SQL Challenge:

Provide Insights to Management in Consumer Goods Domain

Domain: Consumer Goods | **Function:** Executive Management

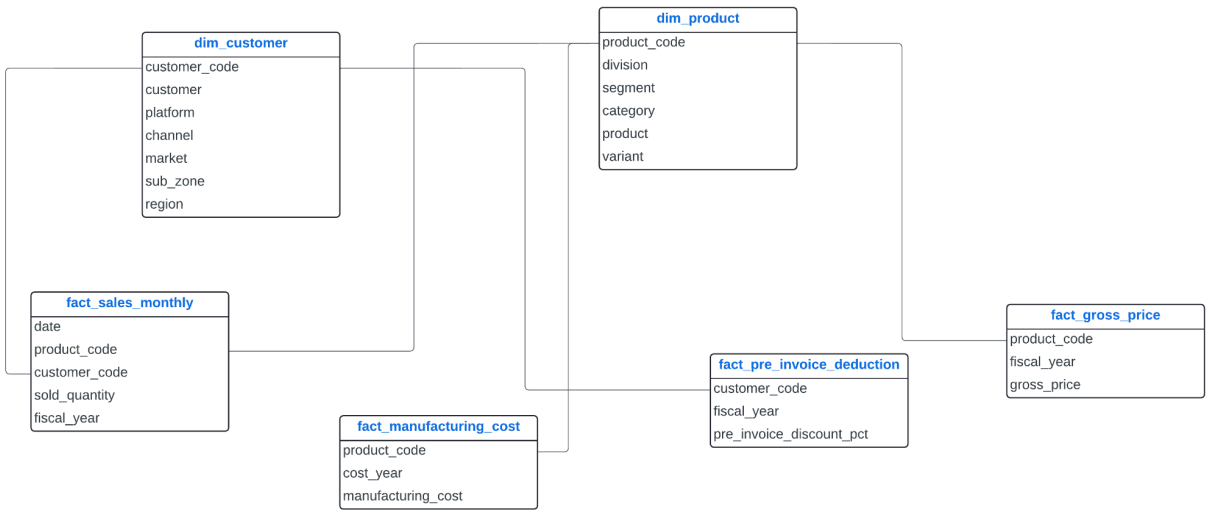
Problem Statement :

Atliq Hardwares 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.

The case study revolves around six key datasets:

- dim_customer
- dim_product
- fact_gross_price
- fact_manufacturing_cost
- fact_pre_invoice_deductions
- fact_sales_monthly

Entity Relationship Diagram



Case Study Questions and Solutions

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

   Query :

   ```
   select distinct(market) from dim_customer
   where customer = "Atliq Exclusive" and region = 'APAC';
   ```

   Output :

market
India
Indonesia
Japan
Philiphines
South Korea
Australia
Newzealand
Bangladesh

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 and percentage_chg**

   Query :

   ```
   SELECT distinct(unique_products_count(2020)) as unique_products_2020,
   unique_products_count(2021) as unique_products_2021,
   ((unique_products_count(2021)-unique_products_count(2020))/
   unique_products_count(2020))*100 as percentage_chg
   FROM gdb023.fact_gross_price;
   ```

   Output :

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

unique_products_2020	unique_products_2021	percentage_chg
245	334	36.3265

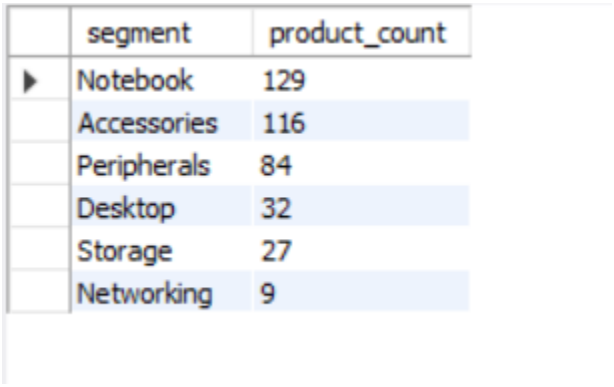
3. **Provide a report with all the unique product counts for each segment and sort them in descending order of product counts.**

The final output contains 2 fields, segment and product_count

Query:

```
select segment,count(distinct(product_code)) as product_count
from gdb023.dim_product
group by segment
order by count(product_code) desc;
```

Output :



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

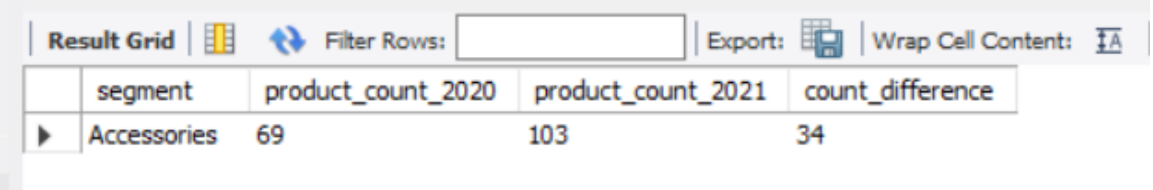
4. **Follow-up: Which segment had the most increase in unique products in 2021 vs 2020? The final output contains these fields, segment product_count_2020 product_count_2021 and difference**

Query :

```
   SELECT
   COALESCE(p1.segment, p2.segment) AS segment,
   COALESCE(product_count_2020, 0) AS product_count_2020,
   COALESCE(product_count_2021, 0) AS product_count_2021,
   COALESCE(product_count_2021, 0) - COALESCE(product_count_2020, 0) AS
count_difference
FROM
   (SELECT
      p.segment, COUNT(fg.product_code) AS product_count_2020
   FROM
      dim_product p
   INNER JOIN
      fact_gross_price fg ON p.product_code = fg.product_code
         AND fg.fiscal_year = 2020
   GROUP BY
      segment) AS p1
LEFT JOIN
   (SELECT
```

```
    p.segment, COUNT(fg.product_code) AS product_count_2021
  FROM
    dim_product p
  INNER JOIN
    fact_gross_price fg ON p.product_code = fg.product_code
      AND fg.fiscal_year = 2021
  GROUP BY
    segment) AS p2 ON p1.segment = p2.segment
ORDER BY count_difference DESC
LIMIT 1;
```

Output :



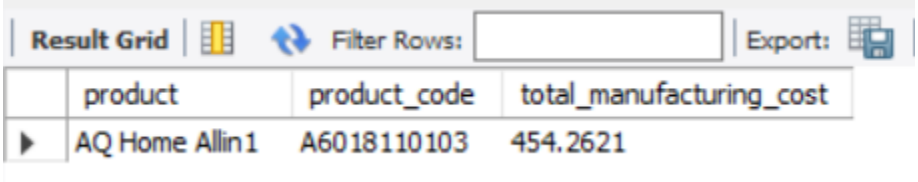
segment	product_count_2020	product_count_2021	count_difference
Accessories	69	103	34

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

Query:

```
select p.product, p.product_code, sum(f.manufacturing_cost) as
total_manufacturing_cost
from fact_manufacturing_cost f inner join dim_product p
on f.product_code = p.product_code
group by p.product_code
order by total_manufacturing_cost desc
limit 1;
```

Output :



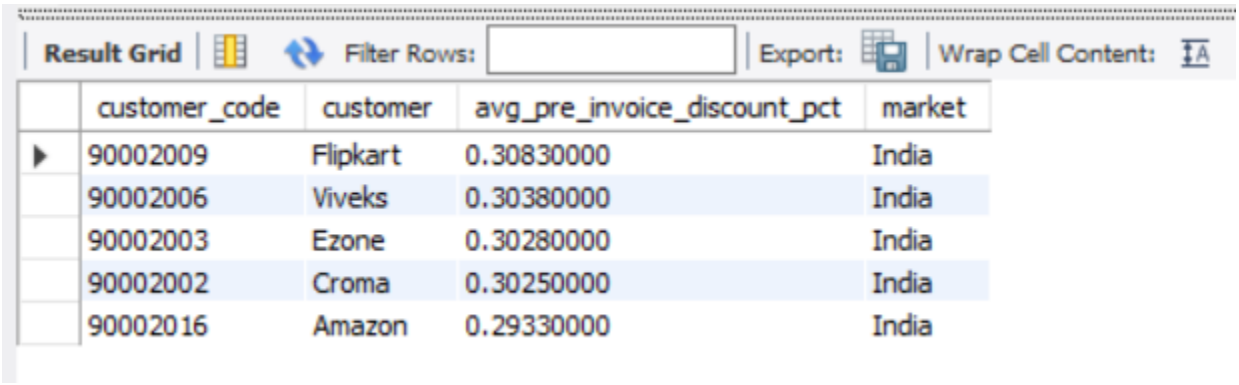
product	product_code	total_manufacturing_cost
AQ Home Allin1	A6018110103	454.2621

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 and average_discount_percentage**

Query:

```
select pre.customer_code,c.customer,avg(pre.pre_invoice_discount_pct) as
avg_pre_invoice_discount_pct,c.market
from fact_pre_invoice_deductions pre
inner join dim_customer c
on pre.customer_code = c.customer_code and c.market = "India" and fiscal_year = 2021
group by pre.customer_code
order by avg_pre_invoice_discount_pct desc
limit 5;
```

Output :



customer_code	customer	avg_pre_invoice_discount_pct	market
90002009	Flipkart	0.30830000	India
90002006	Viveks	0.30380000	India
90002003	Ezone	0.30280000	India
90002002	Croma	0.30250000	India
90002016	Amazon	0.29330000	India

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 and Amount**

Query:

```
select fs.product_code,c.customer_code,c.customer,MONTH(fs.date) as _month,
fs.fiscal_year , sum(gross_price * sold_quantity)  as gross_sales_amount
from  fact_gross_price fg inner join
fact_sales_monthly fs
on fg.product_code = fs.product_code and fs.fiscal_year = fg.fiscal_year
inner join dim_customer c on fs.customer_code = c.customer_code and  c.customer =
"Atliq Exclusive"
group by fs.product_code ;
```

Output:

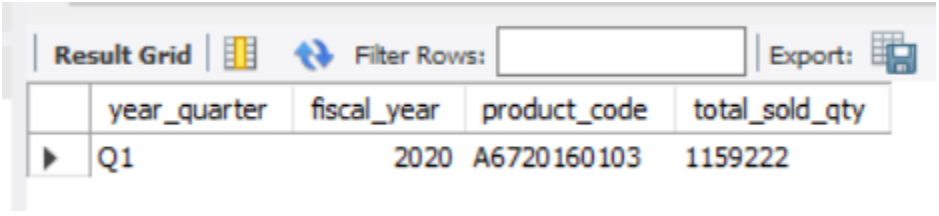
product_code	customer_code	customer	_month	fiscal_year	gross_sales_amount
A0118150101	90002011	Atliq Exclusive	9	2020	482164.5976
A0118150102	90002011	Atliq Exclusive	9	2020	571993.6858
A0118150103	90002011	Atliq Exclusive	9	2020	554842.8770
A0118150104	70002017	Atliq Exclusive	9	2020	577697.3800
A0219150201	70002017	Atliq Exclusive	9	2020	578326.8536
A0219150202	70002017	Atliq Exclusive	9	2020	672717.6288
A0220150203	70002017	Atliq Exclusive	9	2020	625706.7708
A0320150301	70002017	Atliq Exclusive	9	2020	583251.5946
A0321150302	70002017	Atliq Exclusive	9	2021	456734.7004
A0321150303	70002017	Atliq Exclusive	9	2021	434371.5090
A0418150101	70002017	Atliq Exclusive	9	2020	98814.8448
A0418150102	70002017	Atliq Exclusive	9	2020	109445.0084
A0418150103	70002017	Atliq Exclusive	9	2020	384545.4646
A0418150104	70002017	Atliq Exclusive	9	2020	395791.5940
A0418150105	70002017	Atliq Exclusive	9	2020	403629.8916
A0418150106	70002017	Atliq Exclusive	9	2020	381887.5280
A0418150107	70002017	Atliq Exclusive	9	2020	131142.6900
A0418150108	70002017	Atliq Exclusive	9	2020	137599.7952
A0519150201	70002017	Atliq Exclusive	9	2020	462953.8332
A0519150202	70002017	Atliq Exclusive	9	2020	475143.1836
A0519150203	70002017	Atliq Exclusive	9	2020	493899.1540
A0519150204	70002017	Atliq Exclusive	9	2020	486215.6472

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 and total_sold_quantity**

Query:

```
select get_quarter(date) as year_quarter,fiscal_year, product_code,sum(sold_quantity)
as total_sold_qty
from fact_sales_monthly
group by product_code
having fiscal_year = 2020
order by total_sold_qty desc
limit 1;
```

Output:



year_quarter	fiscal_year	product_code	total_sold_qty
Q1	2020	A6720160103	1159222

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 and percentage**

Query :

```
SELECT
   fs.product_code,
   c.channel,
   ROUND((SUM(fs.sold_quantity * fg.gross_price)) / 1000000, 2) AS
gross_sales_amt_mln,
   ROUND((SUM(fs.sold_quantity * fg.gross_price)) / SUM(SUM(fs.sold_quantity *
fg.gross_price)) OVER (PARTITION BY fs.product_code) * 100, 2) AS
channel_contribution_percentage
FROM
   fact_sales_monthly fs
INNER JOIN
   fact_gross_price fg ON fs.product_code = fg.product_code AND fs.fiscal_year = 2021
INNER JOIN
   dim_customer c ON c.customer_code = fs.customer_code
GROUP BY
   fs.product_code, c.channel;
```

Output :

product_code	channel	gross_sales_amt_mln	channel_contribution_percentage
A0118150101	Direct	1.18	16.69
A0118150101	Distributor	0.87	12.35
A0118150101	Retailer	5.01	70.96
A0118150102	Direct	1.33	16.38
A0118150102	Distributor	0.89	10.99
A0118150102	Retailer	5.89	72.64
A0118150103	Direct	1.25	15.05
A0118150103	Distributor	0.86	10.36
A0118150103	Retailer	6.19	74.59
A0118150104	Direct	1.32	15.41
A0118150104	Distributor	1.00	11.65
A0118150104	Retailer	6.25	72.94
A0219150201	Direct	1.40	15.13
A0219150201	Distributor	1.09	11.76
A0219150201	Retailer	6.77	73.11
A0219150202	Direct	1.50	15.76
A0219150202	Distributor	1.03	10.82
A0219150202	Retailer	6.97	73.42
A0220150203	Direct	1.45	15.63

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_qty and rank_order**

Query:

```
with RankedProducts as (
select p.division,fs.product_code, p.product , sum(fs.sold_quantity) as total_sold_qty,
RANK() over(partition by p.division order by sum(fs.sold_quantity) desc) as Rank_Order
from fact_sales_monthly fs inner join dim_product p
on fs.product_code = p.product_code and fs.fiscal_year = 2021
group by fs.product_code
)
select division , product_code, product , total_sold_qty , Rank_Order
from RankedProducts
where Rank_Order <= 3 ;
```

Output :

division	product_code	product	total_sold_qty	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