Mahesh Pandurang Mohite
IIT Kanpur

Swiggy Case Study
Skills: SQL, ER diagram

Problem 1

Find customers who have never ordered.

Solution:

```
SELECT name FROM
users
WHERE user_id NOT IN (SELECT user_id FROM orders)
```

Problem 2

Average Price/dish.

Solution:

```
SELECT f.f_name, AVG(price) AS 'Avg Price'
FROM menu m
JOIN food f
ON m.f_id = f.f_id
GROUP BY m.f_id
```

Problem 3

Find the top restaurant in terms of the number of orders for a given month.

Solution:

```
SELECT r.r_name, COUNT(*) AS 'month'
FROM orders o
JOIN restaurants r
ON o.r_id = r.r_id
WHERE MONTHNAME(date) LIKE 'June'
GROUP BY o.r_id
ORDER BY COUNT(*) DESC LIMIT 1
```

Problem 4

Restaurants with monthly sales greater than x amount.

Solution:

```
SELECT r.r_name, SUM(amount) AS 'revenue'
FROM order o
JOIN restaurants r
ON o.r_id = r.r_id
WHERE MONTHNAME(date) LIKE 'JUNE'
GROUP BY o.r_id
HAVING revenue > 500
```

Problem 5

Show all orders with order details for a particular customer in a particular date range.

Solution:

```
SELECT o.order_id, r.r_name, f.f_name
FROM order o
JOIN restaurants r
ON r.r_id = o.r_id
JOIN order_details od
ON o.order_id = od.order_id
JOIN food f
ON f.f_id = od.f_id
WHERE user_id = ( SELECT user_id FROM users WHERE name LIKE 'Ankit' )
AND ( date > '2022-06-10' AND date < '2022-07-10')
```

Problem 6

Find restaurants with max repeated customers.

Solution:

```
–       SELECT r.r_name, COUNT(*) AS 'loyal_customers'
        FROM (
              SELECT r_id, user_id, COUNT(*) AS 'visits'
              FROM orders
              GROUP BY r_id, user_id
              HAVING visits>1
        ) t
        JOIN restaurants r
        ON r,r_id = t.r_id
        GROUP BY t.r_id
        ORDER BY loyal_customers DESC LIMIT 1
```

Problem 7

Month over month revenue growth of swiggy.

Solution:

```
– SELECT month, ((revenue - prev)/prev) * 100 FROM (
      WITH sales AS
      (
           SELECT MONTHNAME(date) AS 'month', SUM(amount) AS 'revenue'
           FROM orders
           GROUP BY month
           ORDER BY MONTH(date)
       )
  SELECT month, revenue, LAG(revenue,1) OVER(ORDER BY revenue) AS prev FROM sales
  ) t
```

Note: WITH block and LAG function is used

Problem 8

Customer - favorite food.

Solution:

```
–       WITH temp AS
        (
             SELECT o.user_id, od.f_id, COUNT(*) AS 'frequency'
             FROM order o
             JOIN order_details od
             ON o.order_id = od.order_id
             GROUP BY o.user_id, od.f_id
        )
        SELECT * FROM
        temp t1
        JOIN users u
        ON u.user_id = t1.user_id
        JOIN food f
        ON f.f_id = t1.f_id
        WHERE t1.frequency = (
            SELECT MAX(frequency)
            FROM temp t2
            WHERE t2.user_id = t1.user_id
        )
```