



ATLIQ HARDWARE

Consumer Goods Ad_Hoc Insights

S Q L P R O J E C T

Domain: Consumer Goods | Function: Executive Management



ABOUT US

Atliq Hardwares (imaginary company) is one of the leading computer hardware producers in India and well expanded in other countries too.

However, the management noticed that they do not get enough insights to make quick and smart data-informed decisions. They want to expand their data analytics team by adding several junior data analysts. Tony Sharma, their data analytics director wanted to hire someone who is good at both tech and soft skills. Hence, he decided to conduct a SQL challenge which will help him understand both the skills.





Task:

Imagine yourself as the applicant for this role and perform the following task

1. Check 'ad-hoc-requests.pdf' - there are 10 ad hoc requests for which the business needs insights.
2. You need to run a SQL query to answer these requests.
3. The target audience of this dashboard is top-level management - hence you need to create a presentation to show the insights.
4. Be creative with your presentation, audio/video presentation will have more weightage.

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

Input

```
SELECT  
    distinct(market)  
FROM  
    dim_customer  
WHERE  
    customer = "Atliq Exclusive"  
    AND region = "APAC";
```

Output

market
India
Indonesia
Japan
Philippines
South Korea
Australia
Newzealand
Bangladesh

-- 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

Input

```
WITH cte20 AS
(
  SELECT COUNT(DISTINCT (product_code)) AS unique_products_2020
  from fact_sales_monthly
  where fiscal_year = 2020
),
cte21 as
(
  SELECT COUNT(DISTINCT (product_code)) AS unique_products_2021
  from fact_sales_monthly
  where fiscal_year = 2021
)

SELECT
  *,
  ROUND((unique_products_2021 - unique_products_2020) * 100 / unique_products_2020,
        2) AS percentage_chg
FROM cte20
cross join cte21;
```

Output

	unique_products_2020	unique_products_2021	percentage_chg
▶	245	334	36.33

- 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

Input

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

Output

Result Grid | Filter Rows:

	segment	product_count
▶	Notebook	129
	Accessories	116
	Peripherals	84
	Desktop	32
	Storage	27
	Networking	9

- 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

Input

```

WITH cte20 AS
  (SELECT
    p.segment,
    COUNT(DISTINCT (s.product_code)) AS product_count_2020
    FROM
      fact_sales_monthly s
    JOIN
      dim_product p ON s.product_code = p.product_code
    WHERE
      fiscal_year = 2020
    GROUP BY p.segment),
  
  cte21 AS
  (SELECT
    p.segment,
    COUNT(DISTINCT (p.product_code)) AS product_count_2021
    FROM
      fact_sales_monthly s
    JOIN
      dim_product p ON p.product_code = s.product_code
    WHERE
      fiscal_year = 2021
    GROUP BY p.segment)

  SELECT
    segment,
    product_count_2020,
    product_count_2021,
    (product_count_2021 - product_count_2020) AS difference
    FROM
      cte20
    JOIN
      cte21 USING (segment)
    ORDER BY difference DESC;
  
```

Output

	segment	product_count_2020	product_count_2021	difference
▶	Accessories	69	103	34
	Notebook	92	108	16
	Peripherals	59	75	16
	Desktop	7	22	15
	Storage	12	17	5
	Networking	6	9	3

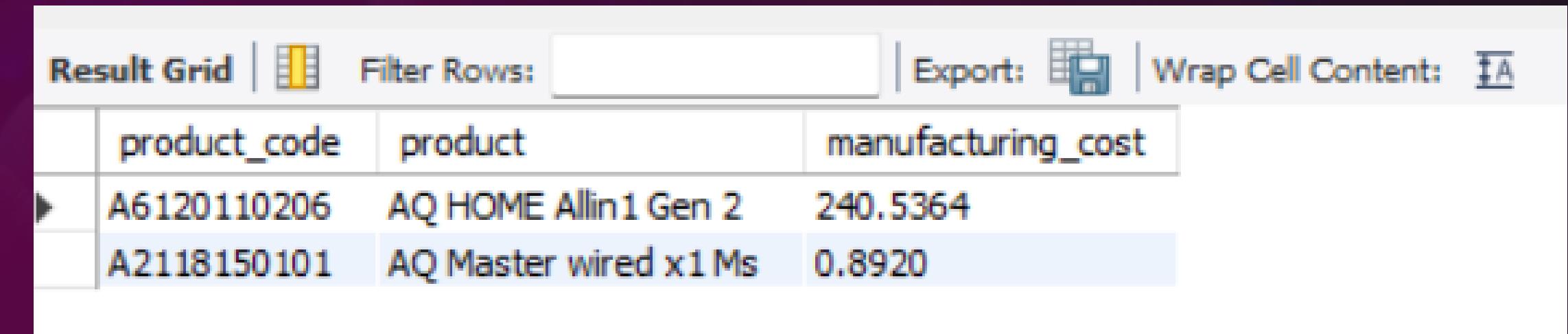
-- Get the products that have the highest and lowest manufacturing costs.
-- The final output should contain these fields, product_code, product, manufacturing_cost

Input

```
WITH CTE1 AS
(
  SELECT
    m.product_code, p.product, m.manufacturing_cost
    FROM
      fact_manufacturing_cost m
    JOIN
      dim_product p ON m.product_code = p.product_code
    ORDER BY m.manufacturing_cost DESC
    LIMIT 1),
CTE2 AS
(
SELECT
  m.product_code, p.product, m.manufacturing_cost
    FROM
      fact_manufacturing_cost m
    JOIN
      dim_product p ON m.product_code = p.product_code
    ORDER BY m.manufacturing_cost ASC
    LIMIT 1)

SELECT
  *
FROM
  CTE1
UNION ALL SELECT
  *
FROM
  CTE2;
```

Output



A screenshot of a database result grid. The grid has three columns: 'product_code', 'product', and 'manufacturing_cost'. The first row shows the highest manufacturing cost product, and the second row shows the lowest manufacturing cost product.

product_code	product	manufacturing_cost
A6120110206	AQ HOME Allin1 Gen 2	240.5364
A2118150101	AQ Master wired x1 Ms	0.8920

-- 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

Input

```

SELECT
    pi.customer_code,
    c.customer,
    ROUND(AVG(pi.pre_invoice_discount_pct), 4) AS average_discount_percentage
FROM
    fact_pre_invoice_deductions pi
        JOIN
    dim_customer c ON pi.customer_code = c.customer_code
WHERE
    fiscal_year = 2021 AND market = 'India'
GROUP BY pi.customer_code , c.customer
ORDER BY average_discount_percentage DESC
LIMIT 5;
    
```

Output

	customer_code	customer	average_discount_percentage
▶	90002009	Flipkart	0.3083
	90002006	Viveks	0.3038
	90002003	Ezone	0.3028
	90002002	Croma	0.3025
	90002016	Amazon	0.2933

- 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

Input

```

SELECT
    MONTHNAME(s.date) AS month,
    s.fiscal_year AS year,
    ROUND(SUM(g.gross_price * s.sold_quantity) / 1000000,
          2) AS gross_sales_amount_mln
FROM
    fact_sales_monthly s
        JOIN
    dim_customer c ON s.customer_code = c.customer_code
        JOIN
    fact_gross_price g ON s.product_code = g.product_code
        AND s.fiscal_year = g.fiscal_year
WHERE
    c.customer LIKE '%Atliq Exclusive%'
GROUP BY month, year
ORDER BY year;
  
```

Output

Result Grid | Filter Rows:

	month	year	gross_sales_amount_mln
▶	September	2020	4.50
	October	2020	5.14
	November	2020	7.52
	December	2020	4.83
	January	2020	4.74
	February	2020	4.00
	March	2020	0.38
	April	2020	0.40
	May	2020	0.78
	June	2020	1.70

Result 1 ×

- 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

Input

```
SELECT  
CASE  
WHEN MONTH(date) IN (9,10,11) THEN 'Q1'  
WHEN MONTH(date) IN (12,1,2) THEN 'Q2'  
WHEN MONTH(date) IN (3,4,5) THEN 'Q3'  
ELSE 'Q4'  
END as Quarter,  
sum(sold_quantity) as total_sold_quantity  
From fact_sales_monthly  
where fiscal_year = 2020  
group by Quarter  
order by total_sold_quantity desc;
```

Output

	Quarter	total_sold_quantity
	Q1	7005619
	Q2	6649642
	Q4	5042541
	Q3	2075087

-- 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

Input

```
WITH ctel AS
(
  SELECT
    c.channel,
    ROUND(SUM(g.gross_price * s.sold_quantity) / 1000000,
          2) AS gross_sales_mln
    FROM
      fact_sales_monthly s
    JOIN
      fact_gross_price g ON s.product_code = g.product_code
      AND s.fiscal_year = g.fiscal_year
    JOIN
      dim_customer c ON s.customer_code = c.customer_code
    WHERE
      s.fiscal_year = 2021
    GROUP BY c.channel
)
```

```
SELECT *,
concat(round((gross_sales_mln*100)/sum(gross_sales_mln) over(), 2),"") AS percentage
FROM ctel
ORDER BY gross_sales_mln DESC;
```

Output

channel	gross_sales_mln	percentage
Retailer	1219.08	73.23%
Direct	257.53	15.47%
Distributor	188.03	11.30%

-- 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_order

Input

```

WITH cte AS
(
    SELECT
        p.division,
        s.product_code,
        p.product,
        SUM(s.sold_quantity) AS total_sold_quantity
    FROM
        fact_sales_monthly s
    JOIN
        dim_product p ON s.product_code = p.product_code
    WHERE
        s.fiscal_year = 2021
    GROUP BY p.division, s.product_code, p.product
),
cte1 AS
(
    SELECT *,
        DENSE_RANK() OVER(PARTITION BY division ORDER BY total_sold_quantity DESC) AS rank_order
    FROM cte
)
SELECT *
FROM
    cte1
WHERE
    rank_order <= 3

```

Output

	division	product_code	product	total_sold_quantity	rank_order
	N & S	A6720160103	AQ Pen Drive 2 IN 1	701373	1
	N & S	A6818160202	AQ Pen Drive DRC	688003	2
	N & S	A6819160203	AQ Pen Drive DRC	676245	3
	P & A	A2319150302	AQ Gamers Ms	428498	1
	P & A	A2520150501	AQ Maxima Ms	419865	2
	P & A	A2520150504	AQ Maxima Ms	419471	3
	PC	A4218110202	AQ Digit	17434	1
	PC	A4319110306	AQ Velocity	17280	2
	PC	A4218110208	AQ Digit	17275	3