

Level 1: Restaurant Data Analytics | Data Analysis

- Task 1: Top Cuisines
- Task 2: City Analysis
- Task 3: Price Range Distribution
- Task 4: Online Delivery

Step 1: Import necessary Python libraries.

```
In [183... import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

Step 2: Load the dataset into a DataFrame

```
In [30]: restaurent_df = pd.read_csv(r"Dataset .csv")
restaurent_df
```

Out[30]:

	Restaurant ID	Restaurant Name	Country Code	City	Address	Locality	Locality Verbose
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..
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..
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..
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..
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..
...
9546	5915730	Naml Gurme	208	İstanbul	Kemankeş Karamustafa Paşası Mahallesi, Rıhtım ...	Karaköy	Karaköy, İstanbul
9547	5908749	Ceviz Aca	208	İstanbul	Koşuyolu Mahallesi, Muhittin Köstendağ Cadd...	Koşuyolu	Koşuyolu, İstanbul
9548	5915807	Huqqa	208	İstanbul	Kuruçeşme Mahallesi, Muallim Naci Caddesi, N...	Kuruçeşme	Kuruçeşme, İstanbul
9549	5916112	Ak Kahve	208	İstanbul	Kuruçeşme Mahallesi, Muallim Naci Caddesi, N...	Kuruçeşme	Kuruçeşme, İstanbul
9550	5927402	Walter's Coffee Roastery	208	İstanbul	Cafea Mahallesi, Bademaltı Sokak, No 21/B, ...	Moda	Moda, İstanbul

9551 rows × 21 columns

Step 3: Basic Inspection on given dataset

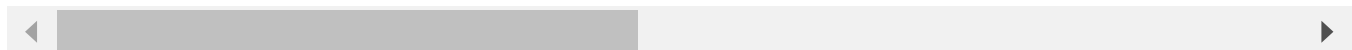
- Top 5 rows - using head

```
In [37]: restaurent_df.head()
```

```
Out[37]:
```

	Restaurant ID	Restaurant Name	Country Code	City	Address	Locality	Locality Verbose	Longitude
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...	121.027535
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...	121.014101
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...	121.056831
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...	121.056475
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...	121.057508

5 rows × 21 columns



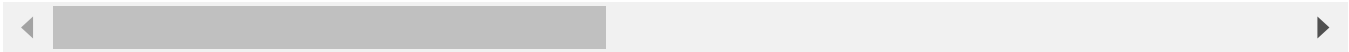
- bottom 5 rows using tail

```
In [43]: restaurent_df.tail()
```

Out[43]:

	Restaurant ID	Restaurant Name	Country Code	City	Address	Locality	Locality Verbose
9546	5915730	Namlı Gurme	208	İstanbul	Kemankeş Karamustafa Paşa Mahallesi, Rıhtım ...	Karaköy	Karaköy, İstanbul
9547	5908749	Ceviz Aca	208	İstanbul	Koşuyolu Mahallesi, Muhittin İsmet Paşa Caddesi	Koşuyolu	Koşuyolu, İstanbul
9548	5915807	Huqqa	208	İstanbul	Kuruçeşme Mahallesi, Muallim Naci Caddesi, N...	Kuruçeşme	Kuruçeşme, İstanbul
9549	5916112	Ak Kahve	208	İstanbul	Kuruçeşme Mahallesi, Muallim Naci Caddesi, N...	Kuruçeşme	Kuruçeşme, İstanbul
9550	5927402	Walter's Coffee Roastery	208	İstanbul	Cafea Mahallesi, Bademaltı Sokak, No 21/B, ...	Moda	Moda, İstanbul

5 rows × 21 columns



• Inspecting Column Names and Data Types

```
In [48]: restaurent_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
```

- **Checking for Missing Values**

```
In [51]: restaurent_df.isnull().sum()
```

```
Out[51]: 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 [61]: cuisines = restaurent_df['Cuisines'].dropna().str.split(", ").explode()
cuisines
```

```
Out[61]: 0          French
0          Japanese
0          Desserts
1          Japanese
2          Seafood
...
9547       Cafe
9548       Italian
9548       World Cuisine
9549       Restaurant Cafe
9550       Cafe
Name: Cuisines, Length: 19710, dtype: object
```

• Basic Statistical Summary

```
In [66]: restaurent_df.describe()
```

Out[66]:

	Restaurant ID	Country Code	Longitude	Latitude	Average Cost for two	Price range	Aggregate rating
count	9.551000e+03	9551.000000	9551.000000	9551.000000	9551.000000	9551.000000	9551.000000
mean	9.051128e+06	18.365616	64.126574	25.854381	1199.210763	1.804837	2.666370
std	8.791521e+06	56.750546	41.467058	11.007935	16121.183073	0.905609	1.516378
min	5.300000e+01	1.000000	-157.948486	-41.330428	0.000000	1.000000	0.000000
25%	3.019625e+05	1.000000	77.081343	28.478713	250.000000	1.000000	2.500000
50%	6.004089e+06	1.000000	77.191964	28.570469	400.000000	2.000000	3.200000
75%	1.835229e+07	1.000000	77.282006	28.642758	700.000000	2.000000	3.700000
max	1.850065e+07	216.000000	174.832089	55.976980	800000.000000	4.000000	4.900000

• Checking Unique Values

```
In [71]: restaurent_df.nunique()
```

```
Out[71]: Restaurant ID      9551
Restaurant Name      7446
Country Code         15
City                 141
Address              8918
Locality             1208
Locality Verbose     1265
Longitude            8120
Latitude             8677
Cuisines             1825
Average Cost for two  140
Currency             12
Has Table booking     2
Has Online delivery   2
Is delivering now     2
Switch to order menu  1
Price range           4
Aggregate rating      33
Rating color          6
Rating text           6
Votes                1012
dtype: int64
```

- **Checking Shape**

```
In [77]: restaurent_df.shape
```

```
Out[77]: (9551, 21)
```

Task-1 Top Cuisines

- **Determine the top three most common cuisines in the dataset.**

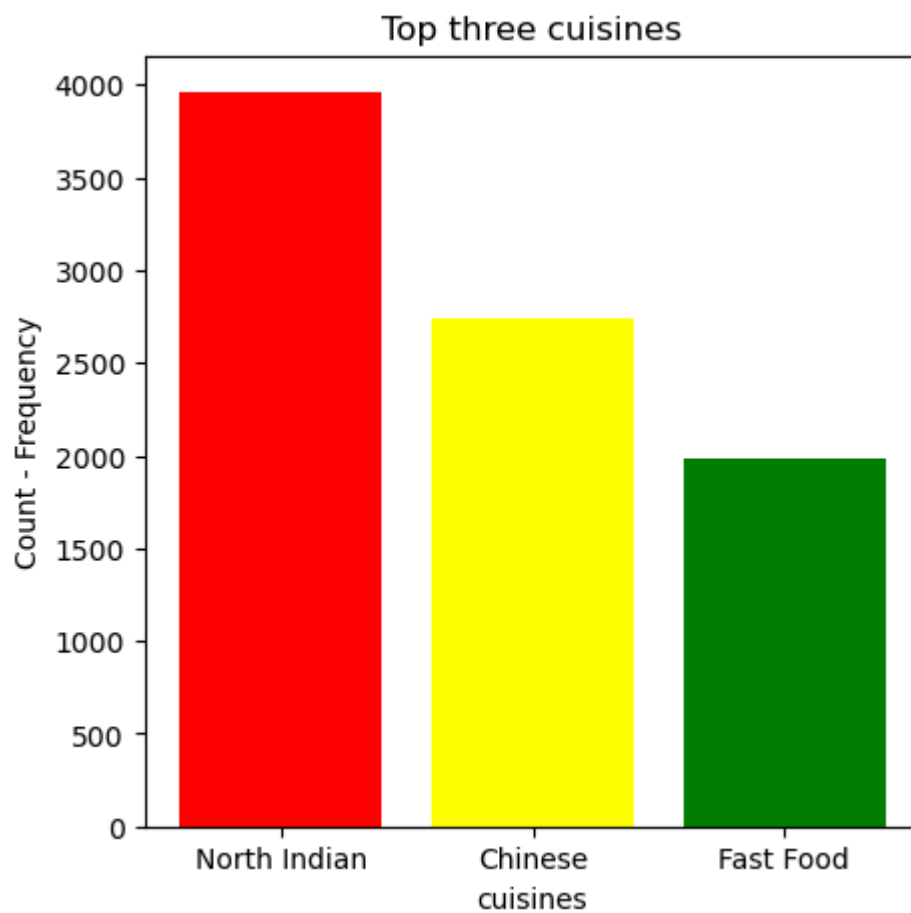
```
In [85]: value_counts = cuisines.value_counts().reset_index().head(3)
value_counts
```

```
Out[85]:
```

	Cuisines	count
0	North Indian	3960
1	Chinese	2735
2	Fast Food	1986

```
In [102... plt.figure(figsize=(5,5))
values=value_counts['Cuisines']
labels=value_counts['count']
plt.bar(values,labels, color=['red','yellow','green'])
plt.title('Top three cuisines')
plt.xlabel('cuisines')
plt.ylabel("Count - Frequency")
plt.show
```

```
Out[102... <function matplotlib.pyplot.show(close=None, block=None)>
```



- Calculate the percentage of restaurants that serve each of the top cuisines

In [108...

```
value_counts['Percentage'] = round((value_counts['count'] / len(restaurent_df))*100,2)
value_counts
```

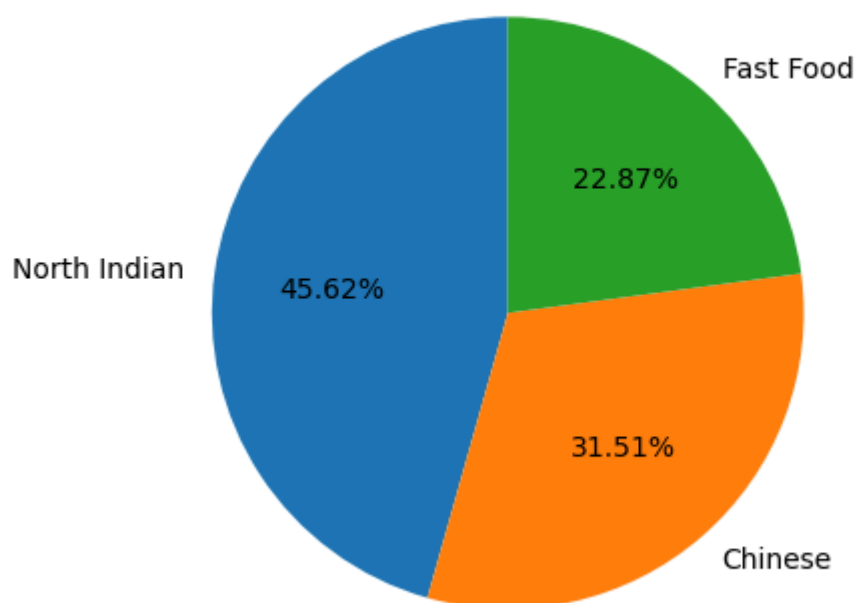
Out[108...

	Cuisines	count	Percentage
0	North Indian	3960	41.46
1	Chinese	2735	28.64
2	Fast Food	1986	20.79

In [134...

```
plt.title('Percentage of restaurantrants that serve each of the top cuisines')
plt.pie(value_counts['Percentage'],labels=value_counts['Cuisines'],autopct='%0.2f%',startang
plt.show()
```


Percentage of restauranttrants that serve each of the top cuisines



Task-2 City Analysis

- **Identify the city with the highest number of restaurants in the dataset.**

```
In [138... city_restaurant_count = restaurent_df.groupby('City')['Restaurant Name'].count()
max_restaurant_city = city_restaurant_count.idxmax()
max_restaurant_count = city_restaurant_count.max()
print(f"{max_restaurant_city} has highest number of restaurants, the count of restaurants is
```

New Delhi has highest number of restaurants, the count of restaurants is 5473 restaurants.

- **Calculate the average rating for restaurants in each city.**

```
In [145... avg_ratting_by_city = restaurent_df.groupby("City")["Aggregate rating"].mean().reset_index()
avg_ratting_by_city
```

Out[145...

	City	Aggregate rating
0	Abu Dhabi	4.300000
1	Agra	3.965000
2	Ahmedabad	4.161905
3	Albany	3.555000
4	Allahabad	3.395000
...
136	Weirton	3.900000
137	Wellington City	4.250000
138	Winchester Bay	3.200000
139	Yorkton	3.300000
140	istanbul	4.292857

141 rows × 2 columns

- **Determine the city with the highest average rating**

In [148...

```
avg_rating = restaurent_df.groupby("City")["Aggregate rating"].mean().reset_index()
avg_rating = avg_rating.sort_values(by="Aggregate rating",ascending=False)
print("City with the highest average rating")
avg_rating.head(1)
```

City with the highest average rating

Out[148...

	City	Aggregate rating
56	Inner City	4.9

Observations

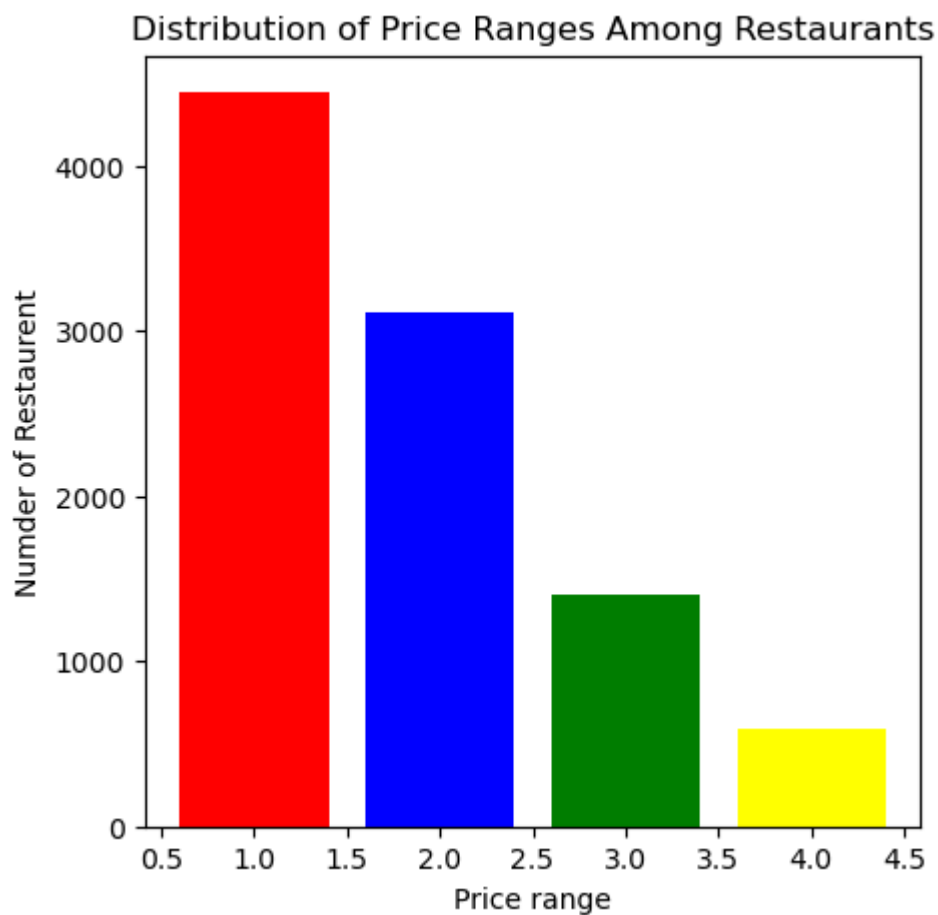
- **City with the highest average rating**
- **Inner City with 4.9 as Avg Rating**

Task-3 Price Range Distribution

- **Create a histogram or bar chart to visualize the distribution of price ranges among the restaurants.**

In [169...

```
price_counts = restaurent_df["Price range"].value_counts()
plt.figure(figsize=(5,5))
plt.bar(price_counts.index, price_counts.values, color=['red','blue','green','yellow'])
plt.ylabel('Numder of Restaurant')
plt.xlabel('Price range')
plt.title('Distribution of Price Ranges Among Restaurants')
plt.show()
```



Observations

- Distribution of price ranges among the restaurants
 - 1
 - 2
 - 3
 - 4
- Calculate the percentage of restaurants in each price range category.

```
In [187... value_counts = restaurant_df["Price range"].value_counts().reset_index()
value_counts.columns = ['Price-Range', 'Count']
total_count = value_counts['Count'].sum()
value_counts['Percentage'] = round((value_counts['Count'] / total_count)*100,2)
df = pd.DataFrame(value_counts)
df
```

Out[187...

	Price-Range	Count	Percentage
0	1	4444	46.53
1	2	3113	32.59
2	3	1408	14.74
3	4	586	6.14

Observations

- Percentage of restaurants in each price range category.

- Price Range :1 Percentage : 46.53%
- Price Range :2 Percentage: 32.59%
- Price Range :3 Percentage: 14.74%
- Price Range: 4 Percentage: 6.14%

Task 4: Online Delivery

- **Determine the percentage of restaurants that offer online delivery**

In [216...

```
total_restaurant_count = restaurent_df.shape[0]
online_restaurant_count = restaurent_df[restaurent_df['Has Online delivery'] == 'Yes']
online_restaurant_count = online_restaurant_count.shape[0]
percentage=round((online_restaurant_count/total_restaurant_count)*100,2)
print("percentage of online order taken by the restaurants")
print(percentage)
```

percentage of online order taken by the restaurants
25.66

Observations

- Percentage of restaurants that offer online delivery - 25.66%
- **Compare the average ratings of restaurants with and without online delivery.**

In [224...

```
print("average rating of restaurant with and without online delivery")
restaurent_df.groupby("Has Online delivery")['Aggregate rating'].mean().round(2).reset_index()
```

average rating of restaurant with and without online delivery

Out[224...

	Has Online delivery	Aggregate rating
0	No	2.47
1	Yes	3.25

Observations

- average ratings of restaurants with and without online delivery
 - No - Online Delivery - Avg Rating - 2.47
 - Online Delivery - Avg Rating - 3.25

In []: