In [41]:
```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

```
Matplotlib is building the font cache; this may take a moment.
```

In [8]:
```
df=pd.read_csv(r'C:\Users\Admin\Desktop\dataset\zomato.csv', encoding='latin-1')
```

In [10]:
```
df.head(5)
```

Out[10]:

	Restaurant ID	Restaurant Name	Country Code	City	Address	Locality	Locality Verbose	Longitu
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...	121.0275
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...	121.0141
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...	121.0568
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...	121.0564
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...	121.0575

5 rows × 21 columns

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

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

In [12]:
```
df.isnull().sum()
```

Out[12]:
```
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
```

In [14]:
```
df.describe()
```

Out[14]:

	Restaurant ID	Country Code	Longitude	Latitude	Average Cost for two	Price range	Aggreg rati
count	9.551000e+03	9551.000000	9551.000000	9551.000000	9551.000000	9551.000000	9551.000(
mean	9.051128e+06	18.365616	64.126574	25.854381	1199.210763	1.804837	2.6663
std	8.791521e+06	56.750546	41.467058	11.007935	16121.183073	0.905609	1.5163
min	5.300000e+01	1.000000	-157.948486	-41.330428	0.000000	1.000000	0.000(
25%	3.019625e+05	1.000000	77.081343	28.478713	250.000000	1.000000	2.500(
50%	6.004089e+06	1.000000	77.191964	28.570469	400.000000	2.000000	3.200(
75%	1.835229e+07	1.000000	77.282006	28.642758	700.000000	2.000000	3.700(
max	1.850065e+07	216.000000	174.832089	55.976980	800000.000000	4.000000	4.900(

In [17]:
```
df.columns
```

Out[17]:
```
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')
```

In [18]:
```
df.shape
```

Out[18]:
```
(9551, 21)
```

In [19]:
```
[features for features in df.columns if df[features].isnull().sum()>0]
```

Out[19]:
```
['Cuisines']
```

In [25]:
```
df_country=pd.read_excel(r"C:\Users\Admin\Desktop\dataset\Country-Code.xlsx")
```

In [30]:
```
all_data=pd.merge(df,df_country,on='Country Code', how='left')
```

In [26]:
```
df_country.columns
```

Out[26]:
```
Index(['Country Code', 'Country'], dtype='object')
```

In [34]:
```
all_data.dtypes
```

Out[34]:
```
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
```

In [37]:
```
all_data.Country.value_counts()
```

Out[37]:
```
Country
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: count, dtype: int64
```

it show that zomato has more customer and businees in india copare to other countries

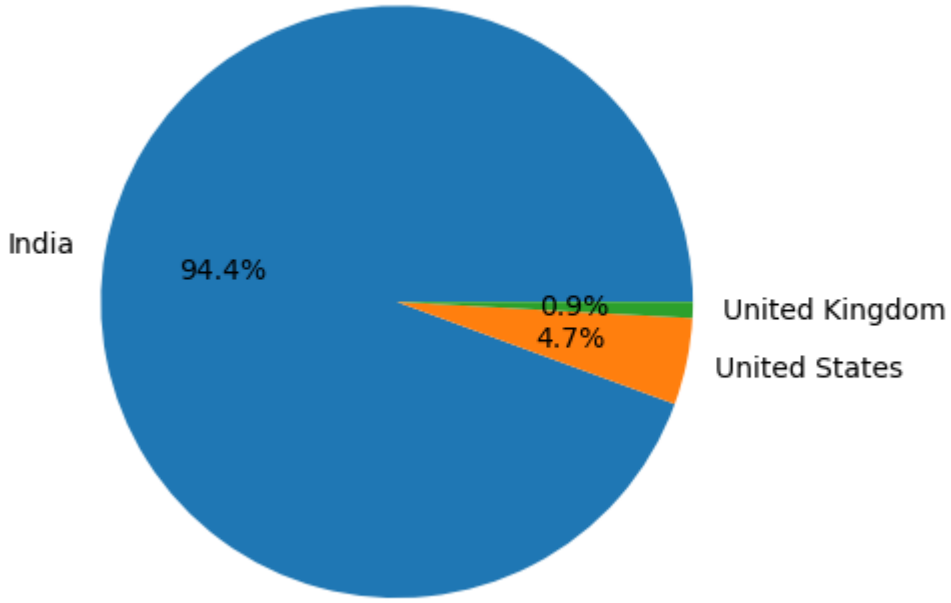
In [38]:
```
country_name=all_data.Country.value_counts().index
```

In [40]:
```
country_val=all_data.Country.value_counts().values
```

In [ ]:
```
plt
```

In [48]:
```
plt.pie(country_val[:3],labels=country_name[:3],autopct="%1.1f%%")
```

Out[48]:
```
([<matplotlib.patches.Wedge at 0x1e936fe9fd0>,
  <matplotlib.patches.Wedge at 0x1e936feb6d0>,
  <matplotlib.patches.Wedge at 0x1e936ffd410>],
 [Text(-1.0829742700952103, 0.19278674827836725, 'India'),
  Text(1.077281715838356, -0.22240527134123297, 'United States'),
  Text(1.0995865153823035, -0.03015783794312073, 'United Kingdom')],
 [Text(-0.590713238233751, 0.10515640815183668, '94.4%'),
  Text(0.5876082086391032, -0.12131196618612707, '4.7%'),
  Text(0.5997744629358018, -0.01644972978715676, '0.9%')])
```



In [49]:
```
all_data.columns
```

Out[49]:
```
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')
```

In [50]:
```
Ratings=all_data.groupby(['Aggregate rating','Rating color','Rating text']).size().
```

In [52]:
```
Ratings
```

Out[52]:

	Aggregate rating	Rating color	Rating text	Ratings 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

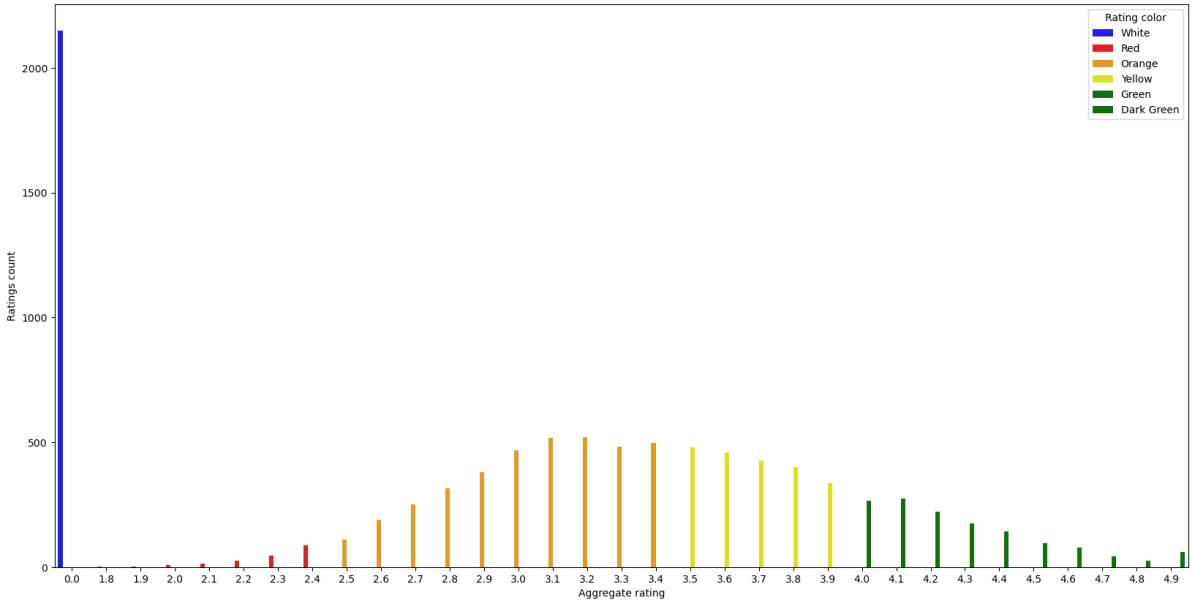
In [53]:
```
Ratings.head()
```

Out[53]:

	Aggregate rating	Rating color	Rating text	Ratings 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

In [57]:
```
fig1,ax1=plt.subplots(figsize=(20,10))
sns.barplot(x='Aggregate rating',y='Ratings count',data=Ratings,ax=ax1,hue='Rating
```

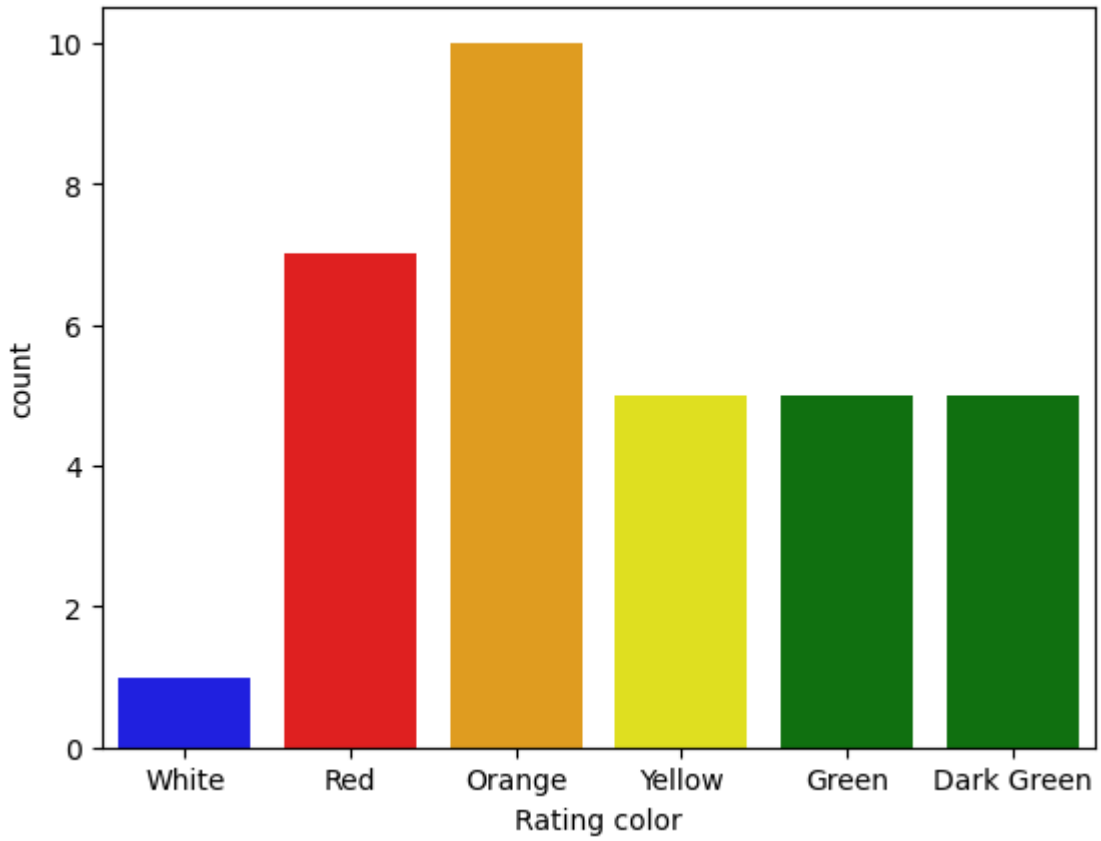
Out[57]:
```
<Axes: xlabel='Aggregate rating', ylabel='Ratings count'>
```



so non rating option is very high and rating count is lie in 2.7 to 4.1

In [58]:
```
sns.countplot(x='Rating color', data=Ratings,palette=['blue','red','orange','yellow
```

Out[58]:
```
<Axes: xlabel='Rating color', ylabel='count'>
```



so if we compare orange rating means "average" is higher then others colour ratings

In [59]:
```
all_data[all_data['Rating color']=='White'].groupby('Country').size().reset_index()
```

Out[59]:

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

max rating zero are come from india

In [60]:
```
all_data.groupby(['Currency','Country']).size().reset_index().rename(columns={0:'cc
```

Out[60]:

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

In [61]:
```
all_data[all_data['Has Online delivery']=='Yes'].groupby('Country').size().reset_ir
```

Out[61]:

	Country	Online Delivery
0	India	2423
1	UAE	28

it shows that max online deleivery in india

In [63]:
```
count_city=all_data[all_data['Has Online delivery']=='Yes'].groupby('City').size().
count_city
```

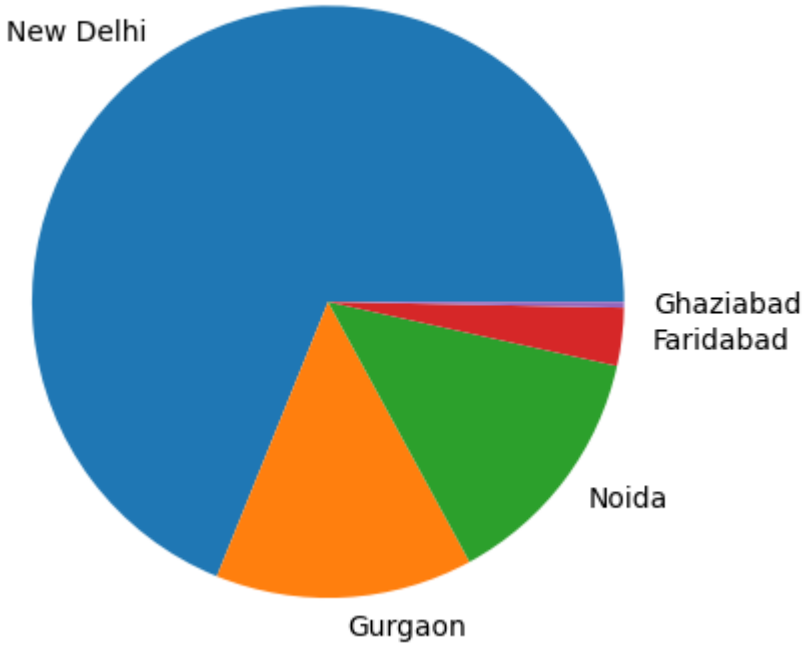
Out[63]:

	City	No of Restaurants
0	Abu Dhabi	11
1	Ahmedabad	11
2	Bangalore	7
3	Chandigarh	6
4	Chennai	13
5	Coimbatore	7
6	Dubai	6
7	Faridabad	35
8	Ghaziabad	10
9	Gurgaon	425
10	Hyderabad	7
11	Jaipur	10
12	Kochi	5
13	Kolkata	8
14	Mohali	1
15	Mumbai	7
16	Nagpur	10
17	New Delhi	1489
18	Noida	364
19	Pune	7
20	Secunderabad	1
21	Sharjah	11

In [67]:
```
count_values=all_data.City.value_counts().values
count_label=all_data.City.value_counts().index

plt.pie(count_values[:5],labels=count_label[:5])
```

Out[67]:
```
([<matplotlib.patches.Wedge at 0x1e939386190>,
  <matplotlib.patches.Wedge at 0x1e93c78e6d0>,
  <matplotlib.patches.Wedge at 0x1e93d3f6ed0>,
  <matplotlib.patches.Wedge at 0x1e93d3f7b50>,
  <matplotlib.patches.Wedge at 0x1e93d404410>],
 [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')])
```



In [68]:
```
cuisine1=all_data.Cuisines.value_counts().index
cuisine1
```

Out[68]:
```
Index(['North Indian', 'North Indian, Chinese', 'Chinese', 'Fast Food',
       'North Indian, Mughlai', 'Cafe', 'Bakery',
       'North Indian, Mughlai, Chinese', 'Bakery, Desserts', 'Street Food',
       ...
       'Cafe, Pizza, Burger',
       'Healthy Food, Continental, Juices, Beverages, Italian, Salad, Lebanese',
       'Goan, American, Portuguese', 'South Indian, Desserts, Beverages',
       'Healthy Food, North Indian, Italian, Salad', 'Bengali, Fast Food',
       'North Indian, Rajasthani, Asian',
       'Chinese, Thai, Malaysian, Indonesian',
       'Bakery, Desserts, North Indian, Bengali, South Indian',
       'Italian, World Cuisine'],
      dtype='object', name='Cuisines', length=1825)
```

In [69]:
```
cuisine=all_data.Cuisines.value_counts()
cuisine.head(10)
```

Out[69]:
```
Cuisines
North Indian                          936
North Indian, Chinese                 511
Chinese                               354
Fast Food                             354
North Indian, Mughlai                 334
Cafe                                  299
Bakery                                218
North Indian, Mughlai, Chinese        197
Bakery, Desserts                      170
Street Food                           149
Name: count, dtype: int64
```

Dhruv Tyagi

In [ ]:
```

```