zomato online transaction analysis

December 15, 2023

[11]:
```
## import the required libraries
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import numpy as np
```

[20]:
```
data = pd.read_csv("C:/Users/Vikas/Downloads/zomato.csv",encoding='latin-1')
```

[21]:
```
data.head(2)
```

[21]:
```
   Restaurant ID   Restaurant Name  Country Code         City  \
0        6317637  Le Petit Souffle           162  Makati City
1        6304287  Izakaya Kikufuji           162  Makati City

                                            Address  \
0  Third Floor, Century City Mall, Kalayaan Avenu…
1  Little Tokyo, 2277 Chino Roces Avenue, Legaspi…

                                     Locality  \
0   Century City Mall, Poblacion, Makati City
1  Little Tokyo, Legaspi Village, Makati City

                                    Locality Verbose   Longitude   Latitude  \
0  Century City Mall, Poblacion, Makati City, Mak…  121.027535  14.565443
1  Little Tokyo, Legaspi Village, Makati City, Ma…  121.014101  14.553708

                     Cuisines  …          Currency Has Table booking  \
0  French, Japanese, Desserts  …  Botswana Pula(P)               Yes
1                    Japanese  …  Botswana Pula(P)               Yes

  Has Online delivery Is delivering now Switch to order menu Price range  \
0                  No                No                   No           3
1                  No                No                   No           3

   Aggregate rating  Rating color Rating text Votes
0               4.8    Dark Green   Excellent   314
1               4.5    Dark Green   Excellent   591
```

```
[2 rows x 21 columns]
```

[22]:
```
data.columns
```

[22]:
```
Index(['Restaurant ID', 'Restaurant Name', 'Country Code', 'City', 'Address',
       'Locality', 'Locality Verbose', 'Longitude', 'Latitude', 'Cuisines',
       'Average Cost for two', 'Currency', 'Has Table booking',
       'Has Online delivery', 'Is delivering now', 'Switch to order menu',
       'Price range', 'Aggregate rating', 'Rating color', 'Rating text',
       'Votes'],
      dtype='object')
```

[23]:
```
data.shape
```

[23]:
```
(9551, 21)
```

[24]:
```
data.info()
```

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

[25]:
```
data.isnull().sum()
```

[25]:
```
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
```

[26]:
```
## description about the data
data.describe()
```

[26]:

	Restaurant ID	Country Code	Longitude	Latitude
count	9.551000e+03	9551.000000	9551.000000	9551.000000
mean	9.051128e+06	18.365616	64.126574	25.854381
std	8.791521e+06	56.750546	41.467058	11.007935
min	5.300000e+01	1.000000	-157.948486	-41.330428
25%	3.019625e+05	1.000000	77.081343	28.478713
50%	6.004089e+06	1.000000	77.191964	28.570469
75%	1.835229e+07	1.000000	77.282006	28.642758
max	1.850065e+07	216.000000	174.832089	55.976980

	Average Cost for two	Price range	Aggregate rating	Votes
count	9551.000000	9551.000000	9551.000000	9551.000000
mean	1199.210763	1.804837	2.666370	156.909748
std	16121.183073	0.905609	1.516378	430.169145
min	0.000000	1.000000	0.000000	0.000000
25%	250.000000	1.000000	2.500000	5.000000
50%	400.000000	2.000000	3.200000	31.000000
75%	700.000000	2.000000	3.700000	131.000000
max	800000.000000	4.000000	4.900000	10934.000000

1 in data analysis what all things we do

1.missing value 2.wxplore abount numerical variable 3.explore about the catogorical variable 4.finding the relatioship between features

[27]:
```
data.isnull().sum()
```

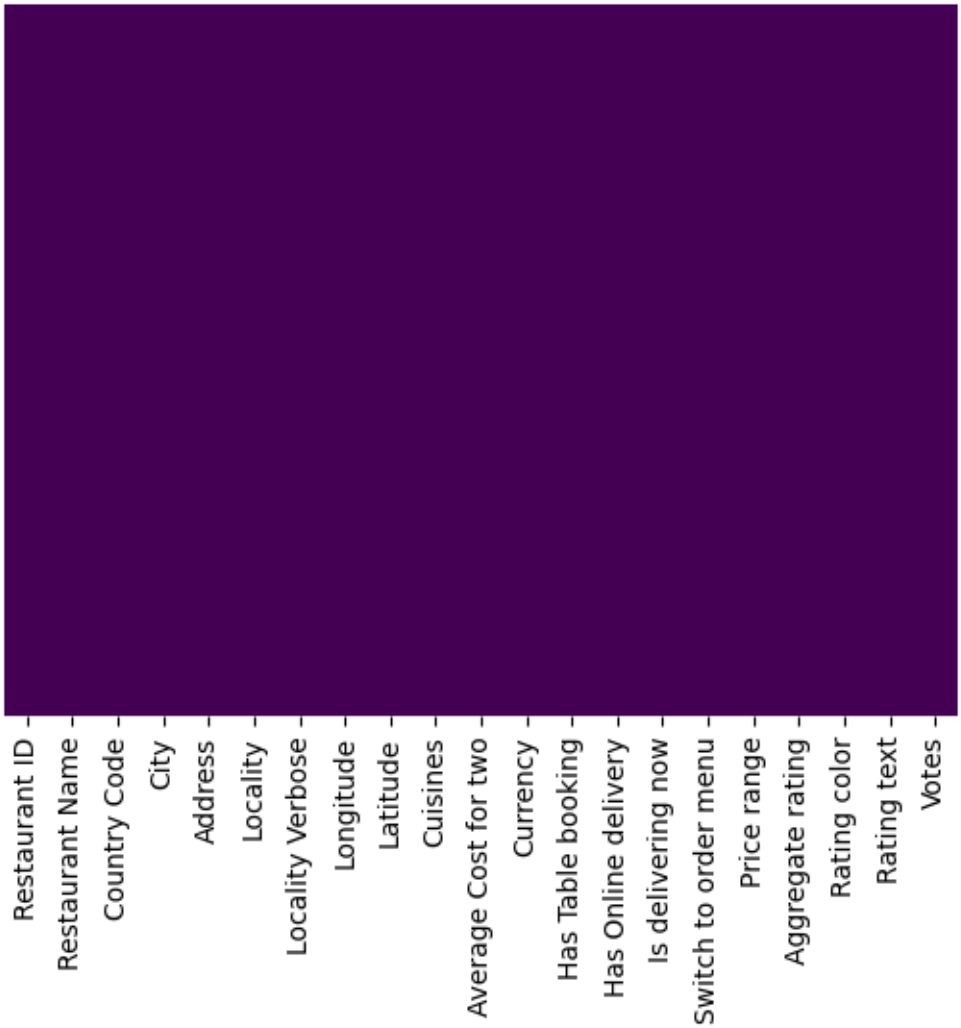
[27]:
```
Restaurant ID         0
Restaurant Name       0
Country Code          0
City                  0
Address               0
Locality              0
Locality Verbose      0
Longitude             0
Latitude              0
Cuisines              9
Average Cost for two  0
Currency              0
Has 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
```

[29]:
```
## another way of finding coloumns missing values
[features for features in data.columns if data[features].isnull().sum()>0]
```

[29]:
```
['Cuisines']
```

[30]:
```
sns.heatmap(data.isnull(),yticklabels=False,cbar=False,cmap='viridis')
```

[30]:
```
<Axes: >
```



[35]:
```
country = pd.read_excel('C:/Users/Vikas/Downloads/Country-Code.xlsx')
```

[42]:
```
country.head()
```

[42]:

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

[44]:
```
country.columns
```

[44]:
```
Index(['Country Code', 'Country'], dtype='object')
```

[45]:
```
## merge two dataframe
final_df=pd.merge(data,country,on='Country Code',how='left')
```

[46]:
```
final_df.head()
```

[46]:

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

```
  Aggregate rating  Rating color  Rating text Votes      Country
0              4.8    Dark Green    Excellent   314  Phillipines
1              4.5    Dark Green    Excellent   591  Phillipines
2              4.4         Green    Very Good   270  Phillipines
3              4.9    Dark Green    Excellent   365  Phillipines
4              4.8    Dark Green    Excellent   229  Phillipines

[5 rows x 22 columns]
```

[48]:
```
## to check datatype
final_df.dtypes
```

[48]:
```
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
Country                  object
dtype: object
```

[56]:
```
## i want to see each country how many records have
country_name = final_df.Country.value_counts()
country_name
```

[56]:
```
India             8652
United States      434
United Kingdom      80
Brazil              60
UAE                 60
South Africa        60
New Zealand         40
```

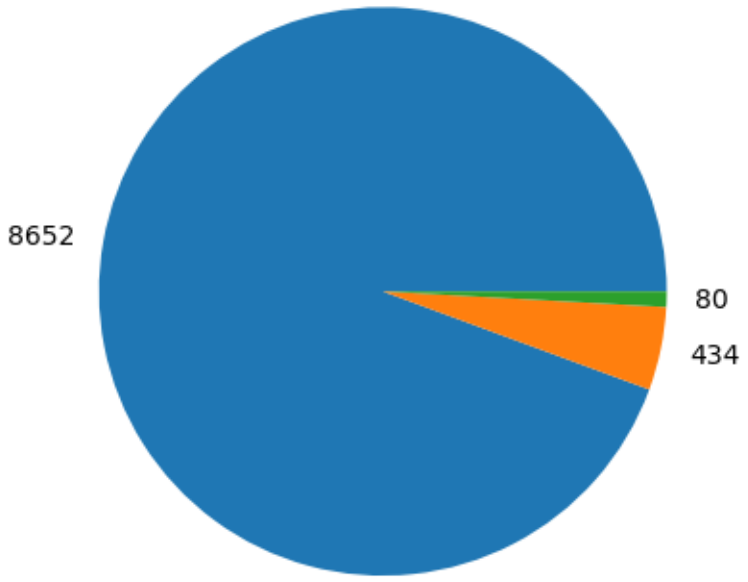
```
Turkey              34
Australia           24
Phillipines         22
Indonesia           21
Singapore           20
Qatar               20
Sri Lanka           20
Canada               4
Name: Country, dtype: int64
```

[57]:
```
country_values = final_df.Country.value_counts().values
country_values
```

[57]:
```
array([8652,  434,   80,   60,   60,   60,   40,   34,   24,   22,   21,
         20,   20,   20,    4], dtype=int64)
```

[59]:
```
## pie chart top 3 country that used as zomato
plt.pie(country_values[:3],labels=country_name[:3],autopct='%1.2f%%')
```

[59]:
```
([<matplotlib.patches.Wedge at 0x2478615a2f0>,
  <matplotlib.patches.Wedge at 0x2478615a200>,
  <matplotlib.patches.Wedge at 0x2478615a9e0>],
 [Text(-1.0829742700952103, 0.19278674827836725, '8652'),
  Text(1.077281715838356, -0.22240527134123297, '434'),
  Text(1.0995865153823035, -0.03015783794312073, '80')])
```



[62]:
```
## obeservations-zomato maximum records or transactions are from india
final_df.columns
```

[62]:
```
Index(['Restaurant ID', 'Restaurant Name', 'Country Code', 'City', 'Address',
       'Locality', 'Locality Verbose', 'Longitude', 'Latitude', 'Cuisines',
       'Average Cost for two', 'Currency', 'Has Table booking',
       'Has Online delivery', 'Is delivering now', 'Switch to order menu',
       'Price range', 'Aggregate rating', 'Rating color', 'Rating text',
       'Votes', 'Country'],
      dtype='object')
```

[72]:
```
Ratings = final_df.groupby(['Aggregate rating','Rating color','Rating text']).
 ↪size().reset_index().rename(columns={0:'Rating Count'})
Ratings
```

[72]:

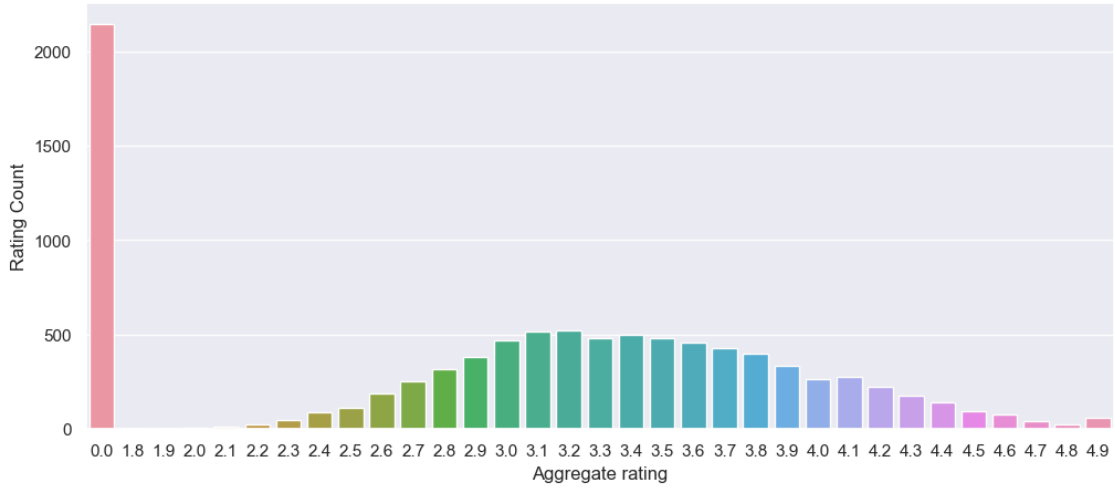
	Aggregate rating	Rating color	Rating text	Rating Count
0	0.0	White	Not rated	2148
1	1.8	Red	Poor	1
2	1.9	Red	Poor	2
3	2.0	Red	Poor	7
4	2.1	Red	Poor	15
5	2.2	Red	Poor	27
6	2.3	Red	Poor	47
7	2.4	Red	Poor	87
8	2.5	Orange	Average	110
9	2.6	Orange	Average	191
10	2.7	Orange	Average	250
11	2.8	Orange	Average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

```
29              4.6   Dark Green   Excellent           78
30              4.7   Dark Green   Excellent           42
31              4.8   Dark Green   Excellent           25
32              4.9   Dark Green   Excellent           61
```

[ ]:
```
## observations
1.when rating is between 4.5 to 4.9----->excelent
2,when rating is between 3.5 to 4.0 ---->very good
3.                        3.0 to 3.4        ----->everage
```

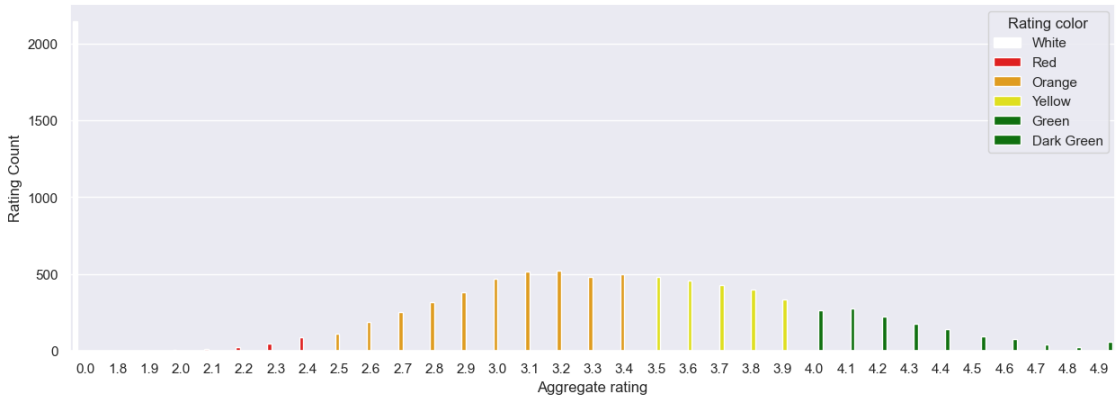
[82]:
```
plt.figure(figsize=(12, 5))
sns.barplot(x='Aggregate rating',y ='Rating Count',data = Ratings)
```

[82]:
```
<Axes: xlabel='Aggregate rating', ylabel='Rating Count'>
```



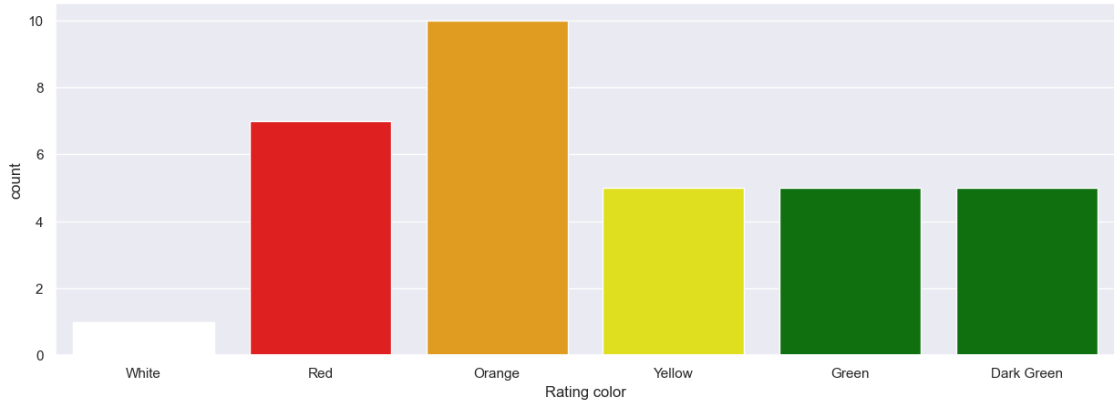
[89]:
```
sns.barplot(x='Aggregate rating',y ='Rating Count',hue='Rating color',data =␣
↪Ratings,palette=['white','red','orange','yellow','green','green'])
```

[89]:
```
<Axes: xlabel='Aggregate rating', ylabel='Rating Count'>
```



[101]:
```
## countplot
sns.countplot(x="Rating color",data=Ratings,palette=['white','red','orange','yellow','green','green'])
```

[101]:
```
<Axes: xlabel='Rating color', ylabel='count'>
```



[106]:
```
final_df[final_df['Rating color']=='White'].groupby('Country').size().reset_index().head(5)
```

[106]:

	Country	0
0	Brazil	5
1	India	2139
2	United Kingdom	1
3	United States	3

[ ]:
```
## Observations
##1.maximum numbers of rating from 0 customers
```

[110]:
```
## find out which currency is used by which coutry
final_df[['Country','Currency']].groupby(['Country','Currency']).size().
 ↪reset_index()
```

[110]:

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

[114]:
```
## which country do have online delivary
final_df[final_df['Has Online delivery']=='Yes'].Country.value_counts()
```

[114]:
```
India    2423
UAE        28
Name: Country, dtype: int64
```

[145]:
```

```

[122]:
```
final_df.columns
```

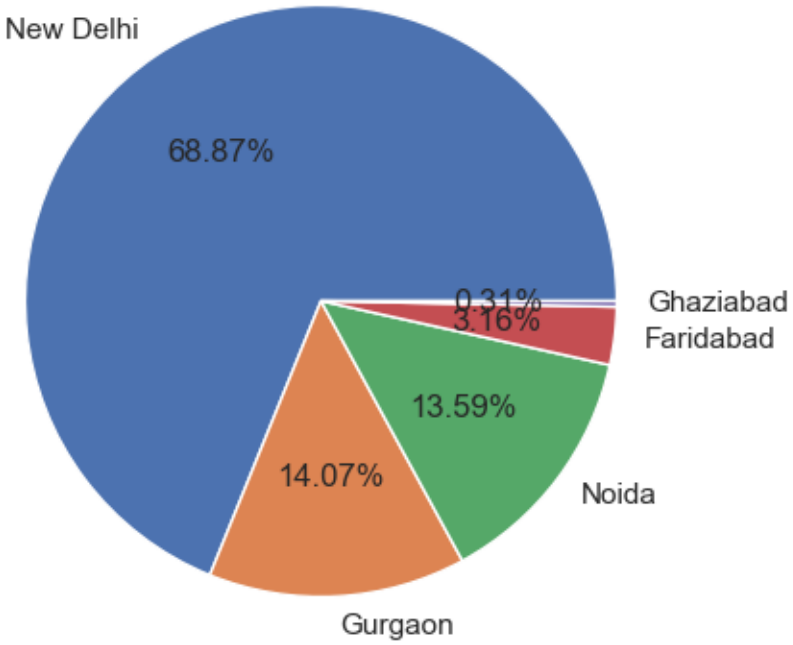
[122]:
```
Index(['Restaurant ID', 'Restaurant Name', 'Country Code', 'City', 'Address',
       'Locality', 'Locality Verbose', 'Longitude', 'Latitude', 'Cuisines',
       'Average Cost for two', 'Currency', 'Has Table booking',
       'Has Online delivery', 'Is delivering now', 'Switch to order menu',
       'Price range', 'Aggregate rating', 'Rating color', 'Rating text',
       'Votes', 'Country'],
      dtype='object')
```

[131]:
```
city_values = final_df.City.value_counts().values
city_labels = final_df.City.value_counts().index
```

[132]:
```
plt.pie(city_values[:5],labels=city_labels[:5],autopct='%1.2f%%')
```

[132]:
```
([<matplotlib.patches.Wedge at 0x2478f834c40>,
  <matplotlib.patches.Wedge at 0x2478f834b50>,
  <matplotlib.patches.Wedge at 0x2478f835900>,
```

```
 <matplotlib.patches.Wedge at 0x2478f835f30>,
 <matplotlib.patches.Wedge at 0x2478f8250c0>],
[Text(-0.6145352824185932, 0.9123301960708633, 'New Delhi'),
 Text(0.0623675251198054, -1.0982305276263407, 'Gurgaon'),
 Text(0.8789045225625368, -0.6614581167535246, 'Noida'),
 Text(1.0922218418223437, -0.13058119407559224, 'Faridabad'),
 Text(1.099946280005612, -0.010871113182029924, 'Ghaziabad')],
[Text(-0.3352010631374145, 0.497634652402289, '68.87%'),
 Text(0.0340186500653484, -0.5990348332507311, '14.07%'),
 Text(0.47940246685229276, -0.36079533641101336, '13.59%'),
 Text(0.5957573682667329, -0.07122610585941394, '3.16%'),
 Text(0.5999706981848791, -0.005929698099289049, '0.31%')])
```



[ ]:
```
observation---.
## Maximum transaction s is happening in dehli
```

[143]:
```
## top 10 cuissines
final_df.groupby('Cuisines').max(10).reset_index()
```

[143]:

	Cuisines	Restaurant ID	Country Code	\
0	Afghani	18481273	1	
1	Afghani, Mughlai, Chinese	18355119	1	

```
2                          Afghani, North Indian       18435789             1
3     Afghani, North Indian, Pakistani, Arabian         301675               1
4                                       African          75027             189
…                                             …             …             …
1820                      Western, Asian, Cafe       18391256              94
1821                 Western, Fusion, Fast Food       18479742             184
1822                               World Cuisine        6004011             208
1823           World Cuisine, Mexican, Italian        6000409             208
1824           World Cuisine, Patisserie, Cafe        5908749             208

       Longitude   Latitude  Average Cost for two  Price range  \
0      77.246764  28.581425                   700            2
1      77.245796  28.558387                   500            2
2      77.045289  28.439285                   900            2
3      77.266754  28.659755                   500            2
4      28.236047 -25.770748                   450            4
…            …         …                     …            …
1820  106.911335  -6.163948                250000            3
1821  103.858430   1.301707                    60            4
1822   32.869800  39.898239                    80            3
1823   32.865683  39.897872                   150            4
1824   29.041297  41.009847                   105            3

      Aggregate rating  Votes
0                  2.9     39
1                  0.0      2
2                  0.0      0
3                  0.0      3
4                  4.7    373
…                  …      …
1820               4.2    259
1821               3.2     32
1822               4.9     95
1823               4.4    115
1824               4.2   1034

[1825 rows x 9 columns]
```

[141]:

[144]:

[140]:

[ ]: