

CONTENTS:

- PROBLEM STATEMENT
 - DATA DESCRIPTION
 - OBJECTIVES
 - ANALYSIS
 - QUERY
 - OUTPUT
 - DESCRIPTION
 - INFERENCE
 - CONCLUSION

➤ PROBLEM STATEMENT OF THE PROJECT:

To derive insights for different scenarios from the composite data of a business organization, confined to the 'sales and delivery' domain is given for the period of the last decade.

➤ DATASET: SALES AND DELIVERY

❖ DATA DESCRIPTION

List Of Tables:

1) **Cust_dimen:**

Column Name	Data type	description	NULL
Customer_name	Varchar	Name of the customer	Yes
Province	Varchar	ID for Province	Yes
Region	Varchar	ID for Region	Yes
Customer_Segment	Varchar	Types of the customer segments	Yes
Cust_id	Varchar	Id to the customers	Yes

Field	Type	Null	Key	Default	Extra
Customer_Name	text	YES		NULL	
Province	text	YES		NULL	
Region	text	YES		NULL	
Customer_Segment	text	YES		NULL	
Cust_id	text	YES		NULL	

There are 5 columns in the table and all the rows contain null values.

2) **Market Fact:**

Column Name	Data Types	description	NULL
Ord id	Varchar	Id for the Order	Yes
Prod id	Varchar	Id for the product	Yes
Ship id	Varchar	ID for the shipping	Yes
Cust id	Varchar	ID for the customer	Yes
Sales	Float	The sales price for the product	Yes
Discount	Float	Discount for the product	Yes
Order Quantity	Float	Number of products have been ordered	Yes
Profit	Float	Profit that has been gained from the product	Yes
Shipping cost	Float	Shipping cost for the product	Yes
Product Base Margin	Float	Base margin value for the product	Yes

Field	Type	Null	Key	Default	Extra
Ord_id	text	YES		NULL	
Prod_id	text	YES		NULL	
Ship_id	text	YES		NULL	
Cust_id	text	YES		NULL	
Sales	double	YES		NULL	
Discount	double	YES		NULL	
Order_Quantity	int	YES		NULL	
Profit	double	YES		NULL	
Shipping_Cost	double	YES		NULL	
Product_Base_Margin	double	YES		NULL	

This table contains 10 columns. All the columns contain null values. The table contains columns with datatypes char, float and int.

3) Orders_Dimen:

Column Name	Data type	Description	NULL
Order id	integer	Id for the order	Yes
Order Date	Varchar	The order date for that order has been ordered	Yes
Order Priority	varchar	Priority for the orders	Yes
Ord id	varchar	Order id as a varchar	Yes

Field	Type	Null	Key	Default	Extra
Order_ID	int	YES		NULL	
Order_Date	text	YES		NULL	
Order_Priority	text	YES		NULL	
Ord_id	text	YES		NULL	

This table consists of 4 columns. Three of the columns have char datatype and one column has integer datatype. All the columns contain null values.

4) Prod_Dimen:

Column Name	Data type	Description	NULL
Product_Category	Varchar	Type of the product	Yes
Product_Sub_category	Varchar	Name of the sub-category	Yes
Prod id	Varchar	Id of Product	Yes

Field	Type	Null	Key	Default	Extra
Product_Category	text	YES		NULL	
Product_Sub_Category	text	YES		NULL	
Prod_id	text	YES		NULL	

This table contains 3 columns. All the three have char datatype and contain null values.

5) Shipping Dimen:

Column Name	Data Type	Description	NULL
Order_ID	Varchar	Id for the orders	Yes
Ship_Mode	Varchar	Type of the shipping	Yes
Ship_Date	Varchar	Shipping date	Yes
Ship_ID	Varchar	ID for the shipping	Yes

Field	Type	Null	Key	Default	Extra
Order_ID	int	YES		NULL	
Ship_Mode	text	YES		NULL	
Ship_Date	text	YES		NULL	
Ship_id	text	YES		NULL	

This table contains 4 columns. Three out of four columns have text values and one column has integer datatype.

Changes made in the tables:

- In the table order_dimen, Changed the datatype of column, order_date from text to date.
- In the table shipping_dimen, Changed the datatype of column, order_date from text to date.

```
ALTER TABLE orders_dimen MODIFY Order_Date DATE;  
ALTER TABLE shipping_dimen MODIFY Ship_Date DATE;
```

- Created new table called Delivery and populated it with data from the orders_dimen and shipping_dimen. In the table delivery, created a new column, daystakenfordelivery that contains the date difference between Order_Date and Ship_Date.

```
CREATE TABLE Delivery AS (SELECT od.order_id,  
    od.order_date,  
    od.ord_id,  
    sd.ship_id,  
    sd.ship_date FROM  
    orders_dimen od  
    INNER JOIN  
    shipping_dimen sd ON od.ORDER_ID = sd.ORDER_ID);
```

```
ALTER TABLE Delivery ADD COLUMN DaysTakenForDelivery INT AS (DATEDIFF(ship_date,order_date)) ;
```

1) Objective:

To find the top 3 customers who have the maximum number of orders.

Query:

```

SELECT
    cd.customer_name, COUNT(mf.ord_id) AS number_of_orders
FROM
    cust_dimen cd
    INNER JOIN
    market_fact mf ON cd.cust_id = mf.cust_id
GROUP BY cd.cust_id
ORDER BY number_of_orders DESC
LIMIT 3;

```

Output:

cust_id	customer_name	number_of_orders
Cust_1140	PATRICK JONES	30
Cust_572	LENA CREIGHTON	21
Cust_444	BILL DONATELLI	21

Description:

This query uses an inner join between the customer dimension table and the market fact table on the common field of the customer ID. The COUNT function is used to count the number of order IDs in the market_fact table that match each customer ID. GROUP BY groups the result set by the customer ID to ensure that (COUNT) is applied to each customer separately.

Inference:

The top three customers with maximum number of orders are Patrick, Lena, and Bill with 30,21 and 21 orders respectively.

this SQL query is helpful in providing valuable insights for sales and marketing strategies.

2) Objective:

Find the customer whose order took the maximum time to get delivered.

Query:

```

SELECT
    c.Cust_id,
    c.Customer_Name,
    d.order_id,
    d.DaysTakenForDelivery
FROM
    delivery d
    INNER JOIN
    market_fact m ON d.ord_id = m.ord_id
    INNER JOIN
    cust_dimen c ON m.Cust_id = c.Cust_id
ORDER BY DaysTakenForDelivery DESC
LIMIT 1:

```

Output:

	Cust_id	Customer_Name	order_id	DaysTakenForDelivery
►	Cust_1460	DEAN PERCER	353	92

Description:

Here we have used inner join between the delivery table and the market fact table on the common field of the order ID (ord_id). It then uses another inner join with the customer dimension table. The ORDER BY clause sorts the result set in descending order of the number of days taken for delivery. 'LIMIT' clause limits the result set to only the first row.

Inference:

The customer whose order took the maximum time to get delivered is Dean Percer, who had to wait for 92 days to get his order delivered. This analysis can be used for further analysis to identify any issues in the delivery process and improve customer satisfaction.

3) Objective:

To retrieve total sales made by each product from the data.

Query:

```

select distinct prod_id,
    sum(sales) OVER(partition by prod_id ) as total_sales
from market_fact order by total_sales DESC;

```

Output:

Number of rows: 17

prod_id	total_sales		
Prod_17	2168697.14	Prod_8	795875.94
Prod_4	1889313.8	Prod_2	736991.54
Prod_11	1786776.75	Prod_5	698093.81
Prod_15	1652823	Prod_6	446452.86
Prod_14	1130361.3	Prod_9	174085.8
Prod_1	1028240.76	Prod_13	167107.22
Prod_3	1022957.59	Prod_16	80996.31
Prod_10	814425.9	Prod_12	38981.55
		Prod_7	15006.63

Description:

Here, DISTINCT keyword is used to retrieve only unique product IDs from the market fact table. SUM is used to calculate the total sales for each product ID by partitioning the result set by product ID using the OVER clause. This creates a separate group of rows for each product ID, and the SUM function is applied to each group to calculate the total sales.

Inference:

- From the output the product with product id 17 has made the maximum sales of more than 21 lakhs. The product with the least sales id product 7, which made a sale of only 15 thousand.
- This query is helps in analysing the sales performance of different products and identifying the products with the highest total sales.

4) Objective:

To retrieve the total profit made from each product from the data.

Query:

```
SELECT DISTINCT prod_id,  
ROUND(SUM(profit) OVER(PARTITION BY prod_id),2) AS total_product_profit  
FROM market_fact order by total_product_profit desc;
```

Output:

Number of rows: 17

prod_id	total_product_profit
Prod_4	316951.62
Prod_17	307712.93
Prod_3	307413.39
Prod_14	167361.49
Prod_15	122738.07
Prod_5	100427.93
Prod_2	97158.06
Prod_8	94287.48
Prod_9	48182.6
Prod_6	45263.2
Prod_12	13677.17

Prod_1	13599.49
Prod_13	7564.78
Prod_7	-102.67
Prod_16	-7799.25
Prod_10	-33729.09
Prod_11	-113468.18

Description:

In this query we used SUM with the OVER clause to calculate the total profit for each product ID by partitioning the result set by product ID. This creates a separate group of rows for each product ID, and the SUM function is applied to each group to calculate the total profit.

Inference:

From the result it can be seen that product 4 did the highest profit of more than 3 lakhs whereas product 7, product 16, product 10 and product 11 lead losses where product 11 was seen to have highest loss of more than 1 lakh . It is evident that product 4 and product 17 are the products with highest sales and profit. It can also be observed that product 10 has fairly high sales but is still undergoing losses. This might be due to high-cost price and low selling price.

5) Objective:

To find the total number of unique customers in January and how many of them came back every month over the entire year in 2011.

Query:

```

AND year(olqel_dafes)=2011)?
WHERE month(olqel_dafes)=1
FROM compitueq_fapje
IN (2EFEC1 DIZIIMCI cufz_fq
WHERE year(olqel_dafes)=2011 AND cufz_fq
FROM compitueq_fapje
COUNT(cufz_fq) OVER(PARTITION BY month(olqel_dafes) ORDER BY month(olqel_dafes)) AS total_unipue_cufzomewerz
2EFEC1 qizfzucf year(olqel_dafes) * month(olqel_dafes)

olqel_dafes ON olqel_dafes = olqel_dafes
WHERE 2011
WHERE 2011
WHERE 2011
WHERE 2011
FROM
2EFEC1 cufzomewer_uame_cufz_fq*olqel_dafes*olqel_dafes
CREATE VIEW compitueq_fapje as(

```


Output:

Year(order_date)	Month(order_date)	Total_Unique_Customers
2011	1	154
2011	2	13
2011	3	14
2011	4	7
2011	5	5
2011	6	7
2011	7	8
2011	8	5
2011	9	6
2011	10	14
2011	11	14
2011	12	11

Description:

The first query creates a view called combined table which combines data from three tables: cust_dimen, market_fact, and orders_dimen. It joins cust_dimen and market_fact on the cust_id column and market_fact and orders_dimen on the ord_id column.

The result is a combined view of the relevant data from these three tables, which is used in subsequent queries. The second query retrieves the total number of unique customers for each month in the year 2011, using a window function and a subquery.

The SELECT statement selects distinct Year and Month values from the order_date column of the combined_table and calculates the Total_Unique_Customers for each month. The count(cust_id) OVER(PARTITION BY month(order_date) order by month(order_date)) calculates the count of unique cust_id values for each month using a window function, and assigns the result to the Total_Unique_Customers column.

The WHERE clause filters the result to only include records from the year 2011 and where the cust_id values are present in the subquery. The subquery selects distinct cust_id values from the combined_table where the month is January and the year is 2011.

Takeaway and Conclusion:

It's seen that a total of 154 customers costomers in January. Out of these customers, minimum 5 customers visit every month. There are 14 customers in January who visited in march, October and November. These are observed to be the highest number of repetitive customers.

➤ CONCLUSION:

From the above analysis, the important insights which can be derived are:

- The top three customers in the business organisations are the customers with id, Cust_1140, Cust_572 and Cust_444. Namely, Patrick Jones, Lena Creighton, and Bill Donatelli with 30,21 and 21 orders respectively. Hence these customers must be retained by maintaining the same service and product quality.
- A customer with the id Cust_1460 and name, DEAN PERCER has experienced maximum time taken for the delivery, which is 92 days. The organisation must make sure to improve the delivery services to customers to maintain customer satisfaction. The organisation must try to give the required benefits to DEAN PERCER to compensate for the delay in the delivery.
- The product with the maximum sales includes product 4 and product 17 which had the maximum profit too. Certain products such as product 10 have high sales but still leading to loss. Since this could be due to very low sale price than the cost price, the business organisation must investigate this product and increase its sales price.
- According to the data, there are 154 customers at the start of the year, that is, in the month of January. Among these customers, there are minimum of 5 customers who have been visiting every month. However, the maximum number of January customers were seen visiting in the month of march, October, and November. The organisation must retain these customers by retaining the services and increase the quality to increase the revisit of customers.