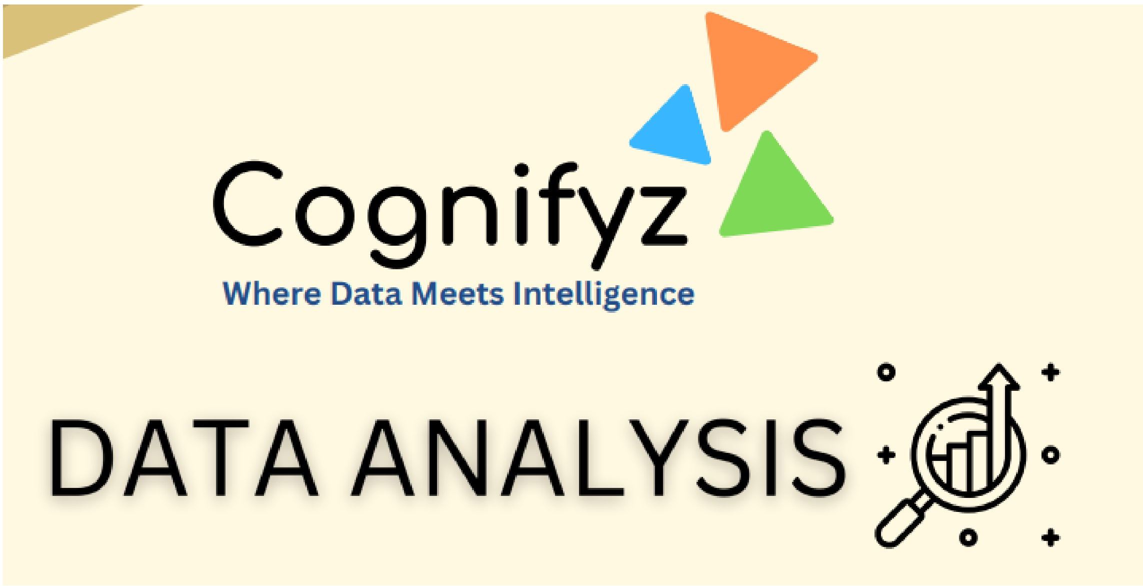


Task - Objective (Level-1)

- Expertise in Python programming and Data Manipulation
- Extract valuable insights from large datasets and drive informed decision-making.
- Data cleaning and preprocessing data, performing statistical analysis, or creating data visualizations,
- Proficiency in Python will play a crucial role in delivering meaningful results.

1. Load Python Modules

In [1]:

```
#  Use Python's import statement to load modules
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from tabulate import tabulate
```

2. Read the Dataset from CSV file - Using Pandas

In [2]:

```
# Read the csv file using pandas read_csv
restaurant_df=pd.read_csv("Dataset.csv")
restaurant_df
```

Out[2]:

	Restaurant ID	Restaurant Name	Country Code	City	Address	Locality	
0	6317637	Le Petit Souffle	162	Makati City	Third Floor, Century City Mall, Kalayaan Avenu...	Century City Mall, Poblacion, Makati City	C Mall,
1	6304287	Izakaya Kikufuji	162	Makati City	Little Tokyo, 2277 Chino Roces Avenue, Legaspi...	Little Tokyo, Legaspi Village, Makati City	I Lega Makat
2	6300002	Heat - Edsa Shangri-La	162	Mandaluyong City	Edsa Shangri-La, 1 Garden Way, Ortigas, Mandal...	Edsa Shangri-La, Ortigas, Mandaluyong City	Edsa M
3	6318506	Ooma	162	Mandaluyong City	Third Floor, Mega Fashion Hall, SM Megamall, O...	SM Megamall, Ortigas, Mandaluyong City	SM M Cit
4	6314302	Sambo Kojin	162	Mandaluyong City	Third Floor, Mega Atrium, SM Megamall, Ortigas...	SM Megamall, Ortigas, Mandaluyong City	SM M Cit
...	...	...	...	...	...	...	
9546	5915730	Namlı Gurme	208	��stanbul	Kemanke�� Karamustafa Pa��a Mahallesi, Rıhtım ...	Karak�_y	.
9547	5908749	Ceviz A��acı	208	��stanbul	Ko��uyolu Mahallesi, Muhittin ��st�_nda�� Cadd...	Ko��uyolu	K .
9548	5915807	Huqqa	208	��stanbul	Kuru�_e��me Mahallesi, Muallim Naci Caddesi, N...	Kuru�_e��me	Kuru� .
9549	5916112	A���k Kahve	208	��stanbul	Kuru�_e��me Mahallesi, Muallim Naci Caddesi, N...	Kuru�_e��me	Kuru� .
9550	5927402	Walter's Coffee Roastery	208	��stanbul	Cafea��a Mahallesi, Bademaltı Sokak, No 21/B, ...	Moda	.

9551 rows × 21 columns

3. Basic Inspection on given dataset

In [3]:

```python
def basic_inspection_dataset(table):
    """Generates a basic inspection dataset from the given table."""

    print("top 5 rows - using head")
    print(table.head())
    print()

    print("bottom 5 rows using tail")
    print(table.tail())
    print()

    print("numbers of samples and columns")
    print(table.shape)
    print()

    print("numbers of samples ")
    print(len(table))
    print()

    print("numbers of entries in the data frame")
    print(table.size)
    print()

    print("Columns Names")
    print(table.columns)
    print()

    print("Columns dtypes")
    print(table.dtypes)
    print()

    print("Dataframe info")
    print(table.info())
    print()

    print()
    print("check the missing value in each column")
    print(table.isnull().sum())

    print()
    print("check the missing value in each column")
    print(table.isna().sum())

basic_inspection_dataset(restaurant_df)
```

```
top 5 rows - using head
   Restaurant ID         Restaurant Name  Country Code              City
\
0        6317637        Le Petit Souffle           162       Makati City
1        6304287        Izakaya Kikufuji           162       Makati City
2        6300002  Heat - Edsa Shangri-La           162  Mandaluyong City
3        6318506                    Ooma           162  Mandaluyong City
4        6314302             Sambo Kojin           162  Mandaluyong City

                                             Address  \
0  Third Floor, Century City Mall, Kalayaan Avenu...
1  Little Tokyo, 2277 Chino Roces Avenue, Legaspi...
2  Edsa Shangri-La, 1 Garden Way, Ortigas, Mandal...
3  Third Floor, Mega Fashion Hall, SM Megamall, O...
4  Third Floor, Mega Atrium, SM Megamall, Ortigas...

                                     Locality  \
0   Century City Mall, Poblacion, Makati City
1  Little Tokyo, Legaspi Village, Makati City
2  Edsa Shangri-La, Ortigas, Mandaluyong City
3      SM Megamall, Ortigas, Mandaluyong City
4      SM Megamall, Ortigas, Mandaluyong City

                                    Locality Verbose   Longitude   Latitud
e  \
0  Century City Mall, Poblacion, Makati City, Mak...  121.027535  14.56544
3
1  Little Tokyo, Legaspi Village, Makati City, Ma...  121.014101  14.55370
8
2  Edsa Shangri-La, Ortigas, Mandaluyong City, Ma...  121.056831  14.58140
4
3  SM Megamall, Ortigas, Mandaluyong City, Mandal...  121.056475  14.58531
8
4  SM Megamall, Ortigas, Mandaluyong City, Mandal...  121.057508  14.58445
0

                           Cuisines  ...          Currency Has Table booki
ng  \
0        French, Japanese, Desserts  ...  Botswana Pula(P)               Y
es
1                          Japanese  ...  Botswana Pula(P)               Y
es
2  Seafood, Asian, Filipino, Indian  ...  Botswana Pula(P)               Y
es
3                   Japanese, Sushi  ...  Botswana Pula(P)
No
4                  Japanese, Korean  ...  Botswana Pula(P)               Y
es

  Has Online delivery Is delivering now Switch to order menu Price range
\
0                  No                No                   No           3
1                  No                No                   No           3
2                  No                No                   No           4
3                  No                No                   No           4
4                  No                No                   No           4

   Aggregate rating  Rating color Rating text Votes
0               4.8    Dark Green   Excellent   314
1               4.5    Dark Green   Excellent   591
2               4.4         Green   Very Good   270
```

```
3               4.9    Dark Green   Excellent   365
4               4.8    Dark Green   Excellent   229

[5 rows x 21 columns]

bottom 5 rows using tail
      Restaurant ID           Restaurant Name  Country Code       City  \
9546        5915730               Naml۱ Gurme           208  ��stanbul
9547        5908749              Ceviz A��ac۱           208  ��stanbul
9548        5915807                     Huqqa           208  ��stanbul
9549        5916112               A����k Kahve           208  ��stanbul
9550        5927402  Walter's Coffee Roastery           208  ��stanbul

                                               Address     Locality  \
9546  Kemanke�� Karamustafa Pa��a Mahallesi, R۱ht۱m ...     Karak�_y
9547  Ko��uyolu Mahallesi, Muhittin ��st�_nda�� Cadd...    Ko��uyol
u
9548  Kuru�_e��me Mahallesi, Muallim Naci Caddesi, N...  Kuru�_e��me
9549  Kuru�_e��me Mahallesi, Muallim Naci Caddesi, N...  Kuru�_e��me
9550  Cafea��a Mahallesi, Bademalt۱ Sokak, No 21/B, ...         Moda

            Locality Verbose  Longitude   Latitude  \
9546     Karak�_y, ��stanbul  28.977392  41.022793
9547    Ko��uyolu, ��stanbul  29.041297  41.009847
9548  Kuru�_e��me, ��stanbul  29.034640  41.055817
9549  Kuru�_e��me, ��stanbul  29.036019  41.057979
9550          Moda, ��stanbul  29.026016  40.984776

                              Cuisines  ...          Currency  \
9546                           Turkish  ...  Turkish Lira(TL)
9547  World Cuisine, Patisserie, Cafe  ...  Turkish Lira(TL)
9548           Italian, World Cuisine  ...  Turkish Lira(TL)
9549                  Restaurant Cafe  ...  Turkish Lira(TL)
9550                             Cafe  ...  Turkish Lira(TL)

     Has Table booking Has Online delivery Is delivering now  \
9546                No                  No                No
9547                No                  No                No
9548                No                  No                No
9549                No                  No                No
9550                No                  No                No

     Switch to order menu Price range  Aggregate rating  Rating color  \
9546                   No           3               4.1         Green
9547                   No           3               4.2         Green
9548                   No           4               3.7        Yellow
9549                   No           4               4.0         Green
9550                   No           2               4.0         Green

     Rating text Votes
9546   Very Good   788
9547   Very Good  1034
9548        Good   661
9549   Very Good   901
9550   Very Good   591

[5 rows x 21 columns]

numbers of samples and columns
(9551, 21)
```

```
numbers of samples
9551

numbers of entries in the data frame
200571

Columns Names
Index(['Restaurant ID', 'Restaurant Name', 'Country Code', 'City', 'Addres
s',
       'Locality', 'Locality Verbose', 'Longitude', 'Latitude', 'Cuisine
s',
       'Average Cost for two', 'Currency', 'Has Table booking',
       'Has Online delivery', 'Is delivering now', 'Switch to order menu',
       'Price range', 'Aggregate rating', 'Rating color', 'Rating text',
       'Votes'],
      dtype='object')

Columns dtypes
Restaurant ID             int64
Restaurant Name          object
Country Code              int64
City                     object
Address                  object
Locality                 object
Locality Verbose         object
Longitude               float64
Latitude                float64
Cuisines                 object
Average Cost for two      int64
Currency                 object
Has Table booking        object
Has Online delivery      object
Is delivering now        object
Switch to order menu     object
Price range               int64
Aggregate rating        float64
Rating color             object
Rating text              object
Votes                     int64
dtype: object

Dataframe 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
None


check the missing value in each column
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

check the missing value in each column
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
```

4. Handling Missing Values

In [4]:

```
#For a categorical variable, determine the most frequent value, known as the mode.
cuisine_mode = restaurant_df['Cuisines'].mode()[0]
print(cuisine_mode)

# fill the missing value with mode
restaurant_df['Cuisines'].fillna(cuisine_mode,inplace=True)

# check for missing values - for confirmation
restaurant_df.isnull().sum()
```

```
North Indian
```

Out[4]:

```
Restaurant ID           0
Restaurant Name         0
Country Code            0
City                    0
Address                 0
Locality                0
Locality Verbose        0
Longitude               0
Latitude                0
Cuisines                0
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
```

Level 1, Task 1: Top Cuisines

1.1.1 Determine the top three most common cuisines in Dataset.

In [5]:

```python
def frequency_table_variable(cat_name,count):
    """Generate a frequency table for the categorical variable 'cat_name' with  top 'count' occurrences."""
    value_counts = restaurant_df[cat_name].value_counts().reset_index().head(count)
    # Give names to the columns
    value_counts.columns = ['Cuisines', 'Frequency']

    # Print the result as a table
    print(tabulate(value_counts, headers='keys', tablefmt='pretty'))
    print()

    print(value_counts["Cuisines"])
```

In [6]:

```python
print("Top Three most common cuisines in Dataset:")
frequency_table_variable("Cuisines",3)
```

```
Top Three most common cuisines in Dataset:
+---+-----------------------+-----------+
|   |       Cuisines        | Frequency |
+---+-----------------------+-----------+
| 0 |     North Indian      |    945    |
| 1 | North Indian, Chinese |    511    |
| 2 |        Chinese        |    354    |
+---+-----------------------+-----------+

0             North Indian
1    North Indian, Chinese
2                  Chinese
Name: Cuisines, dtype: object
```

observations

- Top Three most common cuisines in Dataset
    1. North Indian
    2. North Indian, Chinese
    3. Chinese

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

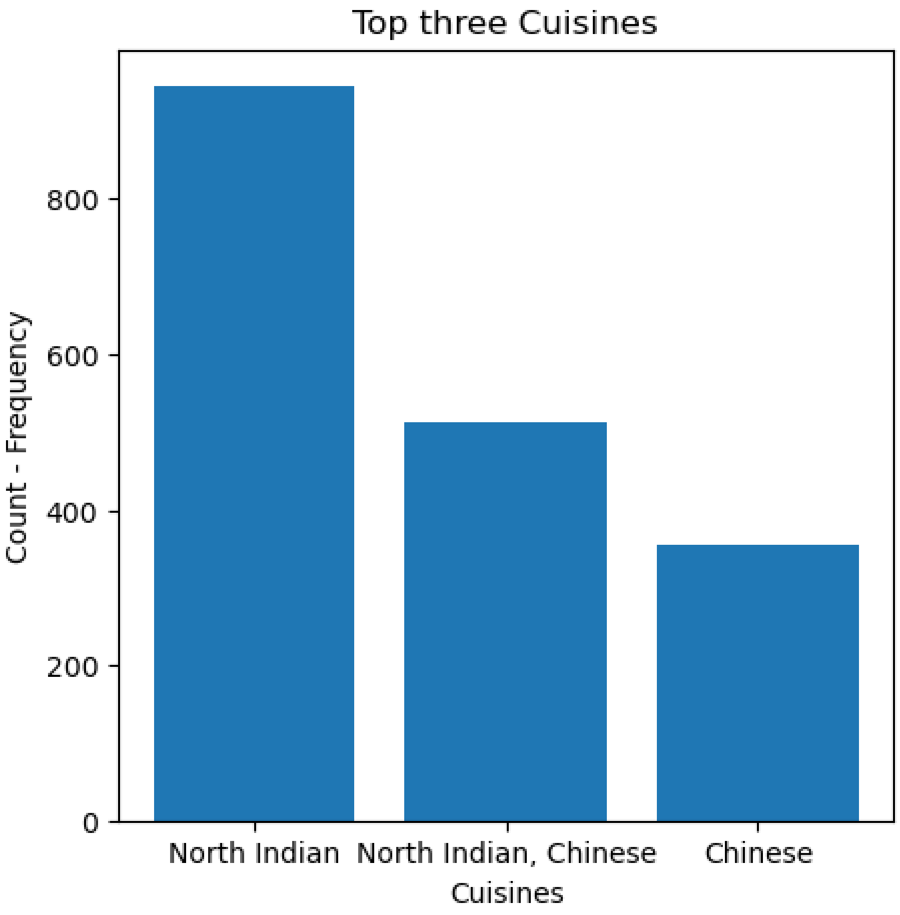
In [7]:

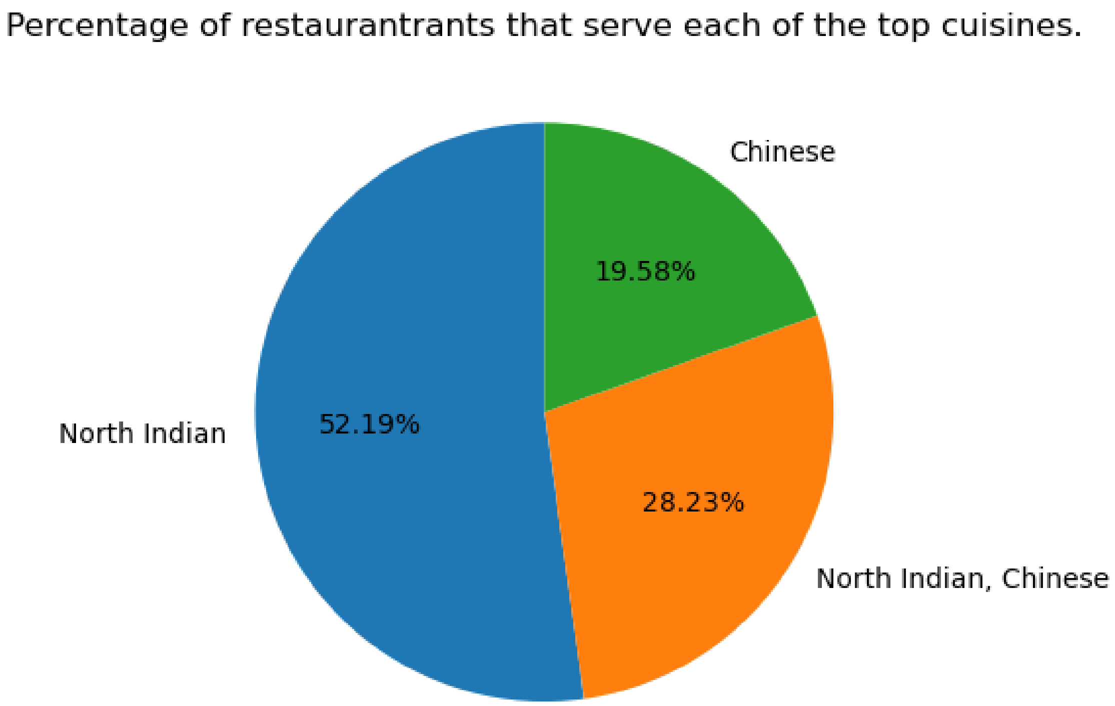
```python
def calculate_relative_frequency(city,count):
    """Calculates the relative frequency of a certain 'city' based on the top 'count' of occurrences."""
    value_counts = restaurant_df[city].value_counts().reset_index().head(count)
    # Rename the columns
    value_counts.columns = ['Cuisines', 'Count']
    value_counts['Percentage'] = round((value_counts['Count'] / len(restaurant_df))*100,2)

    # Print the result as a table
    print(tabulate(value_counts, headers='keys', tablefmt='pretty'))

    plt.figure(figsize=(5,5))
    values = value_counts.Cuisines
    labels = value_counts.Count
    plt.bar(values,labels)
    plt.title('Top three Cuisines')
    plt.xlabel('Cuisines')
    plt.ylabel("Count - Frequency")
    plt.show()

    plt.title('Percentage of restaurantrants that serve each of the top cuisines.')
    plt.pie(value_counts['Percentage'],labels=value_counts['Cuisines'],autopct='%0.2f%%',startangle=90)
    plt.show()
```

In [8]:

```python
calculate_relative_frequency("Cuisines",3)
```

```
+---+----------------------+-------+------------+
|   |       Cuisines       | Count | Percentage |
+---+----------------------+-------+------------+
| 0 |     North Indian     |  945  |    9.89    |
| 1 | North Indian, Chinese |  511  |    5.35    |
| 2 |       Chinese        |  354  |    3.71    |
+---+----------------------+-------+------------+
```





observations

- Percentage of Restaurants that serve each of the top cuisines
  1. North Indian - 52.19%
  2. North Indian, Chinese - 28.23%
  3. Chinese - 19.58%

Level 1, Task 2 : City Analysis

1.2.1 Identify the city with the highest number of restaurants in the dataset

In [9]:

```
print("City with the highest number of restaurants in the dataset.")
frequency_table_variable("City",1)
```

```
City with the highest number of restaurants in the dataset.
+---+-----------+-----------+
|   | Cuisines  | Frequency |
+---+-----------+-----------+
| 0 | New Delhi |   5473    |
+---+-----------+-----------+

0    New Delhi
Name: Cuisines, dtype: object
```

Observations

- City with the highest number of Restaurants in the dataset
  1. New Delhi

1.2.2 Calculate the average rating for restaurants in each city.

In [10]:

```python
def calculate_avg_rating_restaurant_by_city(city, rating):
    """Calculates the average rating of restaurants in the given 'city' based on the provided 'rating' data."""
    avg_ratings_by_city = restaurant_df.groupby(city)[rating].mean().reset_index()
    print(avg_ratings_by_city)
```

In [11]:

```python
print("Calculate the average rating for restaurants in each city")
calculate_avg_rating_restaurant_by_city("City","Aggregate rating")
```

```
Calculate the average rating for restaurants in each city
                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          ��stanbul          4.292857

[141 rows x 2 columns]
```

1.2.3 Determine the city with the highest average rating

In [12]:

```python
def calculate_highest_avg_rating(city,rating):
    """Calculates the highest average rating among restaurants in the specified 'city' using the provided 'rating' data."""
    avg_ratings = restaurant_df.groupby(city)[rating].mean().reset_index()
    avg_ratings = avg_ratings.sort_values(by=rating,ascending=False)
    print("City with the highest average rating")
    print(avg_ratings.head(1))
```

In [13]:

```python
calculate_highest_avg_rating("City","Aggregate rating")
```

```
City with the highest average rating
          City  Aggregate rating
56  Inner City               4.9
```

Observations

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

Level 1, Task 3 : Price Range Distribution

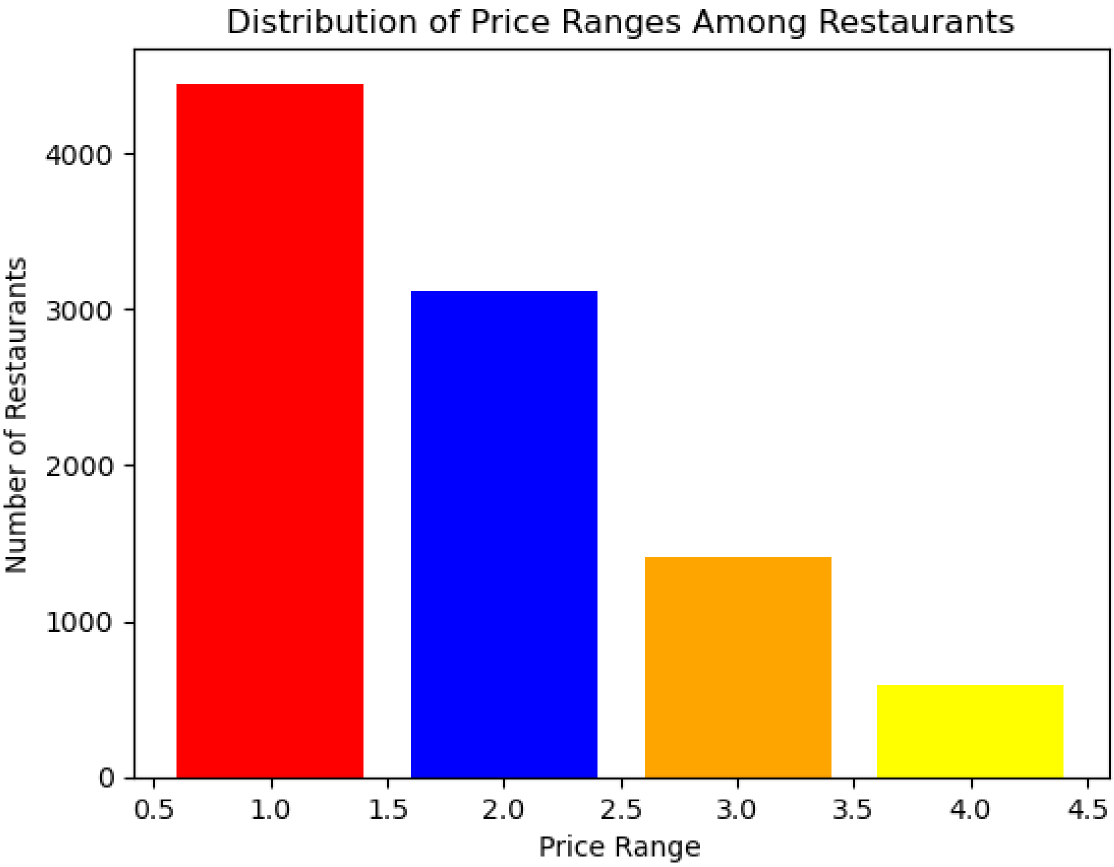
1.3.1 Create a histogram or bar chart to visualize the distribution of price ranges among the restaurants.

In [14]:

```
def draw_bar_plot(price):
    """Draws a bar plot based on the 'price' data."""
    price_counts = restaurant_df[price].value_counts()
    plt.bar(price_counts.index, price_counts.values, color=['red', 'blue', 'orange','yellow'])
    plt.xlabel('Price Range')
    plt.ylabel('Number of Restaurants')
    plt.title('Distribution of Price Ranges Among Restaurants')
    plt.show()
```

In [15]:

```
draw_bar_plot('Price range')
```



Observations

- Distribution of price ranges among the restaurants
  1. 1
  2. 2
  3. 3
  4. 4

1.3.2 Calculate the percentage of restaurants in each price range category.

In [16]:

```
def cal_per_restaurant_price_range(price):
    """Calculates the price range percentage restaurant based on the given 'price' data."""
    value_counts = restaurant_df[price].value_counts().reset_index()

    # Rename the columns
    value_counts.columns = ['Price-Range', 'Count']
    total_count = value_counts['Count'].sum()

    value_counts['Percentage'] = round((value_counts['Count'] / total_count)*100,2)

    # Print the result as a table
    print(tabulate(value_counts, headers='keys', tablefmt='pretty'))
```

In [17]:

```
cal_per_restaurant_price_range("Price range")
```

```
+---+-------------+--------+------------+
|   | Price-Range | Count  | Percentage |
+---+-------------+--------+------------+
| 0 |     1.0     | 4444.0 |   46.53    |
| 1 |     2.0     | 3113.0 |   32.59    |
| 2 |     3.0     | 1408.0 |   14.74    |
| 3 |     4.0     | 586.0  |    6.14    |
+---+-------------+--------+------------+
```

Observations

- Percentage of restaurants in each price range category.
    1. Price Range :1 Percantage : 46.53%
    2. Price Range :2 Percentage: 32.59%
    3. Price Range :3 Percentage: 14.74%
    4. Price Range: 4 Percentage: 6.14%

Level 1 , Task 4: Online Delivery

1.4.1 Determine the percentage of restaurants that offer online delivery

In [18]:

```python
def per_online_delivery_restaurants(online_delivery):
    """Calculates the percentage of restaurants offering online delivery."""
    total_restaurant_count =restaurant_df.shape[0]
    online_restaurnat_count = restaurant_df[restaurant_df[online_delivery]=='Yes']
    online_restaurant_count= online_restaurnat_count.shape[0]
    # percentage of online order taken by the restaurants
 percentage=round((online_restaurant_count/total_restaurant_count)*100,2)
    print(percentage)
```

In [19]:

```python
print("percentage of online order taken by the restaurants")
per_online_delivery_restaurants('Has Online delivery')
```

```
percentage of online order taken by the restaurants
25.66
```

Observations

- Percentage of restaurants that offer online delivery - 25.66%

1.4.2 Compare the average ratings of restaurants with and without online delivery.

In [20]:

```python
def avg_ratings_by_restaurant_with_without_online_delivery(online_delivery,rating):
    # average rating of restaurant with and without online delivery
    print(restaurant_df.groupby(online_delivery)[rating].mean().round(2).reset_index())

```

In [21]:

```python
print("average rating of restaurant with and without online delivery")
avg_ratings_by_restaurant_with_without_online_delivery('Has Online delivery','Aggregate rating')
```

```
average rating of restaurant with and without online delivery
  Has Online delivery  Aggregate rating
0                  No              2.47
1                 Yes              3.25
```

Observations

- average ratings of restaurants with and without online delivery

1. No - Online Delivery - Avg Rating - 2.47
2. Online Delivery - Avg Rating - 3.25