



Provide Insights to Management on Consumer Goods

Domain: Consumer Goods |

Function: Executive Management

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.

### Ad hoc Request - 1

```
1
2 -- Qus. 1
3 -- Provide the list of markets in which customer "Atliq Exclusive" operates its
4 -- business in the APAC region.
5
6 SELECT
7     *
8 FROM
9     dim_customer
10 WHERE
11     customer = 'Atliq Exclusive'
12     AND region = 'APAC';
```

#datawithharsh

Ad hoc Request - 2

```
1 -- Qus. 2
2 -- What is the percentage of unique product increase in 2021 vs. 2020? The
3 -- final output contains these fields,
4 -- unique_products_2020
5 -- unique_products_2021
6 -- percentage_chg
7
8 WITH
9     Unique_products_2020 AS (
10     SELECT
11         COUNT(DISTINCT product_code) as count
12     FROM
13         fact_sales_monthly
14     WHERE
15         fiscal_year = 2020
16     ),
17     Unique_products_2021 AS (
18     SELECT
19         COUNT(DISTINCT product_code) as count
20     FROM
21         fact_sales_monthly
22     WHERE
23         fiscal_year = 2021
24     ),
25     Percentage_change AS (
26     SELECT
27         (((Unique_products_2021.count - Unique_products_2020.count) /
28         Unique_products_2020.count) * 100) AS Percentage_Chng
29     FROM
30         Unique_products_2020,
31         Unique_products_2021
32     )
33     SELECT
34         Unique_products_2020.count AS unique_product_2020,
35         Unique_products_2021.count AS unique_product_2021,
36         Percentage_change.Percentage_Chng
37     FROM
38         Unique_products_2020,
39         Unique_products_2021,
40         Percentage_change;
```

### Ad hoc Request - 3

```
1
2 -- Qus. 3
3 -- Provide a report with all the unique product counts for each segment and
4 -- sort them in descending order of product counts. The final output contains
5 -- 2 fields,
6 -- segment
7 -- product_count
8
9 SELECT
10     segment, COUNT(DISTINCT product_code) AS product_count
11 FROM
12     dim_product
13 GROUP BY segment
14 ORDER BY product_count DESC;
15
```



## Ad hoc Request - 4

```
1
2 -- Qus. 4
3 -- Follow-up: Which segment had the most increase in unique products in
4 -- 2021 vs 2020? The final output contains these fields,
5 -- segment
6 -- product_count_2020
7 -- product_count_2021
8 -- difference
9
10 WITH
11     f_2020 AS (
12         SELECT segment, product_code
13         FROM dim_product
14         JOIN fact_sales_monthly USING(product_code)
15         WHERE fiscal_year = 2020
16     ),
17     f_2021 AS (
18         SELECT segment, product_code
19         FROM dim_product
20         JOIN fact_sales_monthly USING(product_code)
21         WHERE fiscal_year = 2021
22     ),
23     f_2020_agg AS (
24         SELECT segment, COUNT(DISTINCT product_code) as product_code_2020
25         FROM f_2020
26         GROUP BY segment
27     ),
28     f_2021_agg AS (
29         SELECT segment, COUNT(DISTINCT product_code) as product_code_2021
30         FROM f_2021
31         GROUP BY segment
32     )
33     SELECT
34         f_2020_agg.segment,
35         f_2020_agg.product_code_2020,
36         f_2021_agg.product_code_2021,
37         (f_2021_agg.product_code_2021 - f_2020_agg.product_code_2020) as difference
38     FROM
39         f_2020_agg
40     JOIN
41         f_2021_agg USING(segment)
42     ORDER BY difference DESC;
43
```

### Ad hoc Request - 5

```
1 -- Que. 5
2 -- Get the products that have the highest and lowest manufacturing costs.
3 -- The final output should contain these fields,
4 -- product_code
5 -- product
6 -- manufacturing_cost
7
8 SELECT
9     product_code,product,manufacturing_cost
10 FROM dim_product
11 JOIN fact_manufacturing_cost USING(product_code)
12 WHERE manufacturing_cost IN (
13     SELECT MAX(manufacturing_cost) FROM fact_manufacturing_cost
14     UNION
15     SELECT MIN(manufacturing_cost) FROM fact_manufacturing_cost
16 );
17
```

### Ad hoc Request - 6

```
1
2  -- Qus. 6
3  -- Generate a report which contains the top 5 customers who received an
4  -- average high pre_invoice_discount_pct for the fiscal year 2021 and in the
5  -- Indian market. The final output contains these fields,
6  -- customer_code
7  -- customer
8  -- average_discount_percentage
9
10 SELECT
11     dim_customer.customer_code,
12     customer,
13     round(((pre_invoice_discount_pct)*100),2) AS average_discount_percentage
14 FROM
15     fact_pre_invoice_deductions
16     JOIN dim_customer ON fact_pre_invoice_deductions.customer_code =
dim_customer.customer_code
17 WHERE
18     fiscal_year = 2021 AND market = 'India'
19 GROUP BY customer_code, customer
20 ORDER BY average_discount_percentage DESC
21 LIMIT 5;
22
```

### Ad hoc Request - 7

```
1
2 -- Qus. 7
3 -- Get the complete report of the Gross sales amount for the customer "Atliq
4 -- Exclusive" for each month. This analysis helps to get an idea of low and
5 -- high-performing months and take strategic decisions.
6 -- The final report contains these columns:
7 -- Month
8 -- Year
9 -- Gross sales Amount
10
11 SELECT
12     EXTRACT(month FROM fact_sales_monthly.date) AS Month,
13     EXTRACT(year FROM fact_sales_monthly.date) AS Year,
14     ROUND(SUM((gross_price * sold_quantity)), 2) AS gross_sales_amount
15 FROM
16     fact_sales_monthly
17 JOIN   dim_customer USING(customer_code)
18 JOIN   fact_gross_price USING(product_code)
19 WHERE
20     dim_customer.customer = "Atliq Exclusive"
21 GROUP BY Month, Year
22 ORDER BY
23     Year ASC,
24     Month ASC;
25
```



### Ad hoc Request - 8

```
1 -- Qus. 8
2 -- In which quarter of 2020, got the maximum total_sold_quantity? The final
3 -- output contains these fields sorted by the total_sold_quantity,
4 -- Quarter
5 -- total_sold_quantity
6
7
8 WITH
9     Quarters AS (
10     SELECT *,
11     CASE
12         WHEN MONTH(date) IN (9,10,11) THEN 'Q1'
13         WHEN MONTH(date) IN (12,1,2) THEN 'Q2'
14         WHEN MONTH(date) IN (3,4,5) THEN 'Q3'
15         WHEN MONTH(date) IN (6,7,8) THEN 'Q4'
16     END AS Quarter
17 FROM fact_sales_monthly
18 WHERE fiscal_year = 2020
19 )
20
21 SELECT Quarter, SUM(sold_quantity) AS total_sold_quantity
22 FROM Quarters
23 GROUP BY Quarter
24 ORDER BY total_sold_quantity DESC;
25
```

### Ad hoc Request - 9

```
1 -- Qus. 9
2 -- Which channel helped to bring more gross sales in the fiscal year 2021
3 -- and the percentage of contribution? The final output contains these fields,
4 -- channel
5 -- gross_sales_mln
6 -- percentage
7
8 WITH channel_gross AS
9 (
10     SELECT
11         dim_customer.channel,
12         ROUND(SUM(gross_price * sold_quantity), 2) AS gross_sales_mln
13     FROM fact_sales_monthly
14     JOIN dim_customer ON fact_sales_monthly.customer_code = dim_customer.customer_code
15     JOIN fact_gross_price ON fact_sales_monthly.product_code = fact_gross_price.product_code
16     WHERE fact_sales_monthly.fiscal_year = 2021
17     GROUP BY dim_customer.channel
18     ORDER BY gross_sales_mln DESC
19 )
20 SELECT
21     channel,
22     gross_sales_mln,
23     ROUND((gross_sales_mln * 100 / SUM(gross_sales_mln) over()), 3) AS Percentage
24 FROM channel_gross;
25
26
```

Ad hoc Request - 10

```
1 -- Qus 10:
2 -- Get the Top 3 products in each division that have a high
3 -- total_sold_quantity in the fiscal_year 2021? The final output contains these fields,
4 -- division
5 -- product_code
6 -- product
7 -- total_sold_quantity
8 -- rank_order
9
10
11 WITH
12     division_sales AS (
13         SELECT
14             dp.division,
15             fsm.product_code,
16             dp.product,
17             SUM(fsm.sold_quantity) AS total_sold_quantity,
18             Rank() OVER(partition by dp.division ORDER BY SUM(fsm.sold_quantity) DESC) AS
Rank_order
19         FROM fact_sales_monthly fsm
20         JOIN dim_product dp ON fsm.product_code = dp.product_code
21         WHERE fsm.fiscal_year = 2021
22         GROUP BY dp.division, fsm.product_code, dp.product
23     )
24 SELECT
25     division_sales.division,
26     division_sales.product_code,
27     division_sales.product,
28     division_sales.total_sold_quantity,
29     division_sales.rank_order
30 FROM division_sales
31 WHERE division_sales.rank_order <= 3;
```