Debugging Queries

Question 1: How many unique customers are in the city of 'Surat'?

SQL Query:

```
SELECT
    COUNT(DISTINCT customer_id) AS distinct_customers
FORM gdb080.dim_customers
WHERE city == 'surat';
```

Question 2: What are the minimum and maximum order quantities for each product?

SQL Query:

```
SELECT
    p.product_id,
    p.product_name,
    MIN(p.order_qty) as minimum_qty
    MAX(p.order_qty) as maximum_qty
FROM gdb080.fact_order_lines f
JOIN gdb080.dim_products p ON f.product_id = p.product_id
GROUP BY p.product_id;
```

Question 3: Generate a report with month_name and number of unfullfilled_orders(i.e order_qty - delivery_qty) in that respective month.

SQL Query:

```
SELECT
    MONTHNAME(order_placement_date as month_name,
    SUM(order_qty-delivery_qty)
FROM gdb080.fact_orders_lines
GROUP By MONTHNAME(order_placement_date)
ORDER BY unfullfilled_orders DESC;
```

Question 4: What is the percentage breakdown of order_qty by category? The final output includes the following fields:

- category
- order_qty_pct.

SQL Query:

```
with total_order_qty_by_category as
(
    SELECT
        p.category,
        SUM(f.order_qty) as total_quantity
    FROM gdb080.dim_products p
  JOIN gdb080.fact_order_lines f ON p.product_id = f.product_id
    GROUP BY p.category;
)
SELECT
    category,
    ROUND(100 * total_quantity / SUM(total_quantity) OVER (), 2) AS
order_qty_pct
FROM total_order_quantity_by_category
order by order_qty_pct DESC;
```

Question 5: Generate a report that includes the customer ID, customer name, ontime_target_pct, and percentage_category.

The percentage category is divided into four types: 'Above 90' if the ontime_target_pct is greater than 90, 'Above 80' if it is greater than 80, 'Above 70' if it is greater than 70, and 'Less than 70' for all other cases.

SQL Query:

```
SELECT
      customer_id,
      customer_name,
      t.ontime_target_pct,
   CASE
            WHEN t.ontime_target_pct > 90 THEN 'Above 90"
            WHEN t.ontime_target_pct > 80 THEN 'Above 80'
            WHEN t.ontime_target_pct > 70 THEN 'Above 70'
      ELSE "Below 70"
      END AS  percentage_category,
 FROM gdb080.dim_targets_orders t
 JOIN gdb080.dim_customers c
 ON t.customer_id = c.customer_id;
```

Question 6: Generate a report that lists all the product categories, along with the product names and total count of products in each category.
The output should have three columns:
category, products, and product_count.

SQL Query:

```
SELECT category, GROUP_CONCAT(product_name) AS products
COUNT(*) AS product_count
FROM gdb080.dim_products
GROUP category;
```

Question 7: What are the top 3 most demanded products in the 'Dairy' category, and their respective order quantity in millions?

The final output includes the following fields:

- product name
- order_qty_mln.

SQL Query:

```
SELECT
    p.product_name,
    ROUND(SUM(f.order_qt) / 1000000,2) AS order_qty_mln
FROM gdb080.dim_products p
JOIN gdb080.fact_order_lines f
WHERE p.category = 'Dairy'
GROUP BY p.product_name
ORDER BY order_qty_mln DESC
LIMIT 3;
```

Question 8: Calculate the OTIF % for a customer named Vijay Stores

The final output should contain these fields,

customer_name

OTIF_percentage

SQL Query:

```
SELECT
    c.customer_names,
    ROUND((SUM(f.otif) / COUNT(f.order_id) * 100),2) AS
    OTIF_percentage
FROM gdb080.fact_orders_aggregate f
JOIN gdb080.dim_customers c
ON c.customer_id = f.customer_id
GROUP BY c.customer_name
WHERE c.customer_name = "Vijay Stores";
```

Question 9: What is the percentage of 'in full' for each product and which product has the highest percentage, based on the data from the 'fact_order_lines' and 'dim_products' tables?

SQL Query:

```
WITH product_if_target AS (
    SELECT
        p.product_name,
        SUM(CASE WHEN f.in_full = 1 THEN 1 ELSE 0) AS if_count,
        COUNT(f.order_id) AS total_count
    FROM
        gdb080.fact_order_lines f
        JOIN gdb080.dim_products p ON p.product_id = f.product_id
    GROUP BY p.product_name
)
SELECT
    product_name,
    ROUND(if_count / total_count * 100), 2) AS IF_percentage
FROM
    product_if_target
order by IF_percentage DESC;
```