foodhackathon

April 19, 2025

[ ]:

```python
from IPython.display import display, HTML
```

[ ]:

```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from geopy.geocoders import Nominatim
from geopy.exc import GeocoderTimedOut
from IPython.core.display import display, HTML
import time
```

[1]:

```python
!pip install folium plotly geopandas
```

```
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: folium in
c:\users\nagal\appdata\roaming\python\python312\site-packages (0.19.5)
Requirement already satisfied: plotly in c:\programdata\anaconda3\lib\site-
packages (5.22.0)
Requirement already satisfied: geopandas in
c:\users\nagal\appdata\roaming\python\python312\site-packages (1.0.1)
Requirement already satisfied: branca>=0.6.0 in
c:\users\nagal\appdata\roaming\python\python312\site-packages (from folium)
(0.8.1)
Requirement already satisfied: jinja2>=2.9 in c:\programdata\anaconda3\lib\site-
packages (from folium) (3.1.4)
Requirement already satisfied: numpy in c:\programdata\anaconda3\lib\site-
packages (from folium) (1.26.4)
Requirement already satisfied: requests in c:\programdata\anaconda3\lib\site-
packages (from folium) (2.32.2)
Requirement already satisfied: xyzservices in c:\programdata\anaconda3\lib\site-
packages (from folium) (2022.9.0)
Requirement already satisfied: tenacity>=6.2.0 in
c:\programdata\anaconda3\lib\site-packages (from plotly) (8.2.2)
Requirement already satisfied: packaging in c:\programdata\anaconda3\lib\site-
packages (from plotly) (23.2)
Requirement already satisfied: pyogrio>=0.7.2 in
c:\users\nagal\appdata\roaming\python\python312\site-packages (from geopandas)
(0.10.0)
```

```
Requirement already satisfied: pandas>=1.4.0 in
c:\programdata\anaconda3\lib\site-packages (from geopandas) (2.2.2)
Requirement already satisfied: pyproj>=3.3.0 in
c:\users\nagal\appdata\roaming\python\python312\site-packages (from geopandas)
(3.7.1)
Requirement already satisfied: shapely>=2.0.0 in
c:\users\nagal\appdata\roaming\python\python312\site-packages (from geopandas)
(2.1.0)
Requirement already satisfied: MarkupSafe>=2.0 in
c:\programdata\anaconda3\lib\site-packages (from jinja2>=2.9->folium) (2.1.3)
Requirement already satisfied: python-dateutil>=2.8.2 in
c:\programdata\anaconda3\lib\site-packages (from pandas>=1.4.0->geopandas)
(2.9.0.post0)
Requirement already satisfied: pytz>=2020.1 in
c:\programdata\anaconda3\lib\site-packages (from pandas>=1.4.0->geopandas)
(2024.1)
Requirement already satisfied: tzdata>=2022.7 in
c:\programdata\anaconda3\lib\site-packages (from pandas>=1.4.0->geopandas)
(2023.3)
Requirement already satisfied: certifi in c:\programdata\anaconda3\lib\site-
packages (from pyogrio>=0.7.2->geopandas) (2024.8.30)
Requirement already satisfied: charset-normalizer<4,>=2 in
c:\programdata\anaconda3\lib\site-packages (from requests->folium) (2.0.4)
Requirement already satisfied: idna<4,>=2.5 in
c:\programdata\anaconda3\lib\site-packages (from requests->folium) (3.7)
Requirement already satisfied: urllib3<3,>=1.21.1 in
c:\programdata\anaconda3\lib\site-packages (from requests->folium) (2.2.2)
Requirement already satisfied: six>=1.5 in c:\programdata\anaconda3\lib\site-
packages (from python-dateutil>=2.8.2->pandas>=1.4.0->geopandas) (1.16.0)
```

[3]:

```python
import folium
import plotly.express as px
import geopandas as gpd

print("All libraries loaded successfully!  ")
```

```
All libraries loaded successfully!
```

[7]:

```python
data1 = pd.read_csv("C:\\Users\\nagal\\Downloads\\zomato_data.csv")
data2 = pd.read_csv("C:\\Users\\nagal\\Downloads\\Geographical Coordinates.csv")
```

[13]:

```python
data1['rate'] = data1['rate'].replace('-', np.nan)
data1['rate'] = data1['rate'].astype(str).str.replace('/5', '', regex=False).
 ↪str.strip()
data1['rate'] = pd.to_numeric(data1['rate'], errors='coerce')
data1['rate'] = data1['rate'].fillna(data1['rate'].median())
```

[15]:

```
data1['approx_costfor_two_people'] = data1['approx_costfor_two_people'].
 ↪astype(str).str.replace(',', '')
data1['approx_costfor_two_people'] = pd.
 ↪to_numeric(data1['approx_costfor_two_people'], errors='coerce')
data1['approx_costfor_two_people'] = data1['approx_costfor_two_people'].
 ↪fillna(data1['approx_costfor_two_people'].median())
```

[17]:

```
data1['dish_liked'] = data1['dish_liked'].fillna("Not Available")
data1['cuisines'] = data1['cuisines'].fillna("Other")
data1['rest_type'] = data1['rest_type'].fillna("Unknown")
```

[19]:

```
data1['votes'] = pd.to_numeric(data1['votes'], errors='coerce')
data1['votes'] = data1['votes'].fillna(data1['votes'].median())
```

[21]:

```
data1['online_order'] = data1['online_order'].map({'Yes': 1, 'No': 0})
data1['book_table'] = data1['book_table'].map({'Yes': 1, 'No': 0})
```

[23]:

```
data1['rate'] = data1['rate'].astype(float)
data1['votes'] = data1['votes'].astype(int)
data1['approx_costfor_two_people'] = data1['approx_costfor_two_people'].
 ↪astype(int)
```

[25]:

```
print(data1.info())
print(data1.isnull().sum())
print(data1.describe())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 51717 entries, 0 to 51716
Data columns (total 10 columns):
 #   Column                     Non-Null Count  Dtype
---  ------                     --------------  -----
 0   online_order               0 non-null      float64
 1   book_table                 0 non-null      float64
 2   rate                       51717 non-null  float64
 3   votes                      51717 non-null  int32
 4   rest_type                  51717 non-null  object
 5   dish_liked                 51717 non-null  object
 6   cuisines                   51717 non-null  object
 7   approx_costfor_two_people  51717 non-null  int32
 8   listed_intype              51717 non-null  object
 9   listed_incity              51717 non-null  object
dtypes: float64(3), int32(2), object(5)
memory usage: 3.6+ MB
None
online_order                 51717
book_table                   51717
rate                             0
```

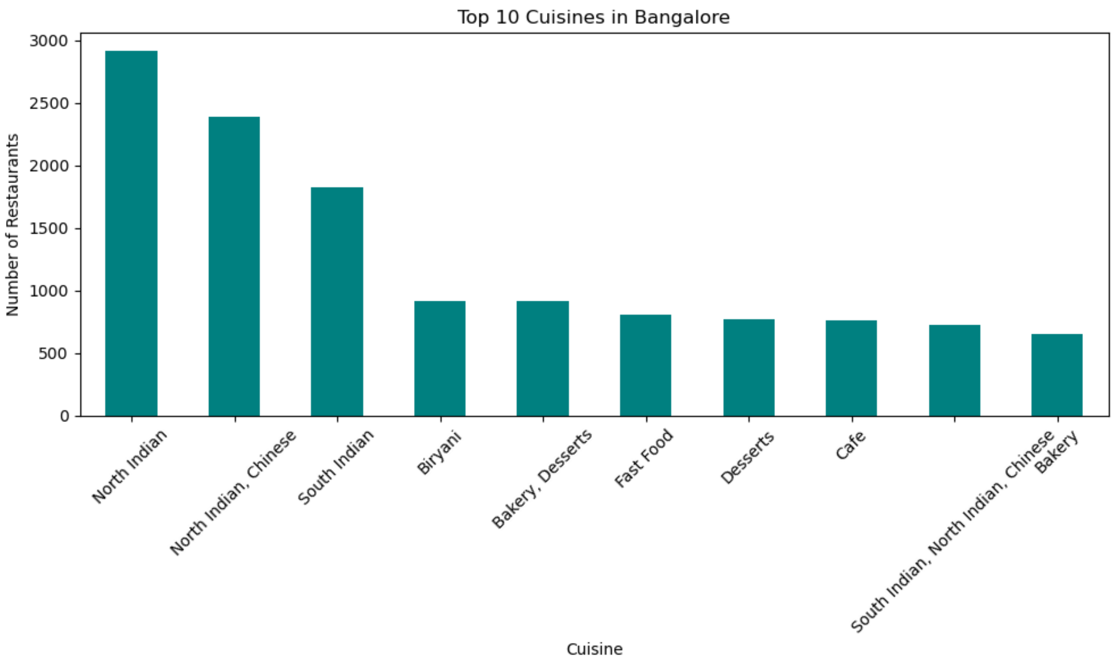
```
votes                          0
rest_type                      0
dish_liked                     0
cuisines                       0
approx_costfor_two_people      0
listed_intype                  0
listed_incity                  0
dtype: int64
       online_order  book_table          rate         votes  \
count           0.0         0.0  51717.000000  51717.000000
mean            NaN         NaN      3.700362    283.697527
std             NaN         NaN      0.395391    803.838853
min             NaN         NaN      1.800000      0.000000
25%             NaN         NaN      3.500000      7.000000
50%             NaN         NaN      3.700000     41.000000
75%             NaN         NaN      3.900000    198.000000
max             NaN         NaN      4.900000  16832.000000

       approx_costfor_two_people
count               51717.000000
mean                  554.391689
std                   437.563723
min                    40.000000
25%                   300.000000
50%                   400.000000
75%                   650.000000
max                  6000.000000
```

[27]:

```python
merged_df = pd.merge(data1, data2, on='listed_incity', how='left')
```

[87]:

```python
import matplotlib.pyplot as plt
import pandas as pd
data1 = pd.read_csv("C:\\Users\\nagal\\Downloads\\zomato_data.csv")
data1['cuisines'] = data1['cuisines'].fillna('Other')
top_cuisines = data1['cuisines'].value_counts().head(10)
plt.figure(figsize=(10, 6))
top_cuisines.plot(kind='bar', color='teal')
plt.title('Top 10 Cuisines in Bangalore')
plt.xlabel('Cuisine')
plt.ylabel('Number of Restaurants')
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```



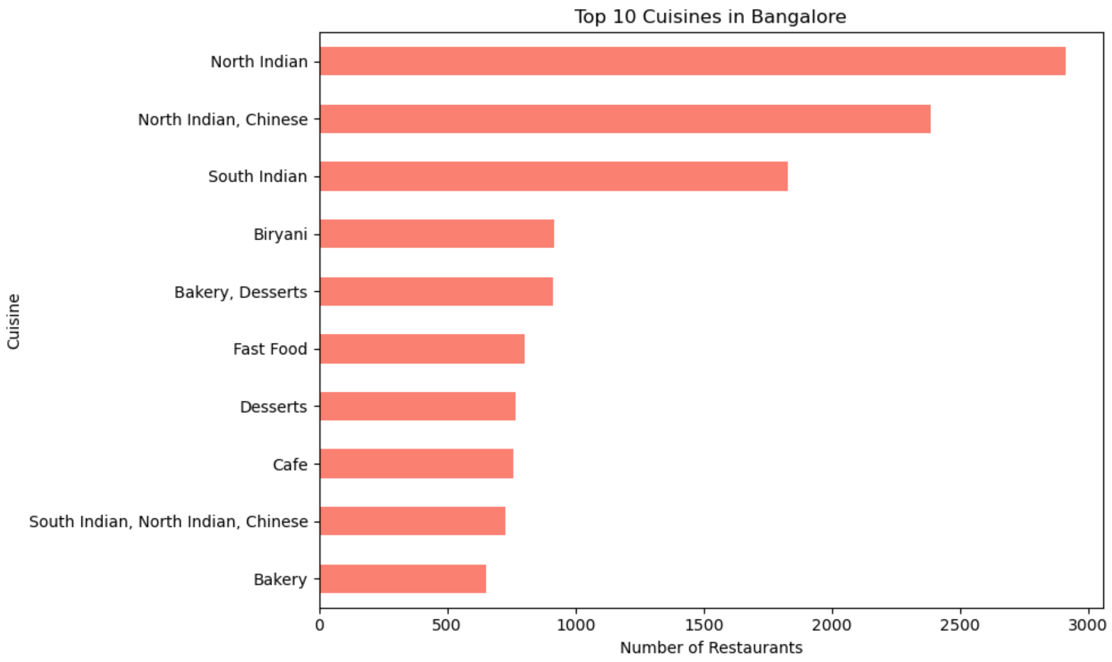
```
[31]: print(merged_df.info())
      print(merged_df[['listed_incity', 'Latitude', 'Longitude']].head())
      print(merged_df.isnull().sum())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 51717 entries, 0 to 51716
Data columns (total 12 columns):
 #   Column                      Non-Null Count  Dtype
---  ------                      --------------  -----
 0   online_order                0 non-null      float64
 1   book_table                  0 non-null      float64
 2   rate                        51717 non-null  float64
 3   votes                       51717 non-null  int32
 4   rest_type                   51717 non-null  object
 5   dish_liked                  51717 non-null  object
 6   cuisines                    51717 non-null  object
 7   approx_costfor_two_people   51717 non-null  int32
 8   listed_intype               51717 non-null  object
 9   listed_incity               51717 non-null  object
 10  Latitude                    46137 non-null  float64
 11  Longitude                   46137 non-null  float64
dtypes: float64(5), int32(2), object(5)
memory usage: 4.3+ MB
None
  listed_incity   Latitude  Longitude
0  Banashankari  12.939333  77.553982
```

```
1  Banashankari  12.939333  77.553982
2  Banashankari  12.939333  77.553982
3  Banashankari  12.939333  77.553982
4  Banashankari  12.939333  77.553982
online_order                 51717
book_table                   51717
rate                             0
votes                            0
rest_type                        0
dish_liked                       0
cuisines                         0
approx_costfor_two_people        0
listed_intype                    0
listed_incity                    0
Latitude                      5580
Longitude                     5580
dtype: int64
```

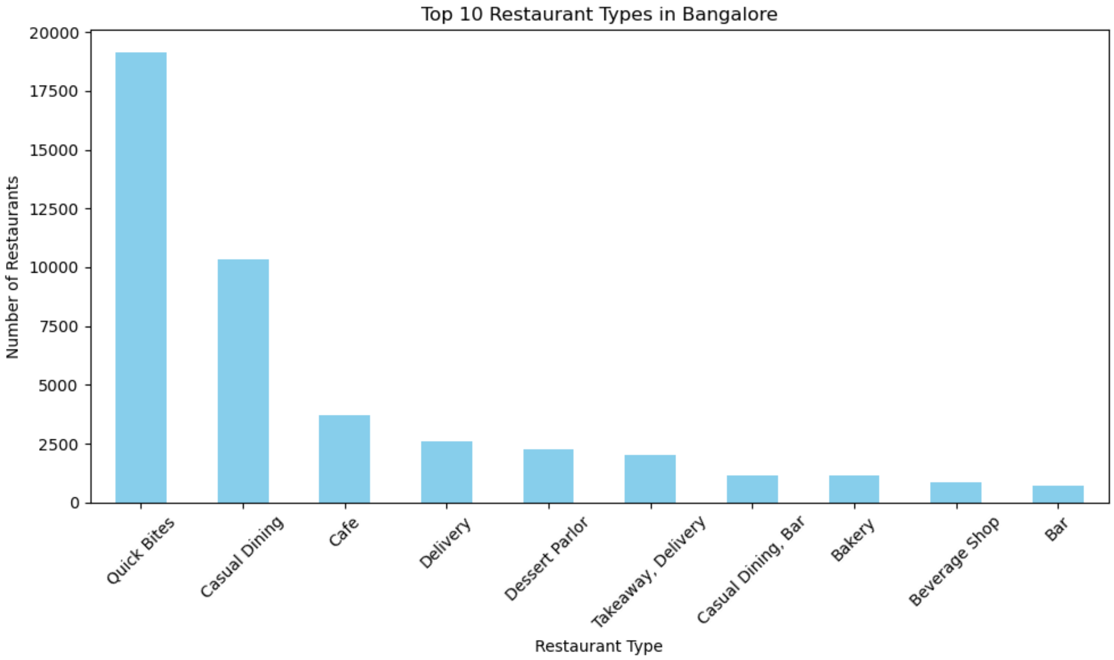
[89]:

```python
import matplotlib.pyplot as plt
import pandas as pd
data1 = pd.read_csv("C:\\Users\\nagal\\Downloads\\zomato_data.csv")
data1['cuisines'] = data1['cuisines'].fillna('Other')
top_cuisines = data1['cuisines'].value_counts().head(10)
plt.figure(figsize=(10, 6))
top_cuisines.plot(kind='barh', color='salmon')
plt.title('Top 10 Cuisines in Bangalore')
plt.xlabel('Number of Restaurants')
plt.ylabel('Cuisine')
plt.gca().invert_yaxis()  # Optional: highest bar on top
plt.tight_layout()
plt.show()
```

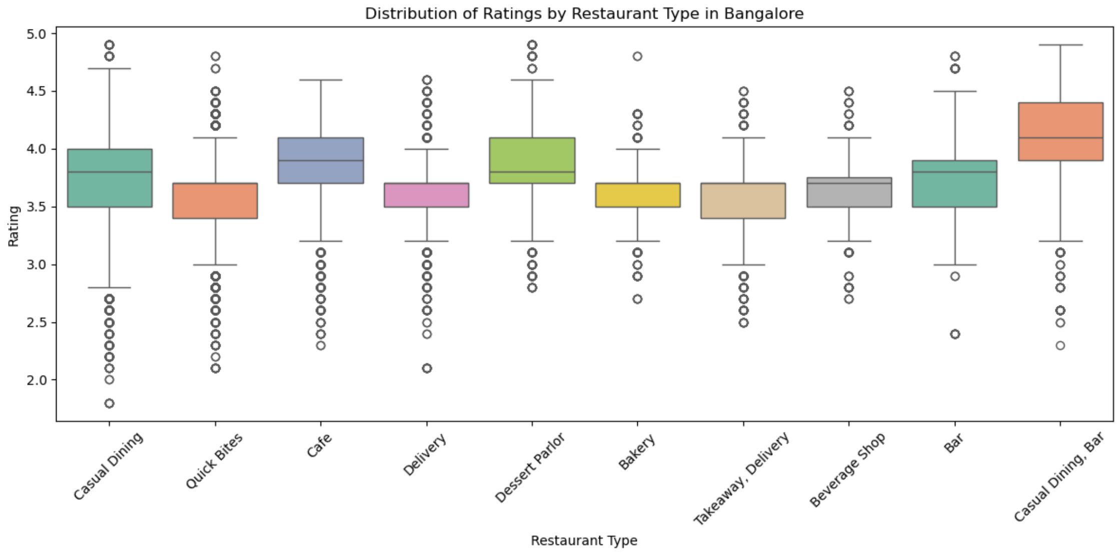


[91]:

```python
import matplotlib.pyplot as plt
merged_df['rest_type'] = merged_df['rest_type'].fillna('Unknown')
top_rest_types = merged_df['rest_type'].value_counts().head(10)
plt.figure(figsize=(10, 6))
top_rest_types.plot(kind='bar', color='skyblue')
plt.title('Top 10 Restaurant Types in Bangalore')
plt.xlabel('Restaurant Type')
plt.ylabel('Number of Restaurants')
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```

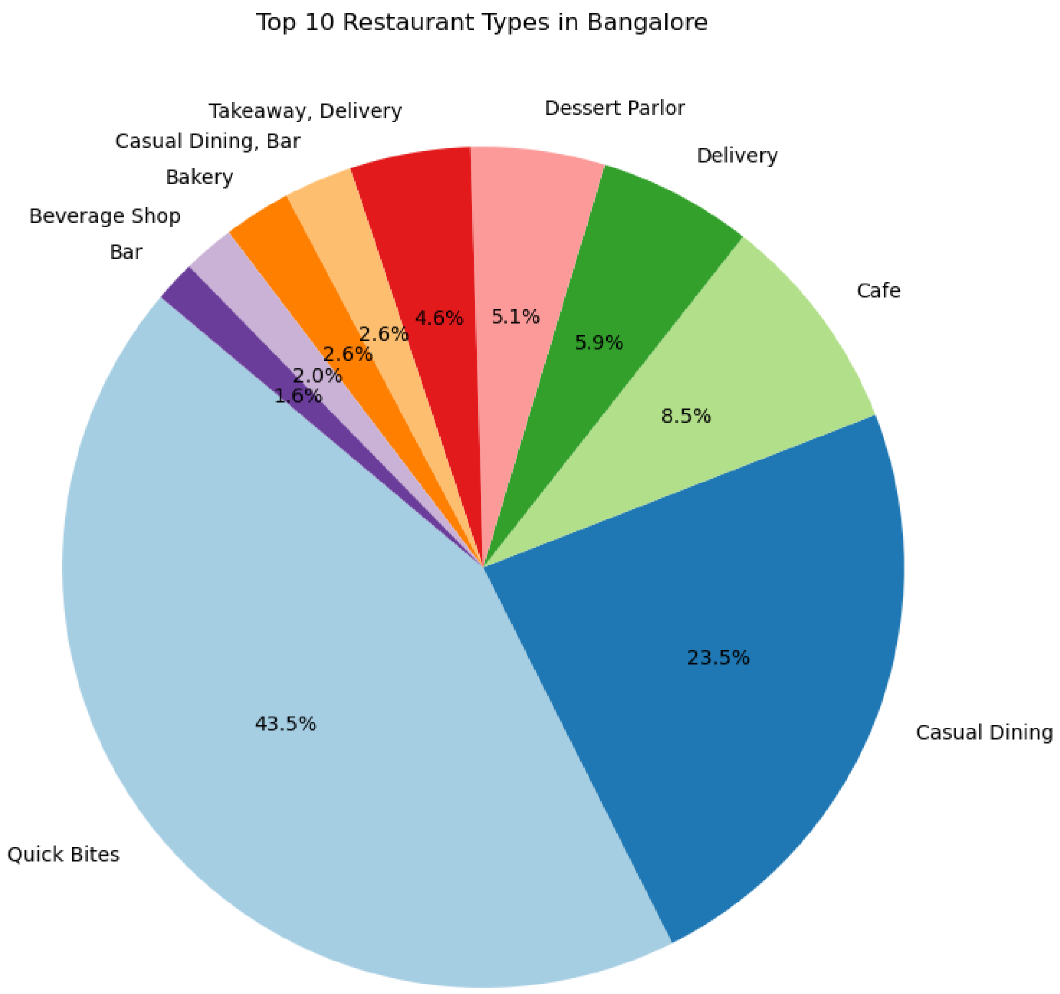


[103]:
```python
import matplotlib.pyplot as plt
import seaborn as sns

merged_df['rest_type'] = merged_df['rest_type'].fillna('Unknown')
merged_df['rate'] = pd.to_numeric(merged_df['rate'], errors='coerce')
top_rest_types = merged_df['rest_type'].value_counts().head(10).index
filtered_df = merged_df[merged_df['rest_type'].isin(top_rest_types)]

plt.figure(figsize=(12, 6))
sns.boxplot(x='rest_type', y='rate', data=filtered_df, hue='rest_type',␣
 ↪palette='Set2', legend=False)
plt.title('Distribution of Ratings by Restaurant Type in Bangalore')
plt.xlabel('Restaurant Type')
plt.ylabel('Rating')
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```



[105]:

```python
import matplotlib.pyplot as plt
merged_df['rest_type'] = merged_df['rest_type'].fillna('Unknown')
top_rest_types = merged_df['rest_type'].value_counts().head(10)
plt.figure(figsize=(8, 8))
top_rest_types.plot(kind='pie', autopct='%1.1f%%', startangle=140, colors=plt.
 ↪cm.Paired.colors)
plt.title('Top 10 Restaurant Types in Bangalore')
plt.ylabel('')  # Hides y-axis label
plt.tight_layout()
plt.show()
```



```
[33]: import folium
      from folium.plugins import HeatMap

      bangalore_map = folium.Map(location=[12.9716, 77.5946], zoom_start=12)

      heat_data = merged_df[['Latitude', 'Longitude']].dropna().values.tolist()

      HeatMap(heat_data).add_to(bangalore_map)


      bangalore_map.save("bangalore_restaurant_density.html")
```

```
[41]: print(merged_df[['Latitude','Longitude']].isnull().sum())
```

```
Latitude     5580
Longitude    5580
dtype: int64
```

```
[43]: merged_df = merged_df.dropna(subset=['Latitude','Longitude'])
```

```
[45]: merged_df = pd.merge(data1, data2, on='listed_incity', how='left')
      merged_df.head()
```

```
[45]:    online_order  book_table  rate  votes           rest_type  \
      0           NaN         NaN   4.1    775       Casual Dining
      1           NaN         NaN   4.1    787       Casual Dining
      2           NaN         NaN   3.8    918  Cafe, Casual Dining
      3           NaN         NaN   3.7     88         Quick Bites
      4           NaN         NaN   3.8    166       Casual Dining

                                               dish_liked  \
      0  Pasta, Lunch Buffet, Masala Papad, Paneer Laja…
      1  Momos, Lunch Buffet, Chocolate Nirvana, Thai G…
      2  Churros, Cannelloni, Minestrone Soup, Hot Choc…
      3                                        Masala Dosa
      4                                Panipuri, Gol Gappe

                               cuisines  approx_costfor_two_people listed_intype  \
      0  North Indian, Mughlai, Chinese                        800        Buffet
      1     Chinese, North Indian, Thai                        800        Buffet
      2          Cafe, Mexican, Italian                        800        Buffet
      3      South Indian, North Indian                        300        Buffet
      4        North Indian, Rajasthani                        600        Buffet

        listed_incity   Latitude  Longitude
      0  Banashankari  12.939333  77.553982
      1  Banashankari  12.939333  77.553982
      2  Banashankari  12.939333  77.553982
      3  Banashankari  12.939333  77.553982
      4  Banashankari  12.939333  77.553982
```

```
[47]: restaurant_map = folium.Map(location=[12.9716, 77.5946], zoom_start=12)

      for idx, row in merged_df.iterrows():
          if not pd.isna(row['Latitude']) and not pd.isna(row['Longitude']):
              folium.CircleMarker(
                  location=[row['Latitude'], row['Longitude']],
                  radius=1,
                  color='blue',
                  fill=True,
                  fill_color='blue',
                  fill_opacity=0.4
              ).add_to(restaurant_map)

      restaurant_map
```

[47]: <folium.folium.Map at 0x259adf82810>

[51]:
```
print(italian_df.columns)
```

```
Index(['online_order', 'book_table', 'rate', 'votes', 'rest_type',
       'dish_liked', 'cuisines', 'approx_costfor_two_people', 'listed_intype',
       'listed_incity', 'Latitude', 'Longitude'],
      dtype='object')
```

[73]:
```
print(italian_df.columns.tolist())
```

```
['online_order', 'book_table', 'rate', 'votes', 'rest_type', 'dish_liked',
'cuisines', 'approx_costfor_two_people', 'listed_intype', 'listed_incity',
'Latitude', 'Longitude']
```

[77]:
```
italian_map = folium.Map(location=[12.9716, 77.5946], zoom_start=12)

for idx, row in italian_df.iterrows():
    if not pd.isna(row['Latitude']) and not pd.isna(row['Longitude']):
        folium.Marker(
            location=[row['Latitude'], row['Longitude']],
            popup=row['rest_type'],  #  using correct column
            icon=folium.Icon(color='green')
        ).add_to(italian_map)

italian_map
```

[77]: <folium.folium.Map at 0x259aec1e300>