Atliq Mart – Supply Chain

Analysis - Harsh Pimpalkar

Problem Statement

Atliq Mart is a Growing FMCG manufacturer headquarter in Gujarat, India. It is currently operational in three cities Surat, Ahmedabad and Vadodara. They want to expand to other metros/ Tier 1 cities in next 2 years.

Atliq Mart is currently facing a problem where a few customers did not extend their annual contracts due to service issues. It is speculates that some of the essential products were either not delivered on time or not delivered in full over a continued period, which could have resulted in bad customer service. Management wants to fix this issue before expanding to other cities and requested their supply chain analytics team to track the „On time", and "In Full"delivery service level for all the customers daily basis so that they can respond swiftly to these issues.

The Supply Chain team decided to use a standard approach to measure the service level in which they will measure „on-time-delivery (OT) %", "In-Full-Delivery (IF) %", And „On Time In Full (OTIF) % " of the customer orders daily basis against the target service level set for each Customer.

I performed 20 SQL queries on the data provided by Atliq Mart.

Harsh Pimpalkar

OVERVIEW

Que. 1 Starting with simple overview of dataset present.

a) Columns presenting each table

b) Total customers present

c) Total Products with their categories available

d) Total cities they are currently operating insert

```
----------------------------OVERVIEW-----------------------------------

-- Overview of the Dataset

SELECT * FROM dim_customer; -- Cusotmer_table
SELECT * FROM dim_dates; -- dates_table
SELECT * FROM dim_products; -- products_table
SELECT * FROM dim_target_orders; -- target_orders_table
SELECT * FROM fact_order_lines; -- orders_lines_table
SELECT * FROM fact_order_aggregate; -- order_aggregated_table

-- To get total customers present
SELECT COUNT(distinct customer_id) AS Total_Cusotmers FROM dim_customers;
-- Total Customers are 35.

-- To get total products with their categories available
SELECT COUNT(DISTINCT product_id) AS Total_products FROM dim_products;
-- Total producta are 18.

-- To get total cities they are currently operating in
SELECT COUNT(DISTINCT city) AS Total_cities FROM dim_customers;
-- Atliq currently operating in 3 cities.
```

Harsh Pimpalkar

OVERVIEW

```sql
----------------------------OVERVIEW---------------------------------
-- What are total number of products and total number of customers?
SELECT COUNT(DISTINCT customer_id) AS Total_Customers,
       COUNT(DISTINCT product_id) AS Total_Products
FROM fact_order_lines;

-- What is the average order quantity by customers?
SELECT customer_id,
       AVG(order_qty) AS Avg_order_qty
FROM fact_order_lines
GROUP BY customer_id;

-- What is the average delivery rate time for orders by city?

SELECT city,
       AVG(DATEDIFF(actual_delivery_date, agreed_delivery_date)) AS Avg_delivery_date
FROM fact_order_lines
JOIN dim_customers ON fact_order_lines.customer_id = dim_customers.customer_id
GROUP BY city;

-- What is the average delivery time for on-time orders by city

SELECT city, AVG(DATEDIFF(actual_delivery_date, agreed_delivery_date)) AS Avg_delivery_date
FROM fact_order_lines
JOIN dim_customers
    ON fact_order_lines.customer_id = dim_customers.customer_id
JOIN fact_orders_aggregate
    ON fact_order_lines.order_id = fact_orders_aggregate.order_id
WHERE fact_orders_aggregate.on_time = 1
GROUP BY city;
```

Harsh Pimpalkar

OVERVIEW

Que. 2What are total orders, total orders on time, total order infull and total orders (on time infull) (OTIF) by city.

```sql
WITH  city_order_data AS
            (
                SELECT
                    dim_customers.city,
                    fact_orders_aggregate.order_id,
                    fact_orders_aggregate.on_time,
                    fact_orders_aggregate.in_full,
                    fact_orders_aggregate.otif
                FROM fact_orders_aggregate
                    JOIN dim_customers ON fact_orders_aggregate.customer_id = dim_customers.customer_id
              ),
all_order_data AS
                (
                    SELECT
                        city_order_data.city,
                        COUNT(DISTINCT city_order_data.order_id) AS Total_orders,
                        SUM(CASE WHEN city_order_data.on_time = 1 THEN 1 ELSE 0 END ) AS total_on_time,
                        SUM(CASE WHEN city_order_data.in_full = 1 THEN 1 ELSE 0 END) AS total_in_full,
                        SUM(CASE WHEN city_order_data.otif = 1 THEN 1 ELSE 0 END) AS total_otif
                    FROM city_order_data
                    GROUP BY city_order_data.city
                    )
SELECT
    all_order_data.city,
    all_order_data.total_orders,
    all_order_data.total_on_time,
    all_order_data.total_in_full,
    all_order_data.total_otif,
    (SELECT
        COUNT(DISTINCT order_id) FROM fact_orders_aggregate) AS overall_total_order
     FROM all_order_data;
```

city	total_orders	total_on_time	total_in_full	total_otif	overall_total_order
Ahmedabad	11061	6433	5995	3244	31729
Surat	9696	5935	5095	2916	31729
Vadodara	10972	6362	5657	3048	31729

Harsh Pimpalkar

Analyzing The Delivery

Que. 3 Provide insight regarding the share distribution of previous question metrics by customers.

```sql
WITH customer_metrics AS
    (
        SELECT
            c.customer_name,
            SUM(ol.order_qty) AS Total_orders,
            SUM(CASE WHEN o.on_time = 1 THEN ol.order_qty ELSE 0 END) AS total_orders_on_time,
            SUM(CASE WHEN o.in_full = 1 THEN ol.order_qty ELSE 0 END) AS total_orders_in_full,
            SUM(CASE WHEN o.otif = 1 THEN ol.order_qty ELSE 0 END) AS total_orders_otif
        FROM fact_order_lines ol
        JOIN dim_customers c  ON ol.customer_id = c.customer_id
        JOIN fact_orders_aggregate o ON ol.order_id = o.order_id
        GROUP BY c.customer_name
    )
  SELECT
        customer_name,
        Total_orders,
        total_orders_on_time,
        total_orders_in_full,
        total_orders_otif,
        ROUND(total_orders_on_time/Total_orders *100,2) AS 'on_time_%',
        ROUND(total_orders_in_full/Total_orders *100,2) AS 'in_full_%',
        ROUND(total_orders_otif/Total_orders *100,2) AS 'otif %'
  FROM customer_metrics
  ORDER BY Total_orders DESC;
```

Customers Name	Total Orders	Total Ordes On Time	Total Ordes In Full	Total Orders OTIF	On Time %	In Full %	OTIF %
Vijay Stores	1176293	998568	406464	304018	84.89	34.55	25.85
Lotus Mart	1157117	300217	560658	158378	25.95	48.45	13.69
Rel Fresh	1155598	980851	550183	424934	84.88	47.61	36.77
Propel Mart	1143763	981179	563551	450220	85.79	49.27	39.36
Acclaimed Stores	1120090	300689	520776	142935	26.85	46.49	12.76
Expert Mart	789698	667646	374604	285655	84.54	47.44	36.17
Coolblue	776624	208655	305960	89823	26.87	39.4	11.57
Elite Mart	772140	657062	226082	172363	85.1	29.28	22.32
Expression Stores	768746	647164	377375	291595	84.18	49.09	37.93
Info Stores	767833	640958	251810	186518	83.48	32.79	24.29
Sorefoz Mart	765536	646450	241100	182104	84.44	31.49	23.79
Atlas Stores	760711	640693	374600	288471	84.22	49.24	37.92
Viveks Stores	760300	636060	386970	301723	83.66	50.9	39.68
Chiptec Stores	756652	632896	376209	283655	83.64	49.72	37.49
Logic Stores	755835	632778	372760	283547	83.72	49.32	37.51

Analyzing The Delivery

Que. 4. Calculate ☒ variance between actual and target from on time(OT), infull(IF) and „ON_Time and In Full" metrics by City.

```sql
WITH actual AS
    (
        SELECT
            dim_customers.city,
            SUM(CASE WHEN fact_orders_aggregate.on_time = 1 THEN 1 ELSE 0 END) /
                COUNT(DISTINCT fact_orders_aggregate.order_id) * 100 AS actual_ot,
            SUM(CASE WHEN fact_orders_aggregate.in_full = 1 THEN 1 ELSE 0 END) /
                COUNT(DISTINCT fact_orders_aggregate.order_id) * 100 AS actual_if,
            SUM(CASE WHEN fact_orders_aggregate.otif = 1 THEN 1 ELSE 0 END) /
                COUNT(DISTINCT fact_orders_aggregate.order_id) * 100 AS actual_otif
        FROM fact_orders_aggregate
        JOIN dim_customers ON fact_orders_aggregate.customer_id = dim_customers.customer_id
        GROUP BY dim_customers.city
    ),
    target AS (
        SELECT
            dim_customers.city,
            SUM(dim_target_orders.ontime_target_pct) / COUNT(DISTINCT dim_target_orders.customer_id) AS target_ot,
            SUM(dim_target_orders.infull_target_pct) / COUNT(DISTINCT dim_target_orders.customer_id) AS target_if,
            SUM(dim_target_orders.otif_target_pct) / COUNT(DISTINCT dim_target_orders.customer_id) AS target_otif
        FROM
            dim_target_orders
        JOIN dim_customers ON dim_target_orders.customer_id = dim_customers.customer_id
        GROUP BY dim_customers.city
    )
        SELECT
            actual.city,
            ROUND((actual.actual_ot - target.target_ot) / target.target_ot * 100, 3) AS ot_varience,
            ROUND((actual.actual_if - target.target_if) / target.target_if * 100, 3) AS if_varience,
            ROUND((actual.actual_otif - target.target_otif) / target.target_otif * 100, 3) AS otif_varience
        FROM actual
        JOIN target ON actual.city = target.city;
```

city	ot_varience	if_varience	otif_varience
Ahmedabad	-32.242	-29.915	-55.897
Surat	-29.05	-31.676	-54.683
Vadodara	-32.707	-31.559	-57.207

Harsh Pimpalkar

Analyzing The Delivery

Que. 5 Calculate the average lead time for each customer

```sql
SELECT
    c.customer_name,
    ROUND(AVG(DATEDIFF(delivery_date, order_date)),2) AS Average_Time
FROM
        fact_order_lines ol
JOIN
            dim_customers c ON ol.customer_id = c.customer_id
GROUP BY
            c.customer_name
ORDER BY
                Average_Time;
```

customer_name	Average_Time
Propel Mart	2.14
Atlas Stores	2.15
Elite Mart	2.15
Rel Fresh	2.16
Sorefoz Mart	2.16
Vijay Stores	2.16
Chiptec Stores	2.18
Expert Mart	2.19
Info Stores	2.2
Logic Stores	2.22
Viveks Stores	2.22
Expression Stores	2.23
Acclaimed Stores	3.23
Coolblue	3.28
Lotus Mart	3.3

Harsh Pimpalkar

Analyzing The Delivery

Que. 6 Provide the average number of days between order placement and delivery for all orders by city.

```sql
SELECT
    c.city,
    ROUND(AVG(DATEDIFF(delivery_date, order_date)),2) AS Average_days
FROM
    fact_order_lines ol
JOIN
    dim_customers c ON ol.customer_id = c.customer_id
GROUP BY
        c.city
ORDER BY
        Average_days DESC;
```

city	Average_days
Ahmedabad	2.45
Vadodara	2.44
Surat	2.37

From the result , we observed the following insights. -

Average delivery days are of 2 and half day for each city.

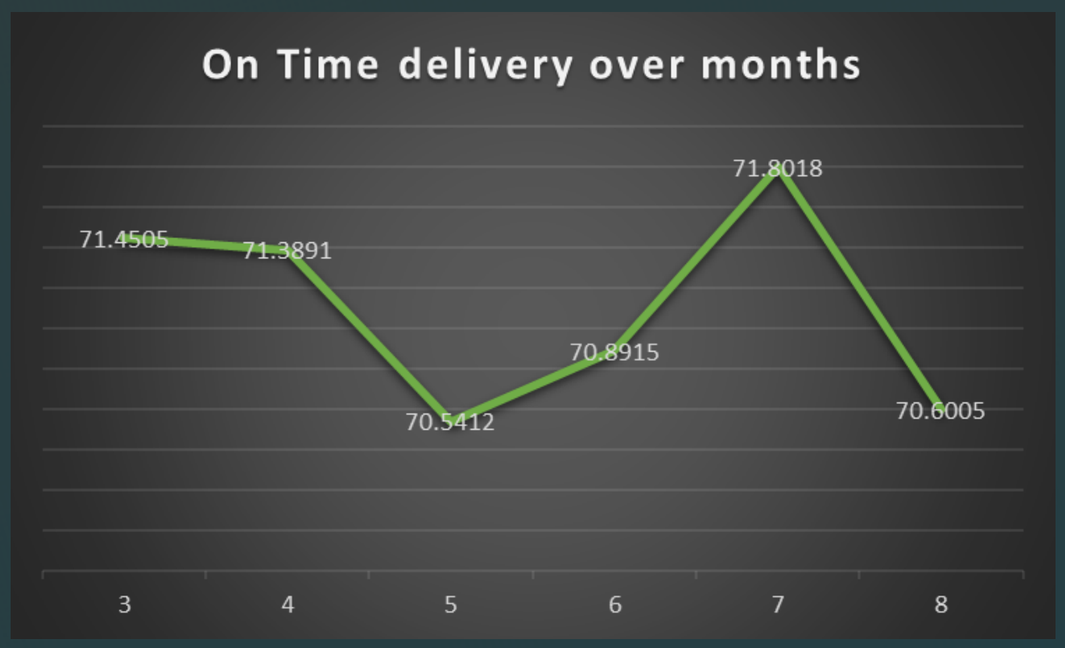
Harsh Pimpalkar

Analyzing The Delivery

Que. 7 Analyze the trend of on time delivery over the months.

```sql
SELECT
    MONTH(order_date) AS month,
    COUNT(order_id) AS Total_orders,
    SUM(CASE WHEN fact_order_lines.On_Time = 1 THEN 1 ELSE 0 END) AS on_time_orders,
    (SUM(CASE WHEN fact_order_lines.On_Time = 1 THEN 1 ELSE 0 END) /
        COUNT(fact_order_lines.order_id) *100 ) AS on_time_pct
FROM
        fact_order_lines
GROUP BY
            MONTH(order_date)
ORDER BY
                MONTH(order_date);
```

month	Total_orders	on_time_orders	on_time_pct
3	9755	6970	71.4505
4	9402	6712	71.3891
5	9756	6882	70.5412
6	9389	6656	70.8915
7	9685	6954	71.8018
8	9109	6431	70.6005



From the visualization,

it clearly represents that month 7 (July Month) as the highest On Time deliveries.

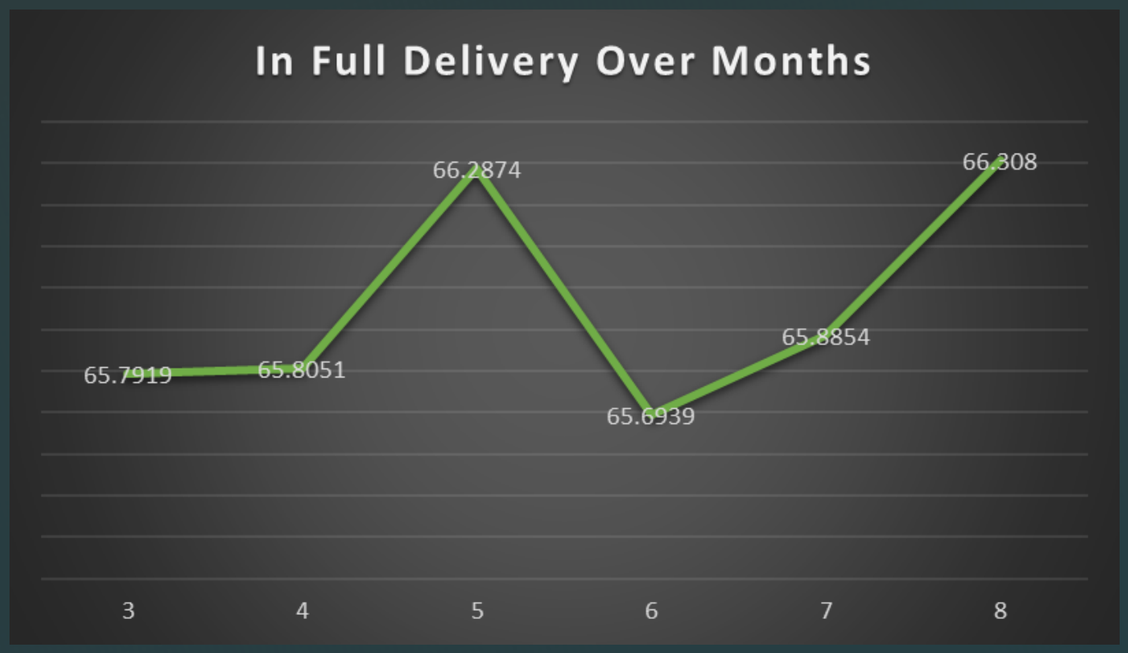
Harsh Pimpalkar

Analyzing The Delivery

Que. 8 Analyze the trend of In Full delivery over the months.

```sql
SELECT
        MONTH(order_date) AS Month,
        COUNT(order_id) AS Total_orders,
        SUM(CASE WHEN fact_order_lines.In_Full = 1 THEN 1 ELSE 0 END ) AS in_full_orders,
        (SUM(CASE WHEN fact_order_lines.In_Full = 1 THEN 1 ELSE 0 END) /
            COUNT(fact_order_lines.order_id) * 100) AS in_full_pct
FROM
            fact_order_lines
GROUP BY
                MONTH(order_date)
ORDER BY
                MONTH(order_date);
```

month	Total_orders	on_time_orders	on_time_pct
3	9755	6970	71.4505
4	9402	6712	71.3891
5	9756	6882	70.5412
6	9389	6656	70.8915
7	9685	6954	71.8018
8	9109	6431	70.6005



From the visualization,
it clearly represents that the May and August Months as the highest In Full deliveries.

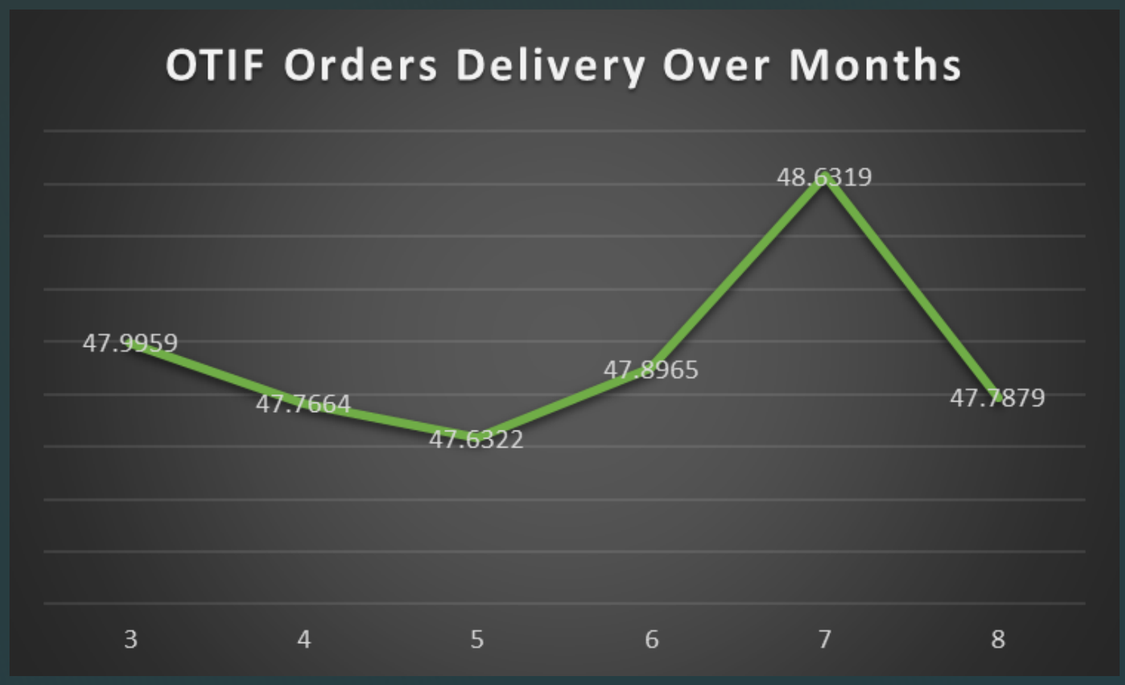
Harsh Pimpalkar

Analyzing The Delivery

Que. 9 Analyze the trend of In Full and On time delivery over months.

```sql
SELECT
        MONTH(order_date) AS Month,
        COUNT(order_id) AS Total_Orders,
        SUM(CASE WHEN fact_order_lines.On_Time_In_Full= 1 THEN 1 ELSE 0 END) AS otif_orders,
        (SUM(CASE WHEN fact_order_lines.On_Time_In_Full = 1 THEN 1 ELSE 0 END) /
            COUNT(fact_order_lines.order_id) * 100)  AS otif_orders_pct
  FROM
        fact_order_lines
  GROUP BY
        MONTH(order_date)
  ORDER BY
        MONTH(order_date);
```

Month	Total_Orders	otif_orders	otif_orders_pct
3	9755	4682	47.9959
4	9402	4491	47.7664
5	9756	4647	47.6322
6	9389	4497	47.8965
7	9685	4710	48.6319
8	9109	4353	47.7879



From the visualization,
it clearly represents that the July Month highest On time and In Full deliveries.

Harsh Pimpalkar

Customer Performance

Que. 10 Top 5 customers by total_quantity_orderd, in full quantity ordered and „on time and infull" quantity ordered.

```sql
-- Top 5 Cusotmers by Total_quantity_ordered

 SELECT
    dim_customers.customer_name,
    SUM(fact_order_lines.order_qty) AS Total_order_qty
FROM
    dim_customers
JOIN fact_order_lines ON dim_customers.customer_id = fact_order_lines.customer_id
GROUP BY dim_customers.customer_name
ORDER BY Total_order_qty DESC
LIMIT 5;
```

customer_name	Total_order_qty
Vijay Stores	1176293
Lotus Mart	1157117
Rel Fresh	1155598
Propel Mart	1143763
Acclaimed Stores	1120090

Harsh Pimpalkar

Customer Performance

```sql
-- Top 5 Cusotmers by in_full_qty_ordered

SELECT
    dim_customers.customer_name,
    SUM(fact_order_lines.delivery_qty) AS Full_qty_ordered
FROM
        dim_customers
JOIN
            fact_order_lines
  ON dim_customers.customer_id = fact_order_lines.customer_id
GROUP BY dim_customers.customer_name
ORDER BY Full_qty_ordered DESC
LIMIT 5;
```

customer_name	Total_order_qty
Vijay Stores	1176293
Lotus Mart	1157117
Rel Fresh	1155598
Propel Mart	1143763
Acclaimed Stores	1120090

customer_name	OTIF_Qty
Info Stores	161531
Expression Stores	161098
Viveks Stores	153525
Propel Mart	152887
Sorefoz Mart	152257

```sql
-- Top 5 Customers by "OTIF" ordered Quantity.

WITH otif_ordered_qty AS
        (
            SELECT
                fact_order_lines.customer_id,
                SUM(CASE WHEN fact_orders_aggregate.otif = 1
                        THEN fact_order_lines.delivery_qty ELSE 0 END) AS OTIF_Qty
            FROM fact_order_lines
            JOIN fact_orders_aggregate ON fact_order_lines.order_id = fact_orders_aggregate.order_id
            GROUP BY fact_order_lines.customer_id
        )
    SELECT
        dim_customers.customer_name,
        otif_ordered_qty.OTIF_Qty
    FROM otif_ordered_qty
    JOIN dim_customers ON otif_ordered_qty.customer_id = dim_customers.customer_id
    ORDER BY OTIF_Qty DESC
    LIMIT 5;
```

Harsh Pimpalkar

Customer Performance

Que. 11 Provide actual OT%, IF%, AND OTIF% by Cusotmers

```sql
WITH actual AS
            (
                SELECT
                    dim_customers.customer_name,
                    SUM(CASE WHEN fact_orders_aggregate.on_time = 1 THEN 1 ELSE 0 END) /
                        COUNT(DISTINCT fact_orders_aggregate.order_id) * 100 AS  actual_ot,
                    SUM(CASE WHEN fact_orders_aggregate.in_full = 1 THEN 1 ELSE 0 END) /
                        COUNT(DISTINCT fact_orders_aggregate.order_id) * 100 AS actual_if,
                    SUM(CASE WHEN fact_orders_aggregate.otif = 1 THEN 1 ELSE 0 END) /
                        COUNT(DISTINCT fact_orders_aggregate.order_id) * 100 AS actual_otif
                FROM fact_orders_aggregate
                JOIN dim_customers ON fact_orders_aggregate.customer_id = dim_customers.customer_id
                GROUP BY dim_customers.customer_name
            )
        SELECT
            actual.customer_name,
            ROUND(actual.actual_ot, 2) AS ot_pct,
            ROUND(actual.actual_if, 2) AS if_pct,
            ROUND(actual.actual_otif,2) AS otif_pct
        FROM
            actual
        ORDER BY actual.customer_name;
```

customer_name	ot_pct	if_pct	otif_pct
Acclaimed Stores	29.43	52.36	15.47
Atlas Stores	71.81	59.78	39.55
Chiptec Stores	71.62	60.35	38.73
Coolblue	29.13	44.73	13.75
Elite Mart	72.45	37.94	24.37
Expert Mart	72.54	59.81	39.11
Expression Stores	69.92	60.83	38.39
Info Stores	70.94	41.16	25.52
Logic Stores	70.82	60.14	38.78
Lotus Mart	28.11	53.35	16.34
Propel Mart	73.64	59.74	40.92
Rel Fresh	72.32	58.69	38.18
Sorefoz Mart	72.67	39.19	25.89
Vijay Stores	72.45	44.98	28.28
Viveks Stores	70.61	60.07	39.44

Harsh Pimpalkar

Customer Performance

Que. 12 Categories the orders by Product category for each customer in descending Order

```sql
WITH customer_orders AS (
        SELECT
            dim_customers.customer_name,
            dim_products.category,
            COUNT(DISTINCT fact_order_lines.order_id) AS Total_Orders
        FROM   fact_order_lines
        JOIN dim_customers ON fact_order_lines.customer_id = dim_customers.customer_id
        JOIN dim_products ON fact_order_lines.product_id = dim_products.product_id
        GROUP BY dim_customers.customer_name, dim_products.category
        )
       SELECT
            customer_orders.customer_name,
            SUM(CASE WHEN customer_orders.category = "dairy"
                    THEN customer_orders.Total_Orders ELSE 0 END) AS 'Dairy',
            SUM(CASE WHEN customer_orders.category = "food"
                    THEN customer_orders.Total_Orders ELSE 0 END) AS 'Food',
            SUM(CASE WHEN customer_orders.category = "beverages"
                    THEN customer_orders.Total_Orders ELSE 0 END) AS 'Beverages',
        SUM(customer_orders.Total_Orders) AS "Total_Orders"
    FROM
        customer_orders
    GROUP BY customer_orders.customer_name
    ORDER BY "Total_Orders" DESC ;
```

customer_name	Dairy	Food	Beverages	Total_Orders
Acclaimed Stores	2603	759	783	4145
Atlas Stores	1322	506	475	2303
Chiptec Stores	1320	488	482	2290
Coolblue	1825	540	526	2891
Elite Mart	1330	497	495	2322
Expert Mart	1366	523	492	2381
Expression Stores	1336	483	512	2331
Info Stores	1361	475	483	2319
Logic Stores	1378	490	474	2342
Lotus Mart	2653	758	751	4162
Propel Mart	1965	720	718	3403
Rel Fresh	1987	731	743	3461
Sorefoz Mart	1352	465	517	2334
Vijay Stores	2023	758	702	3483
Viveks Stores	1339	470	469	2278

Harsh Pimpalkar

Customer Performance

Que. 13 Categories the orders by Product category for each city in descending order

```sql
WITH customer_orders AS
    (
        SELECT
            dim_customers.city,
            dim_products.category,
            COUNT(DISTINCT fact_order_lines.order_id) AS total_orders
        FROM
            fact_order_lines
        JOIN dim_customers ON fact_order_lines.customer_id = dim_customers.customer_id
        JOIN dim_products ON fact_order_lines.product_id = dim_products.product_id
        GROUP BY dim_customers.city, dim_products.category
       )
       SELECT
            customer_orders.city,
            SUM(CASE WHEN customer_orders.category = "dairy"
                    THEN customer_orders.total_orders ELSE 0 END) AS 'Dairy',
            SUM(CASE WHEN customer_orders.category = 'food'
                    THEN customer_orders.total_orders ELSE 0 END) AS 'Food',
            SUM(CASE WHEN customer_orders.category = 'beverages'
                    THEN customer_orders.total_orders ELSE 0 END) AS 'Beverages',
            SUM(customer_orders.total_orders) AS "Total_Orders"
    FROM customer_orders
    GROUP BY customer_orders.city
    ORDER BY "Total_Orders" DESC ;
```

city	Dairy	Food	Beverages	Total_Orders
Ahmedabad	8763	2951	3011	14725
Surat	7728	2742	2630	13100
Vadodara	8669	2970	2981	14620

Harsh Pimpalkar

Customer Performance

Que. 14 Find the top 3 Customers from each city based on thier total orders and what is their OTIF?

```sql
WITH customer_orders AS (
        SELECT
            dim_customers.city,
            dim_customers.customer_name,
            COUNT(fact_orders_aggregate.order_id) AS Total_orders,
            CONCAT((ROUND((COUNT(CASE WHEN otif = 1 THEN (otif) END) /
                        COUNT(otif) * 100),2)), "%") AS "OTIF%",
            ROW_NUMBER() OVER(PARTITION BY dim_customers.city ORDER BY
                        COUNT(fact_orders_aggregate.order_id) DESC) AS Ranking
        FROM fact_orders_aggregate
        JOIN dim_customers ON fact_orders_aggregate.customer_id = dim_customers.customer_id
        GROUP BY dim_customers.city, dim_customers.customer_name
    )
    SELECT * FROM customer_orders WHERE Ranking IN (1,2,3);
```

city	customer_name	Total_orders	OTIF%	Ranking
Ahmedabad	Coolblue	1219	20.34%	1
Ahmedabad	Acclaimed Stores	1194	19.10%	2
Ahmedabad	Lotus Mart	1179	7.97%	3
Surat	Lotus Mart	1203	21.28%	1
Surat	Acclaimed Stores	1126	6.93%	2
Surat	Expression Stores	842	35.27%	3
Vadodara	Coolblue	1218	7.14%	1
Vadodara	Acclaimed Stores	1190	19.92%	2
Vadodara	Lotus Mart	1168	19.69%	3

Harsh Pimpalkar

Product Performance

Que. 15 Which product was most and least ordered by each customer?

```sql
WITH customer_products AS
  (
      SELECT
          dim_customers.customer_name,
          dim_products.product_name,
          COUNT(fact_order_lines.product_id) AS Product_count
       FROM fact_order_lines
       JOIN dim_customers ON fact_order_lines.customer_id = dim_customers.customer_id
       JOIN dim_products ON fact_order_lines.product_id = dim_products.product_id
       GROUP BY dim_customers.customer_name, dim_products.product_name
   )
   SELECT
          customer_products.customer_name,
          MAX(CASE WHEN customer_products.product_count =
                    (SELECT MAX(product_count) FROM customer_products c2
                          WHERE c2.customer_name = customer_products.customer_name) THEN
                    customer_products.product_name ELSE NULL END ) AS most_ordered_product,
        MIN(CASE WHEN customer_products.product_count =
                    (SELECT MIN(product_count) FROM customer_products c2
                        WHERE c2.customer_name = customer_products.customer_name) THEN
                    customer_products.product_name ELSE NULL END) AS least_ordered_product
 FROM customer_products
 GROUP BY customer_products.customer_name
 ORDER BY customer_products.customer_name
 LIMIT 3;
```

customer_name	most_ordered_product	least_ordered_product
Acclaimed Stores	AM Tea 500	AM Butter 250
Atlas Stores	AM Biscuits 250	AM Tea 100
Chiptec Stores	AM Ghee 250	AM Curd 50

Harsh Pimpalkar

Product Performance

Que. 16 Try to distribute the total product orders by their categoriesand their share, also show each city's top and worst selling products

```sql
WITH city_categories AS
    (
        SELECT
                dim_customers.city,
                dim_products.product_name,
                dim_products.category,
                COUNT(fact_order_lines.order_id) AS total_orders
          FROM fact_order_lines
          JOIN dim_customers ON fact_order_lines.customer_id = dim_customers.customer_id
          JOIN dim_products ON fact_order_lines.product_id = dim_products.product_id
          GROUP BY dim_customers.city, dim_products.category
      ),
        categories_totals AS
       (
            SELECT
                city,
                SUM(CASE WHEN category = 'dairy'
                            THEN total_orders ELSE 0 END) AS 'dairy_total',
                SUM(CASE WHEN category = 'food'
                            THEN total_orders ELSE 0 END) AS 'food_total',
                SUM(CASE WHEN category = 'beverages'
                            THEN total_orders ELSE 0 END) AS 'beverages_total',
                SUM(total_orders) AS total_orders
            FROM city_categories
            GROUP BY city
        )
```

Harsh Pimpalkar

Product Performance

```sql
SELECT
    city_categories.city,
    city_categories.category,
    city_categories.total_orders,
    CONCAT(ROUND((city_categories.total_orders/categories_totals.total_orders)*100,2) , "%")AS percent_share,
    MAX(CASE WHEN city_categories.total_orders =
            (SELECT MAX(total_orders) FROM city_categories c2
                    WHERE c2.city = city_categories.city) THEN
                city_categories.product_name  ELSE NULL END) as top_selling_products,
    MIN(CASE WHEN city_categories.total_orders =
            (SELECT MIN(total_orders) FROM city_categories c2
                    WHERE c2.city = city_categories.city) THEN
                city_categories.product_name ELSE NULL END) as least_selling_products
FROM city_categories
JOIN categories_totals ON city_categories.city = categories_totals.city
GROUP BY city_categories.city, city_categories.category
ORDER BY city_categories.city, percent_share DESC;
```

city	category	total_orders	percent_share	top_selling_products	least_selling_products
Ahmedabad	Dairy	13130	66.73%	AM Butter 500	NULL
Ahmedabad	beverages	3294	16.74%	NULL	NULL
Ahmedabad	Food	3252	16.53%	NULL	AM Biscuits 500
Surat	Dairy	11910	66.75%	AM Butter 500	NULL
Surat	Food	3022	16.94%	NULL	NULL
Surat	beverages	2910	16.31%	NULL	AM Tea 500
Vadodara	Dairy	13056	66.69%	AM Butter 500	NULL
Vadodara	Food	3265	16.68%	NULL	NULL
Vadodara	beverages	3257	16.64%	NULL	AM Tea 500

Harsh Pimpalkar

Product Performance

Que. 17 Analyze the customer orders count distribution by day of week.

```sql
SELECT
    WEEKDAY(order_date) AS Day_no,
    CASE WEEKDAY(order_date)
        WHEN 0 THEN 'Sunday'
        WHEN 1 THEN 'Monday'
        WHEN 2 THEN 'Tuesday'
        WHEN 3 THEN 'Wednwsday'
        WHEN 4 THEN 'Thursday'
        WHEN 5 THEN 'Friday'
        WHEN 6 THEN 'Saturday'
     END AS Day_Name  ,
    COUNT(order_id) AS Total_orders
FROM
    fact_order_lines
GROUP BY
    Day_Name
ORDER BY
    Total_orders DESC;
```

Day_no	Day_Name	Total_orders
1	Monday	8343
6	Saturday	8176
5	Friday	8163
2	Tuesday	8157
3	Wednwsday	8098
0	Sunday	8082
4	Thursday	8077

Harsh Pimpalkar

Insights

- The cities with the most customers are Ahmedabad and Vadodara.
- Some customers, such as Sorfex Mart in Vadodara and Info Stores in Surat, have much higher in-full percentages than on-time percentages.
- Among all the customers, Vijay Stores placed the highest number of orders.
- Expert Mart has the highest percentage of on-time orders at 85.54%.
- Coolblue and Elite Mart have the lowest in-full and on-time performance for orders.
- The average delivery time for Delivering is 2 days, and half a day for each city.

Harsh Pimpalkar