Food Delivery Platform Data Analysis: A SQL Case Study Report

Project Goal: This project aims to extract meaningful insights from a food delivery platform's operational data using SQL queries. The analysis covers various aspects of customer behavior, restaurant performance, and sales trends.

Database Schema (Assumed): The analysis is based on an assumed database schema with the following key tables:

- **`users`:** Contains user information (e.g., `user_id`, `name`).
- **`orders`:** Stores order details (e.g., `order_id`, `user_id`, `r_id`, `date`, `amount`).
- **`restaurants`:** Contains restaurant information (e.g., `r_id`, `r_name`).
- **`menu`:** Lists menu items with prices (e.g., `m_id`, `f_id`, `price`, `r_id`).
- **`food`:** Contains food item details (e.g., `f_id`, `f_name`).
- **`order_details`:** Links orders to specific food items (e.g., `order_id`, `f_id`).

Key Findings and Analysis:

1. **Customers Who Have Never Ordered:**
    a. **Insight:** Identified users who have registered but never placed an order. This information is crucial for targeted marketing campaigns to re-engage dormant users.
    b. **Query Logic:** Utilized a **`NOT IN`** subquery to filter users whose `user_id` does not appear in the `orders` table.
2. **Average Price Per Dish:**
    a. **Insight:** Calculated the average price for each food item across all restaurants. This helps in understanding pricing strategies and identifying potential outliers.
    b. **Query Logic: Joined** `menu` and `food` tables, then used **`AVG()`** aggregate function **grouped by** food name.
3. **Top Restaurant by Number of Orders for a Given Month:**
    a. **Insight:** Determined the restaurant with the highest number of orders in a specific month (e.g., June). This highlights top-performing restaurants that could be featured or analyzed for success factors.

b. **Query Logic: Joined** `orders` and `restaurants` tables, filtered by month, **grouped by** restaurant, and **ordered by** count in **descending order** with a **`LIMIT`** 1.

4. **Restaurants with Monthly Sales Exceeding a Threshold:**
    a. **Insight:** Identified restaurants that achieved a sales revenue greater than a specified amount (e.g., 500) in a particular month (e.g., June 2025). This is vital for performance evaluation and identifying high-revenue partners.
    b. **Query Logic: Joined** `orders` and `restaurants` tables, filtered by month and year **(`DATE_TRUNC`), grouped by** restaurant, and used a **`HAVING` clause** to filter by total sales.
5. **Order Details for a Specific Customer within a Date Range:**
    a. **Insight:** Retrieved comprehensive details for all orders placed by a particular customer (e.g., 'Nitish') within a defined date range (e.g., May 10, 2022, to June 10, 2022). This is useful for customer service, order history review, and personalized recommendations.
    b. **Query Logic:** Involved **multiple joins** (`orders`, `restaurants`, `order_details`, `food`) and **filtering by** `user_id` (obtained via a subquery on `users`) and `date`.
6. **Restaurants with the Most Repeated Customers (Loyalty Analysis):**
    a. **Insight:** Identified the restaurant that has the highest number of repeat customers (customers who have ordered more than once). This highlights customer loyalty and satisfaction with a particular restaurant.
    b. **Query Logic:** Used a **subquery** to find `user_id` and `r_id` **combinations** with more than one order, then **counted** the distinct `user_id`s for each `r_id` to find the top restaurant.
7. **Month-over-Month Revenue Growth of the Food Delivery Platform:**
    a. **Insight:** Calculated the percentage month-over-month revenue growth for the food delivery platform. This is a critical business metric for understanding financial performance and trends.
    b. **Query Logic:** Employed a **Common Table Expression (CTE)** to calculate monthly revenue and then used the **`LAG()`** window function to compare current month's revenue with the previous month's to compute growth percentage.
8. **Customer's Favorite Food:**
    a. **Insight:** Determined the most frequently ordered food item for each customer. This information is invaluable for personalized marketing, food recommendations, and understanding individual preferences.

b. **Query Logic:** Used a **CTE** to calculate the frequency of each food item **ordered by** each user. Then, joined with `users` and `food` tables and used a subquery within the **`WHERE`** clause to select the food with the maximum frequency for each user.

Conclusion:

This SQL case study demonstrates the power of relational databases and SQL queries in extracting actionable insights from transactional data. The queries developed provide a solid foundation for understanding customer behavior, evaluating restaurant performance, and monitoring overall business growth for a food delivery platform. These insights can directly inform business strategies related to marketing, customer retention, restaurant partnerships, and menu optimization.