PROJECT OVERVIEW
RESTAURANT DATA ANALYSIS PROJECT

Restaurant datasets encompass various attributes such as names, locations, cuisine types, ratings, review counts, price ranges, and operating hours. They are sourced from online review platforms, food delivery apps, and restaurant websites. Analyses can include descriptive summaries, sentiment analysis, predictive modeling, and geospatial mapping. These insights help restaurant owners improve services, understand customer preferences, and conduct market research. Common challenges involve ensuring data quality, maintaining privacy, and integrating diverse data sources effectively.

```
#importing necessary libraries for data analysis and visualization
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

```
# Loading the restaurant dataset into a dataframe
dataset = pd.read_csv('/content/Dataset  (1).csv')
```

```
# Displaying the first few rows of the dataset to understand the structure
dataset.head()
```

	Restaurant ID	Restaurant Name	Country Code	City	Address	Locality	Locality Verbose	Longitude	Latitude	Cuisines	...	Currency	Tal book
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	14.565443	French, Japanese, Desserts	...	Botswana Pula(P)	
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	14.553708	Japanese	...	Botswana Pula(P)	
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	14.581404	Seafood, Asian, Filipino, Indian	...	Botswana Pula(P)	
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	14.585318	Japanese, Sushi	...	Botswana Pula(P)	
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	14.584450	Japanese, Korean	...	Botswana Pula(P)	

5 rows × 21 columns

```
# Displaying nubers of rows and columns in the dataset
dataset.shape
```

```
(9551, 21)
```

```
# Displaying summary of the datset
dataset.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
dataset.isnull().sum()
```

	0
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

```
# Data cleaning
# Handling missing values
dataset['Cuisines'].fillna('unknown',inplace =True)
```

LEVEL 1

TASK 1 - TOP CUISINES

Determine the top thre most common cuisines in the daraset

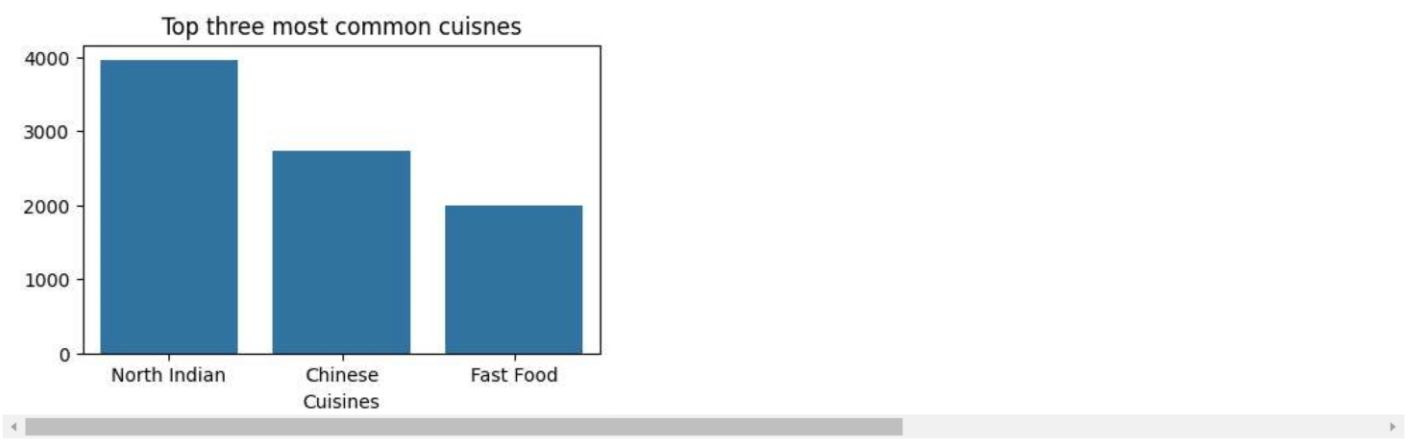
```
# Count the occurence of each cuisine
cuisines = dataset['Cuisines'].str.split(', ').explode().value_counts()

# Get the top 3 most common cuisine
top_cuisines = cuisines.head(3)

# print top 3 common cuisine
print("Top 3 most common cuisines are:")
print(top_cuisines)
```

```
Top 3 most common cuisines are:
Cuisines
North Indian    3960
Chinese         2735
Fast Food       1986
Name: count, dtype: int64
```

```
plt.figure(figsize=(5,3))
sns.barplot(x =top_cuisines.index , y= top_cuisines.values)
plt.title('Top three most common cuisnes')
plt.show()
```



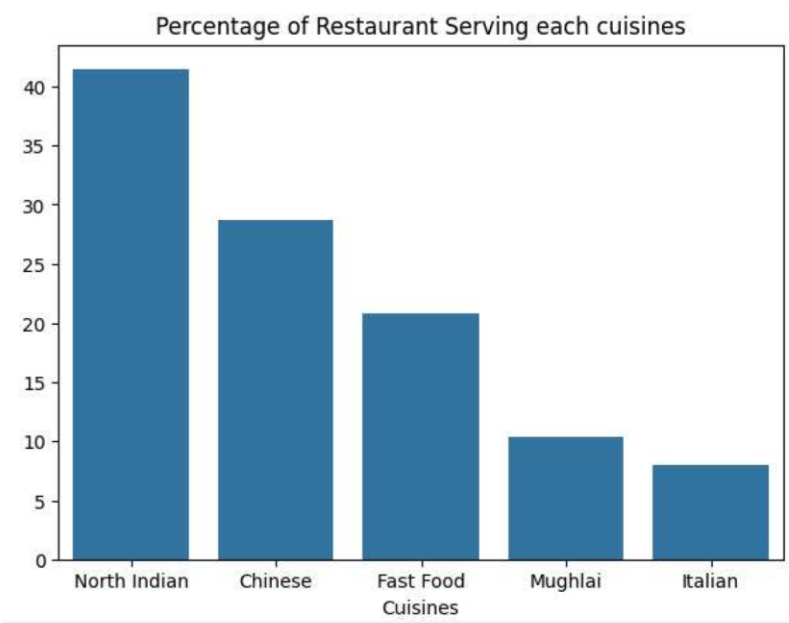
CALCULATE THE PERCENTAGE OF RESTAURANTS THAT SERVE EACH OF THE TOP CUISINES

```
# calculating the percentage of restaurant serving each cuisine
total_restaurant = len(dataset)
cuisines_percentage = (cuisines / total_restaurant) * 100

# print the percentage of restaurant serving each cuisine
print("Percentages of retaurant serving each cuisines:")
top_cuisines_percentages = cuisines_percentage.head(5)
print(top_cuisines_percentages)
```

```
Percentages of retaurant serving each cuisines:
Cuisines
North Indian    41.461627
Chinese         28.635745
Fast Food       20.793634
Mughlai         10.417757
Italian          7.999162
Name: count, dtype: float64
```

```
plt.figure(figsize=(7,5))
sns.barplot(x=top_cuisines_percentages.index,y=top_cuisines_percentages.values)
plt.title('Percentage of Restaurant Serving each cuisines')
plt.show()
```



TASK 2 - CITY ANALYSIS

Identify the city with the highest number of restauarnts in the dataset

```
city = dataset.groupby('City')['Restaurant ID'].count().sort_values(ascending = False)
print(city.head(1))
```

```
City
New Delhi    5473
Name: Restaurant ID, dtype: int64
```

CALCULATE THE AVERAGE RATING FOR RESTAURANT IN EACH CITY

```
average_rating = dataset.groupby('City')['Aggregate rating'].mean()
print(average_rating)
```

```
City
Abu Dhabi          4.300000
Agra               3.965000
Ahmedabad          4.161905
Albany             3.555000
Allahabad          3.395000
                       ...
Weirton            3.900000
Wellington City    4.250000
Winchester Bay     3.200000
Yorkton            3.300000
��stanbul          4.292857
Name: Aggregate rating, Length: 141, dtype: float64
```

DETERMINE THE CITY WITH THE HIGHEST AVERAGE RATING

```
average_rating = dataset.groupby('City')['Aggregate rating'].mean()
print(average_rating.head(1))
```

```
City
Abu Dhabi    4.3
Name: Aggregate rating, dtype: float64
```

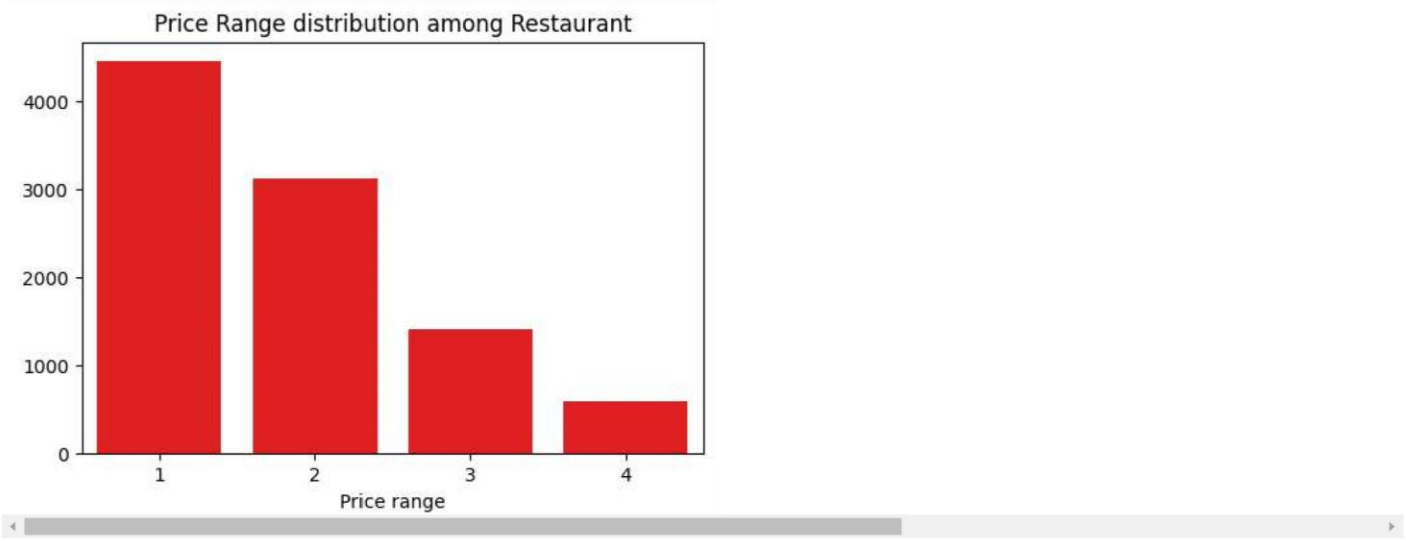
TASK 3 - PRICE RANGE DISTRIBUTION

Create a histogram or bar chart to visualize the distribution of price among the restaurant

```
price_range = dataset['Price range'].value_counts()
print(price_range)
```

```
Price range
1    4444
2    3113
3    1408
4     586
Name: count, dtype: int64
```

```
plt.figure(figsize = (6,4))
sns.barplot(x = price_range.index,y = price_range.values,color= 'red')
plt.title('Price Range distribution among Restaurant')
plt.show()
```

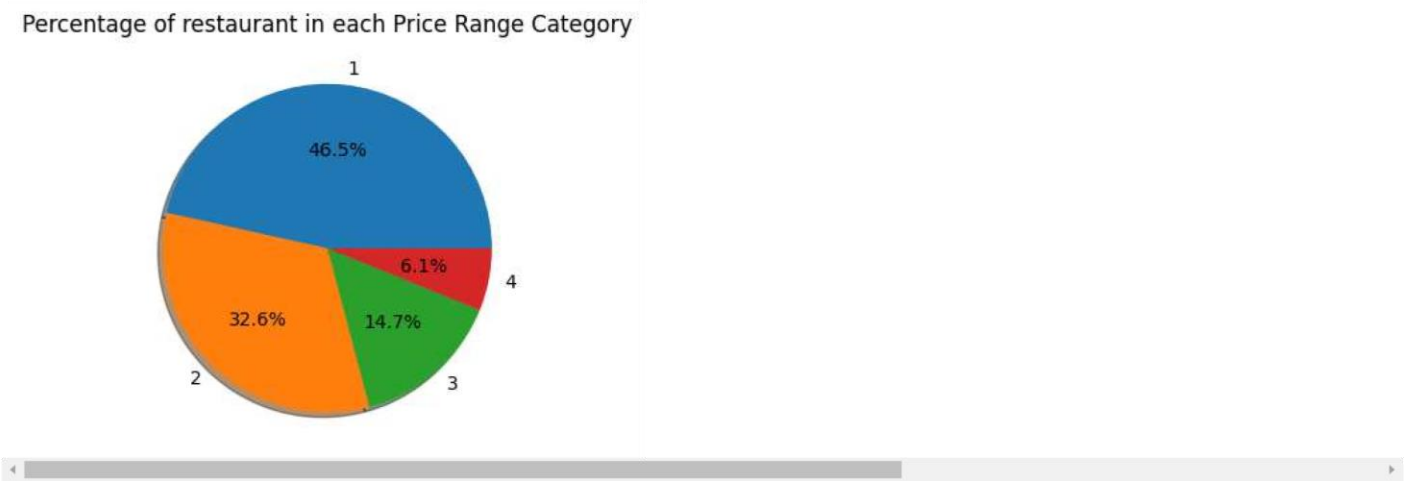


Calculate the percentage of restaurant in each price range category

```
price_range_percentage = (dataset['Price range'].value_counts() / len(dataset)) * 100
print(price_range_percentage)
```

```
Price range
1    46.529159
2    32.593446
3    14.741912
4     6.135483
Name: count, dtype: float64
```

```
plt.figure(figsize =(6,4))
plt.pie(price_range_percentage , labels = price_range_percentage.index ,autopct = '%1.1f%%' ,shadow = True)
plt.title('Percentage of restaurant in each Price Range Category')
plt.show()
```



TASK 4 - ONLINE DELIVERY

Determine the percentage of restaurant that offer online delivery

```
online_delivery = dataset[dataset['Has Online delivery'] == 'Yes']
count = len(online_delivery)
online_delivery_percentage = (count / len(dataset)) * 100
print(online_delivery_percentage)
```

```
25.662234321013504
```

Compare the average rating of restaurant with and without online delivery

```
compare_average_rating = dataset.groupby('Has Online delivery')['Aggregate rating'].mean()
print(compare_average_rating)
```

```
Has Online delivery
No     2.465296
Yes    3.248837
Name: Aggregate rating, dtype: float64
```

Start coding or generate with AI