



BIG DATA AND BUSINESS INTELLIGENCE MODULE



AtliQ - Supply chain Analysis

Name: Preethi Malyala

Student Id: W9566704

Contents

Executive Summary:.....	3
Introduction:	5
Dataset Description:.....	5
Outline of Business Questions:.....	7
KEY FINDINGS and INSIGHTS:.....	8
Conclusions and Recommendations:.....	14
ICA Appendix-BA Design	15
Data Pre-Processing and Data Cleaning:.....	15
1. Data Loading:	15
2. First Row as Header:	18
3. Removing Null Values:	18
4. Renaming Columns:	19
5. Creating Tables in Power Query Editor:.....	20
6. Removing Columns:	26
DATA MODELLING:.....	27
DAX Measures and Calculated Columns:	30
DASHBOARD EXPLORATORY ANALYSIS:.....	36
References:.....	44

Executive Summary:

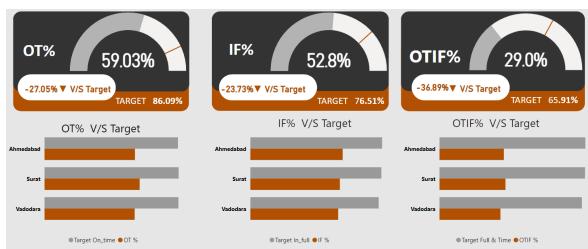
Overview:

The main purpose of creating the business report is to perform the Supply Chain analysis of AtliQ mart based on the Orders and Target data which comprises of Products, Customers, Orders, Delivery details.

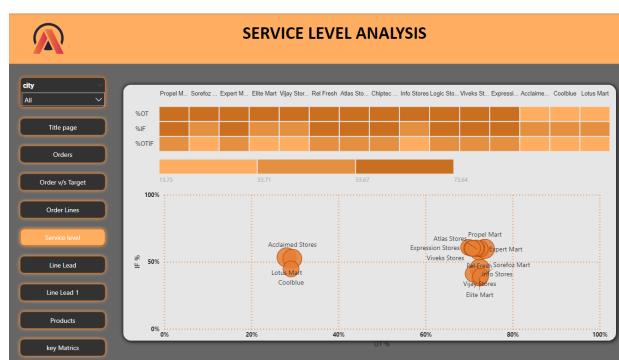
Business Questions:

- What is the average delay of orders from the agreed date?
- Which store has the highest delayed orders?
- Which products have been delayed the highest and what is the highest product quantity delayed?
- Are there any noticeable trends?
- Which city has the highest performance and which city does not?
- Which customers are achieving targets?

Key Insights and Solutions:



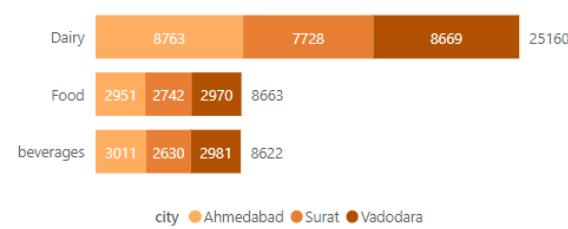
- On Time Delivery is 59.03% compared to the target of 86.09%
- In Full Quantity Delivery is 52.8% vs the target of 76.51%
- On Time and In-Full Delivery is at a surprising low of 29.0% compared to target of 65.91%



- **Acclaimed Stores, Lotus Mart and Coolblue** have the least OT% and OTIF% compared to any other stores which implies the delay is highest from these three stores.



- **Lotus Mart, Acclaimed Stores** has the highest orders of all the stores and comprises of 22.25% of total sales.



- **Dairy products** comprise of almost twice as many orders as both Food and Beverages combined. Also, Dairy products have the highest delay in orders among all products.
- **Ahmedabad and Vadodara** have almost the same number of delayed orders 5902 and 5891.

Conclusions:

Unfortunately, **Acclaimed Stores, Cool Blue & Lotus Mart** are the Key customers who would certainly not be willing to renew the contract as more than **75% of order lines are being delivered late** and they are having the highest number of orders. Moreover, out of **that 75% delayed deliveries, 66% of them are late by 2 or 3 days**. The average **OTIF% is significantly lower than the Targeted OTIF %** Which shows poor delivering capabilities.

Recommendations:

Hence my recommendation for the AtliQ business is to focus on delivery management as delay in deliveries is the key hindrance to business development and retaining the customers.

More resources and infrastructure must be allocated for the Dairy products as they comprise of more than 70% of total orders.

Stores like Coolblue, Lotus Mart & Acclaimed Stores be given more priority on delivering the products as they are the key customers despite having a 75% of order lines delay.

Introduction:

AtliQ mart is a fast-moving consumer goods company in India. It is currently operational in three cities Surat, Ahmedabad, and Vadodara. They want to expand to other metros/tier 1 cities in the next 2 years.

The Company is currently facing a problem where a few key customers did not extend the annual contract due to service issues. It is speculated that some of the essential orders were either not delivered on time or not delivered in full quantity 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 on a 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 OnTime in full (OTIF) % of the customer orders on a daily basis against the target service level set for each customer.

Dataset Description:

Source:

The dataset used here is AtliQ supply chain analysis. The dataset was obtained from Codebasics: "<https://codebasics.io/event/codebasics-resume-project-challenge>".

This dataset contains 2 tables whereas the first dataset consists of 57096 rows and 8 rows with the information on company products, customers, order id, orders placed to date, agreed-on the delivery date, order delivered date, ordered quantity, and delivered Quantity.

Whereas the target orders table contains 35 rows and 4 columns. It contains information on the target of the customers based on on-time delivery, in full quantity, Ontime delivery, and Full quantity delivery.

This dataset is selected because of the information it contains like quantity, dates, list of products, and Customers list. The data looks so clear to work on. Using PowerBi as a challenge in the small-scale company to generate the company performance report is my main aim.

Dataset Description:

The following tables show the description of the columns in the AtliQ-Supply chain analysis dataset:

Index	Column	Description
1.	Customer Id	Unique Id of the Customers
2.	Order Id	Unique Id of the Orders
3.	Product Id	Unique Id of the Products
4.	Ordered quantity	Quantity ordered by the Customers

5.	Delivered Quantity	The quantity delivered to the customers
6.	Ordered Placement Date	The date when the order got placed by the Customers
7.	Agreed Delivery Date	The date agreed upon by the company to deliver the product
8.	Delivered Date	The date when the products got delivered
9.	Product name	Name of the Product
10.	Customer name	Name of the customer
11.	Customer City	City the Customer is located in
12.	Product Category	The category of the product belongs to

Table 1: Orders line table and description

Index	column	Description
1.	customer_id	The unique Id of Customers
2.	ontime_target	Average of On-Time Target
3.	infull_target%	Average of In-Full Target
4.	otif_target%	Average of OTIF Target

Table 2: Target Orders list Table and description

The following Screenshots are the data in excel form

order_id	order_placement_date	customer_id	product_id	order_qty	agreed_delivery_date	actual_delivery_date	delivery_qty	Customer Name	Customer City	Product Name	Product Category	
1	FMR34203601	Tuesday, March 1, 2022	789203	25891601	110	Friday, March 4, 2022	Friday, March 4, 2022	110	Rel Fresh	Vadodara	AM Tea 500	beverages
2	FMR32320302	Tuesday, March 1, 2022	789320	25891203	347	Wednesday, March 2, 2022	Wednesday, March 2, 2022	347	Chiptec Stores	Surat	AM Butter 500	Dairy
3	FMR33320501	Tuesday, March 1, 2022	789320	25891203	187	Thursday, March 3, 2022	Thursday, March 3, 2022	187	Chiptec Stores	Surat	AM Butter 500	Dairy
4	FMR34220601	Tuesday, March 1, 2022	789220	25891203	235	Friday, March 4, 2022	Friday, March 4, 2022	235	Atlas Stores	Surat	AM Butter 500	Dairy
5	FMR33703603	Tuesday, March 1, 2022	789703	25891203	176	Thursday, March 3, 2022	Thursday, March 3, 2022	176	Sorefz Mart	Vadodara	AM Butter 500	Dairy
6	FMR33721603	Tuesday, March 1, 2022	789721	25891203	345	Thursday, March 3, 2022	Thursday, March 3, 2022	345	Logic Stores	Ahmedabad	AM Butter 500	Dairy
7	FMR34202003	Tuesday, March 1, 2022	789420	25891203	138	Thursday, March 3, 2022	Sunday, March 6, 2022	138	Lotus Mart	Surat	AM Butter 500	Dairy
8	FMR34420402	Tuesday, March 1, 2022	789420	25891203	381	Friday, March 4, 2022	Saturday, March 5, 2022	381	Lotus Mart	Surat	AM Butter 500	Dairy
9	FMR32403401	Tuesday, March 1, 2022	789403	25891203	348	Wednesday, March 2, 2022	Wednesday, March 2, 2022	348	Propel Mart	Vadodara	AM Butter 500	Dairy
10	FMR34121203	Tuesday, March 1, 2022	789121	25891203	480	Friday, March 4, 2022	Sunday, March 6, 2022	480	Coolblue	Ahmedabad	AM Butter 500	Dairy
11	FMR32501601	Tuesday, March 1, 2022	789501	25891203	478	Wednesday, March 2, 2022	Wednesday, March 2, 2022	478	Viveks Stores	Surat	AM Butter 500	Dairy
12	FMR34501203	Tuesday, March 1, 2022	789501	25891203	491	Friday, March 4, 2022	Friday, March 4, 2022	491	Viveks Stores	Surat	AM Butter 500	Dairy
13	FMR34102602	Tuesday, March 1, 2022	789102	25891203	407	Friday, March 4, 2022	Friday, March 4, 2022	387	Vijay Stores	Ahmedabad	AM Butter 500	Dairy
14	FMR33902203	Tuesday, March 1, 2022	789902	25891203	299	Thursday, March 3, 2022	Thursday, March 3, 2022	299	Elite Mart	Ahmedabad	AM Butter 500	Dairy
15	FMR34903603	Tuesday, March 1, 2022	789903	25891203	329	Friday, March 4, 2022	Friday, March 4, 2022	329	Elite Mart	Vadodara	AM Butter 500	Dairy
16	FMR32421203	Tuesday, March 1, 2022	789421	25891203	441	Wednesday, March 2, 2022	Thursday, March 3, 2022	419	Lotus Mart	Ahmedabad	AM Butter 500	Dairy
17	FMR33421203	Tuesday, March 1, 2022	789421	25891203	104	Thursday, March 3, 2022	Sunday, March 6, 2022	99	Lotus Mart	Ahmedabad	AM Butter 500	Dairy
18	FMR34212003	Tuesday, March 1, 2022	789421	25891203	491	Thursday, March 3, 2022	Saturday, March 5, 2022	491	Propel Mart	Ahmedabad	AM Butter 500	Dairy
19	FMR33420203	Tuesday, March 1, 2022	789402	25891203	138	Thursday, March 3, 2022	Thursday, March 3, 2022	138	Expert Mart	Ahmedabad	AM Butter 500	Dairy
20	FMR33621603	Tuesday, March 1, 2022	789621	25891203	172	Friday, March 4, 2022	Friday, March 4, 2022	155	Acclaimed Stores	Surat	AM Butter 500	Dairy
21	FMR34520301	Tuesday, March 1, 2022	789520	25891203	38	Thursday, March 3, 2022	Thursday, March 3, 2022	30	Sorefz Mart	Ahmedabad	AM Ghee 150	Dairy
22	FMR33702302	Tuesday, March 1, 2022	789702	25891302	54	Wednesday, March 2, 2022	Wednesday, March 2, 2022	54	Chiptec Stores	Surat	AM Ghee 150	Dairy
23	FMR32320302	Tuesday, March 1, 2022	789320	25891302	37	Friday, March 4, 2022	Friday, March 4, 2022	37	Info Stores	Vadodara	AM Ghee 150	Dairy
24	FMR34603302	Tuesday, March 1, 2022	789603	25891302	63	Friday, March 4, 2022	Saturday, March 5, 2022	60	Logic Stores	Ahmedabad	AM Ghee 150	Dairy
25	FMR34721502	Tuesday, March 1, 2022	789721	25891302	56	Thursday, March 3, 2022	Friday, March 4, 2022	56	Expert Mart	Ahmedabad	AM Ghee 150	Dairy
26	FMR33621302	Tuesday, March 1, 2022	789621	25891302	54	Wednesday, March 2, 2022	Thursday, March 3, 2022	49	Acclaimed Stores	Surat	AM Ghee 150	Dairy
27	FMR32520302	Tuesday, March 1, 2022	789520	25891302	42	Thursday, March 3, 2022	Saturday, March 5, 2022	34	Acclaimed Stores	Surat	AM Ghee 150	Dairy
28	FMR32520601	Tuesday, March 1, 2022	789520	25891302	23	Friday, March 4, 2022	Saturday, March 5, 2022	22	Acclaimed Stores	Surat	AM Ghee 150	Dairy
29	FMR34520302	Tuesday, March 1, 2022	789520	25891302								

Fig: orders line table data in excel

The screenshot shows an Excel spreadsheet titled "dim_targets_orders - Excel". The table has four columns: "customer_id", "ontime_to_infull_targ", "off_targ", and "target%". The data consists of 28 rows of numerical values. The "customer_id" column contains values from 789201 to 789702. The "ontime_to_infull_targ" column contains values like 87, 81, 70, etc. The "off_targ" column contains values like 85, 81, 69, etc. The "target%" column contains values like 92, 76, 70, etc.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
customer_id	ontime_to_infull_targ	off_targ	target%																			
1	customer_id	ontime_to_infull_targ	off_targ	target%																		
2	789201	87	81	70																		
3	789202	85	81	69																		
4	789203	92	76	70																		
5	789301	89	78	69																		
6	789303	88	78	69																		
7	789101	86	80	69																		
8	789102	90	81	73																		
9	789103	92	67	62																		
10	789121	78	77	60																		
11	789122	76	65	49																		
12	789220	91	76	69																		
13	789221	85	80	68																		
14	789320	91	81	74																		
15	789321	87	81	70																		
16	789401	85	79	67																		
17	789402	89	80	71																		
18	789403	86	79	68																		
19	789420	75	79	59																		
20	789421	78	66	51																		
21	789422	79	81	64																		
22	789501	92	80	74																		
23	789503	86	82	71																		
24	789520	75	68	51																		
25	789521	78	79	62																		
26	789522	76	79	60																		
27	789601	92	67	62																		
28	789603	92	75	69																		
29	789702	89	66	59																		

Fig 2: target orders list table in excel

Outline of Business Questions:

This PowerBi project mainly aims at the supply chain analysis of the AtliQ company. This Analysis is going to answer the following questions.

- What is the average of orders delayed from the agreed date?
- Which store has the highest delayed orders?
- Which products have been delayed the highest and what is the highest product quantity delayed?
- Are there any noticeable trends?
- Which city has the highest performance and which city does not?
- Which customers are achieving targets?

KEY FINDINGS and INSIGHTS:



Title page

Orders

Order v/s Target

Products

Order Lines

Service level

Line Lead

Line Lead 1

key Metrics

ORDERS At Glance

Select Year
All

32K

Total Orders

19K

Orders ON Time

17K

Orders in FULL

9208

Orders full & time

Select Month
All

Ahmedabad 3011 8763 2951 14725

Dairy 8763 7728 8669 25160

Vadodara 2981 8669 2970 14620

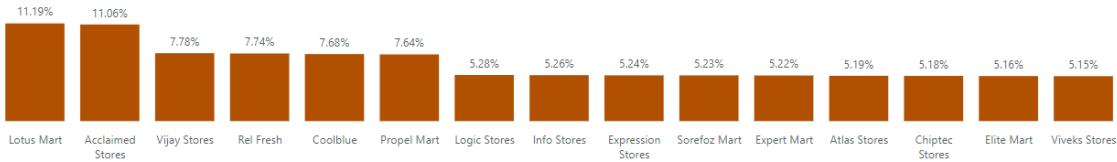
Food 2951 2742 2970 8663

Surat 2630 7728 2742 13100

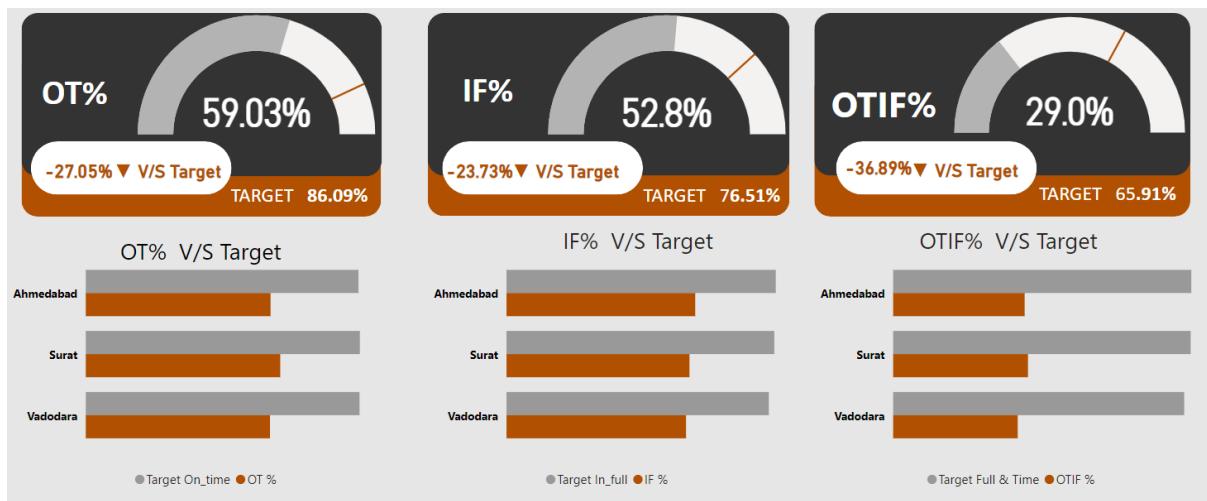
beverages 3011 2630 2981 8622

● Beverages ● Dairy ● Food

city ● Ahmedabad ● Surat ● Vadodara

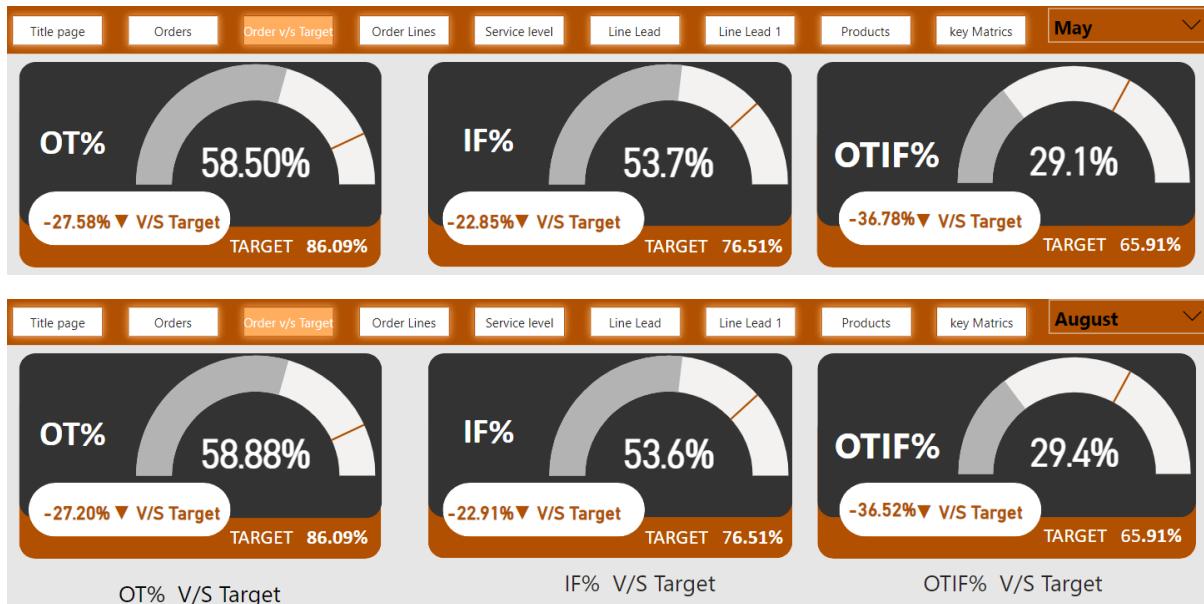


- **Dairy products** orders are almost double that of the Food and Beverages category.
- First 6 Customers Acclaimed Stores, Lotus Mart, Vijay Stores, RelFresh, Cool Blue & Propel Mart constitutes 53% of Total orders & seems to be key customers.
- Used 2 bar graphs that depict the Orders based on Categories in different cities and the Orders based on cities in different categories.
- The last bar graph shows the cities with the highest order count to Lowest.

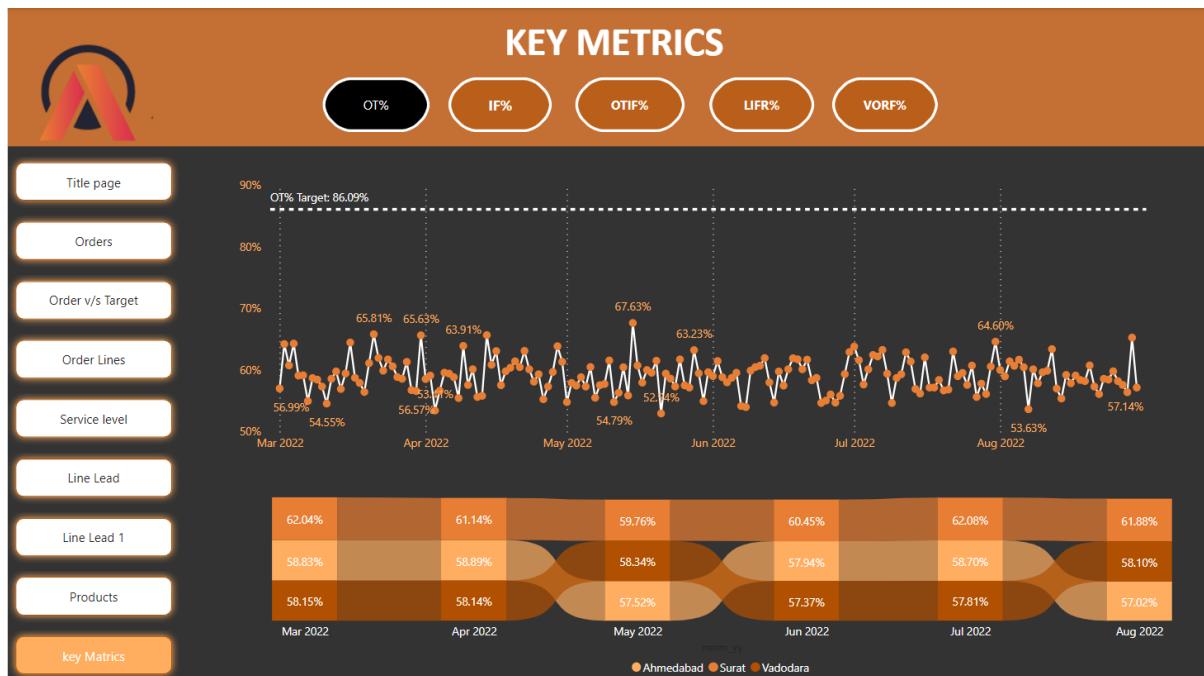


- As we see in the above slide only 59.03% of total orders are delivered on time
- Average OT%, IF%, and OTIF% are noticeably lower than the Targeted levels.

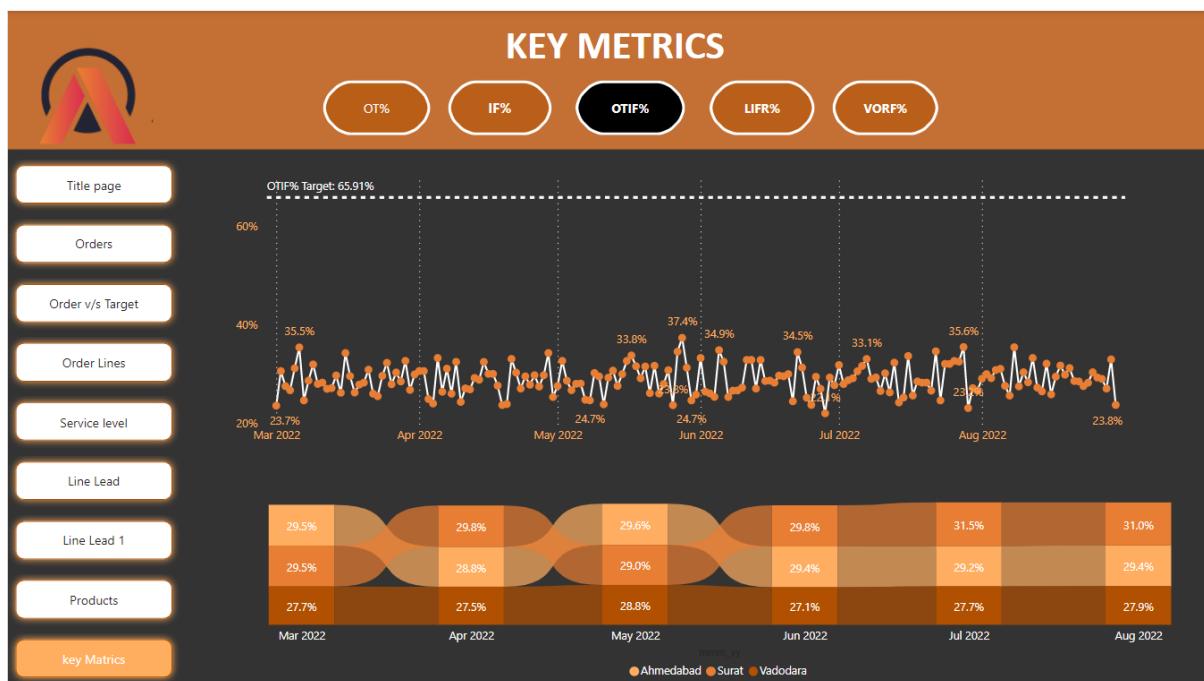
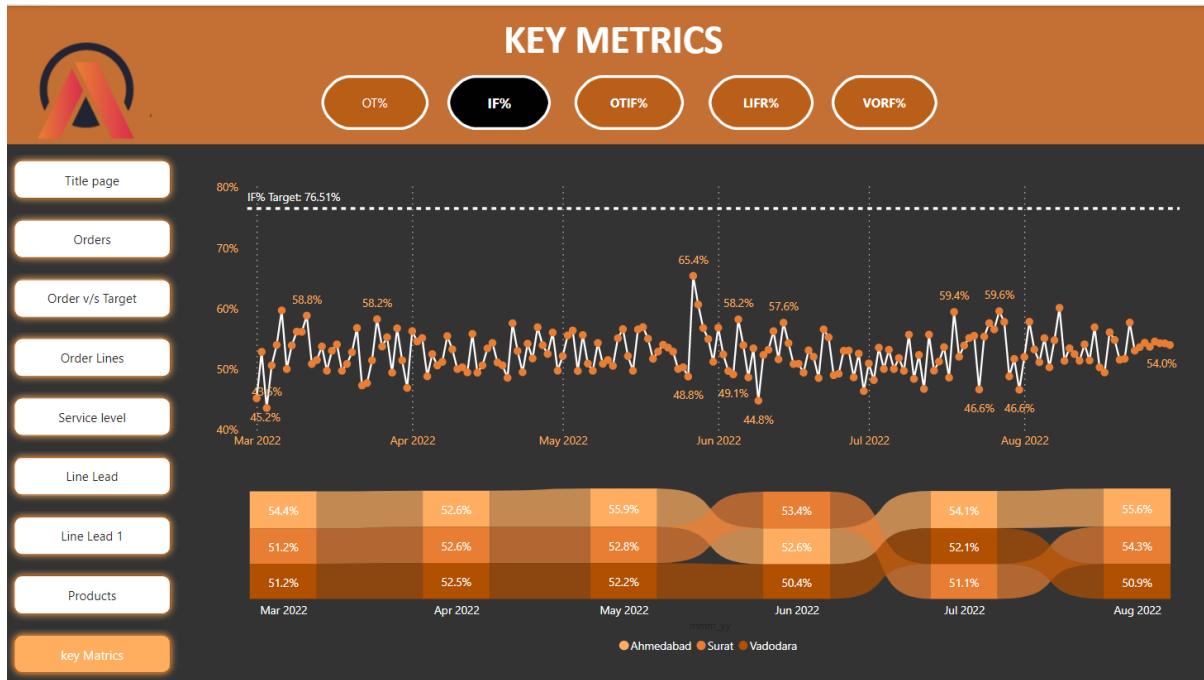
- The gauge axis is used to show the actual OT, IF and OTIF values with the respected target values.
- A card is used to show the difference between the actual and Target values.
- The bar graphs show the difference between OT, IF and OTIF values with the target values based on cities.



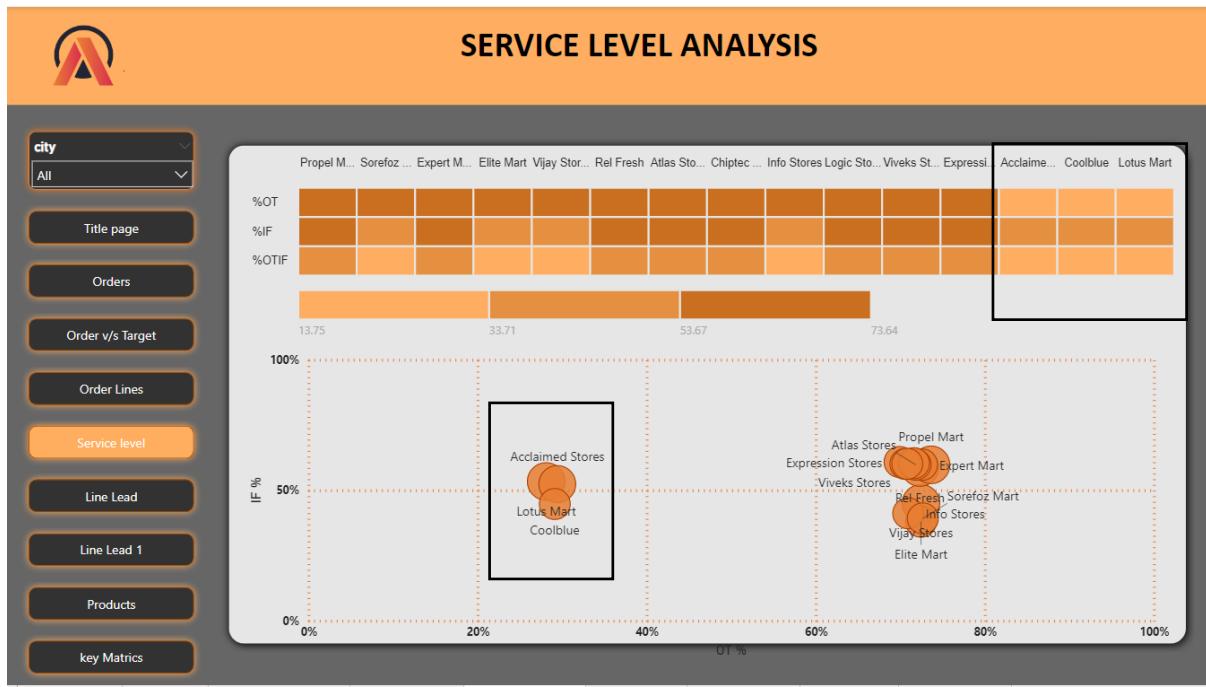
- **MAY'22** and **AUG'22** are the two months where **IF%** and **OTIF %** have shown slight improvement compared to the past month, still they are far away from Target levels



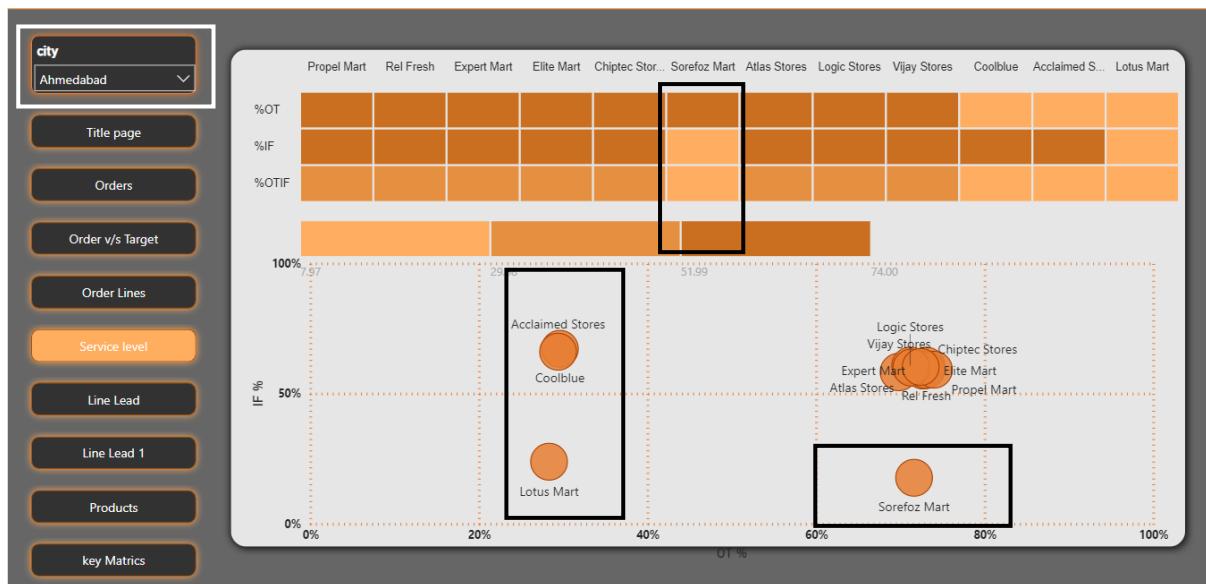
- When the **Daily Trend for OT%** is plotted for Order using **Line Chart**, the difference in the OT% and the Target OT% is significantly high, which depicts that the On Time Delivery Percentage of orders is very delayed.



- Even the Daily trends of the **IF%** and **OTIF % chart** shows that not a single day they are able to touch the Target lines for respective matrices.
- Especially **OTIF% performance is quite disappointing.**



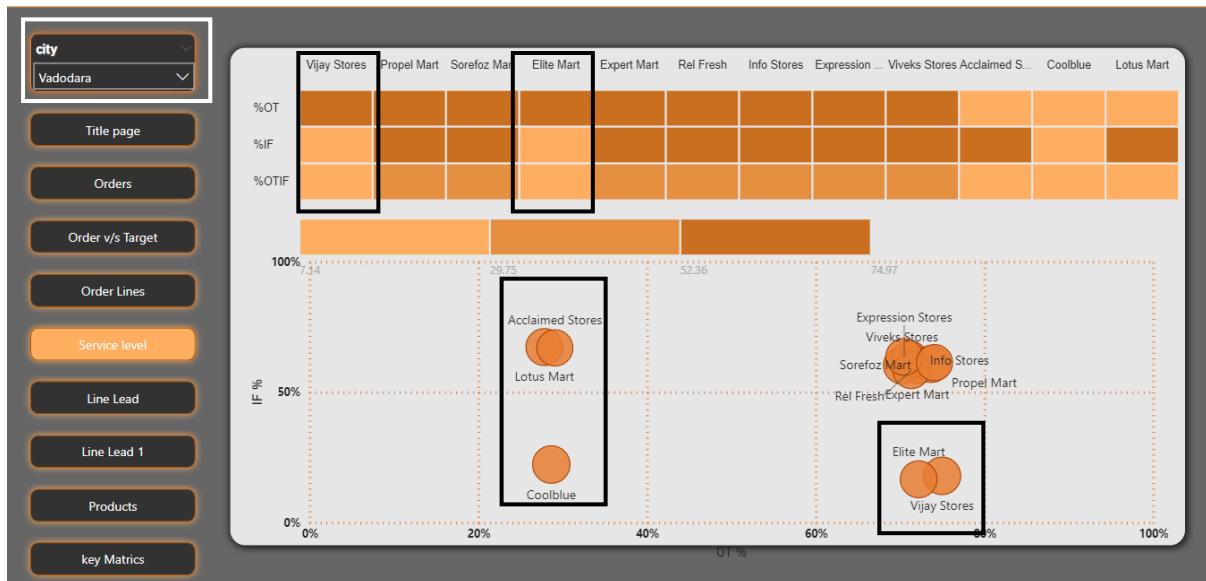
- Service Level Analysis of Orders shows **Acclaimed Stores, Cool Blue & Lotus Mart** have noticeably lower OT% and OTIF % even their IF% too are also not so much convincing.



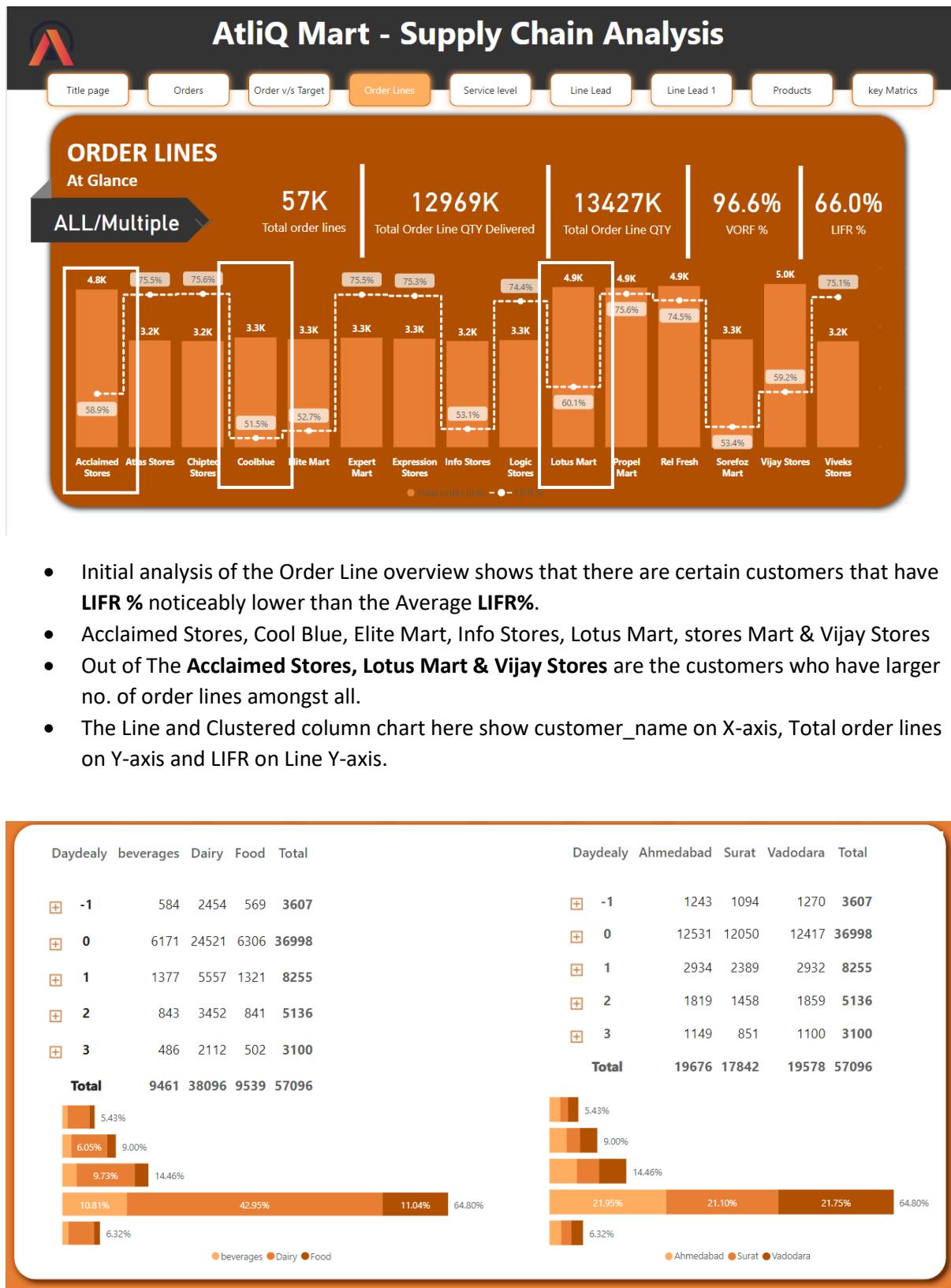
- When service levels are filtered for **AHMEDABAD** city it brings to our notice one more alarming insight that **IF% & OTIF % level for Sorefoz Mart** are severely low, which needs to be taken care.

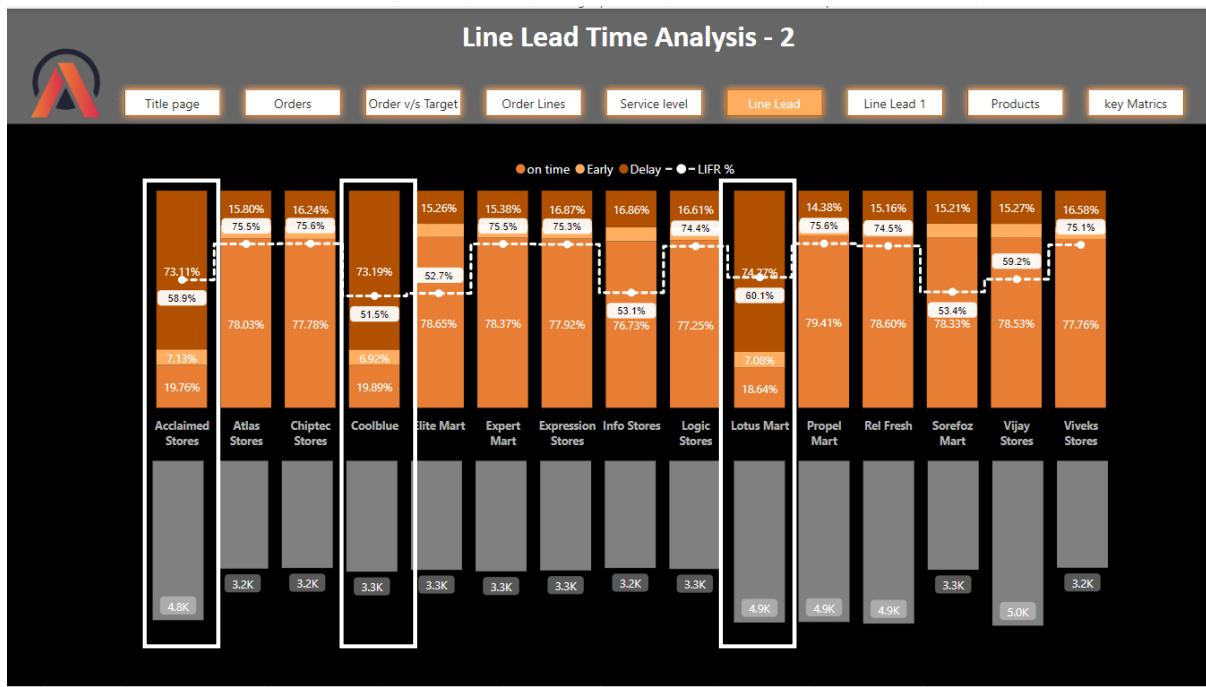


- When service levels are filtered for **SURAT** city Info Stores shows alarming **IF% & OTIF %** levels.



- For **BARODA** city Elite Mart & Vijay Stores' IF% & OTIF % levels are giving alarming signals.
- All these alarming levels of IF % and OTIF % show that distribution centres are running out of stock very quickly and replenishments must be optimized to improve.





Further detailed analysis of Line lead time analysis combined with delayed days analysis reveals some interesting facts.

Conclusions and Recommendations:

Unfortunately, **Acclaimed Stores, Cool Blue & Lotus Mart** are the Key customers who would certainly not be willing to renew the contract as more than 75% of order lines are being delivered late and they are having the highest number of orders. Moreover, out of that **75% delayed deliveries, 66% of them are late by 2 or 3 days**. The average OTIF% is significantly lower than the Targeted OTIF % Which shows poor delivering capabilities.

Hence my recommendation for the AtliQ company is to focus on delivery management as delay in deliveries is the key hindrance to business development and retaining the customers.

ICA Appendix-BA Design

Data Pre-Processing and Data Cleaning:

1. Data Loading:

The first step of data pre-processing is importing data. So, to Import data click on Get data on the PowerBi desktop. Since I already have existing excel data, I have not created a Blank query using M language to create Data that's another option to get data loaded in PowerBi Tool.

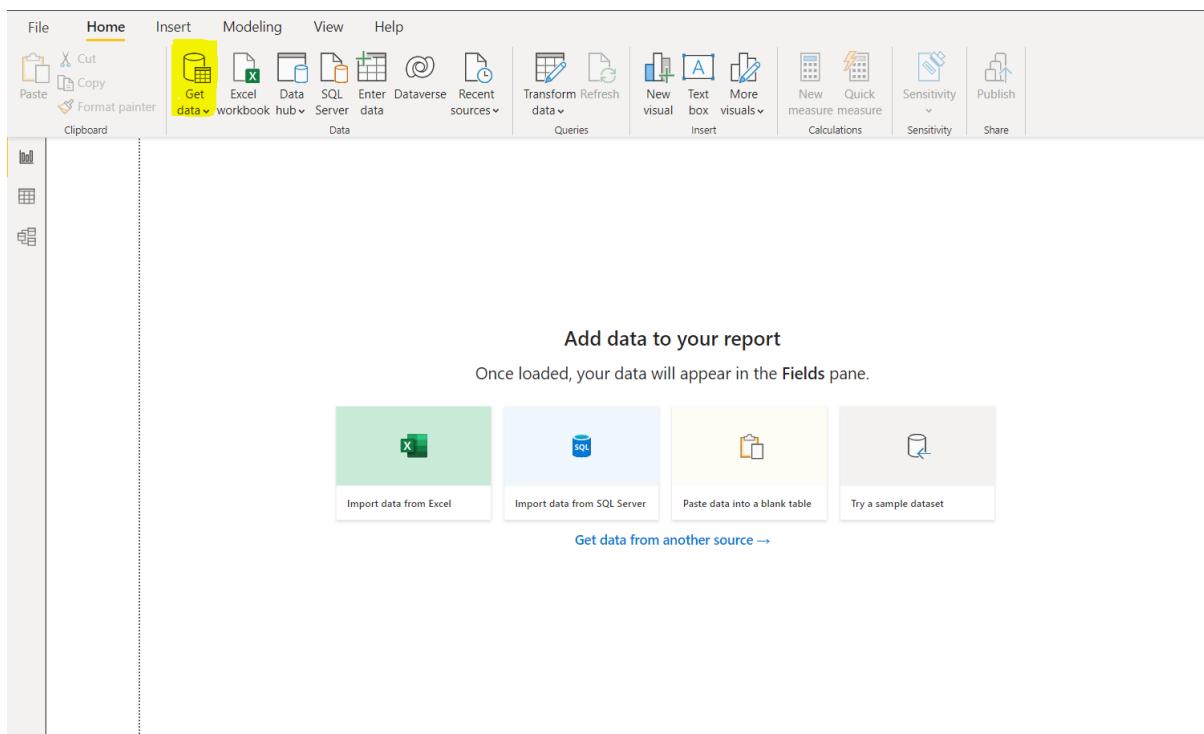


Fig 2: Screenshot to Get Data

To pull in data click on “**Get Data**” on Desktop Menu.

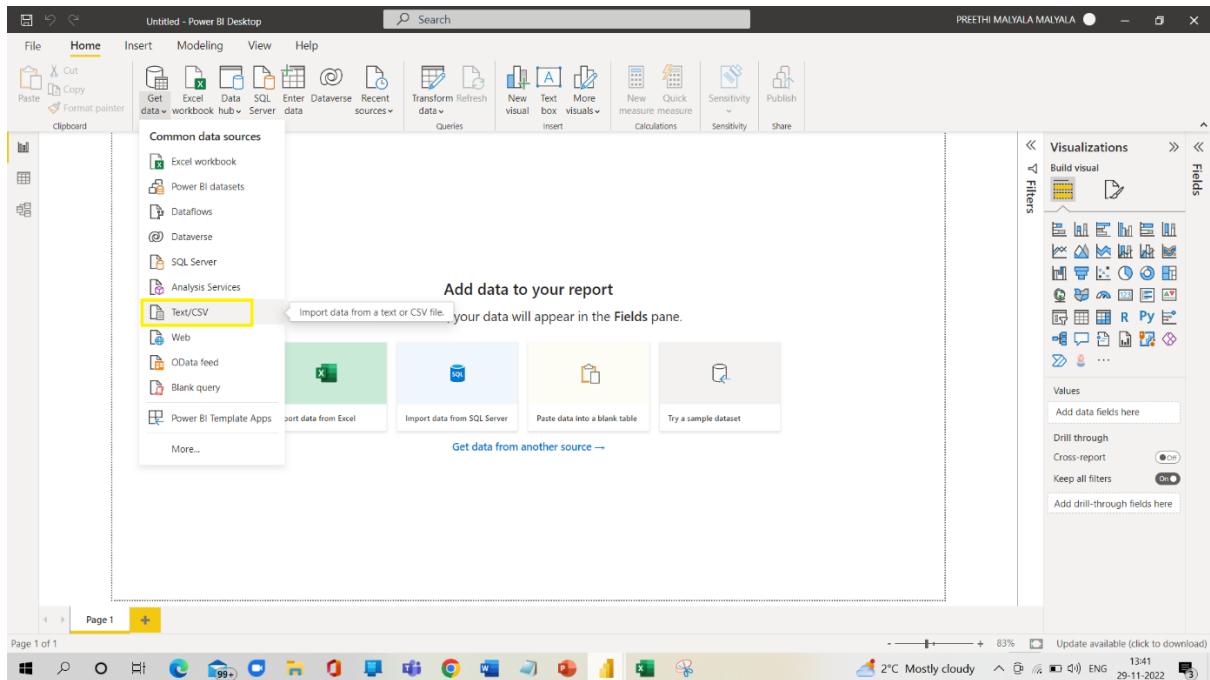


Fig 3: Choosing the data type

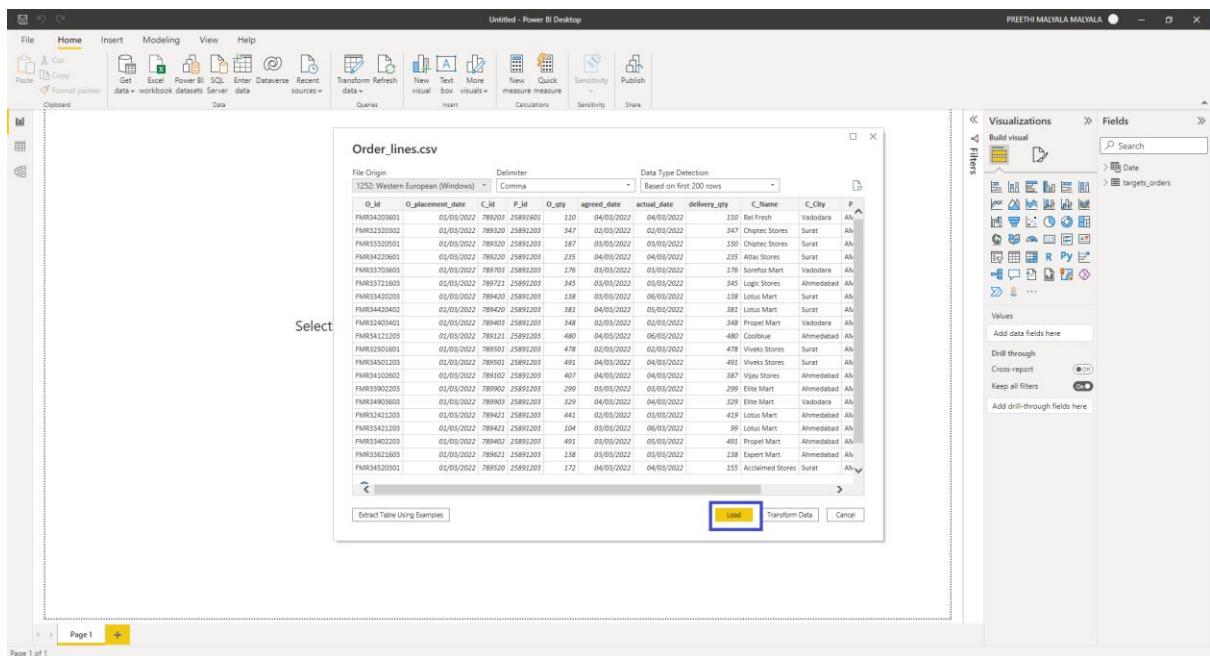


Fig 4: Load data dialog box

The same process was followed by the Targets table as well, Once the data is loaded you will see all the fields of the loaded data available in the Fields Section of the Power BI Desktop as highlighted below in the Fig (5). In the below screenshot, we can see the two tables loaded successfully.

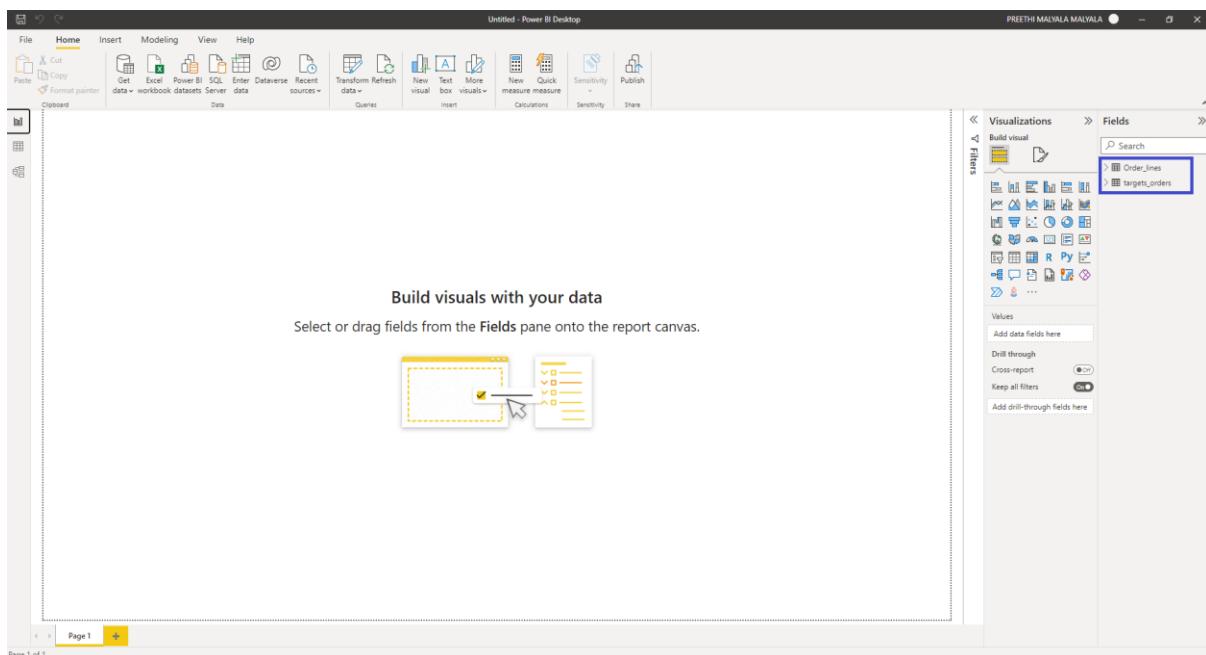


Fig 5: Two tables successfully loaded in PowerBi

Data Cleaning

Cleaning the data by clicking on the “Transform data” in the top view.

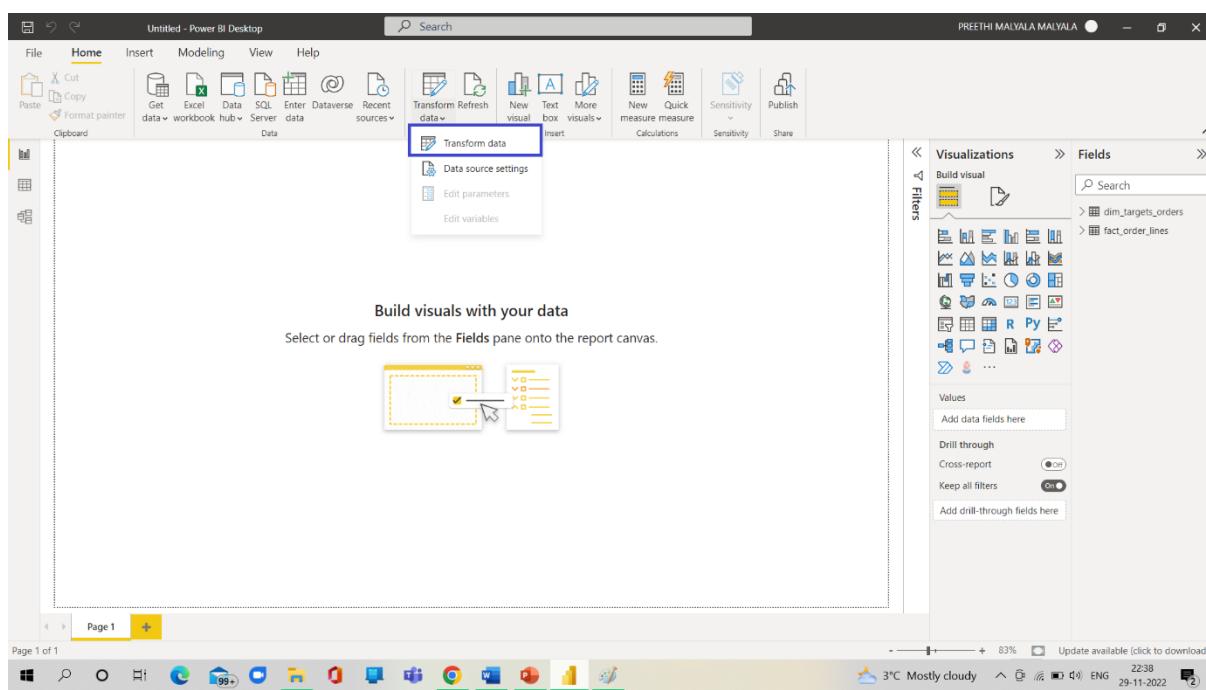


Fig 6: Click on Transform data

The screenshot shows the Power Query Editor interface with the 'fact_order_lines' query selected. The 'Applied Steps' pane on the right highlights the 'Changed Type1' step, which was used to change the first row into headers. The data grid displays 999+ rows of order data.

Fig 7: Power query editor is opened

2. First Row as Header:

As shown in fig (7) the header is one of the rows in the dataset to change it as the header in the power query editor choose the “**Use first row as headers**” option. As shown in fig(8) the first row is changed into a header.

The screenshot shows the Power Query Editor after applying the 'Use First Row as Headers' step. The 'Promoted Headers1' step is highlighted in the 'Applied Steps' pane, indicating it was used to promote the first row into headers. The data grid now correctly identifies the first row as headers and the subsequent rows as data.

Fig 8: First row as header

3. Removing Null Values:

To remove null values the process followed is in the power query editor chosen in the Order_lines table which has null values. Selected the remove blank rows option when right click on the dataset. Found few null values in the data as null values can be seen in the above fig(7). Below is the Screenshot of the table after removing Null values from the data.

The screenshot shows the Power Query Editor interface with the 'fact_order_lines' table selected. The 'Properties' pane on the right shows the 'Customer_Name' column. The 'Applied Steps' pane indicates that the 'Promoted Headers' step was applied. The table itself contains numerous rows of data, with the last row visible being row 999.

Fig 8: Screenshot after removing the null values

4. Renaming Columns:

To not get confused with column names and identify them easily Change the column names as shown in the below table:

Old Column Name	New Column Name
O_id	Order_id
O_placement_date	Order_placement_date,
C_id	Customer_id
P_id	Product_id
O_qty	Ordered_qty
agreed_date	Agreed_Delivery_Date
actual_date	Actual_Delivery_Date
C_Name	Customer_Name
C_City	Customer_City
P_Name	Product_Name
P_Category	Product_Category

Below Fig(9) is the result after changing the column names

The screenshot shows the Microsoft Power Query Editor interface. A table named 'Order_lines' is open, displaying data from row 1 to 38. The columns are: Order_Id, Order_placement_date, Customer_Id, Product_Id, Ordered_qty, Agreed_Delivery_date, Actual_Delivery_date, delivery_qty, and Customer_Name. The 'APPLIED STEPS' pane on the right shows a step named 'Renamed Columns'.

Fig (9): Screenshot after renaming the columns

Click on Close & Apply.

5. Creating Tables in Power Query Editor:

Starting from the data above, I want to categorize the orders table into 4 tables.

Customer Table

Products Table

Orders Aggregate

Date Table.

Customer's table:

- 1) Go to Report View
- 2) Click on “Transform Data”
- 3) Rt. Click on Order_lines table, then choose Duplicate where you can see a duplicate table is created like below.

The screenshot shows the Power Query Editor interface with the 'Order_lines' table selected. In the 'Applied Steps' pane, a step named 'Renamed Columns' is visible under the 'Renamed Columns' section. The table preview shows columns such as Order_id, Order_placement_date, Customer_id, Product_id, Ordered_qty, Agreed_Delivery_date, Actual_Delivery_date, delivery_qty, and Customer_Name.

Here For Customers Table the columns we needed from Order_lines table are Customer name, Customer City, and Customer Id. Then Click on “Choose Columns” then chosen the Columns mentioned above. The pop-up appears as we can see in the image below.

The screenshot shows the 'Choose Columns' dialog box open over the 'Order_lines' table. The dialog allows selecting specific columns to keep, with 'Customer_id', 'Customer_name', and 'Customer_City' checked. The 'Ok' button is highlighted at the bottom right of the dialog.

Click on “Ok”. Then the Table is created like below fig()

The screenshot shows the Power Query Editor interface with the 'Order_lines' query selected. The table contains 39 rows of data with columns: Customer_ID, Customer_Name, and Customer_City. The 'APPLIED STEPS' pane on the right shows the step 'Removed Other Columns'.

As we can see Customers table has been created but there are many duplicate values so on Customer_id column give a right click and choose “Remove Duplicates”. A final Table has been created as Shown Below and Rename the table as Customers.

The screenshot shows the Power Query Editor interface with the 'Customers' query selected. The table contains 35 rows of data with columns: Customer_ID, Customer_Name, and Customer_City. The 'APPLIED STEPS' pane on the right shows the step 'Removed Duplicates'.

As shown in the above Fig() we can see the Customers table is created which Includes the Columns Customer ID, Customer City, and Customer name with 35 rows.

Product Table:

- 1) Go to Report View
- 2) Click on “Transform Data”
- 3) Right click on Order_lines table, then choose Duplicate where you can see a duplicate table is created like below. As explained in the Customers table Click on choose Columns and selected the columns Product_id, Product_name, and Product_Category.

The screenshot shows the Power Query Editor interface. A 'Choose Columns' dialog box is open over the main table view. In the 'Choose the columns to keep' section, three columns are selected: 'Product_id', 'Product_name', and 'Product_Category'. The 'OK' button at the bottom right of the dialog is highlighted with a blue box. The main table view shows a large dataset with columns for Order_Id, Order_Placement_Date, Customer_Id, and so on. The 'Applied Steps' pane on the right shows the step 'Renamed Columns'.

Then click on “OK”. As we can see Customers table has been created but there are many duplicate values so on Product_id column give a right click and choose “Remove Duplicates”. A final Table has been created as Shown Below and Rename the table as Products. As we can see in the below fig() the Final table has been created.

The screenshot shows the Power Query Editor interface again. The 'Products' table is now displayed, showing unique rows for each product. The 'Applied Steps' pane on the right highlights the 'Removed Duplicates' step. The 'Properties' pane shows the table is named 'Products'. The main table view shows the final, cleaned dataset.

As you can see in the above Fig() the Product table has been created with 18 Rows. With the column Product_id, Product Name, Product Category.

Creating Orders_Aggregate Table:

1. Go to Data View
2. Click on “New Table”
3. Write DAX Formula: Here I am using SUMMARIZE function as I am creating a table from an existing table.
 - a. Here For the Product Table the columns we needed from Fact_order Table are Product_id, Product_name and Product Category.

Click on Ok symbol then the below fig()

Creating Date Table Using DAX:

1. Go to Data View
2. Click on “New Table”.
3. Write DAX Formula: Creating a Date table with Columns of Date [01/03/2022-03/09/2022], Month and Week.

The screenshot shows the Power BI Desktop interface with the 'Table tools' ribbon selected. A blue box highlights the DAX code in the formula bar:

```

1 Table <= Date
2 ADDCOLUMNS(
3 CALENDAR(DATE(2022, 3, 1), DATE(2022, 3, 30)),
4 "Month", FORMAT([Date], "mm_y"),
5 "week", WEEKNUMBER([Date]))
    
```

The Fields pane on the right lists various tables: Customers, Order_lines, Orders_Aggregate, Products, Table, and targets_orders. The 'Table' table is currently selected.

4. Click on the tick mark then the Date table is created, and Filtered data is shown like below

The screenshot shows the Power BI Desktop interface with the 'Table tools' ribbon selected. The 'Date' table has been created with 187 rows, as indicated by the status bar at the bottom. The table structure includes columns for Date, Month, and week. The data shows dates from March 1, 2022, to April 2, 2022, with corresponding month and week numbers.

Date	Month	Week
01/03/2022 00:00:00	Mar_22	10
02/03/2022 00:00:00	Mar_22	10
03/03/2022 00:00:00	Mar_22	10
04/03/2022 00:00:00	Mar_22	10
05/03/2022 00:00:00	Mar_22	10
06/03/2022 00:00:00	Mar_22	11
07/03/2022 00:00:00	Mar_22	11
08/03/2022 00:00:00	Mar_22	11
09/03/2022 00:00:00	Mar_22	11
10/03/2022 00:00:00	Mar_22	11
11/03/2022 00:00:00	Mar_22	11
12/03/2022 00:00:00	Mar_22	11
13/03/2022 00:00:00	Mar_22	12
14/03/2022 00:00:00	Mar_22	12
15/03/2022 00:00:00	Mar_22	12
16/03/2022 00:00:00	Mar_22	12
17/03/2022 00:00:00	Mar_22	12
18/03/2022 00:00:00	Mar_22	12
19/03/2022 00:00:00	Mar_22	12
20/03/2022 00:00:00	Mar_22	13
21/03/2022 00:00:00	Mar_22	13
22/03/2022 00:00:00	Mar_22	13
23/03/2022 00:00:00	Mar_22	13
24/03/2022 00:00:00	Mar_22	13
25/03/2022 00:00:00	Mar_22	13
26/03/2022 00:00:00	Mar_22	13
27/03/2022 00:00:00	Mar_22	14
28/03/2022 00:00:00	Mar_22	14
29/03/2022 00:00:00	Mar_22	14
30/03/2022 00:00:00	Mar_22	14
31/03/2022 00:00:00	Mar_22	14
01/04/2022 00:00:00	Apr_22	14
02/04/2022 00:00:00	Apr_22	14
03/04/2022 00:00:00	Apr_22	15
04/04/2022 00:00:00	Apr_22	15

The Fields pane on the right lists various tables: Customers, Order_lines, Orders_Aggregate, Products, and targets_orders. The 'Date' table is currently selected.

5. As you can see in the above Fig() the Date table has been created with 187 Rows. With the column Date, Month, Week.

6. Removing Columns:

1. Go to Report View
2. Click on Transform Data.
3. Select the Columns to remove from the Order_line table they are Customer Name, Customer City, Product Name, and Product Category. Click Ctrl+Select the columns as shown in the below Image.

The screenshot shows the Power Query Editor interface with the 'fact_order_lines' query selected. In the main grid, several columns are listed, including 'set_id', 'order_qty', 'actual_delivery_date', 'customer_name', 'customer_city', 'product_name', and 'product_category'. A selection box is drawn around the four columns: 'Customer Name', 'Customer City', 'Product Name', and 'Product Category'. The 'Applied Steps' pane on the right shows a step named 'Changed Type'.

4. Give a right click where you can see the “Remove Columns” option Select it and the columns are removed now as shown in the below image.

The screenshot shows the Power Query Editor interface. A query named 'fact_order_line' is open, displaying a table with 8 columns. The columns are: order_id, order_placement_date, customer_id, product_id, order_qty, agreed_delivery_date, actual_delivery_date, and delivery_qty. The 'dim_targets_orders' table is also visible in the background. The 'APPLIED STEPS' pane on the right shows the step 'Removed Columns'.

Now there are only 8 Columns in the Order_Line Table. Now Click on “Close and Apply”.

DATA MODELLING:

Before creating the tables the data modelling the data looks like the fig shown below:

The screenshot shows the Power BI Data Model view. It displays two tables: 'Order_lines' and 'targets_orders'. The 'Order_lines' table has columns: actual_delivery_date, agreed_delivery_date, Customer_City, Customer_id, Customer_Name, delivery_qty, Order_id, Order_placement_date, and Order_qty. The 'targets_orders' table has columns: customer_id, infull_target%, ontime_target%, and ofl_target%. The Properties pane on the right shows settings for the cards, including 'Show the database in the header when applicable' set to 'No'.

Now we have one big Order_lines table which is the fact table. Now we need to create dimensional tables to make the database understandable and simple.

Fact Table: In simpler words, this table contains the Metrics and Measures of our Business problems. Below mentioned are the 2 fact tables:

Now we have 2 fact tables whereas the Orders fact table is split into two tables and adding a prefix as Fact before the table name.

1. Fact_Order_lines
2. Fact_Orders_Aggregate.

Dimensional Table: this table provide context around facts.

Now we have created dimensional tables adding a Prefix as dim to the table name to avoid the confusion with other tables. They are

1. dim_customers
2. dim_products
3. dim_target_orders
4. dim_date

Now after all the tables have been created below fig shows the data model before the relationships are created.

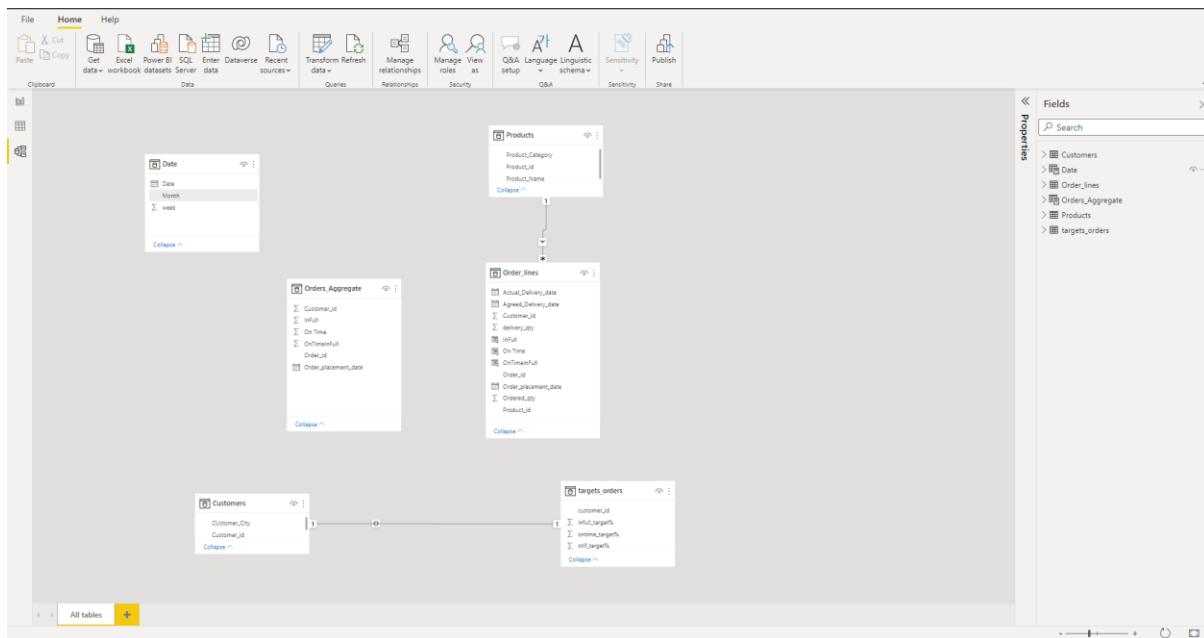


Fig : Data Model before the relationships are created

- Fact_Orders_Aggregate contains the header-level information of the orders.
- Fact_Order_lines contain in-detail information about the order and products.

S.no	Fact table (Order_lines)	Dimensional table
1.	Product_id	Product_id(Product table)
2.	Customer_id	Customer_id(target_orders)
3.	Customer_id	Customer_id (customer table)
4.	Order_placement_date	date (date)

S.no	Fact table (Order_Aggregate)	Dimensional table
1.	Customer_id	Customer_id(target_orders)
2.	Customer_id	Customer_id (customer table)
3.	Order_placement_date	date (date)

S.no	Fact table (Order_Aggregate)	Fact table (Order_lines)
1.	Order_id	Order_id

S.no	Dimensional table(customer)	Dimensional table
1.	Order_id	Order_id

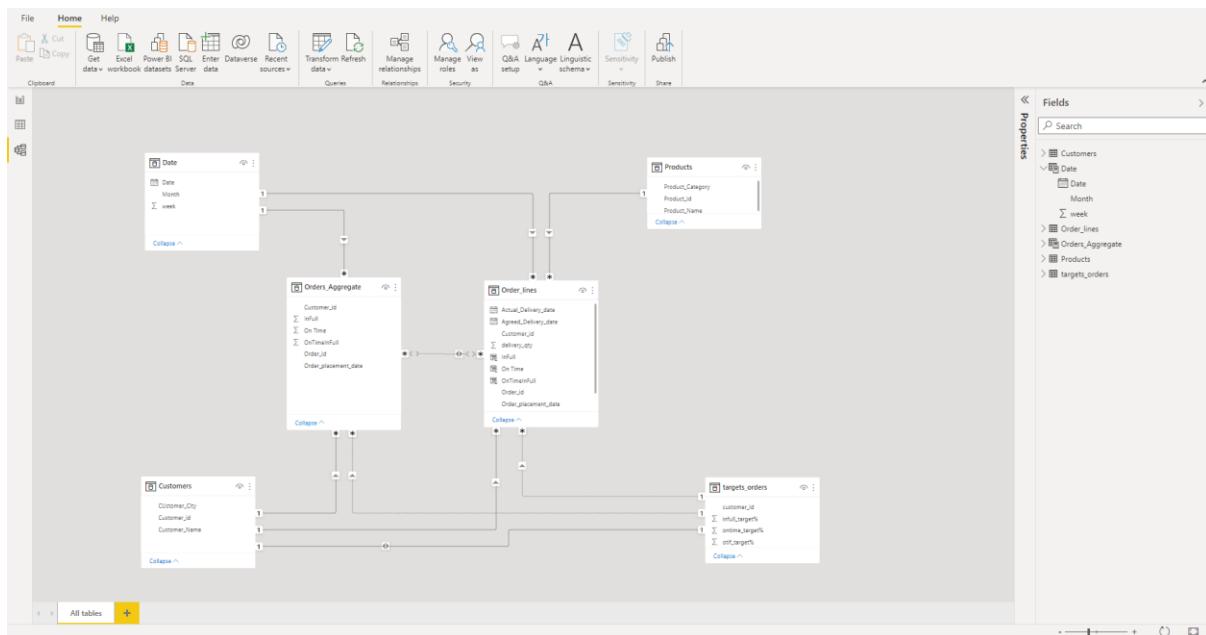
Manage relationships

Active	From: Table (Column)	To: Table (Column)
<input checked="" type="checkbox"/>	dim_targets_orders (customer_id)	dim_customers (customer_id)
<input checked="" type="checkbox"/>	fact_order_lines (customer_id)	dim_customers (customer_id)
<input type="checkbox"/>	fact_order_lines (customer_id)	dim_targets_orders (customer_id)
<input type="checkbox"/>	fact_order_lines (order_id)	fact_orders_aggregate (order_id)
<input checked="" type="checkbox"/>	fact_order_lines (order_placement_date)	dim_date (date)
<input checked="" type="checkbox"/>	fact_order_lines (product_id)	dim_products (product_id)
<input checked="" type="checkbox"/>	fact_orders_aggregate (customer_id)	dim_customers (customer_id)
<input type="checkbox"/>	fact_orders_aggregate (customer_id)	dim_targets_orders (customer_id)
<input checked="" type="checkbox"/>	fact_orders_aggregate (order_placement_date)	dim_date (date)

Fig: Managing the relationships

Star Scheme:

From the below Image we can clearly see the relationship model is Star Scheme whereas all the dimensional tables are connected to the Fact tables. By maintaining the Fact tables in the centre and dimensional tables opposite settled away in the corner from Fact tables clearly depicts it as Star Scheme.



DAX Measures and Calculated Columns:

Calculated Columns:

The below mentioned columns were created using DAX formulae from the columns of table Order_lines.

Sno	Measures	Abbreviation	Description	Table
1.	Total Order Lines		Count of all order lines in Order_lines	Order_lines
2.	On-Time Delivery	OT	No. of orders delivered On Time/ Total no. of Orders.	Order_lines
3.	In Full Delivery	IF	No. of orders delivered in Full quantity/ Total no. of Orders	Order_lines
4.	On-Time in Full	OTIF	No. of orders delivered both IN Full & On Time/ Total no. of Orders.	Order_lines

OT:

Creating a calculated column in Fact_Order_lines table to find out on-time deliveries count. This makes it easier to compare the Agreed delivery date and Actual delivery date and can also compare the delayed days count. Column has been created by using the below DAX function.

OnTime = IF

('Orders Table'[agreed_delivery_date]==='Orders Table'[actual_delivery_date],1,0)

The screenshot shows the 'Order_lines' table in Power BI Desktop. A new calculated column named 'On Time' has been created, defined by the DAX formula: `=IF([Order_lines[Agreed_Delivery_date]]==[Order_lines[Actual_Delivery_date]],1,0)`. The table contains 57,096 rows and 11 columns. The 'On Time' column is of type Whole number and has 2 distinct values (0 and 1). The data includes columns for Order_Id, Order_placement_date, Customer_Id, Product_Id, Ordered_qty, Agreed_Delivery_date, Actual_Delivery_date, delivery_qty, and the newly created 'On Time' column.

Click On “OK”. Then the column is created as shown in the Fig() below.

This screenshot shows the same 'Order_lines' table after the 'On Time' column has been populated with values (0 or 1) based on the DAX formula. The 'On Time' column now has 2 distinct values (0 and 1), indicating whether the actual delivery date matches the agreed delivery date. The rest of the table structure remains the same, with columns for Order_Id, Order_placement_date, Customer_Id, Product_Id, Ordered_qty, Agreed_Delivery_date, Actual_Delivery_date, delivery_qty, and the 'On Time' column.

IF:

Creating a calculated column in Fact_Order_lines table to find out the In full quantity shipped which makes lot easier to compare the quantity ordered and delivered. The Column has been created by using the below DAX function:

1. `InfFull = IF`
2. `('Orders Table'[order_qty] == 'Orders Table'[delivery_qty], 1, 0)`

Untitled - Power BI Desktop

PREETHI MALVALA MALAYALA

File Home Help Table tools Column tools

Summarization Sum

Sort by column Sort by Data groups Group Manage relationships Relationships New column Calculations

Fields

Search

Customer_id Order_lines Actual_Delivery_date Agreed_Delivery_date

Column delivery_qty On Time Order_id Order_placement_date Product_id Products targets_orders

Table: Order_Lines (57,096 rows) Column: Column 1 (distinct values)

The screenshot shows the Power BI desktop interface with a query editor. A calculated column 'InfFull' is being created based on the condition that 'Order_lines'[Ordered_qty] equals 'Order_lines'[Delivery_qty]. The formula is set to 1 if true and 0 if false. The table contains 57,096 rows and one distinct value for the new column.

Click On “OK”. Then the column is created as shown in the Fig() below.

Untitled - Power BI Desktop

PREETHI MALVALA MALAYALA

File Home Help Table tools Column tools

Summarization Sum

Sort by column Sort by Data groups Group Manage relationships Relationships New column Calculations

Fields

Search

Customer_id Order_lines Actual_Delivery_date Agreed_Delivery_date

Column delivery_qty On Time Order_id Order_placement_date Product_id Products targets_orders

Table: Order_Lines (57,096 rows) Column: InfFull (2 distinct values)

The screenshot shows the Power BI desktop interface with the same query editor. The 'InfFull' column has been successfully created and populated with values 1 or 0 based on the condition. The table now has two distinct values in the 'InfFull' column.

OTIF:

Creating a calculated column in the Fact_Order_lines table to find out the order delivery in full and On-time. This helps to find out the count of orders which are delayed in delivery and quantity delivered. Even it help in depth analysis of the orders delayed. Calculated Column has been created by using the below DAX function:

OnTimeInFull = IF('Orders Table'[InFull]==1 && 'Orders Table'[OnTime]=1,1,0)

The screenshot shows the Power BI Desktop interface with the 'Column tools' ribbon selected. A new column 'OnTimeInFull' is being created with the formula: `iif(Order_lines[On Time]==1 && Order_lines[In Full]==1,1,0)`. The Fields pane on the right lists various columns from the 'Order_lines' table, including Customer_id, Order_id, Order_placement_date, Actual_Delivery_date, Agreed_Delivery_date, delivery_qty, and others.

Click On “OK”. Then the column is created as shown in the Fig() below.

The screenshot shows the Power BI Desktop interface with the 'Column tools' ribbon selected. The 'OnTimeInFull' column has been successfully created and populated with binary values (1 or 0) based on the defined DAX formula. The Fields pane on the right lists the columns available for reference.

Measures created using DAX:

S.No	Measure	DAX
1.	One day delay Orders Count	<pre>1day Delayed orders count = CALCULATE (DISTINCTCOUNT (fact_order_lines[order_id]), fact_order_lines[Daydealy] = 1)</pre>

2.	Two day delay Orders Count	2day Delayed orders count = CALCULATE (DISTINCTCOUNT (fact_order_lines[order_id]), fact_order_lines[Daydealy] = 2)
3.	Three day delay Orders Count:	3day Delayed orders count = CALCULATE (DISTINCTCOUNT (fact_order_lines[order_id]), fact_order_lines[Daydealy] = 3)
4.	ADDD:	ADDD = CALCULATE(AVERAGE(fact_order_lines[Daydealy]), fact_order_lines[Daydealy] <> 0, fact_order_lines[Daydealy]<> -1)
5.	Delayed Order:	Delayed Order % = [Delayed orders count]/[Total order lines]*100
6.	Delayed Order count:	Delayed orders count = [1day Delayed orders count] + [2day Delayed orders count] + [3day Delayed orders count]
7.	LIF:	LIF = CALCULATE([Total order lines], fact_order_lines[In Full] = 1)
8.	LIFR:	LIFR % = [LIF]/[Total order lines]
9.	Line Share % Monthly per Delayed Days Status:	Line Share % Monthly per Delayed Days Status = DIVIDE([Total Order Lines], CALCULATE([Total Order Lines], ALLSELECTED(dim_date), ALLSELECTED(fact_order_lines[status])))
10.	Orders full & time:	Orders full & time = CALCULATE([Total Orders], fact_orders_aggregate[in_full]=1, fact_orders_aggregate[on_time]=1)
11.	Orders in Full:	Orders in Full = CALCULATE([Total Orders], fact_orders_aggregate[in_full]=1)
12.	Orders ON Time:	Orders ON Time = CALCULATE([Total Orders], fact_orders_aggregate[on_time]=1)
13.	Target Full & Time:	Target Full & Time = AVERAGE(dim_targets_orders[otif_target%])
14.	Target In_full:	Target In_full = AVERAGE(dim_targets_orders[infull_target%])

15.	Target On_time:	Target On_time = <code>AVERAGE(dim_targets_orders[ontime_target%])</code>
16.	Total order lines:	Total order lines = <code>COUNT(fact_order_lines[order_id])</code>
17.	Total Orders:	Total Orders = <code>DISTINCTCOUNT(fact_orders_aggregate[order_id])</code>
18.	TQO:	TQO = <code>SUM(fact_order_lines[order_qty])</code>
19.	TQS:	TQS = <code>SUM(fact_order_lines[delivery_qty])</code>
20.	VORF:	VORF % = <code>[TQS]/[TQO]</code>

One day delay Orders Count: here we calculate the orders count delayed by a day

Two-day delay Orders Count: here we calculate the orders count delayed by 2 days.

Three-day delay Orders Count: here we calculate the orders count delayed by 3 days.

Delayed Order: Orders which are delayed

Delayed Order count: Count of the orders which are delayed

LIFR: Line-item fill rate

Orders full & time: Count of Orders delivered in full quantity in time

Orders in Full: Count of Orders delivered in full quantity

Orders ON Time: Count of Orders delivered in time

Target Full & Time: Average of in-full and on-time target

Target In_full: Average of in-full target

Target On_time: Average of on-time target

Total order lines: Count of all orderliness in Fact_order_lines table

Total Orders: Count of all the Total orders

TQO: Count of Total Quantity Ordered

TQS: Count of Total Quantity Shipped

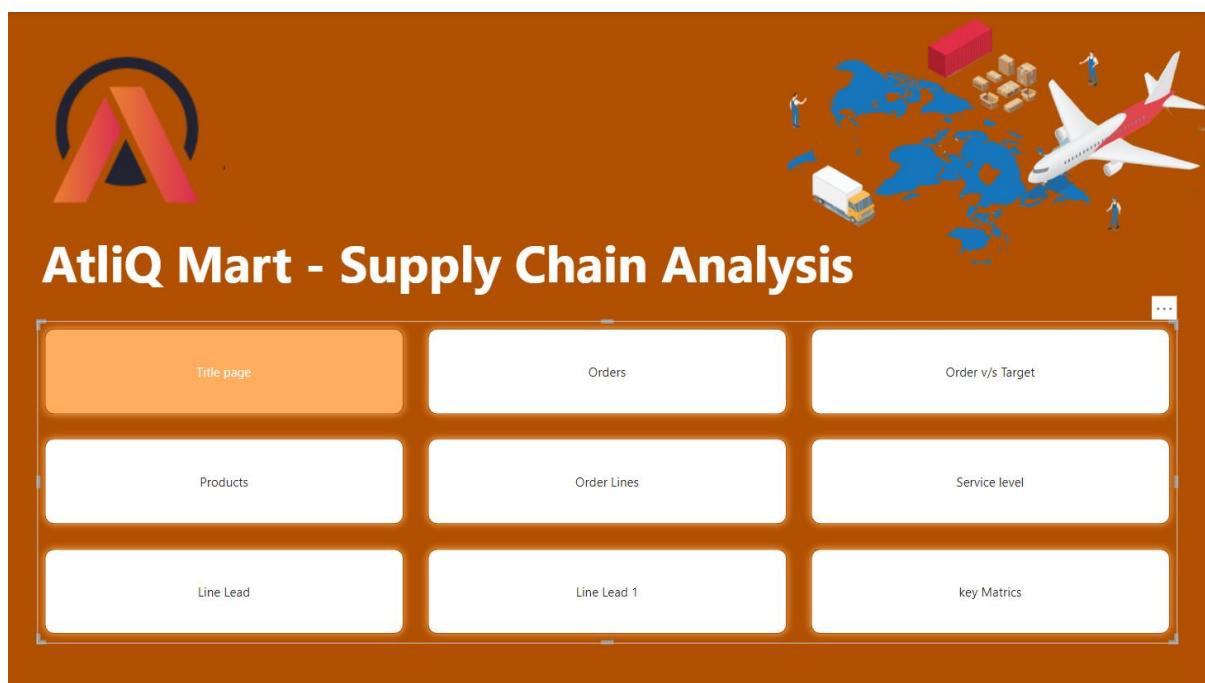
VORF: (Volume Fill Rate)It shows the total quantity ordered

DASHBOARD EXPLORATORY ANALYSIS:

Title Page:

Below fig shows the Title page of the Dashboard which contains the logo of the company. Name of the Project and the Page navigation buttons. Below listed are the pages in the dashboard which I have added in Navigation pane which helps to navigate from one to another.

- Title Page
- Orders
- Orders V/s Target
- Product Analysis
- Order Lines
- Service Level
- Line Lead 1
- Line Lead Analysis
- Key Metrics

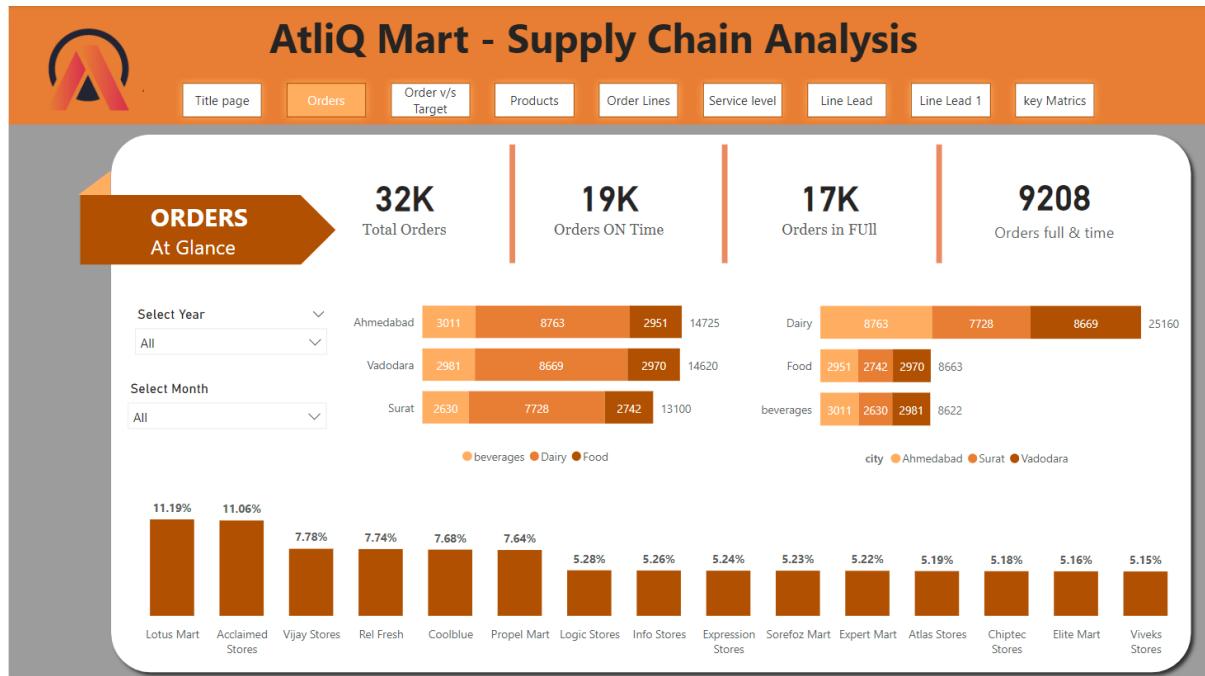


Orders Page:

This page shows the complete orders at a glance. It contains a few cards which show the Total Orders, Orders on time, Orders in Full, Orders in Full, and Time.

Used 2 bar graphs that depict the Orders based on Categories in different cities and the Orders based on cities in different categories.

The last bar graph shows the cities with the highest order count to Lowest.



Fig(): Orders at Glance

Orders V/s Target:

The below page depicts the difference between Actual and target OT%, IF%, and OTIF% which is to check if the company meets the expected target.

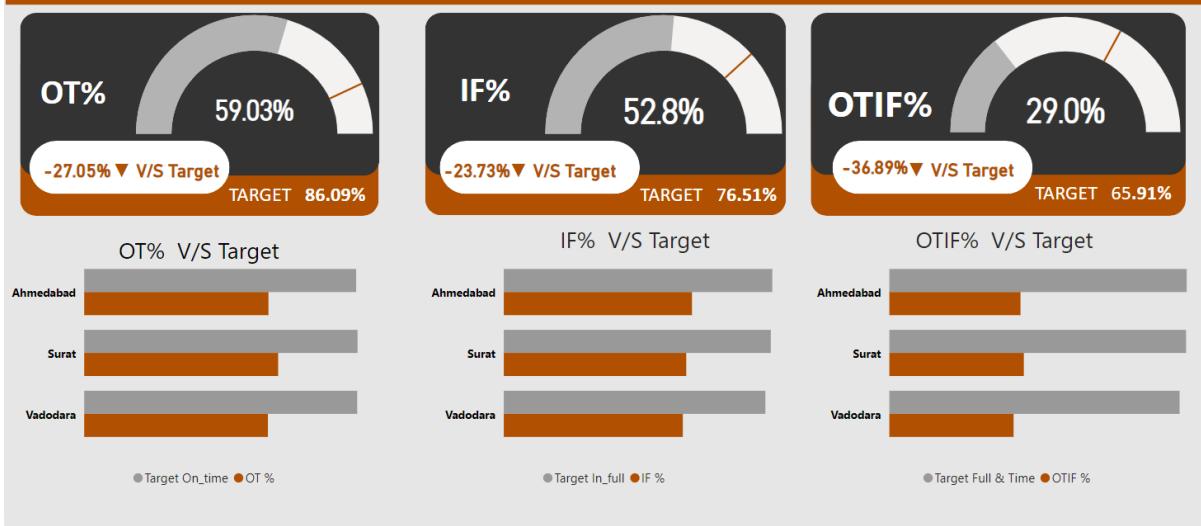
The gauge axis is used to show the actual OT, IF and OTIF values with the respected target values.

A card is used to show the difference between the actual and Target values.

The bar graphs show the difference between OT, IF and OTIF values with the target values based on cities.

AtliQ Mart - Supply Chain Analysis

Title page Orders Order v/s Target Products Order Lines Service level Line Lead Line Lead 1 key Metrics



Order Lines:

This page depicts the entire order lines where the cards used can clearly show the Total Order lines, total order line quantity, total order line quantity delivered, VORF and LIRF.

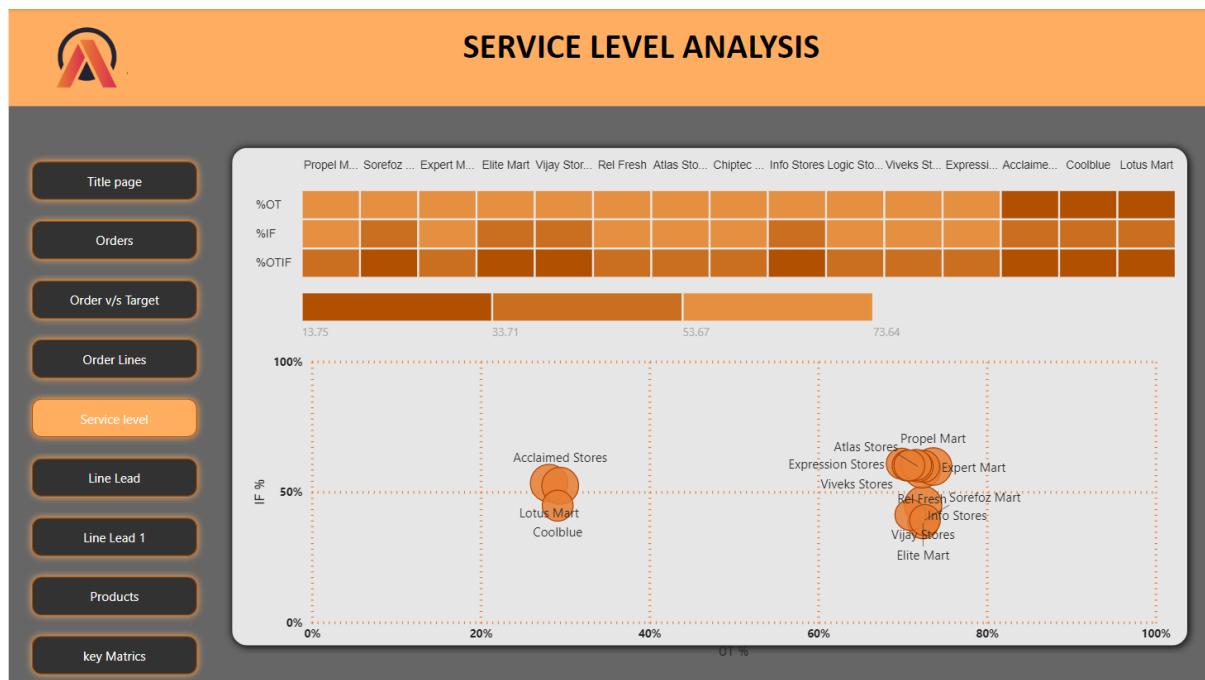
The Line and Clustered column chart here shows customer_name on X-axis, Total order lines on Y-axis and LIRR on Line Y-axis.



Service Level Analysis:

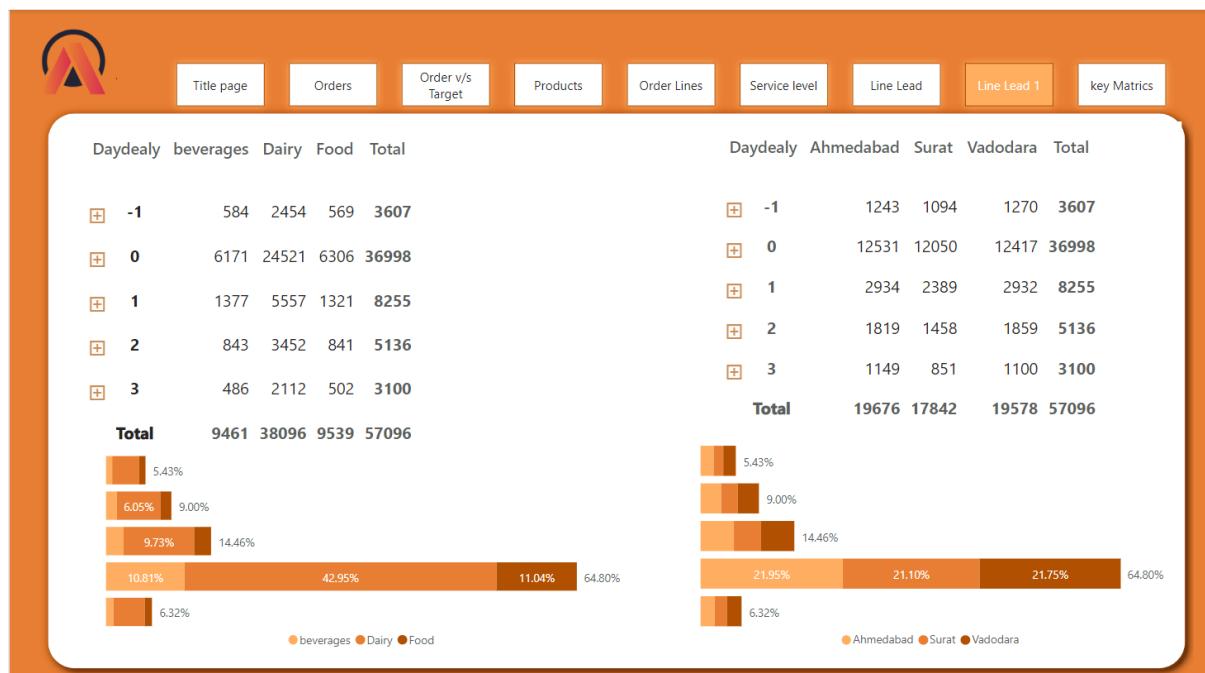
This page depicts the service level analysis based on the Customers against the OT%, IF%, and OTIF%. Table Heat Map here represents the service of the customers from low to high based on the color

code. The scatter chart has the customers as Values and OT% on X-axis and IF% on Y-axis and bubble size depends on the Total Line Order QTY.



Line Lead Time Analysis:

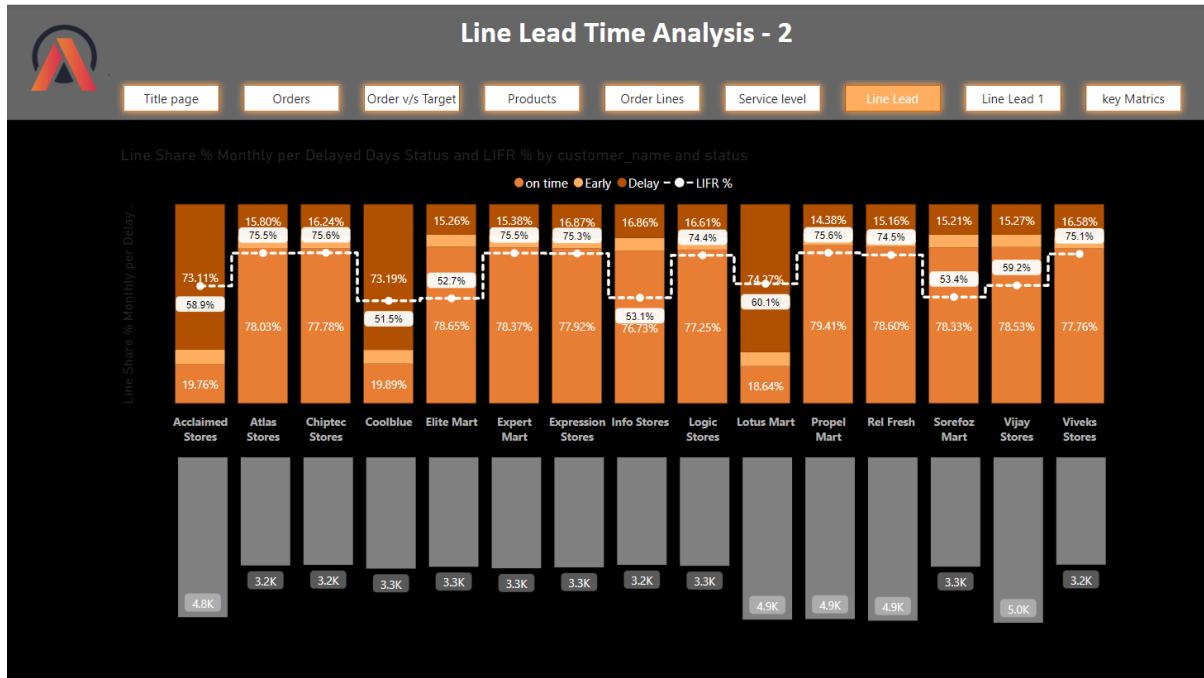
The below page has 2 tables that contain clear data on the orders which are Early, On time and delay in delivery based on the category of food and Cities. Both bar graphs show the percentages of the status of orders based on categories and cities.



Line Lead Time Analysis – 2:

The below page has a Line and stacked column chart which has Customer_name on X-axis and Line share % monthly per delayed day status and LIFR% on Line Y-axis. The legend shows the On time, Early, Delay and LIFR.

The inverted stacked column chart which shows the Customer_name on X-axis and Total Order lines on Y-axis.



Products:

This page shows the matrix table of categories based LIFR,VORF percentages and used Conditional formatting in the table to check the status of LIFR% and VORF% by date in various Categories. The expand/Collapsible icon is placed to show the variety of food discussing in the categories.

The screenshot shows a dashboard titled "AtliQ Mart - Supply Chain Analysis". On the left, there is a vertical navigation menu with the following items:

- Title page
- Orders
- Order v/s Target
- Products
- Order Lines
- Service level
- Line Lead
- Line Lead 1
- key Matrics

The main content area is titled "PRODUCTS At Glance". It contains a table with the following data:

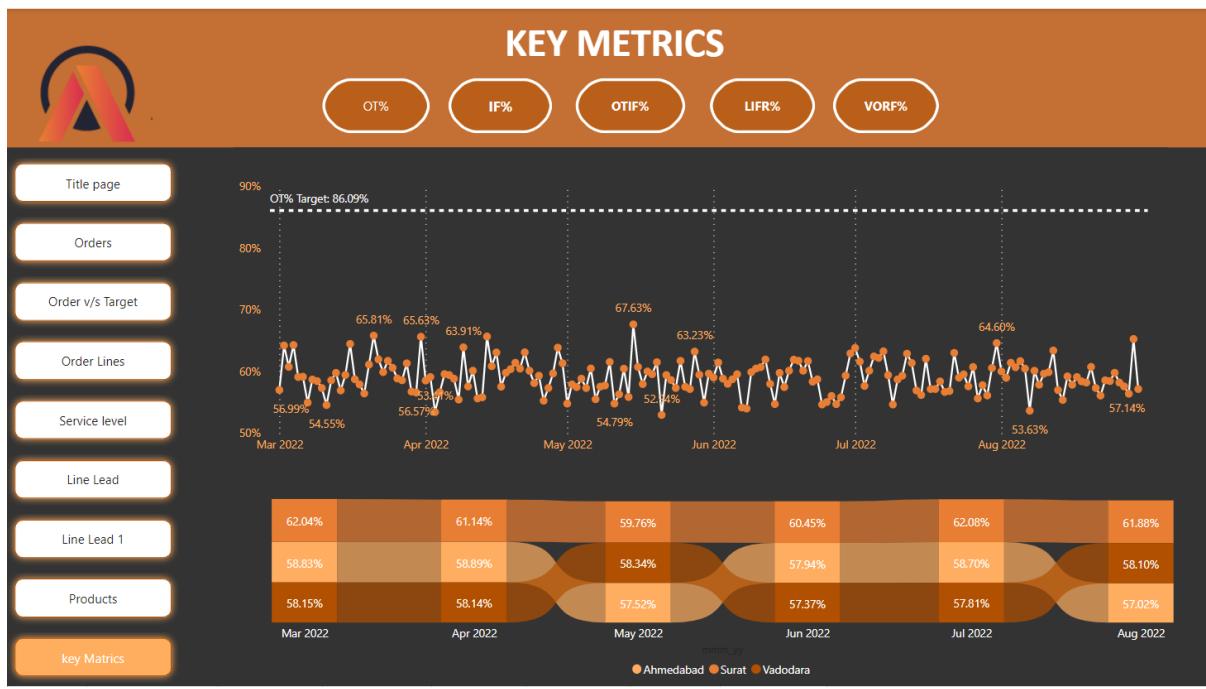
category	LIFR %	VORF %	LIFR % by date	VORF % by date
Food	66.4%	96.6%		
Dairy	65.9%	96.6%		
beverages	65.5%	96.5%		

KEY METRICS:

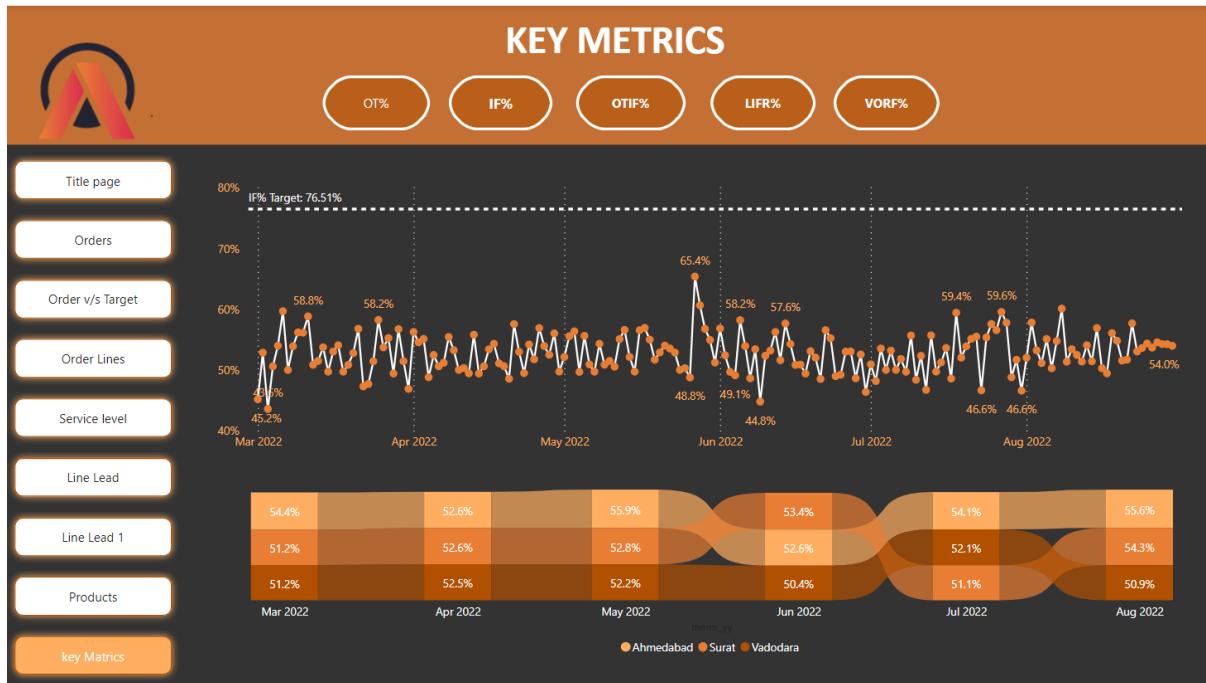
This page depicts the key metrics of OT, IF, OTIF, LIFR and VORF.

To depict the various graphs on one page I have used Buttons that I have bookmarked and added the actions which work for both the line chart and Ribbon Chart. I have used a Line chart here whereas Date on X-axis and the values on Y-axis. The ribbon chart shows the same based on Cities.

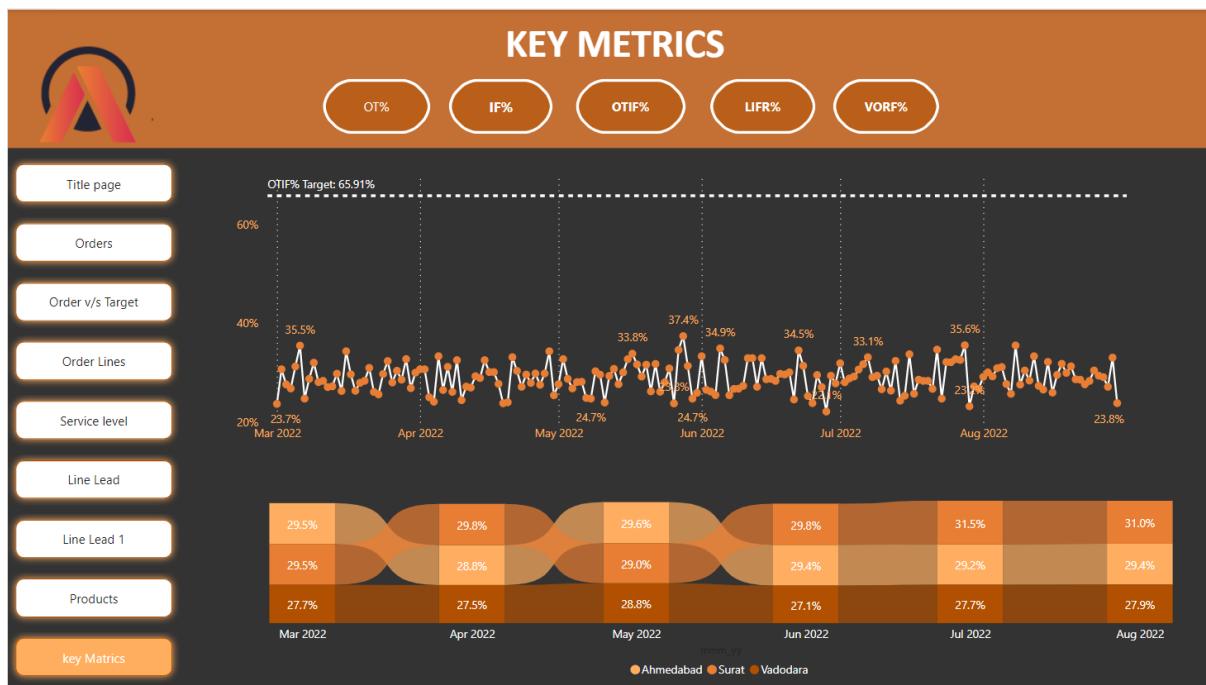
OT%:



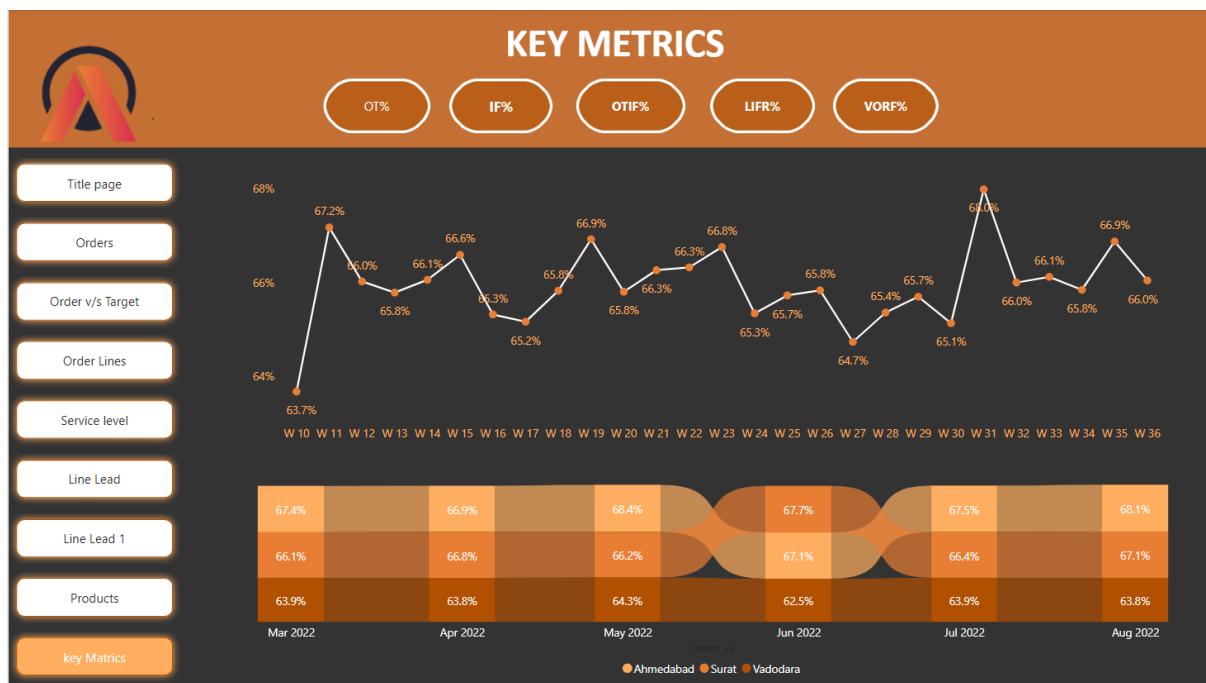
IF%:



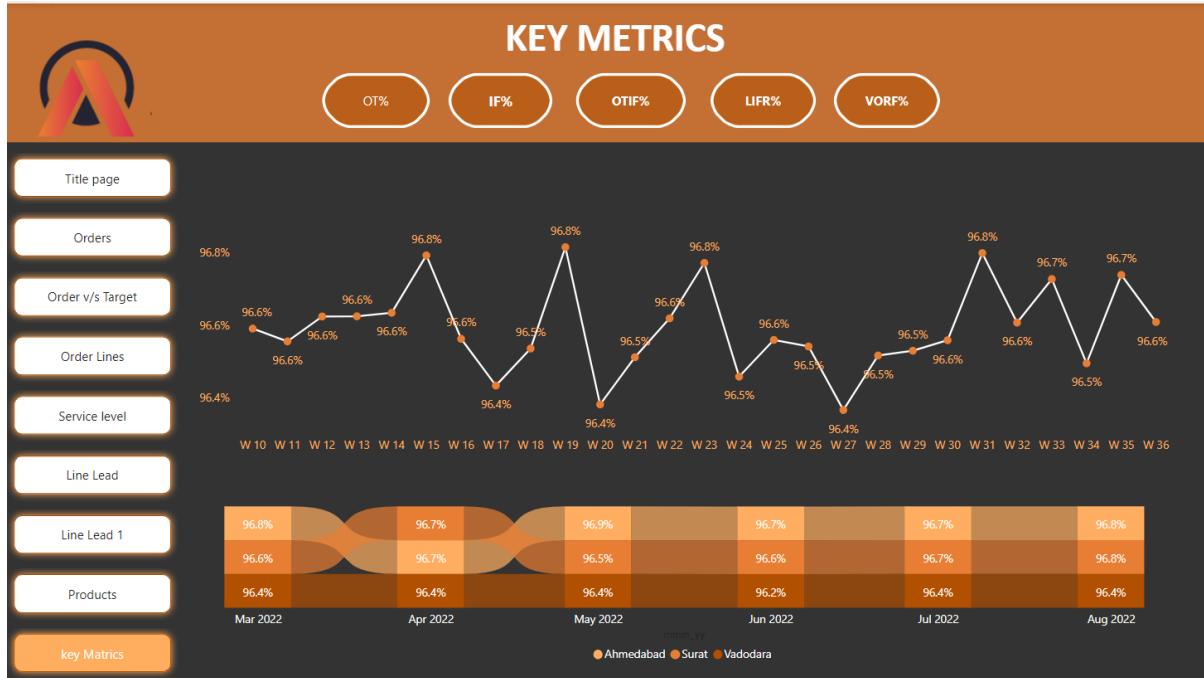
OTIF%:



LIFR%



VORF%



References:

<https://codebasics.io/event/codebasics-resume-project-challenge>

<https://lnkd.in/gfxUiAkc>