1. Provide the list of markets in which customer "Atliq Exclusive" operates its business in the APAC region.  
  
SELECT DISTINCT customer,market,region

FROM gdb023.dim\_customer

WHERE customer='ATLIQ EXCLUSIVE' AND region='APAC';

OUTCOME:



2. What is the percentage of unique product increase in 2021 vs. 2020? The final output contains these fields, unique\_products\_2020, unique\_products\_2021, percentage\_chg.  
SELECT

COUNT(DISTINCT CASE WHEN fiscal\_year=2020 THEN product\_code end) AS unique\_product\_2020 ,

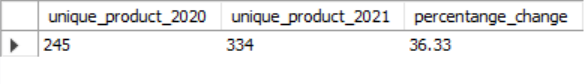
COUNT(DISTINCT CASE WHEN fiscal\_year=2021 THEN product\_code end) AS unique\_product\_2021 ,

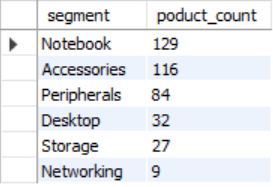
ROUND((COUNT(DISTINCT CASE WHEN fiscal\_year=2021 THEN product\_code end)

-COUNT(DISTINCT CASE WHEN fiscal\_year=2020 THEN product\_code end))\*100

/COUNT(DISTINCT CASE WHEN fiscal\_year=2020 THEN product\_code end),2)   
AS percentange\_change

FROM gdb023.fact\_sales\_monthly;  
OUTCOME:



3. Provide a report with all the unique product counts for each segment and sort them in descending order of product counts. The final output contains 2 fields,  
SELECT segment,COUNT(DISTINCT product\_code) AS poduct\_count FROM gdb023.dim\_product   
GROUP BY segment  
ORDER BY poduct\_count DESC;  
OUTCOME:   


4. Which segment had the most increase in unique products in 2021 vs 2020? The final output contains these fields, segment product\_count\_2020, product\_count\_2021 difference.  
  
SELECT p.segment,

COUNT(DISTINCT CASE WHEN fiscal\_year=2020 THEN p.product\_code END) AS unique\_product\_2020,

COUNT(DISTINCT CASE WHEN fiscal\_year=2021 THEN p.product\_code END) AS unique\_product\_2021,

(COUNT(DISTINCT CASE WHEN fiscal\_year=2021 THEN p.product\_code END)-

COUNT(DISTINCT CASE WHEN fiscal\_year=2020 THEN p.product\_code END))/

COUNT(DISTINCT CASE WHEN fiscal\_year=2020 THEN p.product\_code END)

AS increase\_percentage

FROM gdb023.dim\_product p JOIN gdb023.fact\_sales\_monthly fs

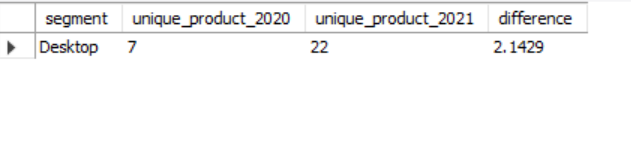
ON p.product\_code=fs.product\_code

GROUP BY p.segment

ORDER BY increase\_percentage DESC

LIMIT 1 ;

OUTCOME:



5. Get the products that have the highest and lowest manufacturing costs. The final output should contain these fields, product\_code ,product ,manufacturing\_cost.  
  
SELECT p.product\_code,product,manufacturing\_cost

FROM gdb023.fact\_manufacturing\_cost mc

JOIN gdb023.dim\_product p

ON mc.product\_code=p.product\_code

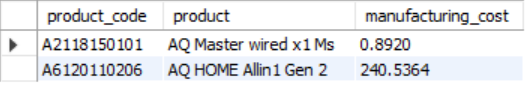
WHERE manufacturing\_cost= (SELECT max(manufacturing\_cost)

FROM gdb023.fact\_manufacturing\_cost)

OR

manufacturing\_cost= (SELECT min(manufacturing\_cost)

FROM gdb023.fact\_manufacturing\_cost ) ;  
  
OUTCOME:



6. Generate a report which contains the top 5 customers who received an average high pre\_invoice\_discount\_pct for the fiscal year 2021 and in the Indian market. The final output contains these fields, customer\_code,   
customer average,\_discount\_percentage.  
   
SELECT c.customer\_code,customer,pre\_invoice\_discount\_pct

FROM gdb023.dim\_customer c

JOIN gdb023.fact\_pre\_invoice\_deductions id

ON c.customer\_code= id.customer\_code

WHERE market='india'AND fiscal\_year=2021

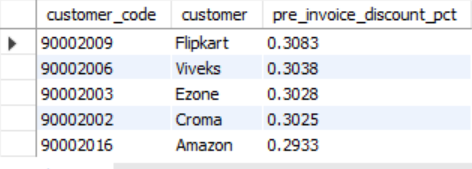
HAVING pre\_invoice\_discount\_pct >

( SELECT AVG(pre\_invoice\_discount\_pct)

FROM gdb023.fact\_pre\_invoice\_deductions

WHERE market='india' AND fiscal\_year=2021 )

ORDER BY pre\_invoice\_discount\_pct DESC

LIMIT 5;  
  
OUTCOME:  


7.Get the complete report of the Gross sales amount for the customer **“Atliq Exclusive”** for each month **.** This analysis helps to get an idea of low and high-performing months and take strategic decisions. The final report contains these columns: Month Year Gross sales Amount   
  
SELECT c.customer,MONTHNAME(fs.date) month,YEAR(fs.date) year,

ROUND(SUM(sold\_quantity\*gross\_price),-2) AS gross\_total\_price

FROM gdb023.fact\_sales\_monthly fs

LEFT JOIN gdb023.dim\_customer c   
ON fs.customer\_code= c.customer\_code

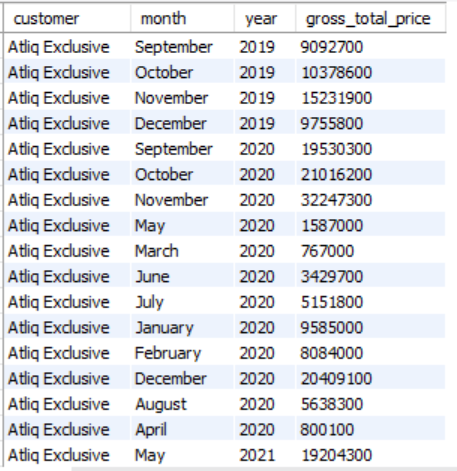
LEFT JOIN gdb023.dim\_product p   
ON fs.product\_code=p.product\_code

LEFT JOIN gdb023.fact\_gross\_price gp   
ON p.product\_code=gp.product\_code

WHERE customer="Atliq Exclusive"

GROUP BY year,month

ORDER BY year,month desc;

OUTCOME:  


8. In which quarter of 2020, got the maximum total\_sold\_quantity? The final output contains these fields sorted by the total\_sold\_quantity, Quarter total\_sold\_quantity.

SELECT SUM(sold\_quantity) Total\_quantity ,

DATE,

CASE

WHEN MONTH(DATE) BETWEEN 9 AND 11 THEN 'Q1'

WHEN MONTH(DATE) IN (12,1,2) THEN 'Q2'

WHEN MONTH(DATE) BETWEEN 3 AND 5 THEN 'Q3'

WHEN MONTH(DATE) BETWEEN 6 AND 8 THEN 'Q4'

END AS QUARTER

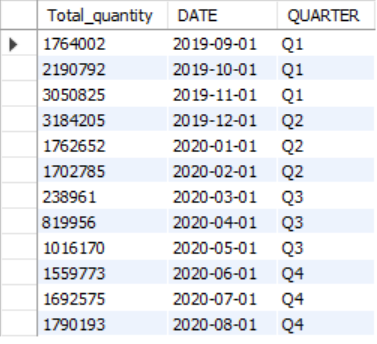
FROM gdb023.fact\_sales\_monthly

WHERE fiscal\_year=2020

GROUP BY QUARTER,DATE

ORDER BY QUARTER,DATE;

OUTCOME:



9. Which channel helped to bring more gross sales in the fiscal year 2021 and the percentage of contribution? The final output contains these fields, channel gross\_sales\_mln ,percentage.

WITH gross\_cte AS(

SELECT c.channel,

SUM(fs.sold\_quantity\*gp.gross\_price)

AS gross\_total\_price

FROM gdb023.fact\_sales\_monthly fs

LEFT JOIN gdb023.fact\_gross\_price gp on fs.product\_code=gp.product\_code

LEFT JOIN gdb023.dim\_customer c on c.customer\_code= fs.customer\_code

GROUP BY c.channel

ORDER BY c.channel DESC)

SELECT channel,gross\_total\_price,

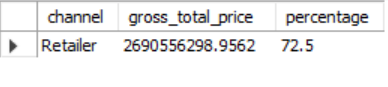
ROUND(gross\_total\_price\*100/ (SELECT SUM(gross\_total\_price) FROM gross\_cte),1)

AS percentage

FROM gross\_cte

LIMIT 1 ;

OUTCOME:



10.Get the Top 3 products in each division that have a high total\_sold\_quantity in the fiscal\_year 2021? The final output contains these fields, division ,product\_code .WITH pro\_cte AS

(SELECT p.division, p.product, p.product\_code, sum(s.sold\_quantity) total\_quantity

FROM gdb023.fact\_sales\_monthly s

LEFT JOIN gdb023.dim\_product p ON p.product\_code=s.product\_code

WHERE s.fiscal\_year=2021

GROUP BY p.division,p.product, p.product\_code

ORDER BY p.division,p.product, p.product\_code )

SELECT division, product\_code, total\_quantity,top3\_rank

FROM ( SELECT pro\_cte.\*,

RANK() OVER (PARTITION BY division ORDER BY total\_quantity DESC) AS top3\_rank

FROM pro\_cte

) ranked

WHERE top3\_rank <=3;  
  
OUTCOME:

