**📊 Food Orders Analytics using SQL**

**🎯 Objective**

The main objective of this project is to **analyze food delivery data** to uncover insights about:

* Customer behavior and preferences
* Restaurant performance and ratings
* Cuisine popularity across demographics
* City-level and time-based sales trends

The analysis helps in **data-driven decision making** for food delivery businesses and restaurants.

**📂 Dataset Overview**

The project is built using **four relational tables**:

1. **Orders**
   * Contains transaction details:
     + order\_id, order\_date, sales\_qty, currency, user\_id, restaurant\_id, quarter, month\_name, year.
   * Helps track sales performance across **time periods** (monthly, quarterly, yearly).
2. **Users**
   * Customer information:
     + user\_id, name, age, gender, marital\_status, occupation.
   * Enables segmentation based on **age, gender, occupation, and marital status**.
3. **Restaurant**
   * Restaurant details:
     + id, name, city, country, rating, rating\_count, cuisine.
   * Used to evaluate **performance, ratings, and cuisine diversity**.
4. **Menu**
   * Menu information:
     + menu\_id, restaurant\_id, food\_id, cuisine, price.
   * Provides **pricing and cuisine category** data for revenue calculation.

**⚙️ Steps Involved**

**🔎 1. Data Exploration**

* Checked the number of **orders, users, restaurants, cuisines, and cities**.
* Verified consistency across primary and foreign keys (user\_id, restaurant\_id).
* Sample query:
* SELECT COUNT(DISTINCT user\_id) AS total\_users FROM users;

**🧹 2. Data Cleaning**

* Identified missing/null values in all tables.
* Replaced them with standard placeholders:
  + cuisine = 'not\_specified'
  + rating\_count = 'no\_rating\_count'
  + name = 'no\_users\_specified'
* Ensured **referential integrity** between orders, users, menu, and restaurant.

**📊 3. Business Analysis (20+ Queries)**

Some key questions answered:

1. **Top 5 restaurants** by revenue in the last year
2. **Top 10 users** with the highest purchases
3. **City-wise revenue** by year and quarter
4. **Bottom 3 restaurants** by performance
5. **Top cuisines** by order count and rating
6. **Age-wise popular cuisines**
7. **Occupation-wise order distribution**
8. **Year-over-year revenue comparison**
9. **High demand cuisines** in the last 2 months
10. **Top-rated vs. lowest-rated cuisines**

**🛠️ Techniques Used**

* **SQL Joins**:
  + Combined data across multiple tables (users, orders, restaurant, menu).
* **Aggregations**:
  + Calculated total revenue, average ratings, and order counts using SUM, COUNT, AVG.
* **Window Functions**:
  + Used RANK() OVER (PARTITION BY ...) to find top restaurants, users, and cuisines.
* **Grouping & Filtering**:
  + Analyzed order trends with GROUP BY, filtered with HAVING.
* **Time-Based Analysis**:
  + Leveraged year, month\_name, and quarter columns for **trend analysis**.

**📈 Insights**

**🔹 Customer Behavior**

* **Occupation matters**: Professionals and students ordered more compared to homemakers.
* **Age groups**: Young adults (18–30) preferred fast food, while older groups preferred traditional cuisines.

**🔹 Restaurant Performance**

* The **top 5 restaurants** contributed significantly to yearly revenue.
* The **bottom 3 restaurants** had consistently low sales despite being in high-demand cities, hinting at poor ratings or limited cuisine options.

**🔹 City & Cuisine Trends**

* Certain **metro cities dominate revenue** due to higher population density and demand.
* Seasonal cuisines (e.g., winter specials) showed spikes in the last 2 months of the year.
* A few **cuisines consistently performed better**, regardless of city or age group.

**🔹 Revenue Patterns**

* Year-over-year growth was visible in most cities.
* Peak demand was seen in **Q4 (October–December)** due to festive seasons.

**💡 Business Impact**

The analysis provides actionable recommendations:

* Focus marketing on **top cities and occupations**.
* Expand the menu in underperforming restaurants to include **high-demand cuisines**.
* Provide **seasonal promotions** during peak months.
* Encourage customer retention by offering **loyalty rewards to top users**.

**🛠️ Tools & Environment**

* **Database**: PostgreSQL
* **Client Tool**: pgAdmin
* **SQL Features**: Joins, Aggregates, Window Functions, Ranking, Grouping