Big Data - Case Study - 1

INDEX

Introduction

This project focuses on analyzing the *Swiggy food delivery dataset* using data analytics and visualization techniques. The aim is to understand restaurant distribution, food pricing, customer ratings, and delivery patterns. By performing data analysis in **MySQL** and visualizing results through **Python (Jupyter Notebook)**, the project highlights how data-driven insights can enhance business decision-making and customer satisfaction in the online food delivery industry.

Details of Dataset

The Swiggy dataset contains 518 records with 10 attributes:

- **ID:** Unique identifier for each restaurant
- **Area:** Locality where the restaurant operates
- **City:** Name of the city
- **Restaurant:** Name of the restaurant
- **Price:** Average meal price (in ₹)
- **Avg Ratings:** Average customer rating
- **Total Ratings:** Total number of customer reviews
- **Food Type:** Type of cuisine offered
- **Address:** Restaurant address
- **Delivery Time:** Average delivery duration (in minutes)

The dataset is clean, with no missing or duplicate values, making it ready for analysis.

Project Scope

The project aims to perform analytical operations on the Swiggy dataset to explore various factors influencing restaurant performance, customer satisfaction, and delivery efficiency. Using MySQL for structured data queries and Python visualization for insights, this project provides a complete data analysis workflow. It demonstrates how food delivery businesses can optimize pricing, improve delivery speed, and enhance overall customer experience through data analytics.

Goals

To perform structured data analysis using **MySQL**.

To visualize restaurant and customer data using **Python (Jupyter)**.

To identify trends in ratings, delivery times, and food pricing.

To derive insights for improving restaurant efficiency and customer satisfaction.

To demonstrate the role of data analytics in the online food delivery industry.

Tools and Working Environment

MySQL: For structured data storage and SQL-based analysis.

Python (Jupyter Notebook): For advanced data visualization using matplotlib and seaborn.

Pandas: For data manipulation and cleaning.

Matplotlib / Seaborn: For graphical representation of insights.

System Environment: Windows 10, Python 3.10+, MySQL 8.0

Performing Analysis On MYSQL

Create & Use Database

```
mysql> CREATE DATABASE swiggy_db;
Query OK, 1 row affected (0.01 sec)

mysql> USE swiggy_db;
Database changed
```

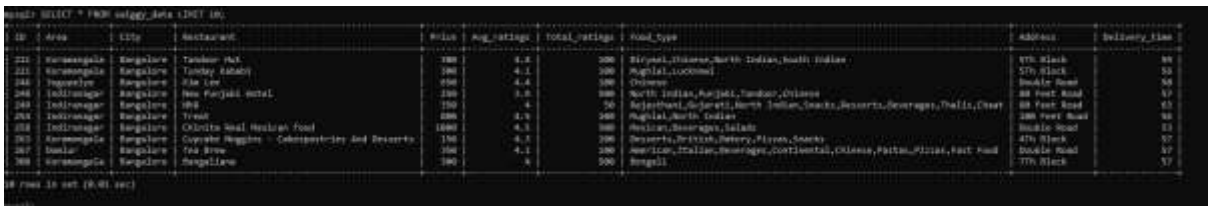
Create table

```
mysql> CREATE TABLE swiggy_data (
    ->     ID INT PRIMARY KEY,
    ->     Area VARCHAR(100),
    ->     City VARCHAR(100),
    ->     Restaurant VARCHAR(200),
    ->     Price INT,
    ->     Avg_ratings FLOAT,
    ->     Total_ratings INT,
    ->     Food_type VARCHAR(300),
    ->     Address VARCHAR(200),
    ->     Delivery_time INT
    -> );
Query OK, 0 rows affected (0.01 sec)
```

Load CSV File into MySQL

```
mysql> LOAD DATA LOCAL INFILE 'E:/da/swiggy dataset .csv'
    -> INTO TABLE swiggy_data
    -> FIELDS TERMINATED BY ','
    -> ENCLOSED BY '"'
    -> LINES TERMINATED BY '\n'
    -> IGNORE 1 ROWS;
Query OK, 518 rows affected (0.03 sec)
Records: 518  Deleted: 0  Skipped: 0  Warnings: 0
```

View First Few Records



Perform Analysis Queries

Total restaurants per city

```
mysql> SELECT City, COUNT(Restaurant) AS Total_Restaurants
    -> FROM swiggy_data
    -> GROUP BY City;
+-----------+-------------------+
| City      | Total_Restaurants |
+-----------+-------------------+
| Bangalore |                86 |
| Hyderabad |                69 |
| Mumbai    |                62 |
| Pune      |                75 |
| Kolkata   |               112 |
| Delhi     |                36 |
| Chennai   |                78 |
+-----------+-------------------+
7 rows in set (0.01 sec)
```

Average rating by area

```
mysql> SELECT Area, ROUND(AVG(Avg_ratings),2) AS Avg_Rating
    -> FROM swiggy_data
    -> GROUP BY Area
    -> ORDER BY Avg_Rating DESC;
+--------------------------------+------------+
| Area                           | Avg_Rating |
+--------------------------------+------------+
| Shankarapura                   |        4.5 |
| Nandanam                       |        4.5 |
| Central Markt Punjabi Bagh     |        4.5 |
| Pashan                         |        4.5 |
| College Square                 |        4.5 |
| Wadgaon Sheri                  |        4.5 |
| Beleghata                      |       4.45 |
| Jogupalya                      |        4.4 |
| Beniapukur                     |        4.4 |
| Bandra Area                    |        4.4 |
| Kotturpuram                    |        4.4 |
| Near Rupbani Cinema            |        4.4 |
| Royapettah                     |        4.4 |
| Chetpet                        |        4.4 |
| Shobhabazar                    |        4.4 |
| Basavanagudi                   |        4.4 |
| Camp                           |       4.35 |
| Kasba                          |       4.35 |
| East Kolkata Township          |       4.35 |
| Egmore                         |       4.35 |
| Machuabazar                    |       4.35 |
| Thousand Lights                |       4.34 |
| Anna Nagar                     |       4.34 |
| New Tippasandra                |       4.33 |
| Mylapore                       |       4.32 |
| Cooke Town                     |        4.3 |
| Kodihalli                      |        4.3 |
| Powai Area                     |        4.3 |
| Commercial Street              |        4.3 |
| Sion                           |        4.3 |
| Erandwane                      |        4.3 |
| Kalyani Nagar                  |        4.3 |
| Abids                          |        4.3 |
| Barabazar Market               |        4.3 |
| Jodhpur Park                   |        4.3 |
| Near 7 Point Crossing          |        4.3 |
| Jayamahal                      |        4.3 |
| Sarat Bose Rd                  |        4.3 |
| West Mambalam                  |        4.3 |
| Narhe                          |        4.3 |
| Shaniwar Peth                  |        4.3 |
| Ghatkopar West                 |        4.3 |
| Tilak Nagar                    |        4.3 |
| Bhowanipore                    |       4.27 |
| Ashok Nagar                    |       4.26 |
| Kothrud                        |       4.25 |
| Basheer Bagh                   |       4.25 |
| Teynampet                      |       4.25 |
| Indiranagar                    |       4.21 |
| T. Nagar                       |       4.21 |
| Punjagutta                     |        4.2 |
| Aundh                          |        4.2 |
| Kalasiguda                     |        4.2 |
| Golpark                        |        4.2 |
| Agarkar Nagar                  |        4.2 |
```

Restaurants taking more than 60 mins delivery time

```
mysql> SELECT Restaurant, Delivery_time, City
    -> FROM swiggy_data
    -> WHERE Delivery_time > 60;
```

Restaurant	Delivery_time	City
Nh8	63	Bangalore
So. The Sky Kitchen	90	Hyderabad
Chinese Pavilion	64	Hyderabad
Chef Inam'S Steak House	69	Hyderabad
Cafe Peterdonuts	71	Pune
Karolbaug	72	Pune
The Punjabi'S Kitchen	65	Mumbai
Aangan - Yatri Nivas	66	Hyderabad
Bluefox	68	Hyderabad
Taaareef	67	Pune
Sai Leela	63	Mumbai
Tandoori Darbar	71	Kolkata
Tero Parbon	72	Kolkata
Wise Owl The Coffee Shop	76	Kolkata
Aminia Restaurant- Golpark	67	Kolkata
Bedouin - Sher E Bengal	70	Kolkata
Tamarind	73	Kolkata
The Grub Club	69	Kolkata
Keshav Reddy Sweets	65	Hyderabad
Kimli	69	Kolkata
Oh So Stoned	68	Hyderabad
Hotel Swagath Grand - Dhanturi Group Of Hotels	67	Hyderabad
Hotel Sitara Grand - Dhanturi Group Of Hotels	70	Hyderabad
Mocambo	62	Kolkata
The Scoop	70	Kolkata
Kaafila	71	Kolkata
Le Coffee Creme	68	Kolkata
Sitara Grand - Nizami Pakwaan	62	Hyderabad
Carnival Restaurant & Bar	64	Pune
Chai - The Way You Like It	73	Pune
Bachan'S Dhaba	61	Kolkata
Mainland China	64	Kolkata
First Innings - The Stadel Hotel	69	Kolkata

Top 5 highest rated restaurants

```
mysql> SELECT Restaurant, Avg_ratings, City
    -> FROM swiggy_data
    -> ORDER BY Avg_ratings DESC
    -> LIMIT 5;
+---------------------------------+-------------+-----------+
| Restaurant                      | Avg_ratings | City      |
+---------------------------------+-------------+-----------+
| Theobroma                       |         4.7 | Mumbai    |
| Mama Mia! - Italian Ice Creams  |         4.7 | Kolkata   |
| Corner House Ice Cream          |         4.7 | Bangalore |
| The Brew Room                   |         4.7 | Chennai   |
| Fresh Baked Goodness            |         4.7 | Chennai   |
+---------------------------------+-------------+-----------+
5 rows in set (0.00 sec)
```

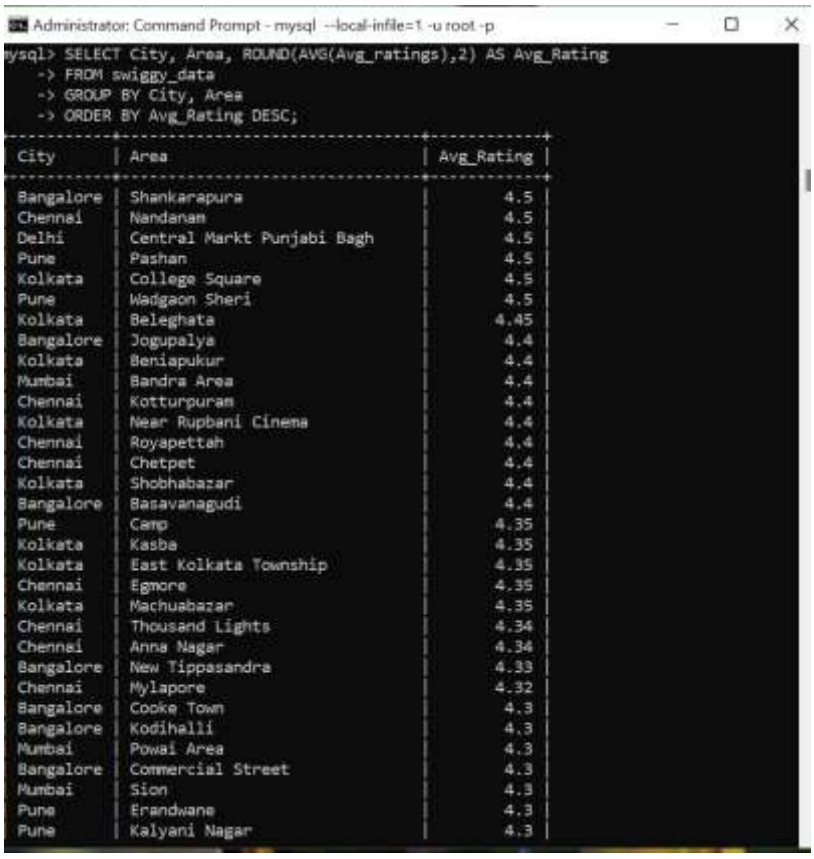
Average price by food type

```
mysql> SELECT Food_type, ROUND(AVG(Price),2) AS Avg_Price
    -> FROM swiggy_data
    -> GROUP BY Food_type
    -> ORDER BY Avg_Price DESC;
+-----------------------------------------------------+-----------+
| Food_type                                           | Avg_Price |
+-----------------------------------------------------+-----------+
| North Indian,Chinese,Biryani,Continental,Mughlai    |   1700.00 |
| North Indian,Biryani,Continental,Italian            |   1500.00 |
| Thai                                                |   1500.00 |
| North Indian,Chinese,Thai,Continental               |   1500.00 |
| Seafood                                             |   1500.00 |
| Continental,European,Salads,Italian                 |   1500.00 |
| American,European                                   |   1500.00 |
| Japanese                                            |   1500.00 |
| Japanese,Korean                                     |   1500.00 |
| Japanese,Asian                                      |   1400.00 |
| Thai,Pan-Asian                                      |   1400.00 |
| North Indian,Mexican,Continental,Italian            |   1300.00 |
| Chinese,Seafood,Thai,Japanese                       |   1300.00 |
| North Indian,Chinese,Italian,Pizzas,Thai            |   1300.00 |
```

Fastest average delivery time by city

```
mysql> SELECT City, ROUND(AVG(Delivery_time),2) AS Avg_Delivery
    -> FROM swiggy_data
    -> GROUP BY City
    -> ORDER BY Avg_Delivery ASC
    -> LIMIT 1;
+---------+--------------+
| City    | Avg_Delivery |
+---------+--------------+
| Mumbai  |        47.94 |
+---------+--------------+
1 row in set (0.00 sec)
```

Highest rated area per city



```
Administrator: Command Prompt - mysql  --local-infile=1 -u root -p
mysql> SELECT City, Area, ROUND(AVG(Avg_ratings),2) AS Avg_Rating
    -> FROM swiggy_data
    -> GROUP BY City, Area
    -> ORDER BY Avg_Rating DESC;
+-----------+------------------------------+------------+
| City      | Area                         | Avg_Rating |
+-----------+------------------------------+------------+
| Bangalore | Shankarapura                 |        4.5 |
| Chennai   | Nandanam                     |        4.5 |
| Delhi     | Central Markt Punjabi Bagh   |        4.5 |
| Pune      | Pashan                       |        4.5 |
| Kolkata   | College Square               |        4.5 |
| Pune      | Wadgaon Sheri                |        4.5 |
| Kolkata   | Beleghata                    |       4.45 |
| Bangalore | Jogupalya                    |        4.4 |
| Kolkata   | Beniapukur                   |        4.4 |
| Mumbai    | Bandra Area                  |        4.4 |
| Chennai   | Kotturpuram                  |        4.4 |
| Kolkata   | Near Rupbani Cinema          |        4.4 |
| Chennai   | Royapettah                   |        4.4 |
| Chennai   | Chetpet                      |        4.4 |
| Kolkata   | Shobhabazar                  |        4.4 |
| Bangalore | Basavanagudi                 |        4.4 |
| Pune      | Camp                         |       4.35 |
| Kolkata   | Kasba                        |       4.35 |
| Kolkata   | East Kolkata Township        |       4.35 |
| Chennai   | Egmore                       |       4.35 |
| Kolkata   | Machuabazar                  |       4.35 |
| Chennai   | Thousand Lights              |       4.34 |
| Chennai   | Anna Nagar                   |       4.34 |
| Bangalore | New Tippasandra              |       4.33 |
| Chennai   | Mylapore                     |       4.32 |
| Bangalore | Cooke Town                   |        4.3 |
| Bangalore | Kodihalli                    |        4.3 |
| Mumbai    | Powai Area                   |        4.3 |
| Bangalore | Commercial Street            |        4.3 |
| Mumbai    | Sion                         |        4.3 |
| Pune      | Erandwane                    |        4.3 |
| Pune      | Kalyani Nagar                |        4.3 |
```

Data Visulization

Import Libraries

In [11]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

Load the Dataset

In [12]:
```
df = pd.read_csv("E:/da/swiggy dataset .csv")
df.head()
```

Out[12]:

	ID	Area	City	Restaurant	Price	Avg ratings	Total ratings	F
0	211	Koramangala	Bangalore	Tandoor Hut	300	4.4	100	Biryani,Chin Indian,Sou
1	221	Koramangala	Bangalore	Tunday Kababi	300	4.1	100	Mughlai
2	246	Jogupalya	Bangalore	Kim Lee	650	4.4	100	
3	248	Indiranagar	Bangalore	New Punjabi Hotel	250	3.9	500	Indian,Punjabi,Tandoc
4	249	Indiranagar	Bangalore	Nh8	350	4.0	50	Rajasthani,Guja Indian,Snack

Check for Missing Values / Duplicates

In [13]:
```
print("Missing values:\n", df.isnull().sum())
print("Duplicate rows:", df.duplicated().sum())
```

```
Missing values:
 ID              0
Area             0
City             0
Restaurant       0
Price            0
Avg ratings      0
Total ratings    0
Food type        0
Address          0
Delivery time    0
dtype: int64
Duplicate rows: 0
```

Basic Info

In [14]:
```
df.info()
df.describe()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 518 entries, 0 to 517
Data columns (total 10 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   ID             518 non-null    int64
 1   Area           518 non-null    object
 2   City           518 non-null    object
 3   Restaurant     518 non-null    object
 4   Price          518 non-null    int64
 5   Avg ratings    518 non-null    float64
 6   Total ratings  518 non-null    int64
 7   Food type      518 non-null    object
 8   Address        518 non-null    object
 9   Delivery time  518 non-null    int64
dtypes: float64(1), int64(4), object(5)
memory usage: 40.6+ KB
```

Out[14]:

	ID	Price	Avg ratings	Total ratings	Delivery time
count	518.000000	518.000000	518.00000	518.000000	518.000000
mean	10795.484556	520.550193	4.06834	349.015444	54.857143
std	6088.851677	315.055324	0.40500	880.899617	13.317825
min	211.000000	100.000000	2.20000	20.000000	24.000000
25%	5315.250000	300.000000	3.92500	80.000000	46.000000
50%	10256.500000	450.000000	4.20000	100.000000	56.000000
75%	16492.000000	600.000000	4.30000	500.000000	65.000000
max	21032.000000	1700.000000	4.70000	10000.000000	90.000000

Total Restaurants per City

In [15]:

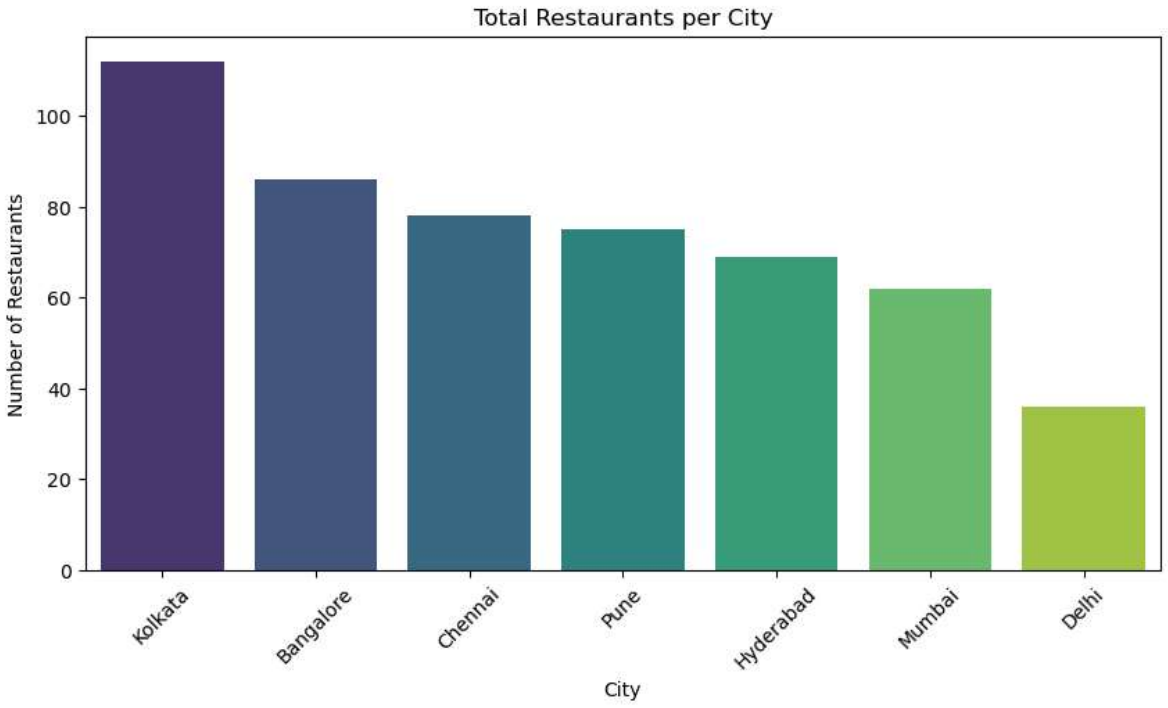
```
city_count = df['City'].value_counts()

plt.figure(figsize=(10,5))
sns.barplot(x=city_count.index, y=city_count.values, palette="viridis")
plt.title("Total Restaurants per City")
plt.xlabel("City")
plt.ylabel("Number of Restaurants")
plt.xticks(rotation=45)
plt.show()
```

```
C:\Users\himan\AppData\Local\Temp\ipykernel_9904\2133847922.py:4: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v
0.14.0. Assign the `x` variable to `hue` and set `legend=False` for the same effe
ct.

  sns.barplot(x=city_count.index, y=city_count.values, palette="viridis")
```



Average Ratings by Area

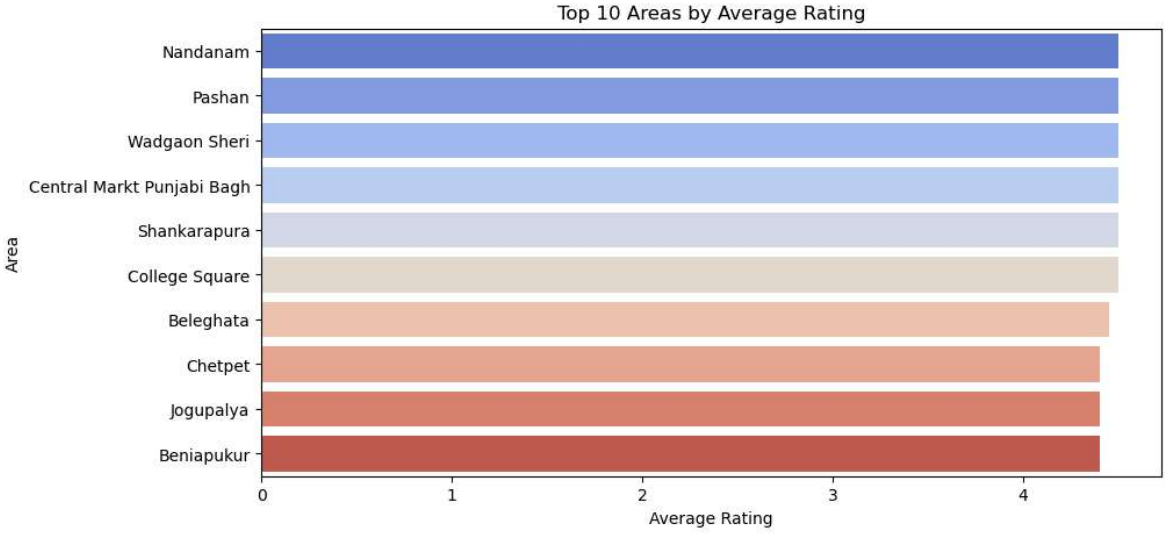
In [16]:
```
avg_rating_area = df.groupby('Area')['Avg ratings'].mean().sort_values(ascending

plt.figure(figsize=(10,5))
sns.barplot(x=avg_rating_area.values, y=avg_rating_area.index, palette="coolwarm
plt.title("Top 10 Areas by Average Rating")
plt.xlabel("Average Rating")
plt.ylabel("Area")
plt.show()
```

```
C:\Users\himan\AppData\Local\Temp\ipykernel_9904\582138821.py:4: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v
0.14.0. Assign the `y` variable to `hue` and set `legend=False` for the same effe
ct.

  sns.barplot(x=avg_rating_area.values, y=avg_rating_area.index, palette="coolwar
m")
```



Delivery Time Distribution

In [17]:
```
plt.figure(figsize=(8,5))
sns.histplot(df['Delivery time'], bins=20, kde=True, color="orange")
plt.title("Distribution of Delivery Times")
plt.xlabel("Delivery Time (minutes)")
plt.ylabel("Number of Restaurants")
plt.show()
```

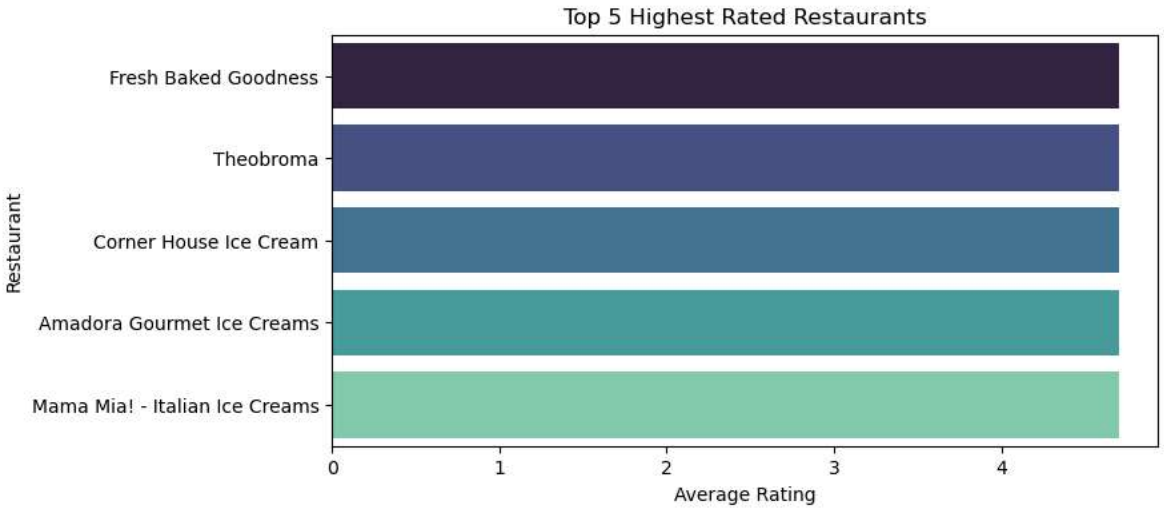
Top 5 Highest Rated Restaurants

In [18]:
```
top_restaurants = df.sort_values(by='Avg ratings', ascending=False).head(5)
plt.figure(figsize=(8,4))
sns.barplot(x='Avg ratings', y='Restaurant', data=top_restaurants, palette="makc
plt.title("Top 5 Highest Rated Restaurants")
plt.xlabel("Average Rating")
plt.ylabel("Restaurant")
plt.show()
```

```
C:\Users\himan\AppData\Local\Temp\ipykernel_9904\571489975.py:3: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v
0.14.0. Assign the `y` variable to `hue` and set `legend=False` for the same effe
ct.

  sns.barplot(x='Avg ratings', y='Restaurant', data=top_restaurants, palette="mak
o")
```



Average Price by Food Type

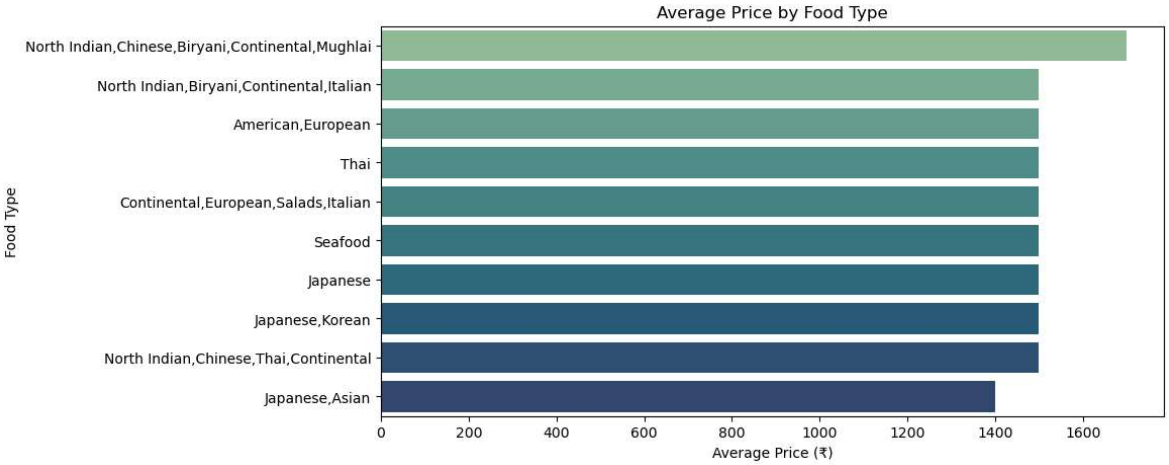
In [19]:
```
avg_price_food = df.groupby('Food type')['Price'].mean().sort_values(ascending=F

plt.figure(figsize=(10,5))
sns.barplot(x=avg_price_food.values, y=avg_price_food.index, palette="crest")
plt.title("Average Price by Food Type")
plt.xlabel("Average Price (₹)")
plt.ylabel("Food Type")
plt.show()
```

```
C:\Users\himan\AppData\Local\Temp\ipykernel_9904\3790784649.py:4: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v
0.14.0. Assign the `y` variable to `hue` and set `legend=False` for the same effe
ct.

  sns.barplot(x=avg_price_food.values, y=avg_price_food.index, palette="crest")
```



City-wise Average Delivery Time

In [20]:
```
avg_delivery_city = df.groupby('City')['Delivery time'].mean().sort_values()

plt.figure(figsize=(10,5))
sns.barplot(x=avg_delivery_city.index, y=avg_delivery_city.values, palette="flar
plt.title("Average Delivery Time by City")
plt.xlabel("City")
plt.ylabel("Average Delivery Time (minutes)")
```

```
plt.xticks(rotation=45)
plt.show()
```

```
C:\Users\himan\AppData\Local\Temp\ipykernel_9904\3574617190.py:4: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v
0.14.0. Assign the `x` variable to `hue` and set `legend=False` for the same effe
ct.

  sns.barplot(x=avg_delivery_city.index, y=avg_delivery_city.values, palette="fla
re")
```