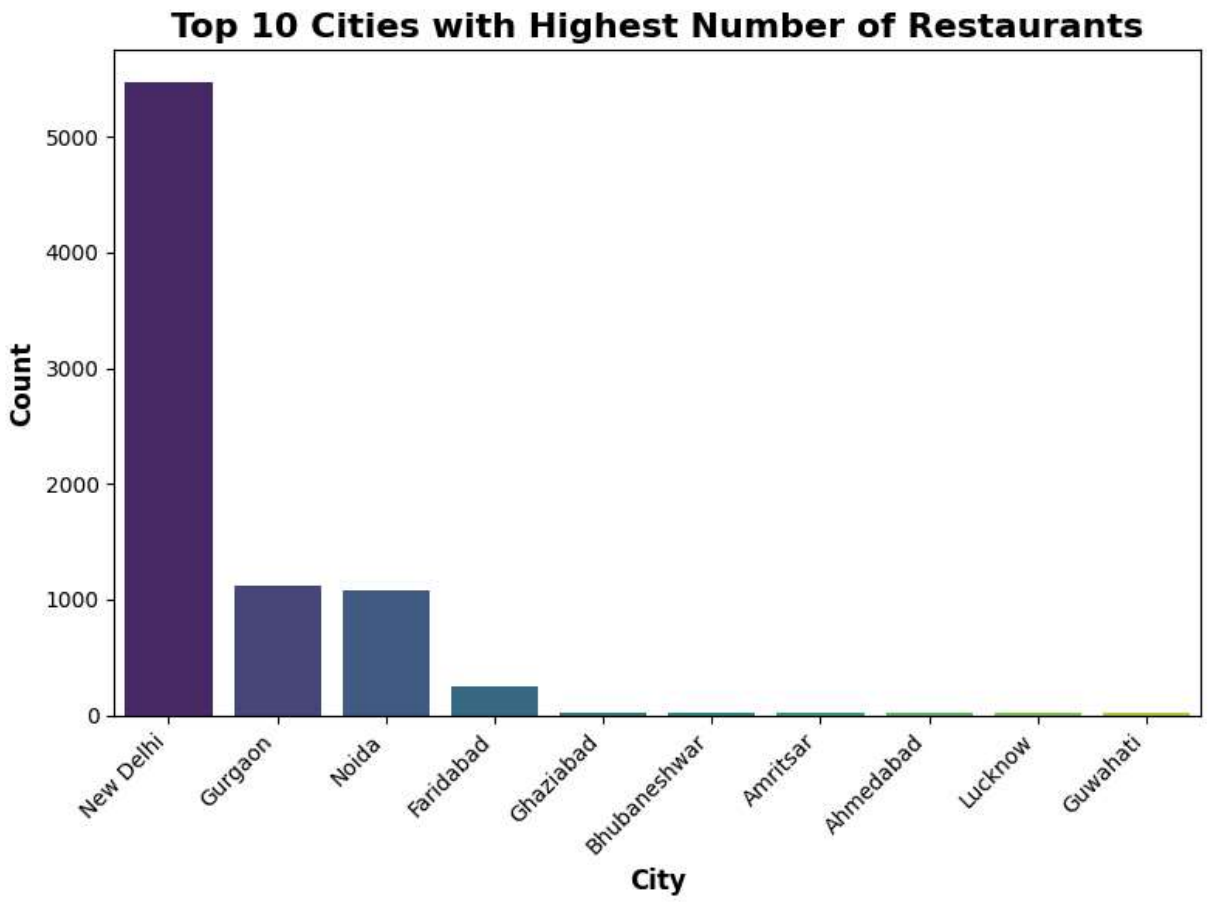
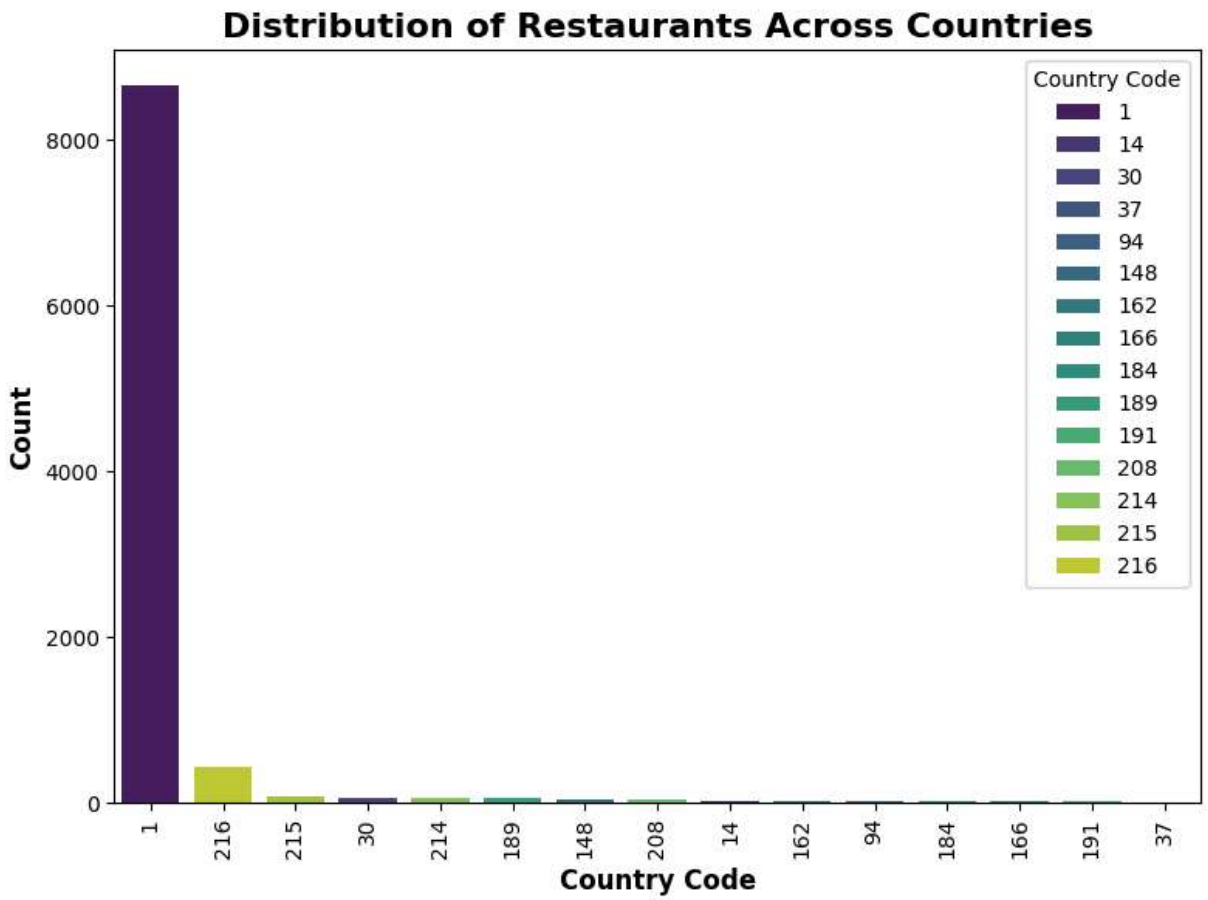
In [25]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
file_path = ('D:/cognifyz/Dataset .csv')
df = pd.read_csv(file_path)
```

In [44]:
```
import pandas as pd
import folium
from folium.plugins import HeatMap
from folium.plugins import MarkerCluster
file_path = 'D:/cognifyz/Dataset .csv'
df = pd.read_csv(file_path)
map_center = [df['Latitude'].mean(), df['Longitude'].mean()]
restaurant_map = folium.Map(location=map_center, zoom_start=10)
marker_cluster = MarkerCluster().add_to(restaurant_map)
heat_data = [[row['Latitude'], row['Longitude']] for index, row in df.iterrows()]
HeatMap(heat_data, radius=10).add_to(restaurant_map)
for _, row in df.iterrows():
    folium.Marker(
        location=[row['Latitude'], row['Longitude']],
        popup=f"Restaurant: {row['Restaurant Name']}<br>Address: {row['Address']}<b
        icon=folium.Icon(color='blue')
    ).add_to(marker_cluster)
restaurant_map.save('restaurant_locations_map_with_heatmap.html')
```

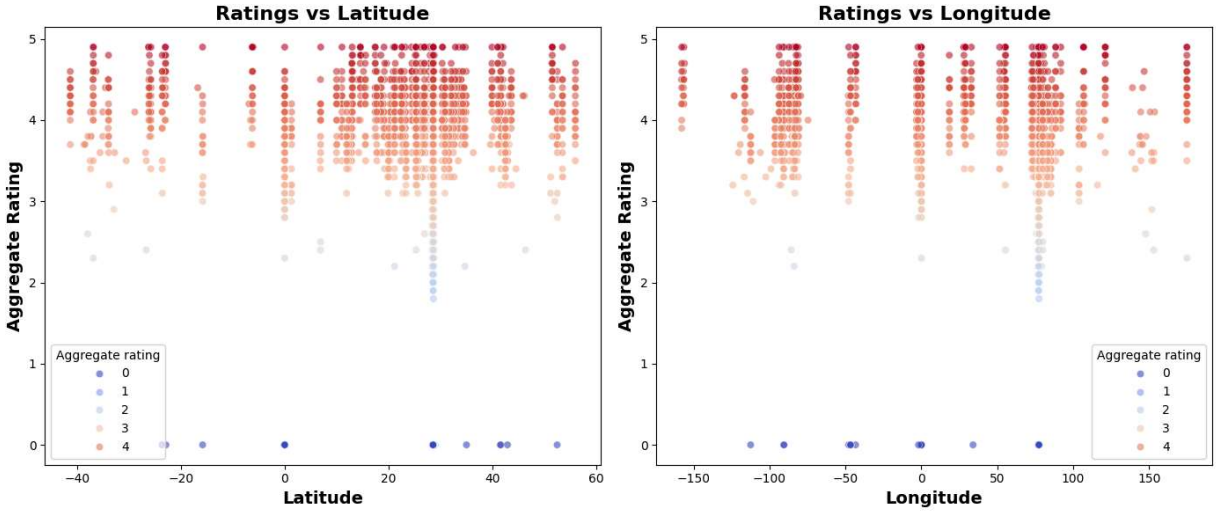
In [62]:
```
top_cities = df['City'].value_counts().head(10)
top_cities_df = pd.DataFrame({
    'City': top_cities.index,
    'Count': top_cities.values
})
palette = sns.color_palette('viridis', len(top_cities_df))
plt.figure(figsize=(8, 6))
sns.barplot(data=top_cities_df, x='City', y='Count', palette=palette, hue='City', l
plt.title('Top 10 Cities with Highest Number of Restaurants', fontsize=16, weight='
plt.xlabel('City', fontsize=12, weight='bold')
plt.ylabel('Count', fontsize=12, weight='bold')
plt.xticks(rotation=45, ha='right', fontsize=10)
plt.yticks(fontsize=10)
plt.tight_layout()
plt.show()
```



In [64]:
```
palette = sns.color_palette('viridis', len(df['Country Code'].value_counts()))
plt.figure(figsize=(8, 6))
sns.countplot(data=df, x='Country Code', order=df['Country Code'].value_counts().in
plt.title('Distribution of Restaurants Across Countries', fontsize=16, weight='bold
plt.xlabel('Country Code', fontsize=12, weight='bold')
plt.ylabel('Count', fontsize=12, weight='bold')
plt.xticks(rotation=90, fontsize=10)
plt.yticks(fontsize=10)
plt.tight_layout()
plt.show()
```



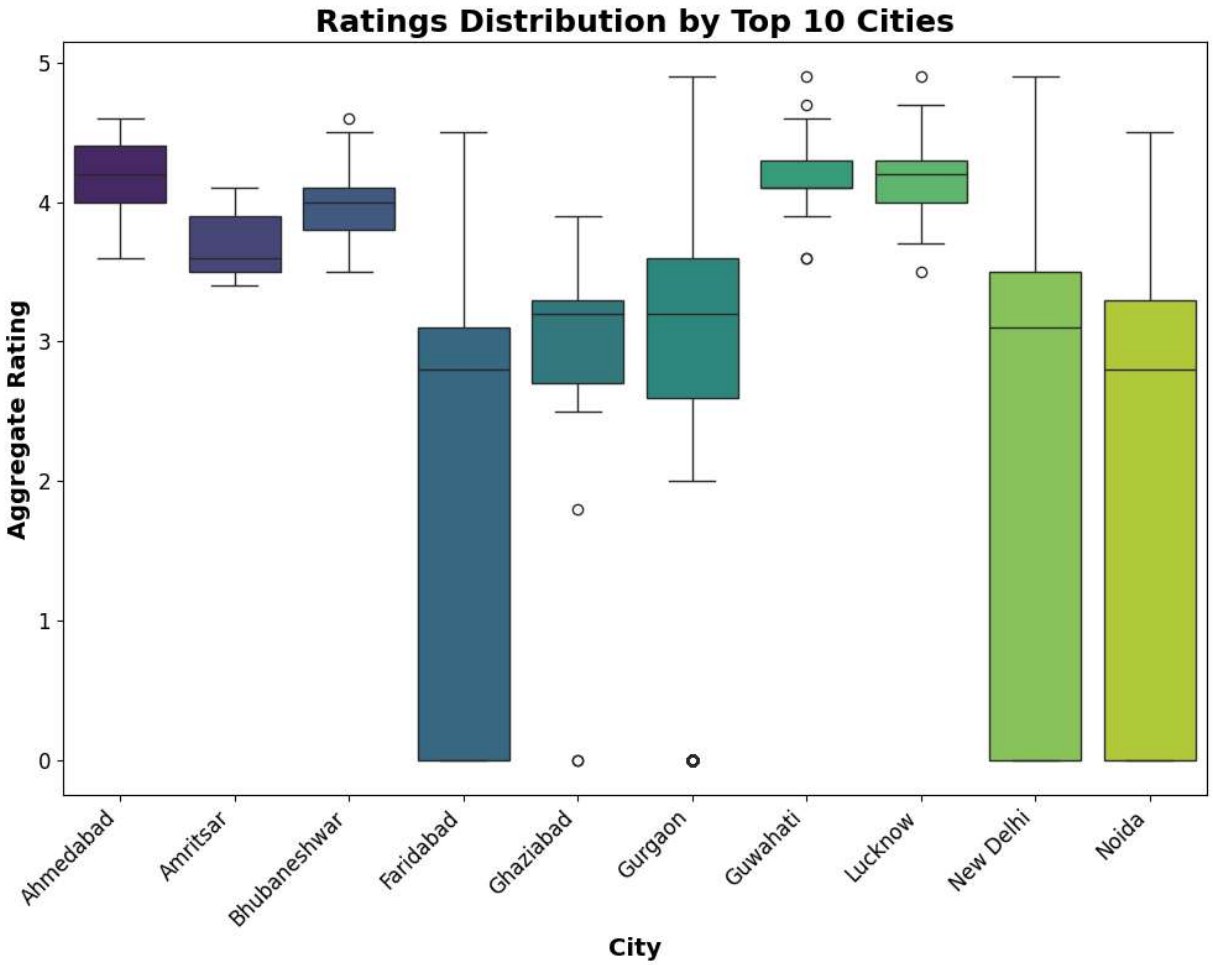
In [41]:
```
plt.figure(figsize=(14, 6))
plt.subplot(1, 2, 1)
sns.scatterplot(data=df, x='Latitude', y='Aggregate rating', hue='Aggregate rating'
plt.title('Ratings vs Latitude', fontsize=16, weight='bold')
plt.xlabel('Latitude', fontsize=14, weight='bold')
plt.ylabel('Aggregate Rating', fontsize=14, weight='bold')
plt.subplot(1, 2, 2)
sns.scatterplot(data=df, x='Longitude', y='Aggregate rating', hue='Aggregate rating
plt.title('Ratings vs Longitude', fontsize=16, weight='bold')
plt.xlabel('Longitude', fontsize=14, weight='bold')
plt.ylabel('Aggregate Rating', fontsize=14, weight='bold')

plt.tight_layout()
plt.show()
```



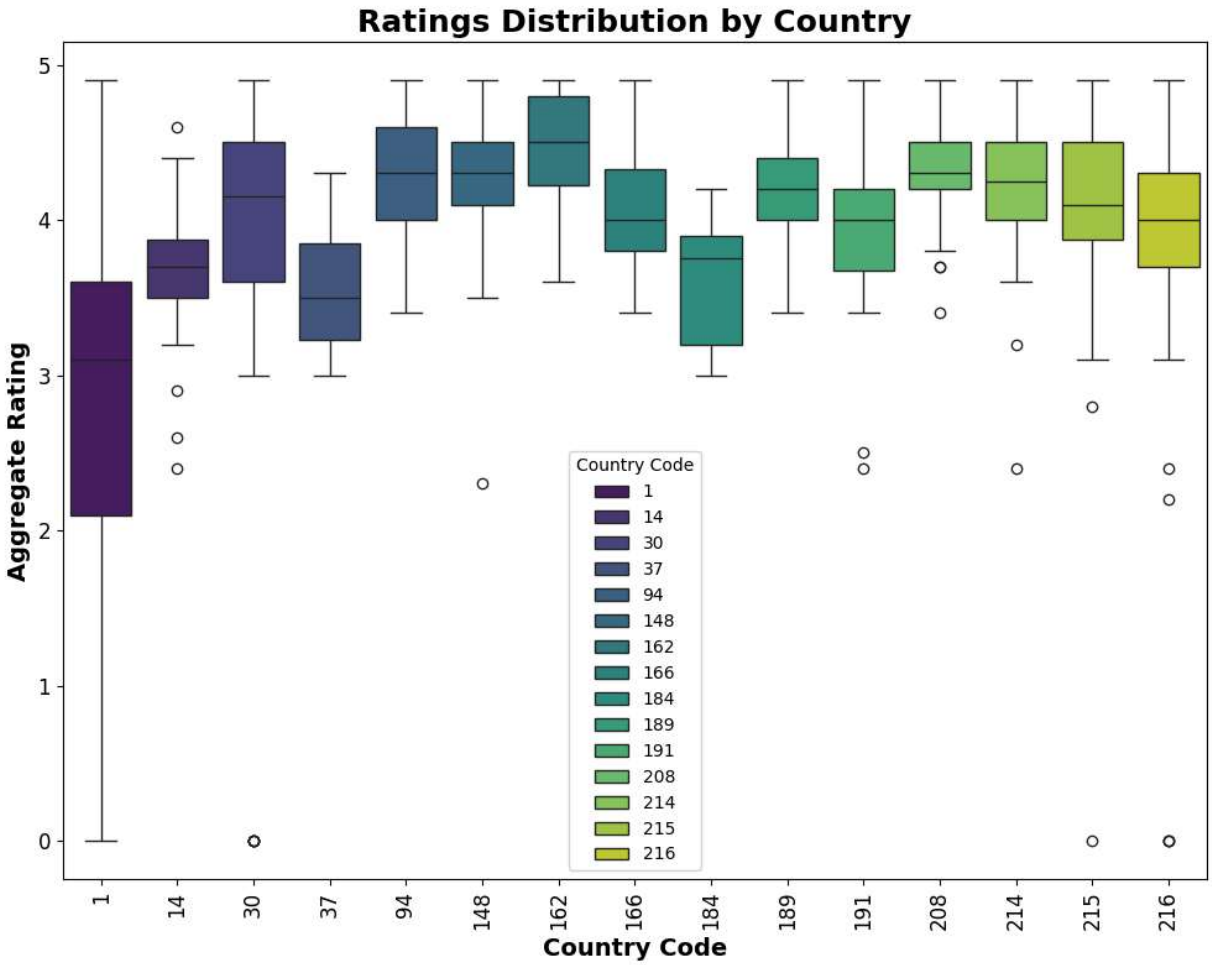
In [40]:
```
correlation_matrix = df[['Latitude', 'Longitude', 'Aggregate rating']].corr()
print("Correlation matrix:\n", correlation_matrix)
```

```
Correlation matrix:
                   Latitude  Longitude  Aggregate rating
Latitude           1.000000   0.043207          0.000516
Longitude          0.043207   1.000000         -0.116818
Aggregate rating   0.000516  -0.116818          1.000000
```

In [80]:
```
top_cities = df['City'].value_counts().head(10).index
filtered_df = df[df['City'].isin(top_cities)]
palette = sns.color_palette('viridis', len(top_cities))
plt.figure(figsize=(10, 8))
sns.boxplot(data=filtered_df, x='City', y='Aggregate rating', palette=palette, hue=
plt.title('Ratings Distribution by Top 10 Cities', fontsize=18, weight='bold')
plt.xlabel('City', fontsize=14, weight='bold')
plt.ylabel('Aggregate Rating', fontsize=14, weight='bold')
plt.xticks(rotation=45, ha='right', fontsize=12)
plt.yticks(fontsize=12)
plt.tight_layout()
plt.show()
```



In [79]:
```
num_countries = df['Country Code'].nunique()
palette = sns.color_palette('viridis', n_colors=num_countries)
plt.figure(figsize=(10, 8))
sns.boxplot(data=df, x='Country Code', y='Aggregate rating', palette=palette, hue='
plt.title('Ratings Distribution by Country', fontsize=18, weight='bold')
plt.xlabel('Country Code', fontsize=14, weight='bold')
plt.ylabel('Aggregate Rating', fontsize=14, weight='bold')
plt.xticks(rotation=90, fontsize=12)
plt.yticks(fontsize=12)
plt.tight_layout()
plt.show()
```



In [48]:
```
import folium
from folium.plugins import HeatMap
map_center = [df['Latitude'].mean(), df['Longitude'].mean()]
restaurant_map = folium.Map(location=map_center, zoom_start=10)
heat_data = [[row['Latitude'], row['Longitude'], row['Aggregate rating']] for index
HeatMap(heat_data, radius=15, blur=10).add_to(restaurant_map)
restaurant_map.save('ratings_heatmap_map.html')
```

In [ ]:
```
```