In [1]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [2]:
```
file_path = ('D:/cognifyz/Dataset .csv')
df = pd.read_csv(file_path)
```

In [3]:
```
numerical_stats = df.describe()
print("Basic statistical measures for numerical columns:\n", numerical_stats)
```

```
Basic statistical measures for numerical columns:
       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
```

In [5]:
```
numerical_df = df.select_dtypes(include='number')
median_values = numerical_df.median()
print("Median values for numerical columns:\n", median_values)
```

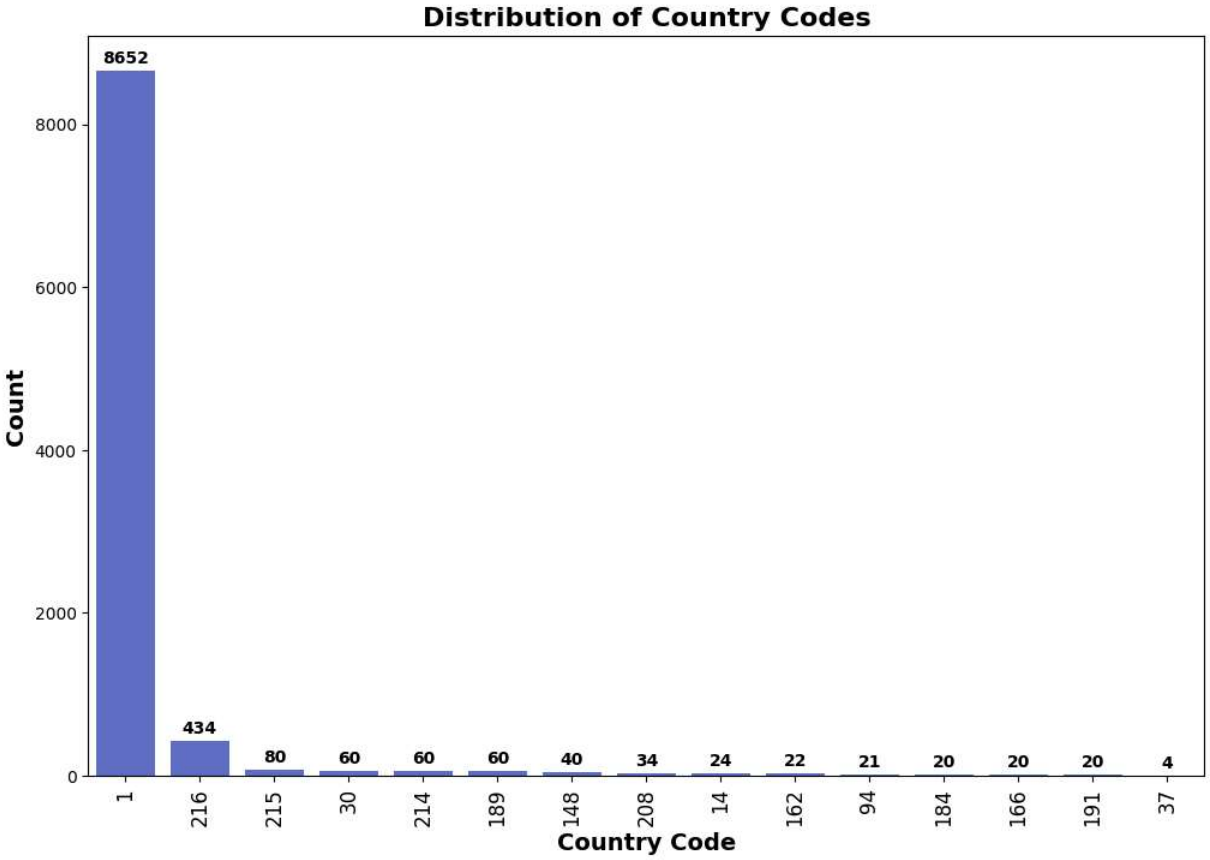
```
Median values for numerical columns:
 Restaurant ID         6.004089e+06
Country Code           1.000000e+00
Longitude              7.719196e+01
Latitude               2.857047e+01
Average Cost for two   4.000000e+02
Price range            2.000000e+00
Aggregate rating       3.200000e+00
Votes                  3.100000e+01
dtype: float64
```

In [32]:
```
country_code_distribution = df['Country Code'].value_counts()
print("Distribution of Country Code:\n", country_code_distribution)

plt.figure(figsize=(12, 8))
colors = sns.color_palette("coolwarm", len(df['Country Code'].value_counts()))
sns.countplot(data=df, x='Country Code', order=df['Country Code'].value_counts().in
              color=colors[0], width=0.8)  # Use a single color for the bars
plt.title('Distribution of Country Codes', fontsize=16, weight='bold')
plt.xlabel('Country Code', fontsize=14, weight='bold')
```

```
plt.ylabel('Count', fontsize=14, weight='bold')
plt.xticks(rotation=90, fontsize=12)
for p in plt.gca().patches:
    plt.text(p.get_x() + p.get_width() / 2, p.get_height() + 50,
             int(p.get_height()), ha='center', va='bottom', fontsize=10, weight='bo
plt.show()
```

```
Distribution of Country Code:
 Country Code
1      8652
216     434
215      80
30       60
214      60
189      60
148      40
208      34
14       24
162      22
94       21
184      20
166      20
191      20
37        4
Name: count, dtype: int64
```



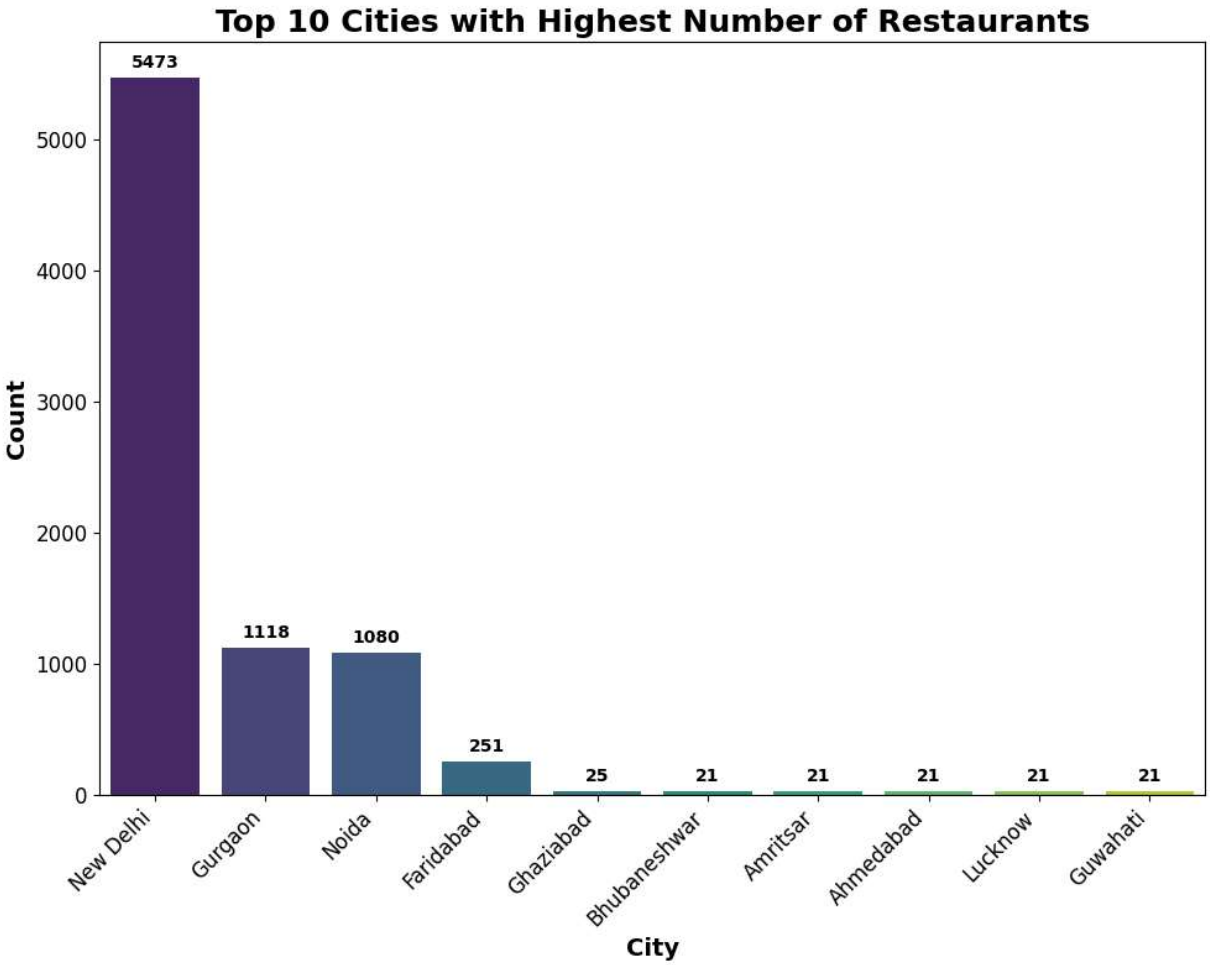
In [39]:
```
city_distribution = df['City'].value_counts()
print("Distribution of City:\n", city_distribution)
top_cities = city_distribution.head(10)
colors = sns.color_palette("viridis", len(top_cities))
top_cities_df = pd.DataFrame({
```

```
    'City': top_cities.index,
    'Count': top_cities.values
})

plt.figure(figsize=(10, 8))
bars = sns.barplot(data=top_cities_df, x='City', y='Count', hue='City', palette=col
plt.title('Top 10 Cities with Highest Number of Restaurants', fontsize=18, weight='
plt.xlabel('City', fontsize=14, weight='bold')
plt.ylabel('Count', fontsize=14, weight='bold')
plt.xticks(rotation=45, ha='right', fontsize=12)
plt.yticks(fontsize=12)
for bar in bars.patches:
    height = bar.get_height()
    bars.text(bar.get_x() + bar.get_width() / 2, height + 50,
              int(height), ha='center', va='bottom', fontsize=10, weight='bold')

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

```
Distribution of City:
 City
New Delhi           5473
Gurgaon             1118
Noida               1080
Faridabad            251
Ghaziabad             25
                    ...
Panchkula              1
Mc Millan              1
Mayfield               1
Macedon                1
Vineland Station       1
Name: count, Length: 141, dtype: int64
```

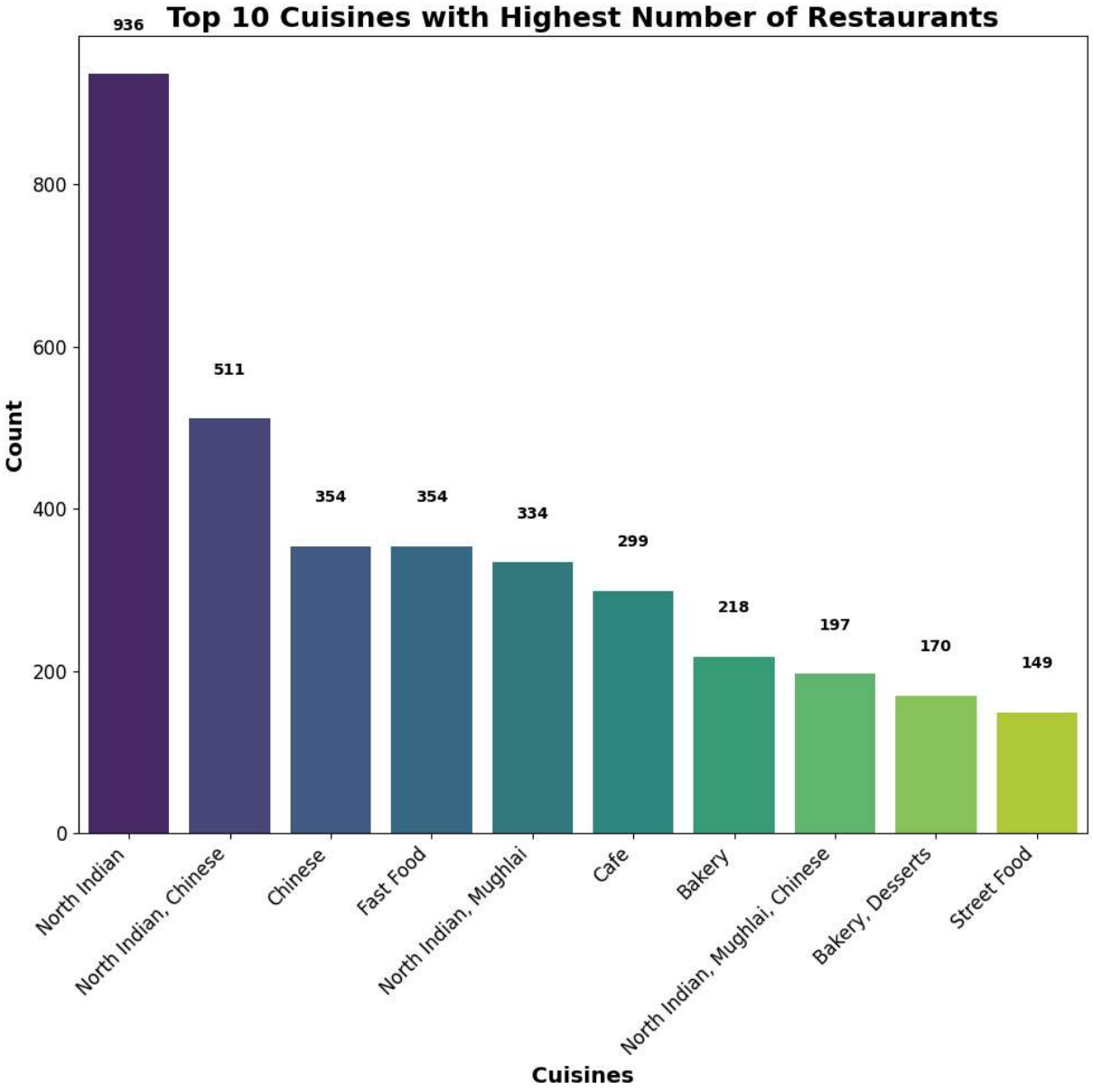


In [40]:
```
cuisines_distribution = df['Cuisines'].value_counts()
top_cuisines = cuisines_distribution.head(10)
print("Top 10 Cuisines with the highest number of restaurants:\n", top_cuisines)
top_cuisines_df = pd.DataFrame({
    'Cuisines': top_cuisines.index,
    'Count': top_cuisines.values
})


plt.figure(figsize=(10, 10))
colors = sns.color_palette("viridis", len(top_cuisines))
bars = sns.barplot(data=top_cuisines_df, x='Cuisines', y='Count', palette=colors, h
plt.title('Top 10 Cuisines with Highest Number of Restaurants', fontsize=18, weight
plt.xlabel('Cuisines', fontsize=14, weight='bold')
plt.ylabel('Count', fontsize=14, weight='bold')
plt.xticks(rotation=45, ha='right', fontsize=12)
plt.yticks(fontsize=12)
for bar in bars.patches:
    height = bar.get_height()
    bars.text(bar.get_x() + bar.get_width() / 2, height + 50,
              int(height), ha='center', va='bottom', fontsize=10, weight='bold')

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

```
Top 10 Cuisines with the highest number of restaurants:
 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
```



In [ ]: