

10 AD HOC QUESTIONS & RELATED SQL QUERIES

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

===

```
SELECT DISTINCT market from dim_customer WHERE customer ="atliq exclusive" AND region ="apac"
```

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

===

```
WITH cte1 AS
(SELECT fiscal_year, COUNT (DISTINCT product_code) AS unique_products_2020
FROM fact_sales_monthly WHERE fiscal_year = 2020),
cte2 AS
(SELECT fiscal_year, COUNT (DISTINCT product_code) AS unique_products_2021
FROM fact_sales_monthly WHERE fiscal_year = 2021)
SELECT cte1.unique_products_2020, cte2.unique_products_2021, (cte2.unique_products_2021 -
cte1.unique_products_2020) / cte1.unique_products_2020 * 100 AS percentage_chg
FROM cte1 CROSS JOIN cte2;
```

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,
segment
product_count

===

```
SELECT segment,COUNT(DISTINCT product_code) AS product_count FROM dim_product GROUP BY segment
ORDER BY product_count DESC
```

4. Follow-up: 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

===

```
WITH cte1 AS(
WITH cte2 AS(
SELECT F.product_code,P.segment,F.fiscal_year FROM fact_sales_monthly AS F JOIN dim_product AS P
ON F.product_code=P.product_code)
SELECT segment,COUNT (DISTINCT CASE WHEN fiscal_year=2020 THEN product_code END) AS
product_count_2020,
COUNT (DISTINCT CASE WHEN fiscal_year=2021 THEN product_code END) AS product_count_2021
FROM cte2 GROUP BY segment)
SELECT segment, product_count_2020, product_count_2021,product_count_2021-product_count_2020 AS
difference FROM cte1 ORDER BY difference DESC LIMIT 1
```

5. Get the products that have the highest and lowest manufacturing costs. The final output should contain these fields,

product_code
product
manufacturing_cost

===

WITH CTE AS

```
(SELECT F.product_code,P.product,F.manufacturing_cost FROM fact_manufacturing_cost AS F
JOIN dim_product AS P ON F.product_code=P.product_code)
SELECT product_code, product, manufacturing_cost FROM CTE
WHERE manufacturing_cost=(SELECT MAX(manufacturing_cost) FROM CTE)
UNION
SELECT product_code, product, manufacturing_cost FROM CTE
WHERE manufacturing_cost=(SELECT MIN(manufacturing_cost) FROM CTE)
```

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

===

WITH CTE AS(

```
SELECT F.customer_code,D.customer,F.pre_invoice_discount_pct,f.fiscal_year,D.market
FROM fact_pre_invoice_deductions AS F JOIN dim_customer AS D ON F.customer_code=D.customer_code)
Retrieve the top 5 customers with the highest average pre_invoice_discount_pct for fiscal year 2021 and in the
Indian market
SELECT customer_code,customer, AVG(pre_invoice_discount_pct) AS average_discount_percentage FROM cte
WHERE fiscal_year = 2021 AND market = 'India'
GROUP BY customer_code, customer
ORDER BY average_discount_percentage DESC LIMIT 5;
```

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

===

WITH CTE AS

```
(SELECT C.customer,F.date, F.fiscal_year,G.gross_price*F.sold_quantity AS Gross_Sales FROM fact_sales_monthly
AS F
JOIN fact_gross_price AS G
ON F.product_code=G.product_code AND F.fiscal_year=G.fiscal_year
JOIN dim_customer AS C ON C.customer_code=F.customer_code)
SELECT (MONTH(DATE) + 3) % 12 + 1 AS Sales_Month,fiscal_year,round(sum(Gross_Sales)/1000000,2) AS
Gross_Sales_Milion
FROM CTE WHERE customer="Atliq Exclusive" GROUP BY Sales_Month, fiscal_year
ORDER BY fiscal_year,Sales_Month
```

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

===

WITH CTE1 AS

**(WITH CTE AS(SELECT MONTH(DATE) AS MNTH,SUM(sold_quantity) AS total_sold_quantity
FROM fact_sales_monthly WHERE fiscal_year=2020 GROUP BY MNTH ORDER BY MNTH)**

SELECT MNTH,total_sold_quantity ,

CASE WHEN MNTH IN (9,10,11) THEN "Q1"

WHEN MNTH IN (12,1,2) THEN "Q2"

WHEN MNTH IN (3,4,5) THEN "Q3"

WHEN MNTH IN (6,7,8) THEN "Q4"

END AS Quater FROM CTE)

SELECT Quater,sum(total_sold_quantity) AS total_sold_quantity FROM CTE1

GROUP BY Quater ORDER BY Quater ASC LIMIT 1

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 CTE1 AS

**(WITH CTE AS(SELECT date, F.product_code, F.customer_code,C.channel, F.sold_quantity*G.gross_price As
Gross_Sales, F.fiscal_year FROM fact_sales_monthly AS F JOIN fact_gross_price AS G ON
F.product_code=G.product_code AND F.fiscal_year=G.fiscal_year**

JOIN dim_customer AS C ON F.customer_code=C.customer_code)

SELECT channel,ROUND(SUM(Gross_Sales)/1000000,2) AS Gross_Sales_Mln FROM CTE

WHERE fiscal_year=2021 GROUP BY channel)

**SELECT channel,Gross_Sales_Mln,(Gross_Sales_Mln/(SELECT SUM(Gross_Sales_Mln) AS Tot_Gross_Sales FROM
CTE1)) AS percentage FROM CTE1**

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,

product

total_sold_quantity

rank_orde

===

```
WITH CTE1 AS
(WITH CTE AS
(SELECT P.division, P.product_code,P.product,F.sold_quantity,F.fiscal_year FROM fact_sales_monthly AS F
JOIN dim_product AS P ON F.product_code=P.product_code WHERE fiscal_year = 2021)
SELECT division, product_code, product,SUM(sold_quantity) AS Total_sold_quantity FROM CTE GROUP BY
division, product_code, product)
SELECT division,product_code,product,total_sold_quantity,rank_order
FROM (SELECT division,product_code,product,total_sold_quantity,
RANK() OVER (PARTITION BY division ORDER BY total_sold_quantity DESC) AS rank_order FROM CTE1)
AS RANK_PRODUCTS WHERE rank_order <= 3
```