

```
In [1]: 1 import pandas as pd
2 import numpy as np
3 import matplotlib.pyplot as plt
4 import seaborn as sns
5 import plotly.express as px
```

```
In [2]: 1 df = pd.read_csv(r"X:\Data Science\Udemy Projects\Zomato/zomato.csv")
```

```
In [3]: 1 df.head()
```

Out[3]:

		url	address	name	online_order	book_table	rate	votes	phone	location	rest_type	dish_li
0	https://www.zomato.com/bangalore/jalsa-banash...	942, 21st Main Road, 2nd Stage, Banashankari, ...	Jalsa	Yes	Yes	4.1/5	775	42297555\r\n+91 9743772233	080	Banashankari	Casual Dining	Pa Lu Bu Ma: Paç Par La
1	https://www.zomato.com/bangalore/spice-elephan...	2nd Floor, 80 Feet Road, Near Big Bazaar, 6th ...	Spice Elephant	Yes	No	4.1/5	787	080 41714161	080 41714161	Banashankari	Casual Dining	Mon Lu Bu Choco Nirv Thai
2	https://www.zomato.com/SanchurroBangalore?cont...	1112, Next to KIMS Medical College, 17th Cross...	San Churro Cafe	Yes	No	3.8/5	918	+91 9663487993	+91 9663487993	Banashankari	Cafe, Casual Dining	Chur Cannell Minestr Soup, Ch
3	https://www.zomato.com/bangalore/addhuri-udipi...	1st Floor, Annakuteera, 3rd Stage, Banashankar...	Addhuri Udupi Bhojana	No	No	3.7/5	88	+91 9620009302	+91 9620009302	Banashankari	Quick Bites	Ma: D
4	https://www.zomato.com/bangalore/grand-village...	10, 3rd Floor, Lakshmi Associates, Gandhi Baza...	Grand Village	No	No	3.8/5	166	8026612447\r\n+91 9901210005	+91 9901210005	Basavanagudi	Casual Dining	Pani Gol Ga

```
In [4]: 1 df.columns
```

```
Out[4]: Index(['url', 'address', 'name', 'online_order', 'book_table', 'rate', 'votes',
   'phone', 'location', 'rest_type', 'dish_liked', 'cuisines',
   'approx_cost(for two people)', 'reviews_list', 'menu_item',
   'listed_in(type)', 'listed_in(city)'],
  dtype='object')
```

```
In [5]: 1 df.dtypes
```

```
Out[5]: url          object
address        object
name           object
online_order    object
book_table     object
rate            object
votes           int64
phone           object
location        object
rest_type       object
dish_liked      object
cuisines        object
approx_cost(for two people)    object
reviews_list    object
menu_item       object
listed_in(type)  object
listed_in(city)  object
dtype: object
```

```
In [6]: 1 df.shape
```

```
Out[6]: (51717, 17)
```

## Data Cleaning

```
In [7]: 1 df.isnull().sum()
```

```
Out[7]: url 0
address 0
name 0
online_order 0
book_table 0
rate 7775
votes 0
phone 1208
location 21
rest_type 227
dish_liked 28078
cuisines 45
approx_cost(for two people) 346
reviews_list 0
menu_item 0
listed_in(type) 0
listed_in(city) 0
dtype: int64
```

```
In [8]: 1 feature_na = [feature for feature in df.columns if df[feature].isnull().sum()>0]
```

```
In [9]: 1 feature_na
```

```
Out[9]: ['rate',
'phone',
'location',
'rest_type',
'dish_liked',
'cuisines',
'approx_cost(for two people)']
```

```
In [10]: 1 for feature in feature_na:
2     print(' {} has {} missing values'.format(feature,np.round(df[feature].isnull().sum()/len(df)*100,3)))
```

```
rate has 15.034 missing values
phone has 2.336 missing values
location has 0.041 missing values
rest_type has 0.439 missing values
dish_liked has 54.292 missing values
cuisines has 0.087 missing values
approx_cost(for two people) has 0.669 missing values
```

```
In [11]: 1 df['rate'].unique()
```

```
Out[11]: array(['4.1/5', '3.8/5', '3.7/5', '3.6/5', '4.6/5', '4.0/5', '4.2/5',
'3.9/5', '3.1/5', '3.0/5', '3.2/5', '3.3/5', '2.8/5', '4.4/5',
'4.3/5', 'NEW', '2.9/5', '3.5/5', nan, '2.6/5', '3.8 /5', '3.4/5',
'4.5/5', '2.5/5', '2.7/5', '4.7/5', '2.4/5', '2.2/5', '2.3/5',
'3.4 /5', '-', '3.6 /5', '4.8/5', '3.9 /5', '4.2 /5', '4.0 /5',
'4.1 /5', '3.7 /5', '3.1 /5', '2.9 /5', '3.3 /5', '2.8 /5',
'3.5 /5', '2.7 /5', '2.5 /5', '3.2 /5', '2.6 /5', '4.5 /5',
'4.3 /5', '4.4 /5', '4.9/5', '2.1/5', '2.0/5', '1.8/5', '4.6 /5',
'4.9 /5', '3.0 /5', '4.8 /5', '2.3 /5', '4.7 /5', '2.4 /5',
'2.1 /5', '2.2 /5', '2.0 /5', '1.8 /5'], dtype=object)
```

```
In [12]: 1 df.dropna(axis='index',subset=['rate'],inplace = True)
```

```
In [13]: 1 df.shape
```

```
Out[13]: (43942, 17)
```

```
In [14]: 1 def split(x):
2     return x.split('/')[0]
```

```
In [15]: 1 df['rate'] = df['rate'].apply(split)
```

In [16]: 1 df.head()

Out[16]:

		url	address	name	online_order	book_table	rate	votes	phone	location	rest_type	dish_lik
0		https://www.zomato.com/bangalore/jalsa-banashankari	942, 21st Main Road, 2nd Stage, Banashankari, ...	Jalsa	Yes	Yes	4.1	775	080 42297555\n+91 9743772233	Banashankari	Casual Dining	Pas Lur Buf Mas Pap Pan Laj
1		https://www.zomato.com/bangalore/spice-elephant	2nd Floor, 80 Feet Road, Near Big Bazaar, 6th ...	Spice Elephant	Yes	No	4.1	787	080 41714161	Banashankari	Casual Dining	Mom Lur Buf Chocol Nirv Thai
2		https://www.zomato.com/SanchurroBangalore?cont...	1112, Next to KIMS Medical College, 17th Cross...	San Churro Cafe	Yes	No	3.8	918	+91 9663487993	Banashankari	Cafe, Casual Dining	Churr Cannell Minestr Soup, f Cho
3		https://www.zomato.com/bangalore/addhuri-udipi...	1st Floor, Annakuteera, 3rd Stage, Banashankari...	Addhuri Udupi Bhojana	No	No	3.7	88	+91 9620009302	Banashankari	Quick Bites	Mas Dc
4		https://www.zomato.com/bangalore/grand-village...	10, 3rd Floor, Lakshmi Associates, Gandhi Baza...	Grand Village	No	No	3.8	166	+91 8026612447\n+91 9901210005	Basavanagudi	Casual Dining	Panip Gol Gap

In [17]: 1 df['rate'].unique()

Out[17]: array(['4.1', '3.8', '3.7', '3.6', '4.6', '4.0', '4.2', '3.9', '3.1', '3.0', '3.2', '3.3', '2.8', '4.4', '4.3', 'NEW', '2.9', '3.5', '2.6', '3.8 ', '3.4', '4.5', '2.5', '2.7', '4.7', '2.4', '2.2', '2.3', '3.4 ', '-', '3.6 ', '4.8', '3.9 ', '4.2 ', '4.0 ', '4.1 ', '3.7 ', '3.1 ', '2.9 ', '3.3 ', '2.8 ', '3.5 ', '2.7 ', '2.5 ', '3.2 ', '2.6 ', '4.5 ', '4.3 ', '4.4 ', '4.9', '2.1', '2.0', '1.8', '4.6 ', '4.9 ', '3.0 ', '4.8 ', '2.3 ', '4.7 ', '2.4 ', '2.1 ', '2.2 ', '2.0 ', '1.8 '], dtype=object)

In [18]: 1 df.replace('NEW',0, inplace = True)

In [19]: 1 df.replace('-',0,inplace = True)

In [20]: 1 df['rate'].dtype

Out[20]: dtype('O')

In [21]: 1 df['rate'] = df['rate'].astype(float)

In [22]: 1 df['rate'].dtype

Out[22]: dtype('float64')

## Calculate Average rating of each restaurant

In [23]: 1 df\_rate = df.groupby('name')['rate'].mean().to\_frame().reset\_index()

In [24]: 1 df\_rate.columns=['restaurant','avg\_rating']

In [25]: 1 df\_rate.head(10)

Out[25]:

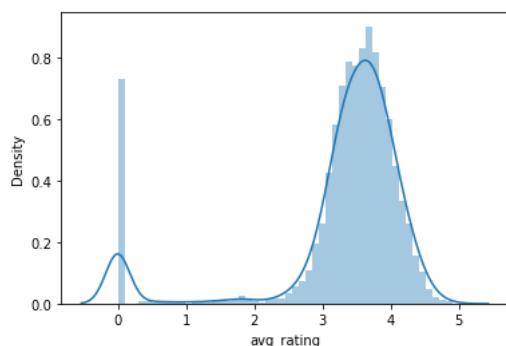
	restaurant	avg_rating
0	#FeelTheROLL	3.400
1	#L-81 Cafe	3.900
2	#refuel	3.700
3	1000 B.C	3.200
4	100°C	3.700
5	11 to 11 Express Biriyanis	3.500
6	1131 Bar + Kitchen	4.500
7	12th Main - Grand Mercure	4.100
8	1441 Pizzeria	4.100
9	1522 - The Pub	4.212

**Get distribution of rating column & try to find out what distribution this feature support ?**

In [26]: 1 sns.distplot(df\_rate['avg\_rating'])

```
C:\Users\Amir\AppData\Roaming\Python\Python39\site-packages\seaborn\distributions.py:2551: FutureWarning: `distplot` is a deprecated function and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).
    warnings.warn(msg, FutureWarning)
```

Out[26]: <AxesSubplot:xlabel='avg\_rating', ylabel='Density'>



**Top Restaurant chains in Bengalore**

In [27]: 1 df\_rate.shape

Out[27]: (7162, 2)

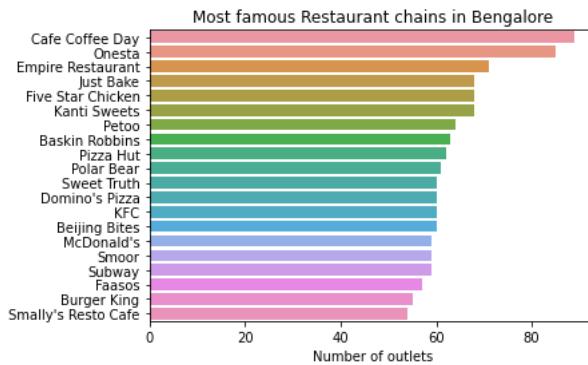
In [28]: 1 chains = df['name'].value\_counts()[0:20]

```
In [29]: 1 chains
```

```
Out[29]: Cafe Coffee Day      89
Onesta          85
Empire Restaurant  71
Just Bake       68
Five Star Chicken  68
Kanti Sweets     68
Petoo           64
Baskin Robbins    63
Pizza Hut        62
Polar Bear       61
Sweet Truth      60
Domino's Pizza    60
KFC              60
Beijing Bites     60
McDonald's        59
Smoor            59
Subway           59
Faasos           57
Burger King      55
Smally's Resto Cafe 54
Name: name, dtype: int64
```

```
In [30]: 1 sns.barplot(x=chains,y=chains.index)
2 plt.title('Most famous Restaurant chains in Bengalore')
3 plt.xlabel('Number of outlets')
```

```
Out[30]: Text(0.5, 0, 'Number of outlets')
```



## How many of the restaurant do not accept Online orders

```
In [31]: 1 df['online_order'].value_counts()
```

```
Out[31]: Yes    28308
No     15634
Name: online_order, dtype: int64
```

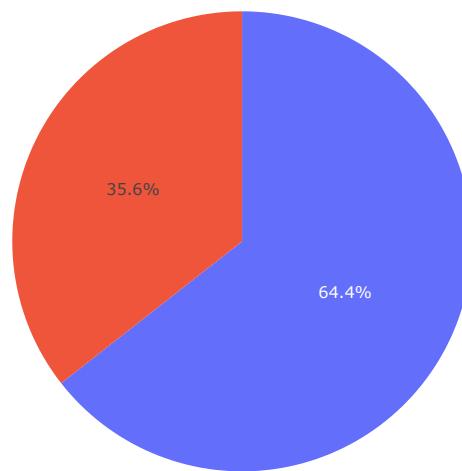
```
In [32]: 1 x = df['online_order'].value_counts()
2 x
```

```
Out[32]: Yes    28308
No     15634
Name: online_order, dtype: int64
```

```
In [33]: 1 labels = ['accepted', 'Not accepted']
```

```
In [34]: 1 px.pie(df, values=x, labels=labels, title= 'Pie Chart')
```

Pie Chart



## Ratio between Restaurant that provide table & do not provide table ?

```
In [35]: 1 df.head()
```

Out[35]:

	url	address	name	online_order	book_table	rate	votes	phone	location	rest_type	dish_lik
0	https://www.zomato.com/bangalore/jalsa-banashankari...	942, 21st Main Road, 2nd Stage, Banashankari, ...	Jalsa	Yes	Yes	4.1	775	080 42297555\\n+91 9743772233	Banashankari	Casual Dining	Pas Lur Buf Mas Pap Pan Laj
1	https://www.zomato.com/bangalore/spice-elephant...	2nd Floor, 80 Feet Road, Near Big Bazaar, 6th ...	Spice Elephant	Yes	No	4.1	787	080 41714161	Banashankari	Casual Dining	Mom Lur Buf Chocol Nirva Thai
2	https://www.zomato.com/SanchurroBangalore?cont...	1112, Next to KIMS Medical College, 17th Cross...	San Churro Cafe	Yes	No	3.8	918	+91 9663487993	Banashankari	Cafe, Casual Dining	Churr Cannell Minestr Soup, f Cho
3	https://www.zomato.com/bangalore/addhuri-udupi...	1st Floor, Annakuteera, 3rd Stage, Banashankar...	Addhuri Udupi Bhojana	No	No	3.7	88	+91 9620009302	Banashankari	Quick Bites	Mas Dc
4	https://www.zomato.com/bangalore/grand-village...	10, 3rd Floor, Lakshmi Associates, Gandhi Baza...	Grand Village	No	No	3.8	166	+91 8026612447\\n+91 9901210005	Basavanagudi	Casual Dining	Panip Gol Gap

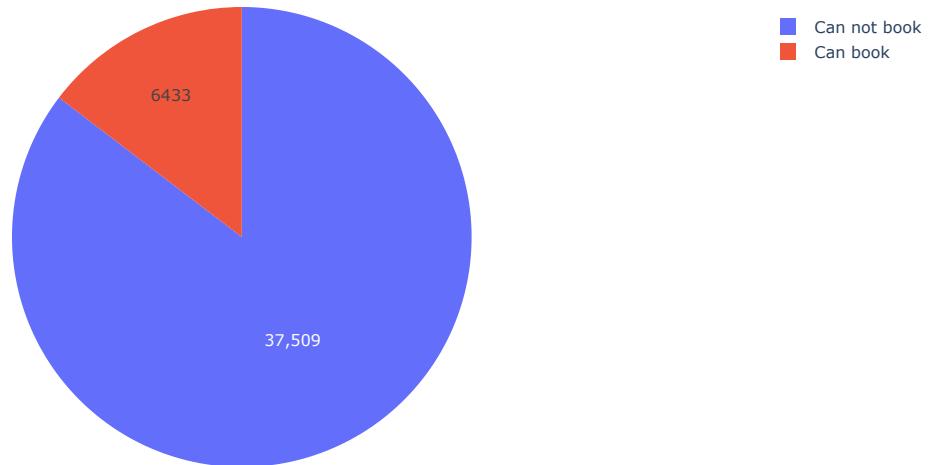
```
In [36]: 1 x = df['book_table'].value_counts()
2 x
```

Out[36]: No 37509  
Yes 6433  
Name: book\_table, dtype: int64

```
In [37]: 1 labels = ['Can not book', 'Can book']
```

```
In [38]: 1 import plotly.graph_objs as go  
2 from plotly.offline import iplot
```

```
In [39]: 1 trace1 = go.Pie(labels= labels, values= x, hoverinfo= 'label+percent', textinfo= 'value')  
2 iplot([trace1])
```



## Indepth Analysis of type of Restaurant we have

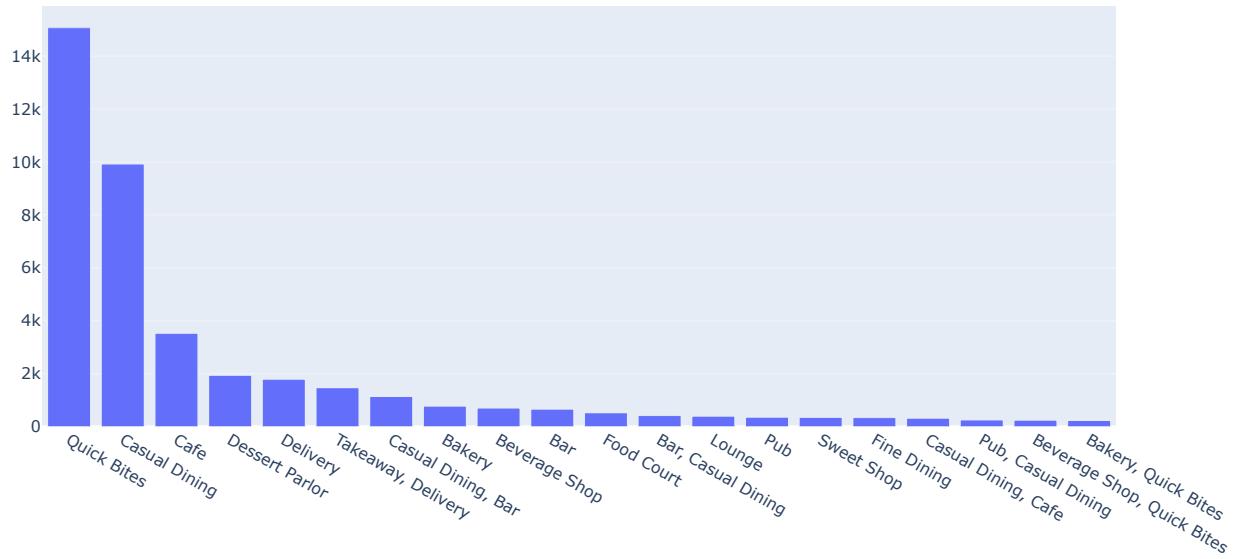
```
In [40]: 1 df['rest_type'].isna().sum()
```

Out[40]: 151

```
In [41]: 1 df['rest_type'].dropna(inplace=True)
```

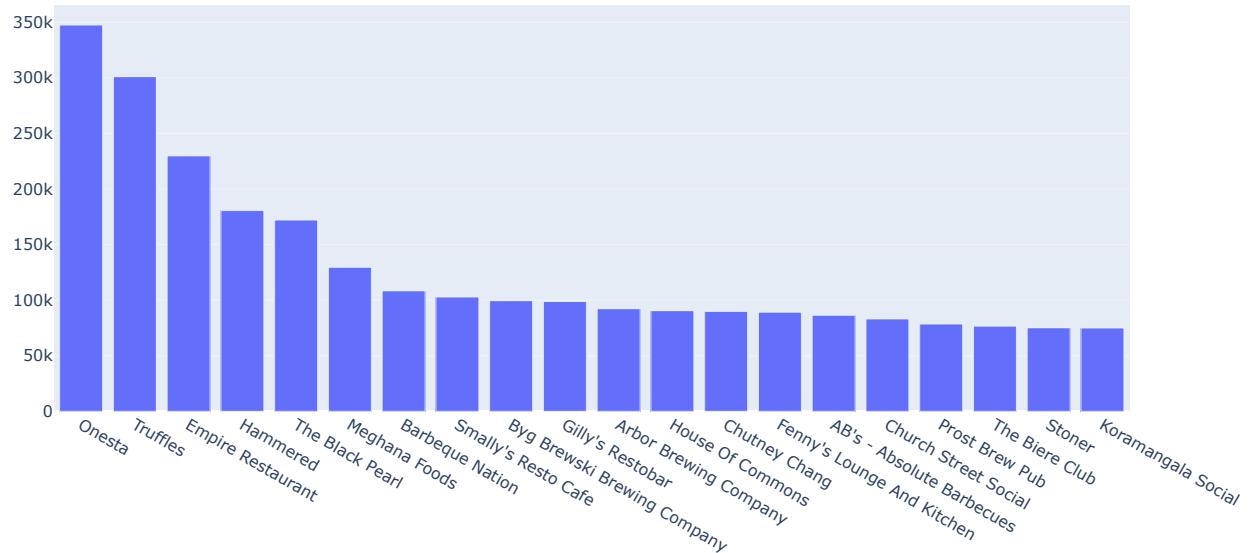
```
In [42]: 1 trace1 = go.Bar(x= df['rest_type'].value_counts().nlargest(20).index,  
2                      y = df['rest_type'].value_counts().nlargest(20))
```

```
In [43]: 1 iplot([trace1])
```



## Highest Voted Restaurant

```
In [44]: 1 trace2= go.Bar(x = df.groupby('name')[ 'votes'].sum().nlargest(20).index,
2                  y = df.groupby('name')[ 'votes'].sum().nlargest(20))
3 iplot([trace2])
```



## Total restaurant at different locations of Bengalore

```
In [45]: 1 restaurant = []
2 location = []
3 for key,location_df in df.groupby('location'):
4     location.append(key)
5     restaurant.append(len(location_df['name'].unique()))
```

```
In [46]: 1 df_total = pd.DataFrame(zip(location, restaurant))
2 df_total.columns= ['location','restaurant']
3 df_total.head()
```

Out[46]:

	location	restaurant
0	BTM	581
1	Banashankari	238
2	Banaswadi	151
3	Bannerghatta Road	362
4	Basavanagudi	195

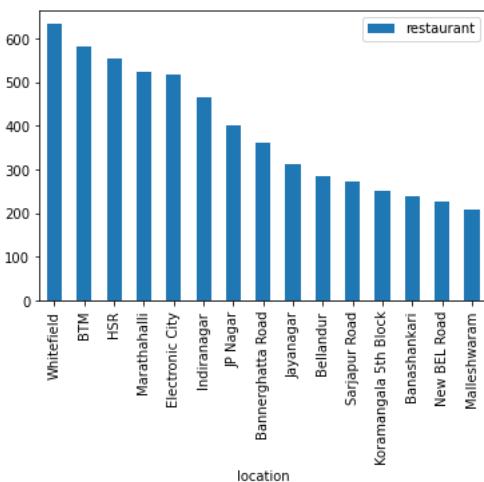
```
In [47]: 1 df_total.set_index('location', inplace = True)
2 df_total.head()
```

Out[47]:

	restaurant
location	
BTM	581
Banashankari	238
Banaswadi	151
Bannerghatta Road	362
Basavanagudi	195

```
In [48]: 1 df_total.sort_values(by= 'restaurant', ascending = False).head(15).plot.bar()
```

Out[48]: <AxesSubplot:xlabel='location'>



## Total Number of variety of restaurants in Bengalore

In [49]: 1 df.head()

Out[49]:

		url	address	name	online_order	book_table	rate	votes	phone	location	rest_type	dish_lik
0		https://www.zomato.com/bangalore/jalsa-banashankari...	942, 21st Main Road, 2nd Stage, Banashankari, ...	Jalsa	Yes	Yes	4.1	775	080 42297555\n+91 9743772233	Banashankari	Casual Dining	Pas Lur Buf Mas Pap Pan Laj
1		https://www.zomato.com/bangalore/spice-elephant...	2nd Floor, 80 Feet Road, Near Big Bazaar, 6th ...	Spice Elephant	Yes	No	4.1	787	080 41714161	Banashankari	Casual Dining	Mom Lur Buf Chocol Nirv Thai
2		https://www.zomato.com/SanchurroBangalore?cont...	1112, Next to KIMS Medical College, 17th Cross...	San Churro Cafe	Yes	No	3.8	918	+91 9663487993	Banashankari	Cafe, Casual Dining	Churr Cannell Minestr Soup, f Cho
3		https://www.zomato.com/bangalore/addhuri-udipi...	1st Floor, Annakuteera, 3rd Stage, Banashankari...	Addhuri Udupi Bhojana	No	No	3.7	88	+91 9620009302	Banashankari	Quick Bites	Mas Dc
4		https://www.zomato.com/bangalore/grand-village...	10, 3rd Floor, Lakshmi Associates, Gandhi Baza...	Grand Village	No	No	3.8	166	+91 8026612447\n+91 9901210005	Basavanagudi	Casual Dining	Panip Gol Gap

In [50]: 1 cuisines = df['cuisines'].value\_counts()[0:10]

In [51]: 1 cuisines

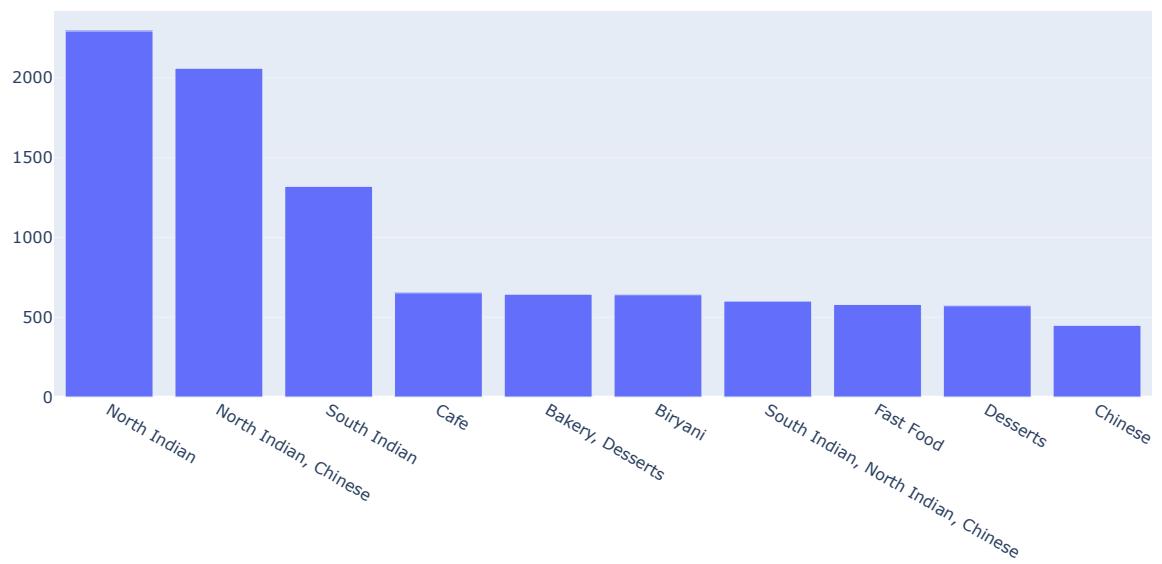
Out[51]:

North Indian	2294
North Indian, Chinese	2060
South Indian	1320
Cafe	653
Bakery, Desserts	644
Biryani	641
South Indian, North Indian, Chinese	601
Fast Food	580
Desserts	572
Chinese	449

Name: cuisines, dtype: int64

In [52]: 1 trace3 = go.Bar(x = df['cuisines'].value\_counts()[0:10].index,  
2 y = df['cuisines'].value\_counts()[0:10])

```
In [53]: 1 iplot([trace3])
```



```
In [54]: 1 df.columns
```

```
Out[54]: Index(['url', 'address', 'name', 'online_order', 'book_table', 'rate', 'votes',
       'phone', 'location', 'rest_type', 'dish_liked', 'cuisines',
       'approx_cost(for two people)', 'reviews_list', 'menu_item',
       'listed_in(type)', 'listed_in(city)'],
      dtype='object')
```

```
In [55]: 1 df['approx_cost(for two people)'].isna().sum()
```

```
Out[55]: 252
```

```
In [56]: 1 df.dropna(axis='index', subset = ['approx_cost(for two people)'], inplace = True)
```

```
In [57]: 1 df['approx_cost(for two people)'].isna().sum()
```

```
Out[57]: 0
```

```
In [58]: 1 sns.distplot(df['approx_cost(for two people)'])

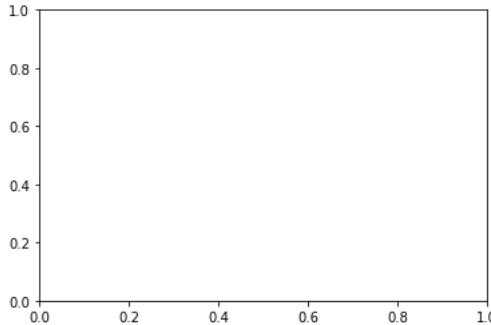
C:\Users\Amir\AppData\Roaming\Python\Python39\site-packages\seaborn\distributions.py:2551: FutureWarning:
`distplot` is a deprecated function and will be removed in a future version. Please adapt your code to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).

-----
ValueError                                Traceback (most recent call last)
Input In [58], in <cell line: 1>()
----> 1 sns.distplot(df['approx_cost(for two people)'])

File ~\AppData\Roaming\Python\Python39\site-packages\seaborn\distributions.py:2568, in distplot(a, bins, hist, kde, rug, fit, h
ist_kws, kde_kws, rug_kws, fit_kws, color, vertical, norm_hist, xlabel, label, ax, x)
 2565     a = x
 2566     # Make a a 1-d float array
-> 2568     a = np.asarray(a, float)
 2569     if a.ndim > 1:
 2570         a = a.squeeze()

File ~\anaconda3\lib\site-packages\pandas\core\series.py:872, in Series.__array__(self, dtype)
 825 def __array__(self, dtype: npt.DTypeLike | None = None) -> np.ndarray:
 826     """
 827     Return the values as a NumPy array.
 828
 829     (...)
 830     dtype='datetime64[ns]')
 831     """
--> 872     return np.asarray(self._values, dtype)

ValueError: could not convert string to float: '1,200'
```



```
In [59]: 1 df['approx_cost(for two people)'].dtype
```

```
Out[59]: dtype('O')
```

```
In [60]: 1 df['approx_cost(for two people)'].unique()
```

```
Out[60]: array(['800', '300', '600', '700', '550', '500', '450', '650', '400',
   '900', '200', '750', '150', '850', '100', '1,200', '350', '250',
   '950', '1,000', '1,500', '1,300', '199', '80', '1,100', '160',
   '1,600', '230', '130', '1,700', '1,400', '1,350', '2,200', '2,000',
   '1,800', '1,900', '180', '330', '2,500', '2,100', '3,000', '2,800',
   '3,400', '50', '40', '1,250', '3,500', '4,000', '2,400', '2,600',
   '1,450', '70', '3,200', '560', '240', '360', '6,000', '1,050',
   '2,300', '4,100', '120', '5,000', '3,700', '1,650', '2,700',
   '4,500'], dtype=object)
```

```
In [61]: 1 df['approx_cost(for two people)'] = df['approx_cost(for two people)'].apply(lambda x : x.replace(',', ''))
```

```
In [62]: 1 df['approx_cost(for two people)'].unique()
```

```
Out[62]: array(['800', '300', '600', '700', '550', '500', '450', '650', '400',
   '900', '200', '750', '150', '850', '100', '1200', '350', '250',
   '950', '1000', '1500', '1300', '199', '80', '1100', '160', '1600',
   '230', '130', '1700', '1400', '1350', '2200', '2000', '1800',
   '1900', '180', '330', '2500', '2100', '3000', '2800', '3400', '50',
   '40', '1250', '3500', '4000', '2400', '2600', '1450', '70', '3200',
   '560', '240', '360', '6000', '1050', '2300', '4100', '120', '5000',
   '3700', '1650', '2700', '4500'], dtype=object)
```

```
In [63]: 1 df['approx_cost(for two people)'] = df['approx_cost(for two people)'].astype(int)
```

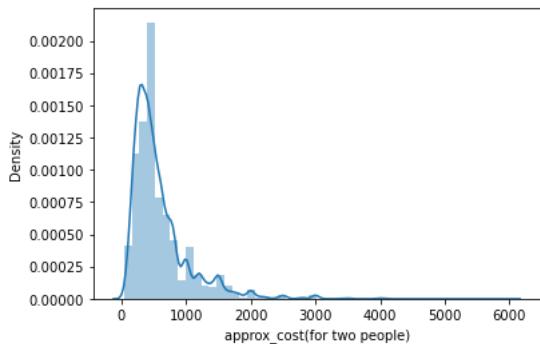
```
In [64]: 1 df['approx_cost(for two people)'].dtype
```

```
Out[64]: dtype('int32')
```

## Analyse Approx cost of 2 people feature

```
In [65]: 1 sns.distplot(df['approx_cost(for two people)'])
```

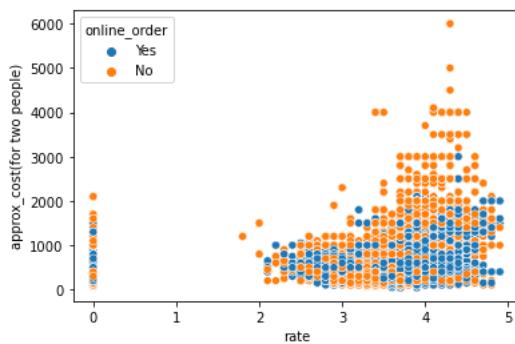
```
Out[65]: <AxesSubplot:xlabel='approx_cost(for two people)', ylabel='Density'>
```



## Analyse 'Approx cost of 2 People' Vs 'Rating' and understand the relationship among them

```
In [66]: 1 sns.scatterplot(x ='rate', y='approx_cost(for two people)', hue='online_order', data = df)
```

```
Out[66]: <AxesSubplot:xlabel='rate', ylabel='approx_cost(for two people)'>
```



**Is there any difference between votes of restaurants accepting and not accepting the online orders ?**

```
In [67]: 1 px.box(df, x='online_order', y = 'votes')
```

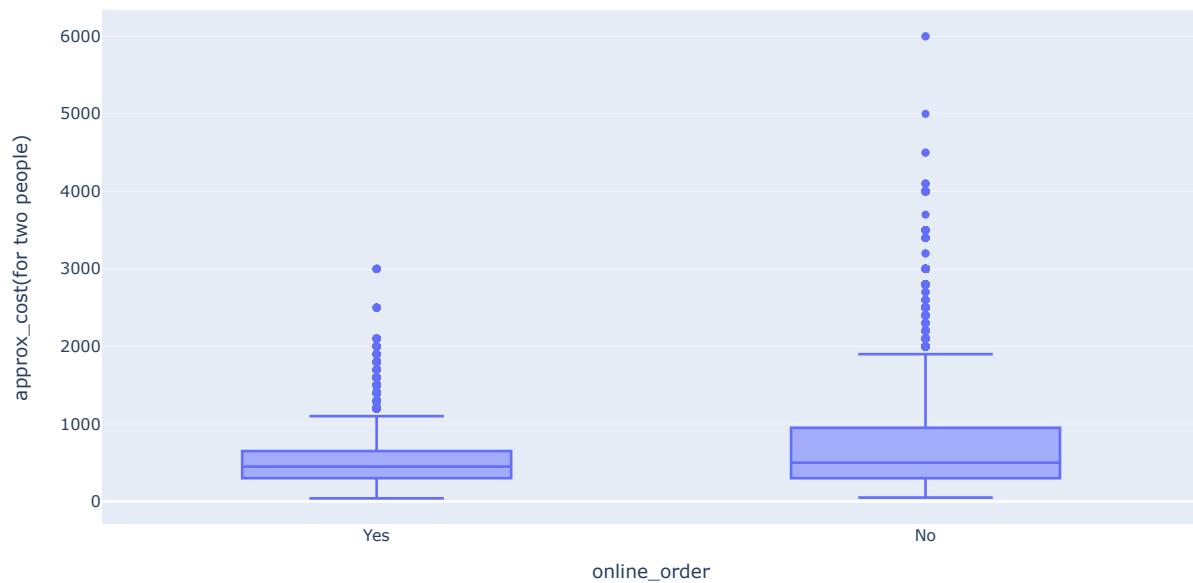


## Is there any difference in price among restaurant accepting and not accepting online order

```
In [68]: 1 df.columns
```

```
Out[68]: Index(['url', 'address', 'name', 'online_order', 'book_table', 'rate', 'votes',
       'phone', 'location', 'rest_type', 'dish_liked', 'cuisines',
       'approx_cost(for two people)', 'reviews_list', 'menu_item',
       'listed_in(type)', 'listed_in(city)'],
      dtype='object')
```

```
In [69]: 1 px.box(df, x='online_order', y ='approx_cost(for two people)')
```



## Find out the most Expensive restaurant

```
In [70]: 1 df[ 'approx_cost(for two people)'].min()
```

```
Out[70]: 40
```

```
In [71]: 1 df[ 'approx_cost(for two people)'].max()
```

```
Out[71]: 6000
```

```
In [72]: 1 df[df[ 'approx_cost(for two people)']==6000][ 'name']
```

```
Out[72]: 19139    Le Cirque Signature - The Leela Palace
45618     Le Cirque Signature - The Leela Palace
Name: name, dtype: object
```

## Top 10 Most Expensive restaurant with approx cost for 2 people

```
In [73]: 1 df[ 'approx_cost(for two people)'].sort_values(ascending= False)
```

```
Out[73]: 45618    6000
19139    6000
40266    5000
41591    4500
42141    4500
...
18891     40
5270      40
27091     40
12232     40
14819     40
Name: approx_cost(for two people), Length: 43690, dtype: int32
```

```
In [74]: 1 data= df.copy()
```

```
In [75]: 1 data.set_index('name', inplace = True)
```

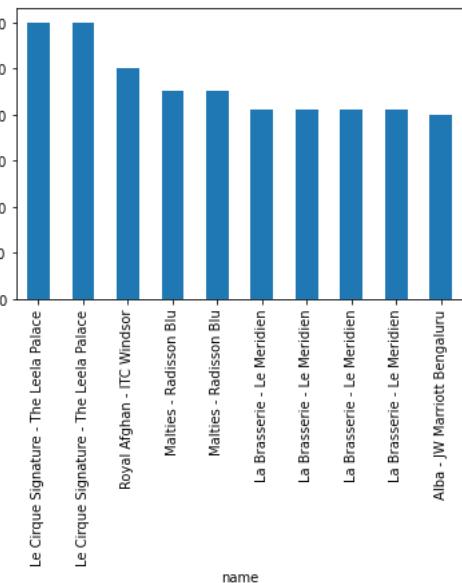
```
In [76]: 1 data.head()
```

```
Out[76]:
```

		url	address	online_order	book_table	rate	votes	phone	location	rest_type	dish_liked
name											
Jalsa	https://www.zomato.com/bangalore/jalsabana... ...	942, 21st Main Road, 2nd Stage, Banashankari, ...	Yes	Yes	4.1	775	080 42297555\r\n+91 9743772233	Banashankari	Casual Dining	Pasta, Lunch Buffet, Masala Papad, Paneer Laja...	
Spice Elephant	https://www.zomato.com/bangalore/spice-elephant... Bazaar, 6th ...	2nd Floor, 80 Feet Road, Near Big Bazaar, 6th ...	Yes	No	4.1	787	080 41714161	Banashankari	Casual Dining	Momos, Lunch Buffet, Chocolate Nirvana, Thai G...	
San Churro Cafe	https://www.zomato.com/SanchurroBangalore?cont... ...	1112, Next to KIMS Medical College, 17th Cross...	Yes	No	3.8	918	+91 9663487993	Banashankari	Cafe, Casual Dining	Churros, Cannelloni, Minestrone Soup, Hot Choc...	
Addhuri Udupi Bhojana	https://www.zomato.com/bangalore/addhuri-udipi... Banashankar...	1st Floor, Annakuteera, 3rd Stage, Banashankar...	No	No	3.7	88	+91 9620009302	Banashankari	Quick Bites	Masala Dosa	
Grand Village	https://www.zomato.com/bangalore/grand-village... Gandhi Baza...	10, 3rd Floor, Lakshmi Associates, Gandhi Baza...	No	No	3.8	166	+91 8026612447\r\n+91 9901210005	Basavanagudi	Casual Dining	Panipuri, Gol Gappe	

```
In [77]: 1 data['approx_cost(for two people)'].nlargest(10).plot.bar()
```

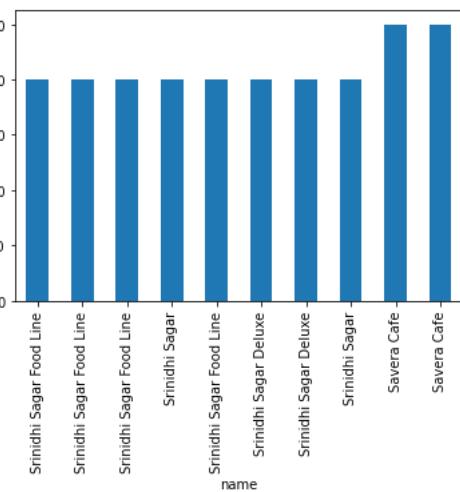
```
Out[77]: <AxesSubplot:xlabel='name'>
```



## Top 10 Cheapest restaurant of Bengalore

```
In [78]: 1 data['approx_cost(for two people)'].nsmallest(10).plot.bar()
```

```
Out[78]: <AxesSubplot:xlabel='name'>
```



**Find all restaurant that are below than 500(budget total) as well as Affordable**

In [79]: 1 data[data['approx\_cost(for two people)']<=500]

Out[79]:

			url	address	online_order	book_table	rate	votes
	name							
0	Addhuri Udupi Bhojana	https://www.zomato.com/bangalore/addhuri-udipi...	1st Floor, Annakuteera, 3rd Stage, Banashankar...	No	No	3.7	88	+91 962
1	CafÃ Ä Ã Ä Ã Ä Ã Ä Ã Ä Ã Ä Ã Ä Ã Ä Ã Ä Ã Ä Down The Alley	https://www.zomato.com/bangalore/caf%C3%A9-dow...	12,29 Near PES University Back Gate, D'Souza N...	Yes	No	4.1	402	267244€ 740
2	The Coffee Shack	https://www.zomato.com/bangalore/the-coffee-sh...	6th Block, 3rd Stage, Banashankari, Bangalore	Yes	Yes	4.2	164	+91 973
3	Caf-Eleven	https://www.zomato.com/bangalore/caf-eleven-ba...	111, Sapphire Toys Building, 100 Feet Ring Roa...	No	No	4.0	424	080 4
4	T3H Cafe	https://www.zomato.com/bangalore/t3h-cafe-bana...	504, CJ Venkata Das Road, Padmanabhangar, 2nd ...	No	No	3.9	93	+91 888
...	...	...	...	...	...	...	...	...
5	Venice Kerala Restaurant	https://www.zomato.com/bangalore/venice-kerala...	Opposite Brookfield Mall, Shiva Temple Road, B...	No	No	3.7	34	+91 888
6	Thai & Chinese Fast Food	https://www.zomato.com/bangalore/thai-chinese-...	7, Anurhh Arcade, Hoodi Road, Kodigehalli Road...	No	No	3.4	7	9206260( 761
7	The Crunch	https://www.zomato.com/bangalore/the-crunch-br...	33/1, Kundalahalli Road, Brookefield, Bangalore	No	No	3.3	6	+91 901
8	Punjabi Charchay	https://www.zomato.com/bangalore/punjabi-charc...	7, Anurhh Arcade, Hoodi Road, Kodigehalli Road...	No	No	3.6	31	9341341€ 761
9	Raapchick	https://www.zomato.com/bangalore/raapchick-bro...	Opposite Brookefield Mall, Kundanhalli, Brooke...	Yes	No	3.6	49	41717€ 957

26330 rows × 16 columns

In [80]: 1 df\_budget = data[data['approx\_cost(for two people)']<=500].loc[:,('approx\_cost(for two people)')]  
2 df\_budget.head()

Out[80]: name

Addhuri Udupi Bhojana	300
CafÃ Ä Ã Ä Ã Ä Ã Ä Ã Ä Ã Ä Ã Ä Ã Ä Ã Ä Down The Alley	500
The Coffee Shack	500
Caf-Eleven	450
T3H Cafe	300

Name: approx\_cost(for two people), dtype: int32

In [81]: 1 df\_budget = df\_budget.reset\_index()  
2 df\_budget.head()

Out[81]:

	name	approx_cost(for two people)
0	Addhuri Udupi Bhojana	300
1	CafÃ Ä Ã Ä Ã Ä Ã Ä Ã Ä Ã Ä Ã Ä Ã Ä Ã Ä Down The A...	500
2	The Coffee Shack	500
3	Caf-Eleven	450
4	T3H Cafe	300

## Total Restaurant that have good rating(>4) and budget(<500)

```
In [82]: 1 df[(df['rate']>4) & (df['approx_cost(for two people)']<=500)].shape
```

Out[82]: (2436, 17)

```
In [83]: 1 len(df[(df['rate']>4) & (df['approx_cost(for two people)']<=500)]['name'].unique())
```

Out[83]: 372

## Total various affordable restaurants at all locations of Bengalore

```
In [84]: 1 df_new = df[(df['rate']>4) & (df['approx_cost(for two people)']<500)]
2 df_new.head()
```

Out[84]:

		url	address	name	online_order	book_table	rate	votes	phone	location	rest_type	dish
51		https://www.zomato.com/bangalore/shree-cool-po...	1514, 4th Cross, 7th Main, RPC layout, 2nd Sta...	Shree Cool Point	Yes	No	4.1	28	+91 9900617124	Vijay Nagar	Quick Bites	
52		https://www.zomato.com/bangalore/corner-house-...	808/6-1, 24th A Cross, K.R Road, 2nd Stage, Ba...	Corner House Ice Cream	No	No	4.3	345	080 26713965\n+91 9845444155	Banashankari	Dessert Parlor	Chocof Van C Fu
65		https://www.zomato.com/bangalore/kabab-magic-b...	31, Opposite Vijaya College, RV Road, Basavanna...	Kabab Magic	Yes	No	4.1	1720	080 26570381	Basavanagudi	Quick Bites	Cl Cl
71		https://www.zomato.com/bangalore/the-biryani-c...	476, 2nd Floor, 9th Cross, Avalahalli BDA Park...	The Biryani Cafe	No	No	4.1	520	+91 7010846043	Banashankari	Quick Bites	Veg B ( Ch Cl
76		https://www.zomato.com/bangalore/sri-guru-kott...	49/1, Subbaraoeshetty Road, Netkallappa Circle,...	Sri Guru Kottureshwara Davangere Benne Dosa	Yes	No	4.1	558	+91 9448305384\n+91 9060079480	Basavanagudi	Quick Bites	Bajji CP

```
In [85]: 1 location = []
2 total = []
3
4 for loc, location_df in df_new.groupby('location'):
5     location.append(loc)
6     total.append(len(location_df['name'].unique()))
```

```
In [86]: 1 location_df = pd.DataFrame(zip(location, total))
2 location_df.head()
```

Out[86]:

	0	1
0	BTM	24
1	Banashankari	11
2	Banaswadi	1
3	Bannerghatta Road	5
4	Basavanagudi	21

```
In [87]: 1 location_df.columns = ['location', 'restaurant']
```

```
In [88]: 1 location_df = location_df.sort_values(by = 'restaurant', ascending = False).head()
```

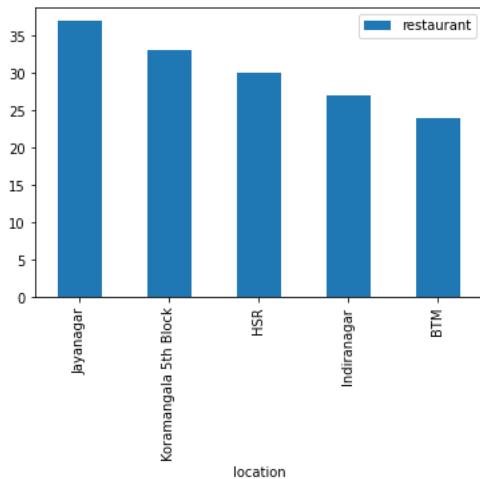
In [89]: 1 location\_df # Jayanagar has 37 restaurant which cost below 500 for two people and has rated more than 4

Out[89]:

	location	restaurant
22	Jayanagar	37
30	Koramangala 5th Block	33
16	HSR	30
20	Indiranagar	27
0	BTM	24

In [90]: 1 location\_df.head().plot.bar(x='location', y ='restaurant')

Out[90]: <AxesSubplot:xlabel='location'>



## Find the best budget restaurant in any location

In [91]: 1 def return\_budget(location,restaurant):  
2 budget = df[(df['approx\_cost(for two people)']<=400) & (df['location']==location) & (df['rate']>4) & (df['rest\_type']==  
3 return (budget['name'].unique())

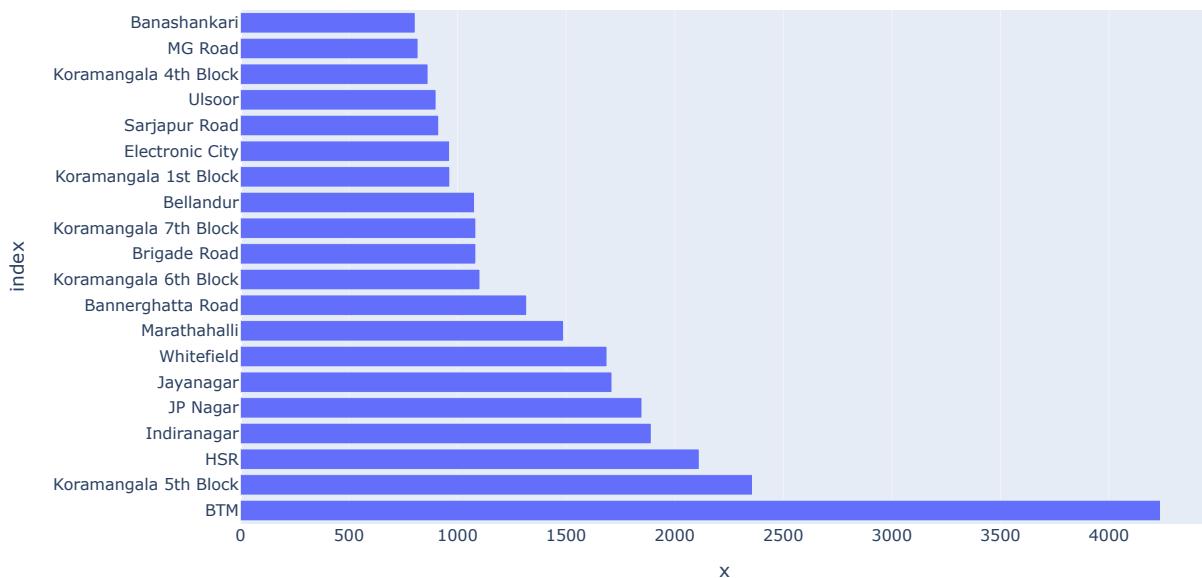
In [92]: 1 return\_budget('BTM','Quick Bites')

Out[92]: array(['Swadista Aahan', 'Litti Twist', 'The Shawarma Shop', 'Gorbandh',  
 'Yum In My Tum', 'Chaatimes', "Muthashy's", 'Swad Punjab Da',  
 "Domino's Pizza", 'Roti Wala', 'Andhra Kitchen'], dtype=object)

## Which are the foodie areas in the Bengaluru ?

In [93]: 1 restaurant\_location = df['location'].value\_counts()[0:20]

```
In [94]: 1 fig = px.bar(restaurant_location, x=restaurant_location, y=restaurant_location.index)
2 fig.show()
```



```
In [95]: 1 df.head()
```

Out[95]:

		url	address	name	online_order	book_table	rate	votes	phone	location	rest_type	dish_lik
0		https://www.zomato.com/bangalore/jalsa-banashankari	942, 21st Main Road, 2nd Stage, Banashankari, ...	Jalsa	Yes	Yes	4.1	775	080 42297555\n+91 9743772233	Banashankari	Casual Dining	Pas Lur Buf Mas Pap Pan Laj
1		https://www.zomato.com/bangalore/spice-elephant	2nd Floor, 80 Feet Road, Near Big Bazaar, 6th ...	Spice Elephant	Yes	No	4.1	787	080 41714161	Banashankari	Casual Dining	Mom Lur Buf Chocol Nirv Thai C
2		https://www.zomato.com/SanchurroBangalore?cont...	1112, Next to KIMS Medical College, 17th Cross...	San Churro Cafe	Yes	No	3.8	918	+91 9663487993	Banashankari	Cafe, Casual Dining	Churr Cannell Minestr Soup, I Cho
3		https://www.zomato.com/bangalore/addhuri-udupi...	1st Floor, Annakuteera, 3rd Stage, Banashankari...	Addhuri Udupi Bhojana	No	No	3.7	88	+91 9620009302	Banashankari	Quick Bites	Mas Dc
4		https://www.zomato.com/bangalore/grand-village...	10, 3rd Floor, Lakshmi Associates, Gandhi Baza...	Grand Village	No	No	3.8	166	+91 8026612447\n+91 9901210005	Basavanagudi	Casual Dining	Panip Gol Gap

## Find Out Latitudes and Longitudes for each of the location of Bengaluru

```
In [96]: 1 locations = pd.DataFrame({'Name':df['location'].unique()})
```

In [97]: 1 locations

Out[97]:

	Name
0	Banashankari
1	Basavanagudi
2	Mysore Road
3	Jayanagar
4	Kumaraswamy Layout
...	...
87	Sahakara Nagar
88	Jalahalli
89	Nagarbhavi
90	Peenya
91	KR Puram

92 rows × 1 columns

In [98]: 1 !pip install geopy

```
Requirement already satisfied: geopy in c:\users\amir\anaconda3\lib\site-packages (2.3.0)
Requirement already satisfied: geographiclib<3,>=1.52 in c:\users\amir\anaconda3\lib\site-packages (from geopy) (2.0)
```

In [99]: 1 from geopy.geocoders import Nominatim

In [100]: 1 geolocator = Nominatim(user\_agent='app')

In [101]: 1 lat\_lon = []
2 for location in locations['Name']:
3 location = geolocator.geocode(location)
4 if location is None:
5 lat\_lon.append(np.nan)
6 else:
7 geo = (location.latitude, location.longitude)
8 lat\_lon.append(geo)

In [102]: 1 locations['geo\_loc'] = lat\_lon

In [103]: 1 locations.head()

Out[103]:

	Name	geo_loc
0	Banashankari	(15.8876779, 75.7046777)
1	Basavanagudi	(12.9417261, 77.5755021)
2	Mysore Road	(12.9466619, 77.5300896)
3	Jayanagar	(27.64392675, 83.05280519687284)
4	Kumaraswamy Layout	(12.9081487, 77.5553179)

In [104]: 1 locations.shape

Out[104]: (92, 2)

```
In [105]: 1 rest_locations = pd.DataFrame(df['location'].value_counts().reset_index())
2 rest_locations
```

Out[105]:

	index	location
0	BTM	4237
1	Koramangala 5th Block	2358
2	HSR	2113
3	Indiranagar	1892
4	JP Nagar	1849
...	...	...
87	West Bangalore	5
88	Yelahanka	4
89	Rajarajeshwari Nagar	2
90	Nagarbhavi	1
91	Peenya	1

92 rows × 2 columns

```
In [106]: 1 rest_locations.columns=['Name', 'count']
2 rest_locations.head()
```

Out[106]:

	Name	count
0	BTM	4237
1	Koramangala 5th Block	2358
2	HSR	2113
3	Indiranagar	1892
4	JP Nagar	1849

```
In [107]: 1 restaurant_locations = rest_locations.merge(locations, on = 'Name', how='left').dropna()
2 restaurant_locations.head()
```

Out[107]:

	Name	count	geo_loc
0	BTM	4237	(45.95485055, -112.49659530324134)
1	Koramangala 5th Block	2358	(12.9348429, 77.6189768)
2	HSR	2113	(18.1475, 41.538889)
3	Indiranagar	1892	(12.9732913, 77.6404672)
4	JP Nagar	1849	(12.2655944, 76.6465404)

```
In [108]: 1 np.array(restaurant_locations['geo_loc'])
```

```
Out[108]: array([(45.95485055, -112.49659530324134), (12.9348429, 77.6189768),
 (18.1475, 41.538889), (12.9732913, 77.6404672),
 (12.2655944, 76.6465404), (27.64392675, 83.05280519687284),
 (53.5533682, -2.2969019), (12.9552572, 77.6984163),
 (12.8754958, 77.5949946), (12.9390255, 77.6238477),
 (-34.0881284, 18.4585622), (12.9364846, 77.6134783),
 (12.93103185, 77.6782471), (12.9277245, 77.6327822),
 (12.848759900000001, 77.64825295827616), (12.920441, 77.6653284),
 (12.9778793, 77.6246697), (12.9327778, 77.6294052),
 (12.9755264, 77.6067902), (15.8876779, 75.7046777),
 (13.0221416, 77.6493368), (13.0027353, 77.5703253),
 (12.9417261, 77.5755021), (40.575894, -74.1251342),
 (34.0732341, 74.8305765), (33.5935063, -79.0345627),
 (12.996845, 77.6138165), (40.7171341, -74.0062127),
 (13.0093455, 77.6377094), (13.0343869, 77.5683929),
 (40.7652844, -76.373824), (13.0141618, 77.6518539),
 (12.9882338, 77.554883), (36.5348643, -79.0905056),
 (12.9624669, 77.6381958), (22.478459, 88.3541291),
 (1.3066135, 103.883234), (33.117255, -84.837311),
 (51.519619, -0.0744749), (12.945245, 77.6269144),
 (18.5322493, 73.8499601124847), (12.9678074, 77.6568367),
 (12.988721250000001, 77.58516877601824), (12.9408685, 77.617338),
 (12.9489339, 77.5968273), (12.9282918, 77.6254034),
 (12.9081487, 77.5553179), (13.2227, 78.5541977),
 (12.9089453, 77.6239038), (12.973936, 77.6509982),
 (12.9931876, 77.5753419), (12.2950874, 76.6149845),
 (30.3141045, -89.8035331), (1.3147772, 103.8567062),
 (13.0258087, 77.6305067), (1.2847055, 103.84320655721689),
 (13.02383, 77.5529215), (13.0358698, 77.6323597),
 (12.9243692, 77.6242433), (12.9846713, 77.6790908),
 (12.8845448, 77.5701276), (12.9413238, 77.7471103),
 (12.9804194, 77.7275164), (39.76880625, -86.15345077251979),
 (12.7687114, 77.78836), (18.5384853, 75.5569267),
 (17.2518682, 80.1651978), (13.0227204, 77.595715),
 (15.8782951, 74.5084834), (23.1485712, 81.6048241),
 (18.4900796, 73.8475301), (12.9340114, 77.6222304),
 (12.9845687, 77.73766474151809), (12.957998, 77.6037312),
 (38.7801076, -121.5056438), (12.9756527, 77.5553548),
 (13.0464531, 77.5483803), (13.0180639, 77.7061087),
 (12.5560169, 77.4249605), (12.9466619, 77.5300896),
 (12.9767936, 77.590082), (13.007516, 77.695935),
 (13.0382184, 77.5919), (12.9176571, 77.4837568),
 (12.9055682, 77.5455438), (13.0101286, 77.5548006),
 (12.9848519, 77.5400626), (13.1006982, 77.5963454),
 (12.9274413, 77.5155224), (12.95961755, 77.51126721318181),
 (13.0329419, 77.5273253)], dtype=object)
```

```
In [109]: 1 lat, lon = zip(*np.array(restaurant_locations['geo_loc']))
```

```
In [110]: 1 type(lat)
```

```
Out[110]: tuple
```

```
In [111]: 1 restaurant_locations['lat'] = lat
 2 restaurant_locations['lon'] = lon
```

```
In [112]: 1 restaurant_locations.head()
```

```
Out[112]:
```

	Name	count	geo_loc	lat	lon
0	BTM	4237	(45.95485055, -112.49659530324134)	45.954851	-112.496595
1	Koramangala 5th Block	2358	(12.9348429, 77.6189768)	12.934843	77.618977
2	HSR	2113	(18.1475, 41.538889)	18.147500	41.538889
3	Indiranagar	1892	(12.9732913, 77.6404672)	12.973291	77.640467
4	JP Nagar	1849	(12.2655944, 76.6465404)	12.265594	76.646540

```
In [113]: 1 restaurant_locations.drop('geo_loc', axis=1, inplace=True)
```

## Generate Basemap of Bengaluru

```
In [114]: 1 !pip install folium
```

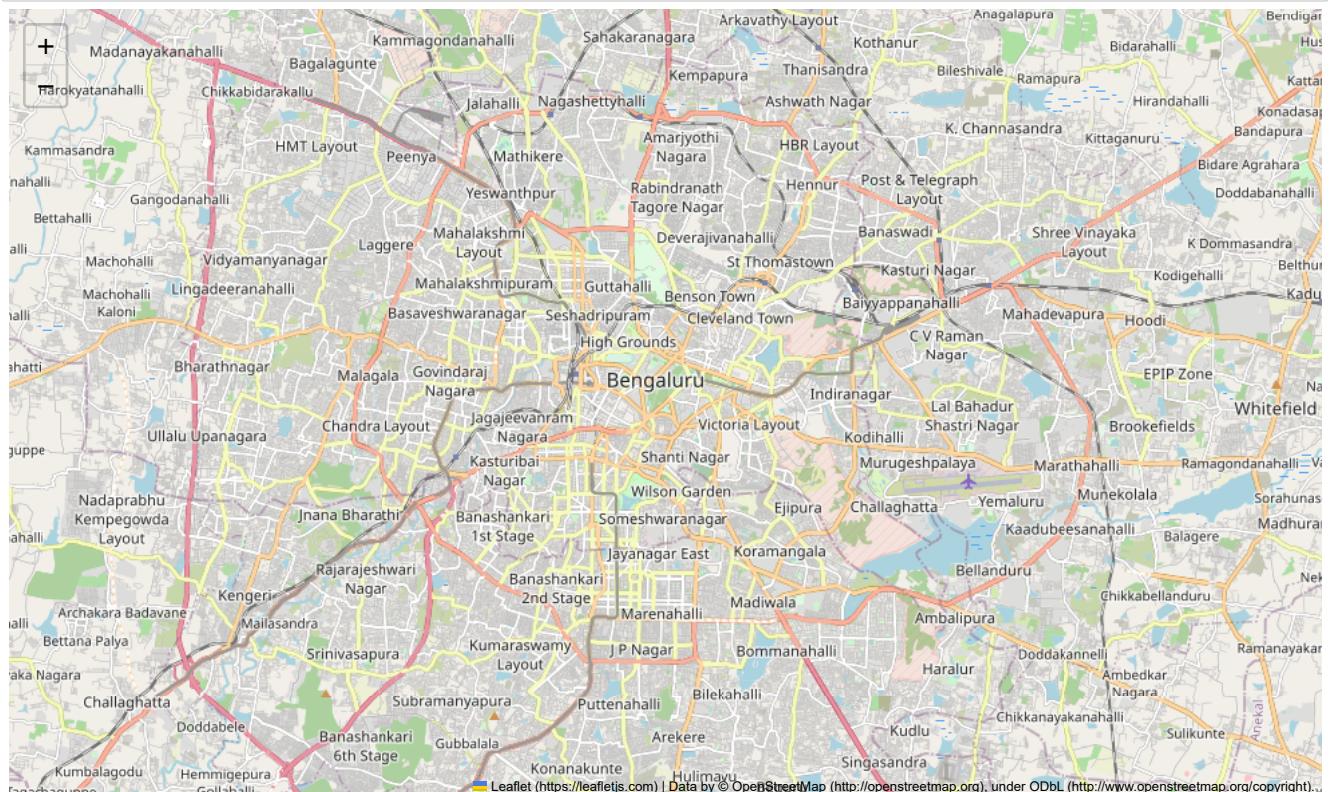
```
Requirement already satisfied: folium in c:\users\amir\anaconda3\lib\site-packages (0.14.0)
Requirement already satisfied: jinja2>=2.9 in c:\users\amir\anaconda3\lib\site-packages (from folium) (2.11.3)
Requirement already satisfied: branca>=0.6.0 in c:\users\amir\anaconda3\lib\site-packages (from folium) (0.6.0)
Requirement already satisfied: numpy in c:\users\amir\anaconda3\lib\site-packages (from folium) (1.21.5)
Requirement already satisfied: requests in c:\users\amir\anaconda3\lib\site-packages (from folium) (2.27.1)
Requirement already satisfied: MarkupSafe>=0.23 in c:\users\amir\anaconda3\lib\site-packages (from jinja2>=2.9->folium) (2.0.1)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\amir\anaconda3\lib\site-packages (from requests->folium) (2021.1.0.8)
Requirement already satisfied: charset-normalizer~=2.0.0 in c:\users\amir\anaconda3\lib\site-packages (from requests->folium) (2.0.4)
Requirement already satisfied: idna<4,>=2.5 in c:\users\amir\anaconda3\lib\site-packages (from requests->folium) (3.3)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in c:\users\amir\anaconda3\lib\site-packages (from requests->folium) (1.26.9)
```

```
In [115]: 1 import folium  
          2 from folium.plugins import HeatMap
```

```
In [116]: 1 def generatebasemap(default_location = [12.97,77.59], default_zoom_start = 12):  
2     basemap = folium.Map(location = default_location, zoom_start = default_zoom_start)  
3     return basemap
```

```
In [117]: 1 basemap = generatebasemap()
```

In [118]: 1 basemap



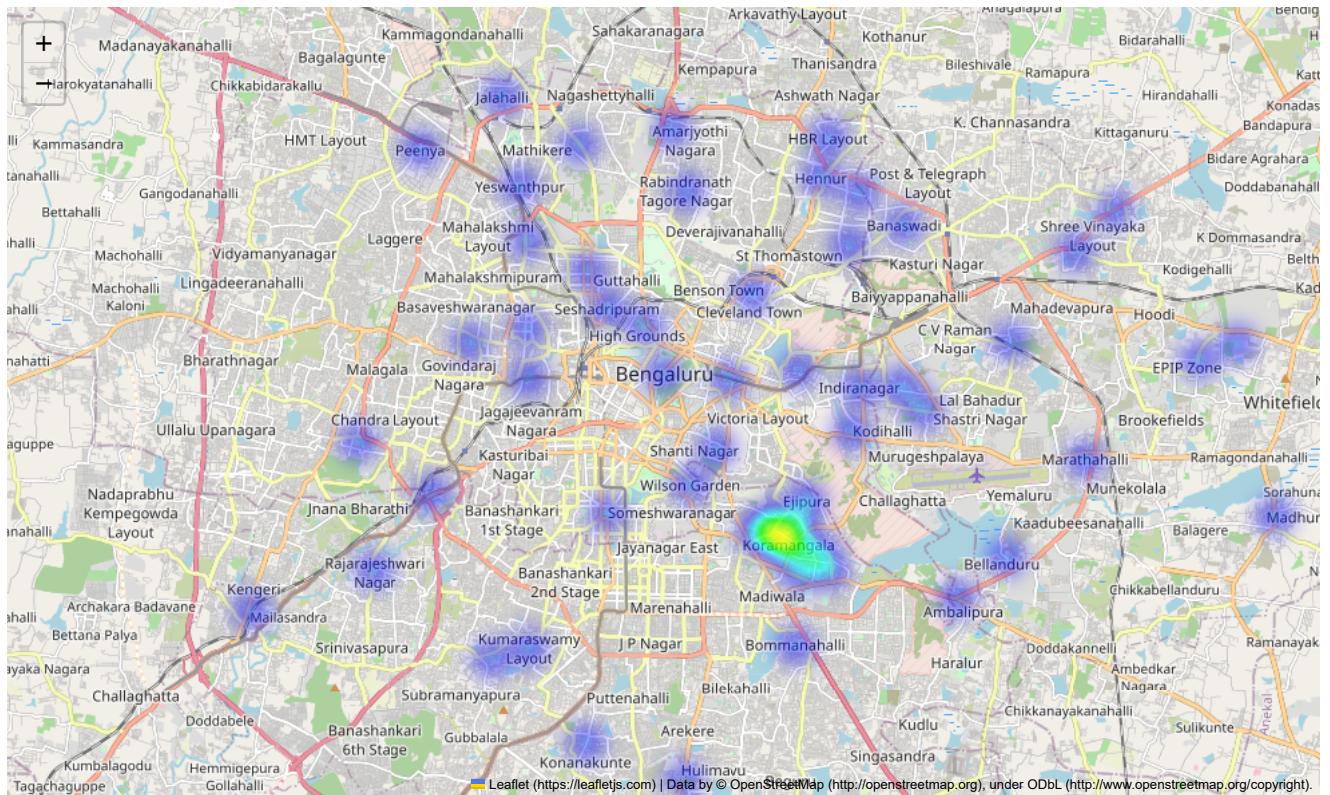
## HeatMap for Restaurants

```
In [119]: 1 HeatMap(restaurant_locations[['lat','lon','count']].values.tolist(),zoom = 20, radius = 15).add_to(basemap)
```

Out[119]: <folium.plugins.heat\_map.HeatMap at 0x259cd93eeb0>

In [120]: 1 basemap

Out[120]:



## HeatMap for North Indian Restaurant

In [121]: 1 df.head()

Out[121]:

	url	address	name	online_order	book_table	rate	votes	phone	location	rest_type	dish_lik
0	https://www.zomato.com/bangalore/jalsa-banash... ...	942, 21st Main Road, 2nd Stage, Banashankari, ...	Jalsa	Yes	Yes	4.1	775	080 42297555\n+91 9743772233	Banashankari	Casual Dining	Pas Lur Buf Mas Pap Pan Laj
1	https://www.zomato.com/bangalore/spice-elephant... ...	2nd Floor, 80 Feet Road, Near Big Bazaar, 6th ...	Spice Elephant	Yes	No	4.1	787	080 41714161	Banashankari	Casual Dining	Mom Lur Buf Chocol Nirv Thai C
2	https://www.zomato.com/SanchurroBangalore?cont... ...	1112, Next to KIMS Medical College, 17th Cross...	San Churro Cafe	Yes	No	3.8	918	+91 9663487993	Banashankari	Cafe, Casual Dining	Churr Cannell Minestr Soup, f Cho
3	https://www.zomato.com/bangalore/addhuri-udupi... ...	1st Floor, Annakuteera, 3rd Stage, Banashankar...	Addhuri Udupi Bhojana	No	No	3.7	88	+91 9620009302	Banashankari	Quick Bites	Mas Dc
4	https://www.zomato.com/bangalore/grand-village... ...	10, 3rd Floor, Lakshmi Associates, Gandhi Baza...	Grand Village	No	No	3.8	166	+91 8026612447\n+91 9901210005	Basavanagudi	Casual Dining	Panip Gol Gap

In [122]: 1 df2 = df[df['cuisines']=='North Indian']

In [123]: 1 north\_india = df2.groupby(['location'], as\_index = False)[['url']].count()

In [124]: 1 north\_india.head()

Out[124]:

	location	url
0	BTM	262
1	Banashankari	35
2	Banaswadi	9
3	Bannerghatta Road	60
4	Basavanagudi	17

In [125]: 1 north\_india.columns = ['Name', 'count']

In [126]: 1 north\_india.head()

Out[126]:

	Name	count
0	BTM	262
1	Banashankari	35
2	Banaswadi	9
3	Bannerghatta Road	60
4	Basavanagudi	17

In [127]: 1 north\_india = north\_india.merge(locations, on='Name', how='left').dropna()

In [128]: 1 north\_india.head()

Out[128]:

	Name	count	geo_loc
0	BTM	262	(45.95485055, -112.49659530324134)
1	Banashankari	35	(15.8876779, 75.7046777)
2	Banaswadi	9	(13.0141618, 77.6518539)
3	Bannerghatta Road	60	(12.8754958, 77.5949946)
4	Basavanagudi	17	(12.9417261, 77.5755021)

In [129]: 1 north\_india['lat'], north\_india['lon'] = zip(\*north\_india['geo\_loc'].values)

In [130]: 1 north\_india.head()

Out[130]:

	Name	count	geo_loc	lat	lon
0	BTM	262	(45.95485055, -112.49659530324134)	45.954851	-112.496595
1	Banashankari	35	(15.8876779, 75.7046777)	15.887678	75.704678
2	Banaswadi	9	(13.0141618, 77.6518539)	13.014162	77.651854
3	Bannerghatta Road	60	(12.8754958, 77.5949946)	12.875496	77.594995
4	Basavanagudi	17	(12.9417261, 77.5755021)	12.941726	77.575502

In [131]: 1 north\_india.drop('geo\_loc', axis=1, inplace=True)

In [132]: 1 north\_india.head()

Out[132]:

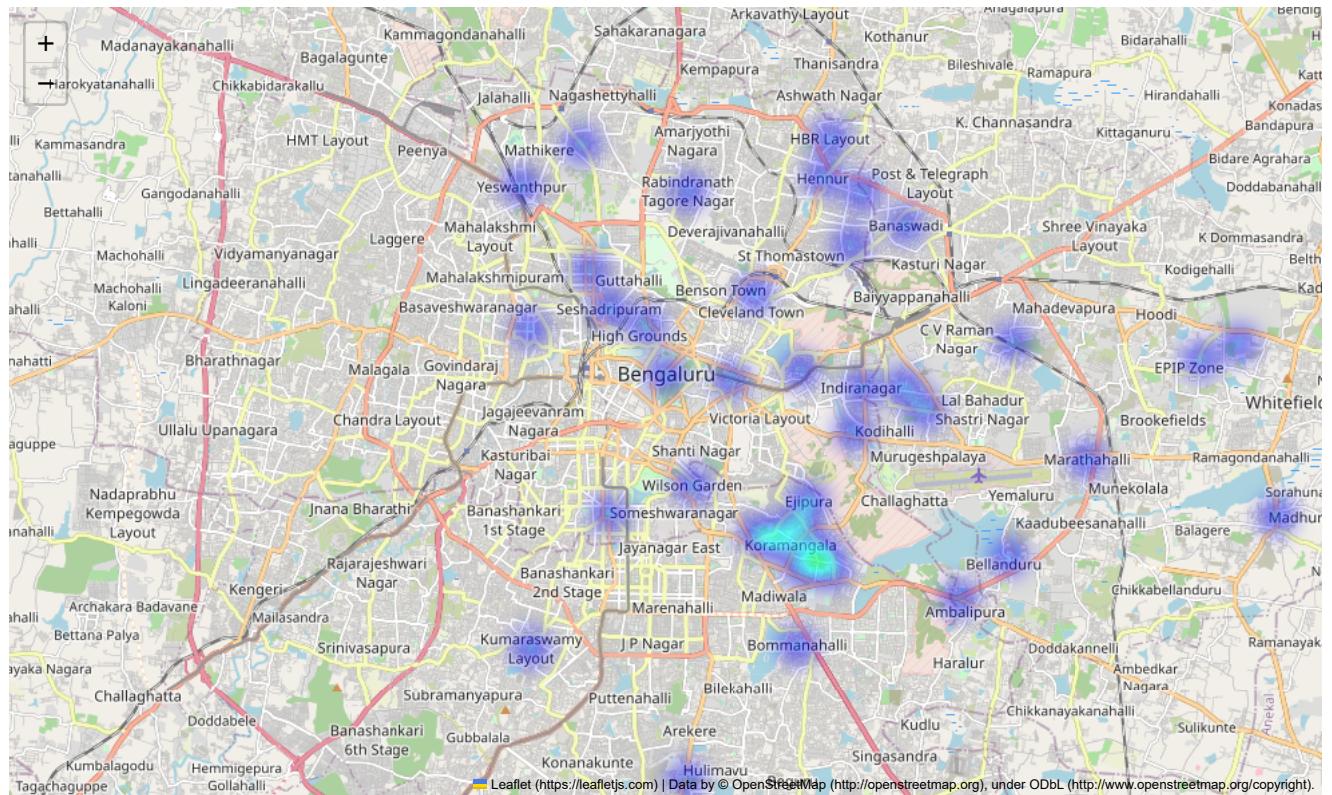
	Name	count	lat	lon
0	BTM	262	45.954851	-112.496595
1	Banashankari	35	15.887678	75.704678
2	Banaswadi	9	13.014162	77.651854
3	Bannerghatta Road	60	12.875496	77.594995
4	Basavanagudi	17	12.941726	77.575502

In [133]: 1 basemap = generatebasemap()  
2 HeatMap(north\_india[['lat', 'lon', 'count']].values.tolist(), zoom=20, radius=15).add\_to(basemap)

Out[133]: <folium.plugins.heat\_map.HeatMap at 0x259cd950b50>

In [134]: 1 basemap

Out[134]:



## Which are the most popular Casual dining Restaurant chains ?

```
In [135]: 1 df_1 = df.groupby(['rest_type', 'name']).count()
```

In [136]: 1 df\_1.head()

Out[136]:

```
In [137]: 1 df_1.sort_values(['url'], ascending = False).groupby(['rest_type'], as_index = False).apply(lambda x: x.sort_values(by ='url',
```

Out[137]:

	level_0	rest_type	name	url
0	0	Bakery	Just Bake	44
1	0	Bakery	Warm Oven	28
2	0	Bakery	INDULGE by InnerChef	28
3	0	Bakery	Karachi Bakery	26
4	0	Bakery	CakeZone	21
...	...	...	...	...
7707	86	Takeaway, Delivery	Agraharam Meal	1
7708	86	Takeaway, Delivery	Adyar Ananda Bhavan Sweets	1
7709	86	Takeaway, Delivery	Faasos	1
7710	86	Takeaway, Delivery	@North Parontha Hut	1
7711	86	Takeaway, Delivery	Lucky Singh & Co	1

7712 rows × 4 columns

```
In [138]: 1 dataset = df_1.sort_values(['url'], ascending = False).groupby(['rest_type'], as_index = False).apply(lambda x: x.sort_values(
```

In [139]: 1 dataset

Out[139]:

	level_0	rest_type	name	count
0	0	Bakery	Just Bake	44
1	0	Bakery	Warm Oven	28
2	0	Bakery	INDULGE by InnerChef	28
3	0	Bakery	Karachi Bakery	26
4	0	Bakery	CakeZone	21
...	...	...	...	...
7707	86	Takeaway, Delivery	Agraharam Meal	1
7708	86	Takeaway, Delivery	Adyar Ananda Bhavan Sweets	1
7709	86	Takeaway, Delivery	Faasos	1
7710	86	Takeaway, Delivery	@North Parontha Hut	1
7711	86	Takeaway, Delivery	Lucky Singh & Co	1

7712 rows × 4 columns

```
In [140]: 1 casual = dataset[dataset['rest_type']=='Casual Dining']
```

```
In [141]: 1 casual.head()
```

Out[141]:

	level_0	rest_type	name	count
1001	24	Casual Dining	Empire Restaurant	58
1002	24	Casual Dining	Beijing Bites	48
1003	24	Casual Dining	Mani's Dum Biryani	47
1004	24	Casual Dining	Chung Wah	46
1005	24	Casual Dining	Oye Amritsar	41

In [ ]:

1