Codebasics SQL Challenge

By João Pedro – Github

Requests:

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

Answer:

```
SELECT DISTINCT market
        FROM fact_sales_monthly INNER JOIN dim_customer
        ON fact_sales_monthly.customer_code = dim_customer.customer_code
        WHERE customer = "Atliq Exclusive" AND region = "APAC"
```

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, percentage_chg

Answer:

```
WITH
        result1 (unique_products_2020) AS (
                SELECT COUNT(DISTINCT product_code) AS unique_products_2020
                FROM fact_sales_monthly
                WHERE fiscal_year = "2020"
    ),
        result2 (unique_products_2021) AS (
                SELECT COUNT(DISTINCT product_code) AS unique_products_2020
                FROM fact_sales_monthly
                WHERE fiscal_year = "2021"
    ),
        result3 (percentage_chg) AS (
                SELECT ROUND((((unique_products_2021/unique_products_2020)-1)*100),2)
                FROM result1 JOIN result2
    )
SELECT * FROM result1 JOIN result2 JOIN result3
```

unique_products_2020	unique_products_2021	percentage_chg
245	334	36.33

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

Answer:

```
WITH
	count_column (segment, product_count) AS (
		SELECT segment, COUNT(distinct product_code)
		FROM dim_product
		GROUP BY segment
	)
SELECT * FROM count_column ORDER BY product_count DESC
```

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 difference

Answer:

```
WITH
	segment20_sel (segment, product_count_2020) AS (
		SELECT segment, count(DISTINCT fact_gross_price.product_code)
		FROM fact_gross_price JOIN dim_product
			ON fact_gross_price.product_code=dim_product.product_code
		WHERE fact_gross_price.fiscal_year="2020"
		GROUP BY segment
  ),
	segment21_sel (segment, product_count_2021) AS (
		SELECT segment, count(DISTINCT fact_gross_price.product_code)
		FROM fact_gross_price JOIN dim_product
			ON fact_gross_price.product_code=dim_product.product_code
		WHERE fact_gross_price.fiscal_year="2021"
		GROUP BY segment
  )
SELECT segment20_sel.segment, product_count_2020, product_count_2021,
	product_count_2021-product_count_2020 as difference,
	ROUND(100*((product_count_2021/product_count_2020)-1)) as percentage
FROM segment20_sel INNER JOIN segment21_sel
	ON segment20_sel.segment = segment21_sel.segment
ORDER BY difference DESC
```

segment	product_count_2020	product_count_2021	difference	percentage
Accessories	69	103	34	49
Notebook	92	108	16	17
Peripherals	59	75	16	27
Desktop	7	22	15	214
Storage	12	17	5	42
Networking	6	9	3	50

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

Answer:

```
WITH big_table(product_code, product, manufacturing_cost) AS (
        SELECT fact_manufacturing_cost.product_code, product, manufacturing_cost
                FROM fact_manufacturing_cost INNER JOIN dim_product
                        ON fact_manufacturing_cost.product_code = dim_product.product_code
)
SELECT product_code, product, ROUND(manufacturing_cost,2) as manufacturing_cost
        FROM big_table
        WHERE manufacturing_cost = (SELECT max(manufacturing_cost) FROM big_table)
UNION ALL
SELECT product_code, product, ROUND(manufacturing_cost,2) as manufacturing_cost
        FROM big_table
        WHERE manufacturing_cost = (SELECT min(manufacturing_cost) FROM big_table)
```

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

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

Answer:

```
SELECT dim_customer.customer_code, customer, ROUND(pre_invoice_discount_pct*100,2) AS
average_discount_percentage
FROM fact_pre_invoice_deductions INNER JOIN dim_customer
        ON fact_pre_invoice_deductions.customer_code = dim_customer.customer_code
WHERE fiscal_year="2021"
        AND pre_invoice_discount_pct>=(SELECT AVG(pre_invoice_discount_pct) FROM
fact_pre_invoice_deductions where fiscal_year="2021")
        AND market = "India"
ORDER BY pre_invoice_discount_pct DESC
LIMIT 5
```

customer_code	customer	average_discount_percentage
90002009	Flipkart	30.83
90002006	Viveks	30.38
90002003	Ezone	30.28
90002002	Croma	30.25
90002016	Amazon	29.33

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

Answer:

```
SELECT month(date) as month,year(date) as year, round(sum(sold_quantity*gross_price)) as
Gross_sales_Amount
   FROM fact_sales_monthly INNER JOIN fact_gross_price
        ON fact_gross_price.product_code=fact_sales_monthly.product_code
              AND fact_gross_price.fiscal_year=fact_sales_monthly.fiscal_year
        INNER JOIN dim_customer
              ON dim_customer.customer_code=fact_sales_monthly.customer_code
   WHERE dim_customer.customer="Atliq Exclusive"
   GROUP BY date ORDER BY year and month ASC
```

month	year	Gross_sales_Amount
9	2019	4496260
10	2019	5135902
11	2019	7522893
12	2019	4830405
1	2020	4740600
2	2020	3996228
3	2020	378771
4	2020	395035
5	2020	783813
6	2020	1695217

month	year	Gross_sales_Amount
11	2020	20464999
12	2020	12944660
1	2021	12399393
2	2021	10129736
3	2021	12144061
4	2021	7312000
5	2021	12150225
6	2021	9824521
7	2021	12092346
8	2021	7178708

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

Answer:

```
SELECT quarter(DATE_ADD(date, INTERVAL 4 MONTH)) 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;
```

quarter	total_sold_quantity
1	7005619
2	6649642
4	5042541
3	2075087

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

Answer:

```
WITH tabela (channel, gross_sales_mln) AS (
	SELECT channel, round(sum(sold_quantity*gross_price)) AS gross_sales_mln
	FROM fact_sales_monthly
		INNER JOIN fact_gross_price
			ON fact_sales_monthly.product_code=fact_gross_price.product_code
				AND fact_sales_monthly.fiscal_year=fact_gross_price.fiscal_year
		INNER JOIN dim_customer
			ON fact_sales_monthly.customer_code=dim_customer.customer_code
	WHERE year(date)="2021"
	GROUP BY channel
)
SELECT channel, gross_sales_mln, ROUND(gross_sales_mln/(SELECT sum(gross_sales_mln) FROM tabela)*100) AS percentage
	FROM tabela
	ORDER BY percentage DESC
```

channel	gross_sales_mln	percentage
Retailer	705532519	73
Direct	150664256	16
Distributor	107332599	11

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_order

Answer:

```
WITH tabela (division, product_code, product, total_sold_quantity) AS (
SELECT  division, fact_sales_monthly.product_code, product, sum(sold_quantity) AS total_sold_quantity
	FROM fact_sales_monthly INNER JOIN dim_product
			ON fact_sales_monthly.product_code=dim_product.product_code
	WHERE fiscal_year="2021"
	GROUP BY fact_sales_monthly.product_code, division, product
	ORDER BY division DESC, total_sold_quantity DESC
)
SELECT * FROM ( SELECT *, ROW_NUMBER() OVER (PARTITION BY division ORDER BY total_sold_quantity DESC) AS rank_order FROM tabela)
	RANKED
	WHERE rank_order<= 3;
```

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