1
2624	Ginger Garlic	1
2623	Gibson's Gourmet Burgers & Ribs	1
2622	Giapo	1
2621	Giani's di Hatti	1
2618	Gian Ji Punjabi Dhaba	1
2617	Ghungroo Club & Bar - By Gautam Gambhir	1
2616	Ghar Ki Handi	1
2615	Ghar Ka Swad	1
2613	Ghar Bistro Cafe	1
7444	Ìàukura€Ùa Sofras€±	1

7445 rows × 2 columns

```
The Maximum numbers of cuisines serves by a restaurant is 83 by 'Cafe Coffee Day' and the
least number of cuisines serves by a restaurant is 1.
There are many such restaurants with only 1 cuisine.
```

In [58]:

```
# Currency wise distribution of cost
Currency_dis = pd.DataFrame(df.groupby('Currency').Average_Cost_for_two.count()).reset_inde
Currency_dis.sort_values(by='Average_Cost_for_two', ascending = False)
```

Out[58]:

	Currency	Average_Cost_for_two
4	Indian Rupees(Rs.)	8651
2	Dollar($)	482
7	Pounds(â£)	80
1	Brazilian Real(R$)	60
3	Emirati Diram(AED)	60
9	Rand(R)	60
6	NewZealand($)	40
11	Turkish Lira(TL)	34
0	Botswana Pula(P)	22
5	Indonesian Rupiah(IDR)	21
8	Qatari Rial(QR)	20
10	Sri Lankan Rupee(LKR)	20

In [59]:

```
##  Distribution cost accross the restaurants
Cost_per_restaurants = pd.DataFrame(df.groupby('Restaurant_Name').Average_Cost_for_two.sum(
Cost_per_restaurants.sort_values(by = 'Average_Cost_for_two', ascending = False)
```

Out[59]:

	Restaurant_Name	Average_Cost_for_two
5897	Skye	800000
5594	Satoo - Hotel Shangri-La	800000
6262	Talaga Sampireun	600000
6170	Sushi Masa	500000
41	3 Wise Monkeys	450000
505	Avec Moi Restaurant and Bar	350000
3856	Lucky Cat Coffee & Kitchen	300000
4703	Onokabe	300000
3884	MONKS	250000
3762	Lemongrass	250000
7410	Zenbu	200000
7061	Union Deli	200000
2336	Fish Streat	200000
4642	OJJU	200000
6957	Toodz House	165000
4599	Noah's Barn Coffeenery	150000
2373	Flip Burger	120000
4249	Momo Milk	70000
2098	Domino's Pizza	55300
663	Barbeque Nation	38950
1098	Cafe Coffee Day	37350
6105	Subway	31400
4960	Pizza Hut	26130
4076	McDonald's	23550
4923	Pind Balluchi	18350
2716	Green Chick Chop	17850
4961	Pizza Hut Delivery	15200
670	Barista	14300
4300	Moti Mahal Delux	13550
2145	Dunkin' Donuts	13200
...	...	...
731	Berry Patch Restaurant	10
6993	Tu-Do Vietnamese Restaurant	10

	Restaurant_Name	Average_Cost_for_two
629	Bandit Burrito	10
6838	The Thai Bowl	10
3145	Ingleside Village Pizza	10
5984	Southern Bliss Bakery	10
3101	Ike & Jane	10
3252	Jim's Burgers	10
6607	The Giggling Goat	7
693	Beach Box Cafe	7
3662	La Trattoria of Lavandula	7
2540	Funkey Monkey	7
7112	VNS Live Studio	0
5625	Senor Iguanas	0
6458	The BrewMaster	0
1913	Deena Chat Bhandar	0
5725	Sheroes Hangout	0
5409	Royal Hotel	0
511	Azteca	0
2205	El Vaquero Mexican Restaurant	0
1973	Deorio's	0
1373	Chapter 1 Cafe	0
4869	Pearly's Famous Country Cookng	0
2808	HI Lite Bar & Lounge	0
2852	Happy Joe's Pizza & Ice Cream	0
7096	UrbanCrave	0
486	Atmosphere Grill Cafe Sheesha	0
1746	Cookie Shoppe	0
6691	The Latitude - Radisson Blu	0
522	BMG - All Day Dining	0

7445 rows × 2 columns

In [68]:

```
# Restaurants wise distribution of cost - By Currencies
df3=pd.DataFrame(df.groupby(['Currency','Restaurant_Name']).Average_Cost_for_two.sum()).res
df3.sort_values(by = 'Average_Cost_for_two', ascending = False)
```

Out[68]:

	Currency	Restaurant_Name	Average_Cost_for_two
7219	Indonesian Rupiah(IDR)	Satoo - Hotel Shangri-La	800000
7220	Indonesian Rupiah(IDR)	Skye	800000
7222	Indonesian Rupiah(IDR)	Talaga Sampireun	600000
7221	Indonesian Rupiah(IDR)	Sushi Masa	500000
7208	Indonesian Rupiah(IDR)	3 Wise Monkeys	450000
7209	Indonesian Rupiah(IDR)	Avec Moi Restaurant and Bar	350000
7218	Indonesian Rupiah(IDR)	Onokabe	300000
7213	Indonesian Rupiah(IDR)	Lucky Cat Coffee & Kitchen	300000
7212	Indonesian Rupiah(IDR)	Lemongrass	250000
7214	Indonesian Rupiah(IDR)	MONKS	250000
7224	Indonesian Rupiah(IDR)	Union Deli	200000
7210	Indonesian Rupiah(IDR)	Fish Streat	200000
7225	Indonesian Rupiah(IDR)	Zenbu	200000
7217	Indonesian Rupiah(IDR)	OJJU	200000
7223	Indonesian Rupiah(IDR)	Toodz House	165000
7216	Indonesian Rupiah(IDR)	Noah's Barn Coffeenery	150000
7211	Indonesian Rupiah(IDR)	Flip Burger	120000
7215	Indonesian Rupiah(IDR)	Momo Milk	70000
2494	Indian Rupees(Rs.)	Domino's Pizza	55300
1209	Indian Rupees(Rs.)	Barbeque Nation	38800
1580	Indian Rupees(Rs.)	Cafe Coffee Day	37350
6041	Indian Rupees(Rs.)	Subway	31400
4998	Indian Rupees(Rs.)	Pizza Hut	26050
4208	Indian Rupees(Rs.)	McDonald's	23550
4963	Indian Rupees(Rs.)	Pind Balluchi	18350
3005	Indian Rupees(Rs.)	Green Chick Chop	17850
4999	Indian Rupees(Rs.)	Pizza Hut Delivery	15200
1215	Indian Rupees(Rs.)	Barista	14300
4401	Indian Rupees(Rs.)	Moti Mahal Delux	13550
2533	Indian Rupees(Rs.)	Dunkin' Donuts	13200
...	...	...	...
302	Dollar($)	Jehova es Mi Pastor Tacos y Burritos	10

	Currency	Restaurant_Name	Average_Cost_for_two
306	Dollar($)	Jim's Burgers	10
307	Dollar($)	Jimmie's Hot Dogs	10
309	Dollar($)	Jimmy's Pancake House	10
310	Dollar($)	Johnnie Mars	10
313	Dollar($)	Kihei Caffe	10
319	Dollar($)	La Jalisco Supermercato	10
211	Dollar($)	El Kiosco Mexican Restaurant	10
111	Dollar($)	Beach Box Cafe	7
487	Dollar($)	The Giggling Goat	7
320	Dollar($)	La Trattoria of Lavandula	7
248	Dollar($)	Funkey Monkey	7
1055	Indian Rupees(Rs.)	Atmosphere Grill Cafe Sheesha	0
100	Dollar($)	Azteca	0
269	Dollar($)	Happy Joe's Pizza & Ice Cream	0
268	Dollar($)	HI Lite Bar & Lounge	0
390	Dollar($)	Pearly's Famous Country Cookng	0
179	Dollar($)	Cookie Shoppe	0
192	Dollar($)	Deorio's	0
1082	Indian Rupees(Rs.)	BMG - All Day Dining	0
435	Dollar($)	Senor Iguanas	0
6919	Indian Rupees(Rs.)	VNS Live Studio	0
1828	Indian Rupees(Rs.)	Chapter 1 Cafe	0
5691	Indian Rupees(Rs.)	Sheroes Hangout	0
2322	Indian Rupees(Rs.)	Deena Chat Bhandar	0
418	Dollar($)	Royal Hotel	0
6558	Indian Rupees(Rs.)	The Latitude - Radisson Blu	0
214	Dollar($)	El Vaquero Mexican Restaurant	0
6351	Indian Rupees(Rs.)	The BrewMaster	0
6904	Indian Rupees(Rs.)	UrbanCrave	0

7472 rows × 3 columns

In [70]:

```
df3.groupby(['Currency'], sort=False).max()
```

	Average_Cost_for_two
Currency	
Botswana Pula(P)	6000
Brazilian Real(R$)	460
Dollar($)	500
Emirati Diram(AED)	750
Indian Rupees(Rs.)	55300
Indonesian Rupiah(IDR)	800000
NewZealand($)	200
Pounds(å£)	230
Qatari Rial(QR)	550
Rand(R)	3210
Sri Lankan Rupee(LKR)	4500

In [75]:

```
# Rating distribution count on the scale of 0-5, where 5 being the best
Rating_count=pd.DataFrame(df.groupby('Aggregate_rating').Restaurant_Name.count()).reset_inc
Rating_count.sort_values(by='Aggregate_rating', ascending = False)
```

Out[75]:

	Aggregate_rating	Restaurant_Name
32	4.9	61
31	4.8	25
30	4.7	42
29	4.6	78
28	4.5	95
27	4.4	144
26	4.3	174
25	4.2	221
24	4.1	273
23	4.0	266
22	3.9	335
21	3.8	400
20	3.7	427
19	3.6	458
18	3.5	480
17	3.4	498
16	3.3	483
15	3.2	522
14	3.1	519
13	3.0	468
12	2.9	381
11	2.8	315
10	2.7	250
9	2.6	191
8	2.5	110
7	2.4	87
6	2.3	47
5	2.2	27
4	2.1	15
3	2.0	7
2	1.9	2
1	1.8	1
0	0.0	2148

The above table shows that 61 restaurants have the highest & best ratings. On the other hand a big number of restaurants (2148) do not even have any ratings.

In [95]:

```
Country_Rating_count=pd.DataFrame(df.groupby(['Country','Aggregate_rating']).Restaurant_Nam
Country_Rating_count.sort_values(by='Aggregate_rating', ascending = False)
```

Out[95]:

	Country	Aggregate_rating	Restaurant_Name
221	United States	4.9	14
92	New Zealand	4.9	2
139	South Africa	4.9	3
151	Sri Lanka	4.9	1
32	Brazil	4.9	3
163	Turkey	4.9	3
101	Phillipines	4.9	3
114	Qatar	4.9	1
179	UAE	4.9	4
79	Indonesia	4.9	4
199	United Kingdom	4.9	4
69	India	4.9	19
91	New Zealand	4.8	1
198	United Kingdom	4.8	1
178	UAE	4.8	2
100	Phillipines	4.8	4
68	India	4.8	9
31	Brazil	4.8	3
220	United States	4.8	3
138	South Africa	4.8	2
90	New Zealand	4.7	2
99	Phillipines	4.7	1
113	Qatar	4.7	1
67	India	4.7	16
30	Brazil	4.7	2
137	South Africa	4.7	1
177	UAE	4.7	2
219	United States	4.7	9
197	United Kingdom	4.7	5
162	Turkey	4.7	3
...	...	...	...
50	India	3.0	465
13	Brazil	3.0	1
33	Canada	3.0	1

	Country	Aggregate_rating	Restaurant_Name
115	Singapore	3.0	1
2	Australia	2.9	1
49	India	2.9	380
48	India	2.8	314
181	United Kingdom	2.8	1
47	India	2.7	250
46	India	2.6	190
1	Australia	2.6	1
45	India	2.5	109
141	Sri Lanka	2.5	1
44	India	2.4	83
0	Australia	2.4	1
202	United States	2.4	1
140	Sri Lanka	2.4	1
164	UAE	2.4	1
43	India	2.3	46
80	New Zealand	2.3	1
42	India	2.2	26
201	United States	2.2	1
41	India	2.1	15
40	India	2.0	7
39	India	1.9	2
38	India	1.8	1
200	United States	0.0	3
12	Brazil	0.0	5
180	United Kingdom	0.0	1
37	India	0.0	2139

222 rows × 3 columns

In [98]:

```
# Countrywise best rated restaurants
Country_wise_top_rating=pd.DataFrame(Country_Rating_count[Country_Rating_count.Aggregate_ra
Country_wise_top_rating.sort_values(by='Aggregate_rating' , ascending = False)
```

Out[98]:

	index	Country	Aggregate_rating	Restaurant_Name
0	32	Brazil	4.9	3
1	69	India	4.9	19
2	79	Indonesia	4.9	4
3	92	New Zealand	4.9	2
4	101	Phillipines	4.9	3
5	114	Qatar	4.9	1
6	139	South Africa	4.9	3
7	151	Sri Lanka	4.9	1
8	163	Turkey	4.9	3
9	179	UAE	4.9	4
10	199	United Kingdom	4.9	4
11	221	United States	4.9	14

```
The above tell us that the maximum number of best rated restaurants are in India, then
United States. Out of 15, 12 Countries have at least 1 or above best rated restaurant/s.
```

In [100]:

```
# Rating distribution basis Rating Type (Excellent, Very Good, Good, Average, Poor and Not
RatingType_count=pd.DataFrame(df.groupby('Rating_text').Restaurant_Name.count()).reset_inde
RatingType_count
```

Out[100]:

	Rating_text	Restaurant_Name
0	Average	3737
1	Excellent	301
2	Good	2100
3	Not rated	2148
4	Poor	186
5	Very Good	1078

```
It states that accross countries & cities only 301 restaurants have 'Excellent' rating.
```

In [102]:

```
Country_RatingType_count=pd.DataFrame(df.groupby(['Country','Rating_text']).Restaurant_Name
Country_RatingType_count
```

Out[102]:

	Country	Rating_text	Restaurant_Name
0	Australia	Average	4
1	Australia	Excellent	1
2	Australia	Good	13
3	Australia	Poor	1
4	Australia	Very Good	5
5	Brazil	Average	8
6	Brazil	Excellent	16
7	Brazil	Good	11
8	Brazil	Not rated	5
9	Brazil	Very Good	20
10	Canada	Average	2
11	Canada	Good	1
12	Canada	Very Good	1
13	India	Average	3678
14	India	Excellent	116
15	India	Good	1847
16	India	Not rated	2139
17	India	Poor	180
18	India	Very Good	691
19	Indonesia	Average	1
20	Indonesia	Excellent	7
21	Indonesia	Good	3
22	Indonesia	Very Good	10
23	New Zealand	Excellent	12
24	New Zealand	Good	2
25	New Zealand	Poor	1
26	New Zealand	Very Good	25
27	Phillipines	Excellent	12
28	Phillipines	Good	1
29	Phillipines	Very Good	9
...	...	...	...
36	Singapore	Very Good	3
37	South Africa	Average	1
38	South Africa	Excellent	12

	Country	Rating_text	Restaurant_Name
39	South Africa	Good	12
40	South Africa	Very Good	35
41	Sri Lanka	Average	2
42	Sri Lanka	Excellent	2
43	Sri Lanka	Good	4
44	Sri Lanka	Poor	1
45	Sri Lanka	Very Good	11
46	Turkey	Average	1
47	Turkey	Excellent	10
48	Turkey	Good	3
49	Turkey	Very Good	20
50	UAE	Average	1
51	UAE	Excellent	18
52	UAE	Good	9
53	UAE	Poor	1
54	UAE	Very Good	31
55	United Kingdom	Average	5
56	United Kingdom	Excellent	23
57	United Kingdom	Good	20
58	United Kingdom	Not rated	1
59	United Kingdom	Very Good	31
60	United States	Average	23
61	United States	Excellent	68
62	United States	Good	159
63	United States	Not rated	3
64	United States	Poor	2
65	United States	Very Good	179

66 rows × 3 columns

In [105]:

```
Country_wise_ratingType=pd.DataFrame(Country_RatingType_count[Country_RatingType_count.Rati
Country_wise_ratingType.sort_values(by= 'Rating_text', ascending = False)
```

Out[105]:

	index	Country	Rating_text	Restaurant_Name
0	1	Australia	Excellent	1
1	6	Brazil	Excellent	16
2	14	India	Excellent	116
3	20	Indonesia	Excellent	7
4	23	New Zealand	Excellent	12
5	27	Phillipines	Excellent	12
6	31	Qatar	Excellent	4
7	38	South Africa	Excellent	12
8	42	Sri Lanka	Excellent	2
9	47	Turkey	Excellent	10
10	51	UAE	Excellent	18
11	56	United Kingdom	Excellent	23
12	61	United States	Excellent	68

```
After disbuting the rating by its type, its clear that India has maximum numbers of
restaurants with 'Excellent' rating.
```

In [106]:

```
Rating_corr=df[['Restaurant_ID','Restaurant_Name','Country','City','Aggregate_rating','Aver
                'Price_range','Has_Table_booking','Has_Online_delivery']]
```

In [107]:

```
dummy = ['Has_Table_booking','Has_Online_delivery'] # 0 indicates 'NO' and 1 indicates 'YES
Rating_corr = pd.get_dummies(Rating_corr,columns=dummy,drop_first=True)
Rating_corr = Rating_corr.merge(CuisineCount_by_res,left_on='Restaurant_Name',right_on='Res
Rating_corr.head()
```

Out[107]:

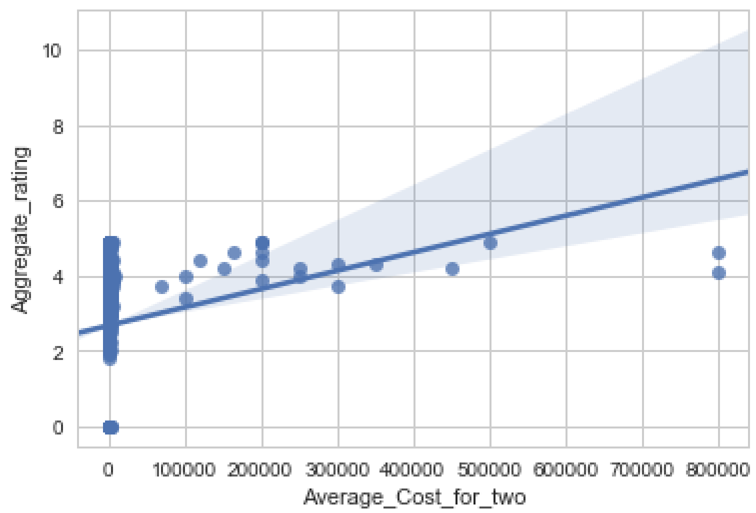
	Restaurant_ID	Restaurant_Name	Country	City	Aggregate_rating	Average_Cost_for_two
0	7402935	Skye	Indonesia	Jakarta	4.1	800000
1	7410290	Satoo - Hotel Shangri-La	Indonesia	Jakarta	4.6	800000
2	7420899	Sushi Masa	Indonesia	Jakarta	4.9	500000
3	7421967	3 Wise Monkeys	Indonesia	Jakarta	4.2	450000
4	7422489	Avec Moi Restaurant and Bar	Indonesia	Jakarta	4.3	350000

In [108]:

```
# Correlation between "Rating" and "Average cost for two"
sns.regplot(x='Average_Cost_for_two',y='Aggregate_rating',data=Rating_corr)
Rating_corr[["Average_Cost_for_two", "Aggregate_rating"]].corr()
```

Out[108]:

	Average_Cost_for_two	Aggregate_rating
Average_Cost_for_two	1.000000	0.051797
Aggregate_rating	0.051797	1.000000



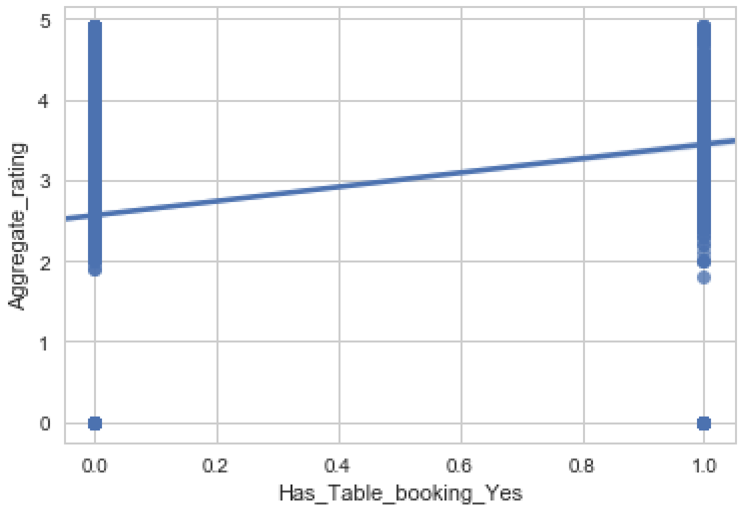
The above charts shows a weak positive correlation between Average cost for two and Rating of the restaurants

In [109]:

```
# Correlation between "Rating" and "Has Table Booking" - 0 indicates 'NO' and 1 indicates '
sns.regplot(x='Has_Table_booking_Yes',y='Aggregate_rating',data=Rating_corr)
Rating_corr[['Has_Table_booking_Yes','Aggregate_rating']].corr()
```

Out[109]:

	Has_Table_booking_Yes	Aggregate_rating
Has_Table_booking_Yes	1.000000	0.190045
Aggregate_rating	0.190045	1.000000



There is a good positive correlation between table booking option and Rating. This could be one of the basis for good rating

In [110]:

```
# Correlation between "Rating" and "Has Online delivery" - 0 indicates 'NO' and 1 indicates
sns.regplot(x='Has_Online_delivery_Yes',y='Aggregate_rating',data=Rating_corr)
Rating_corr[['Has_Online_delivery_Yes','Aggregate_rating']].corr()
```

Out[110]:

	Has_Online_delivery_Yes	Aggregate_rating
Has_Online_delivery_Yes	1.000000	0.225772
Aggregate_rating	0.225772	1.000000

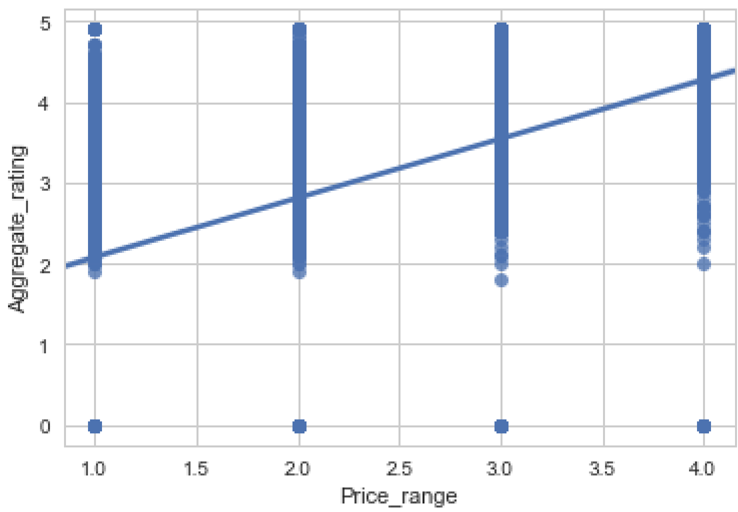
There is a good positive correlation between online delivry option and Rating. This could be another big factor for giving a good rating to a restaurant

In [111]:

```
# Correlation between "Rating" and "Price Range"
sns.regplot(x='Price_range',y='Aggregate_rating',data=Rating_corr)
Rating_corr[['Price_range','Aggregate_rating']].corr()
```

Out[111]:

	Price_range	Aggregate_rating
Price_range	1.000000	0.437874
Aggregate_rating	0.437874	1.000000



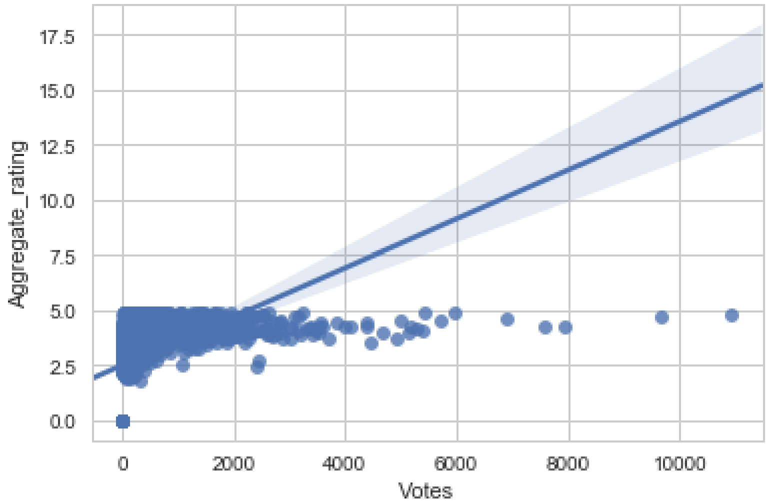
Price range has very positive correlation with rating, as with increasing price range rating also gets better. This is clearly a major factor for deciding the rating.

In [112]:

```
# Correlation between "Rating" and "Votes"
sns.regplot(x='Votes',y='Aggregate_rating',data=Rating_corr)
Rating_corr[['Votes','Aggregate_rating']].corr()
```

Out[112]:

	Votes	Aggregate_rating
Votes	1.000000	0.313598
Aggregate_rating	0.313598	1.000000



Similar to price, Votes also have a very positive correlation with rating, as with increasing price range rating also gets better.

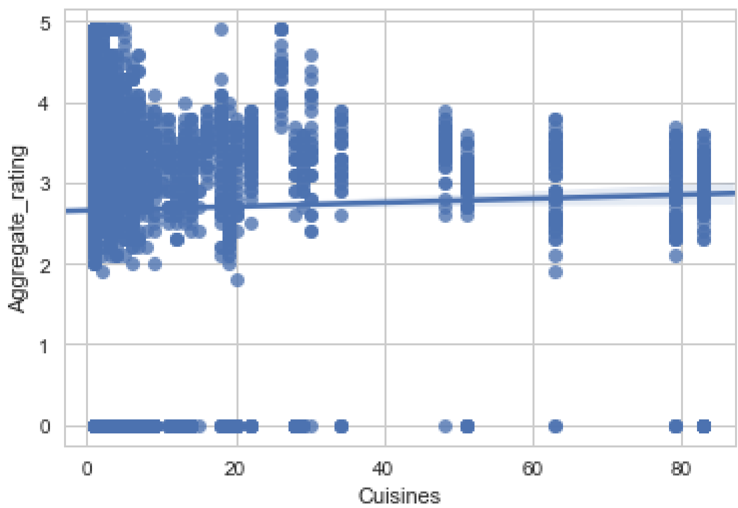
This also a major factor for deciding the rating.

In [113]:

```
# Correlation between "Rating" and "Cuisines Count"
sns.regplot(x='Cuisines',y='Aggregate_rating',data=Rating_corr)
Rating_corr[['Cuisines','Aggregate_rating']].corr()
```

Out[113]:

	Cuisines	Aggregate_rating
Cuisines	1.000000	0.021097
Aggregate_rating	0.021097	1.000000



Number of Cuisines provided by a restaurant is worst correlated to Rating. There is very minimal relation. Hence this cannot be the deciding factor for Rating.