CONSUMER-AD-HOC INSIGHTS

PRESENTED BY : BHAVANI

Objectives

- **AtliQ Hardware** (fictitious corporation) is one of the major computer hardware manufacturers in India, with a strong presence in other nations.
- Nevertheless, the management did note that they do not have sufficient insights to make prompt, wise, and data-informed judgments.
- Plan to expand the data analytics team by adding junior data analysts.
- To assess candidates, Data analytics director, Tony Sharma plans to conduct a **SQL Challenge** to evaluate both tech and soft skills.
- Thecompanyseeksinsights for 10 ad hoc requests.

Company's Markets

AtliQ Hardware operates across four major regions: **North America (NA)**, **Latin America (LATAM)**, **Europe (EU)**, and **Asia-Pacific (APAC)**. This global presence allows AtliQ to serve diverse customer needs with products in Networking and Storage, PCs, Peripherals and Accessories.

In **NA** and **EU**, AtliQ benefits from strong demand for **Advanced Computing Solutions**. **LATAM** and **APAC**, with their growing economies, offer **Significant Opportunities for Expansion**. This strategic positioning ensures AtliQ's sustained growth and customer satisfaction worldwide.

AtliQ Hardware

HOME ABOUT CONTACT ME

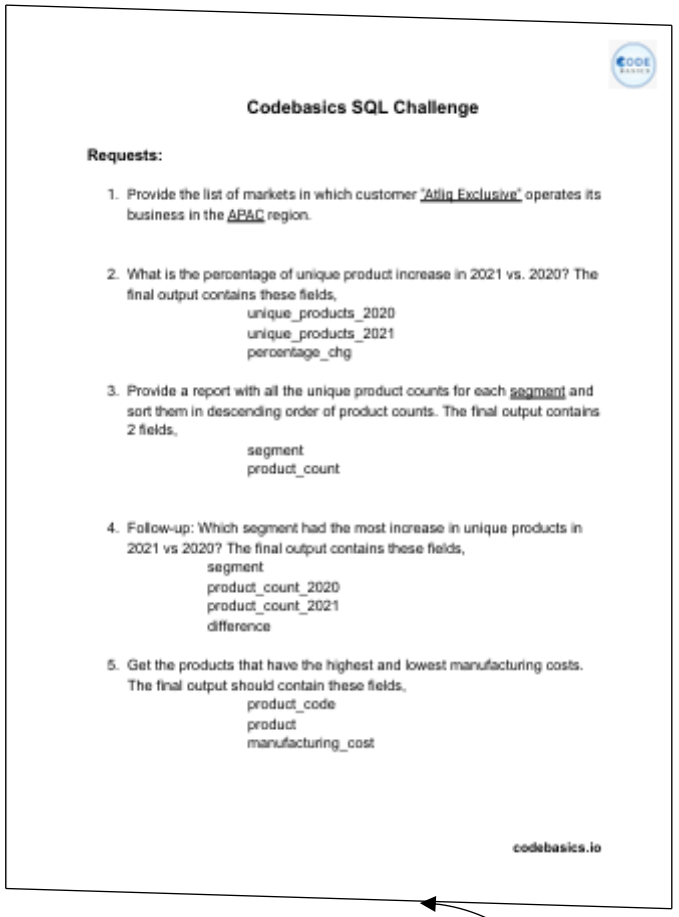
Requests & Tools

Power BI

For Analysis and Visualisations

MySQL

For Ad Hoc Queries



Codebasics SQL Challenge

Requests:

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

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

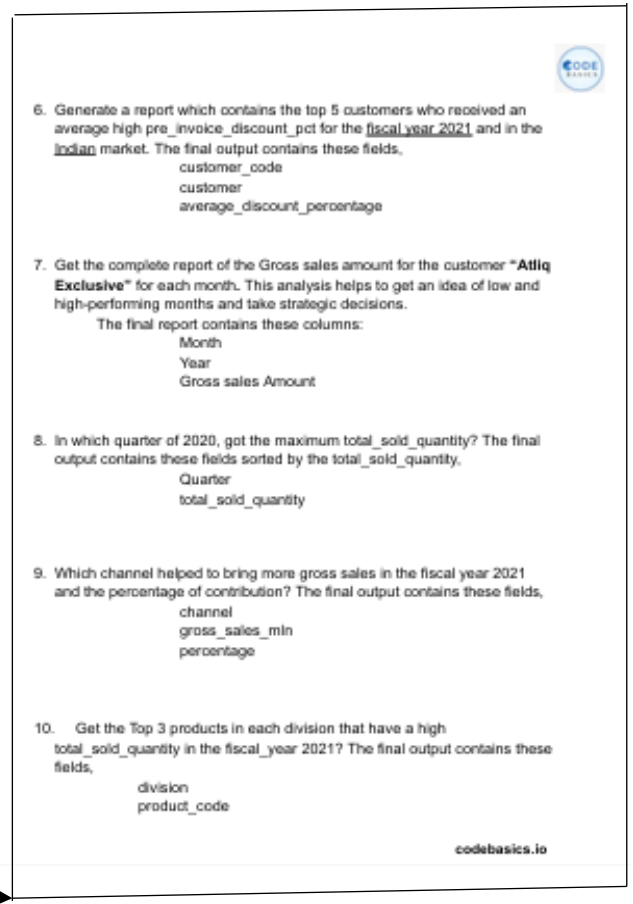
3. Provide a report with all the unique product counts for each segment and sort them in descending order of product counts. The final output contains 2 fields,
   segment
   product_count

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

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

codebasics.io

10 Ad Hoc Requests



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

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

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

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

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

codebasics.io

PRESENTED BY : BHAVANI

AtliQ Hardware

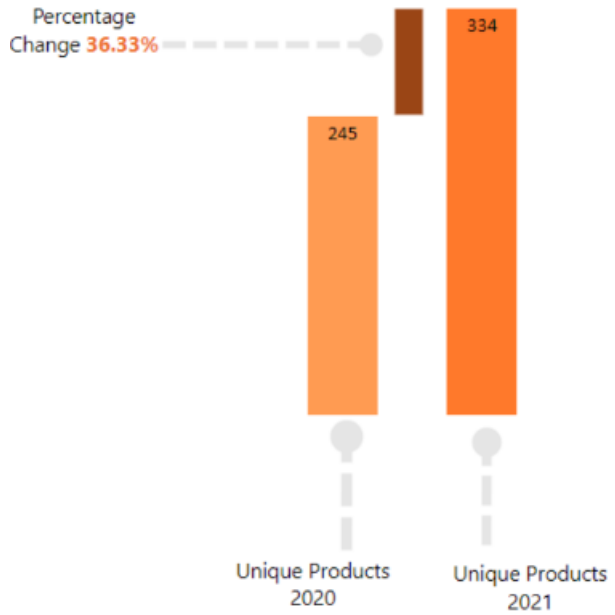
HOME ABOUT CONTACT ME

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

```sql
select
  distinct market
from
  dim_customer
where
  customer = "Atliq Exclusive"
  and region = "APAC"
order by
  market;
```

PRESENTED BY : BHAVANI

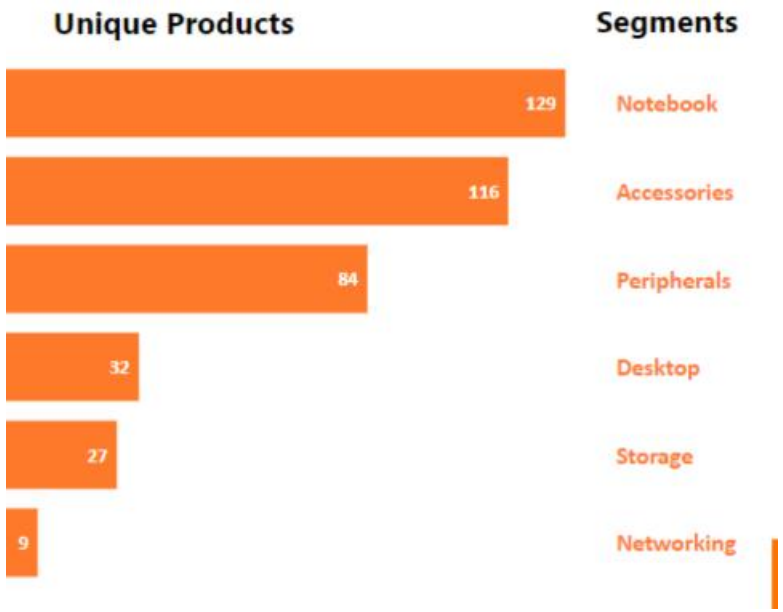
2Q. What is the percentage of unique product increase in 2021 vs. 2020? The final output contains these fields,unique_products_2020 ,unique_products_2021,percentage_chg.




```sql
WITH unique_products_2020 AS (
  SELECT
    COUNT(DISTINCT product_code) AS unique_products_2020
  FROM
    fact_sales_monthly
  WHERE
    fiscal_year = 2020
),
unique_products_2021 AS (
  SELECT
    COUNT(DISTINCT product_code) AS unique_products_2021
  FROM
    fact_sales_monthly
  WHERE
    fiscal_year = 2021
)
SELECT
  up20.unique_products_2020,
  up21.unique_products_2021,
  ROUND(
    (
      (
        unique_products_2021 - unique_products_2020
      ) * 100
    ) / unique_products_2020,
    2
  ) AS percentage_chg
FROM
  unique_products_2020 up20,
  unique_products_2021 up21;
```

3Q. 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

```sql
select
  segment,
  count(
    distinct(product_code)
  ) as unique_products
from
  dim_product
group by
  segment
order by
  unique_products desc;
```

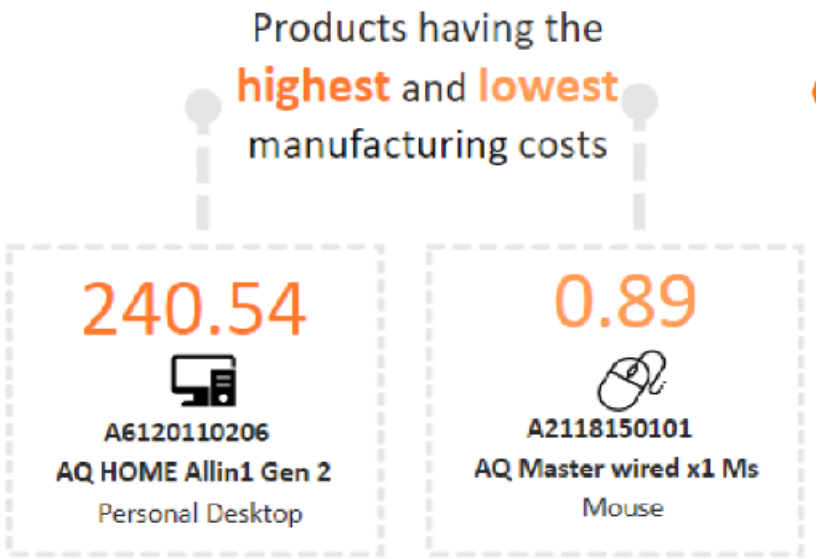


4Q. Follow-up: Which segment had the most increase in unique products in 2021 vs 2020? The final output contains these fields – segment,product_count_2020,product_count_2021 difference.

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

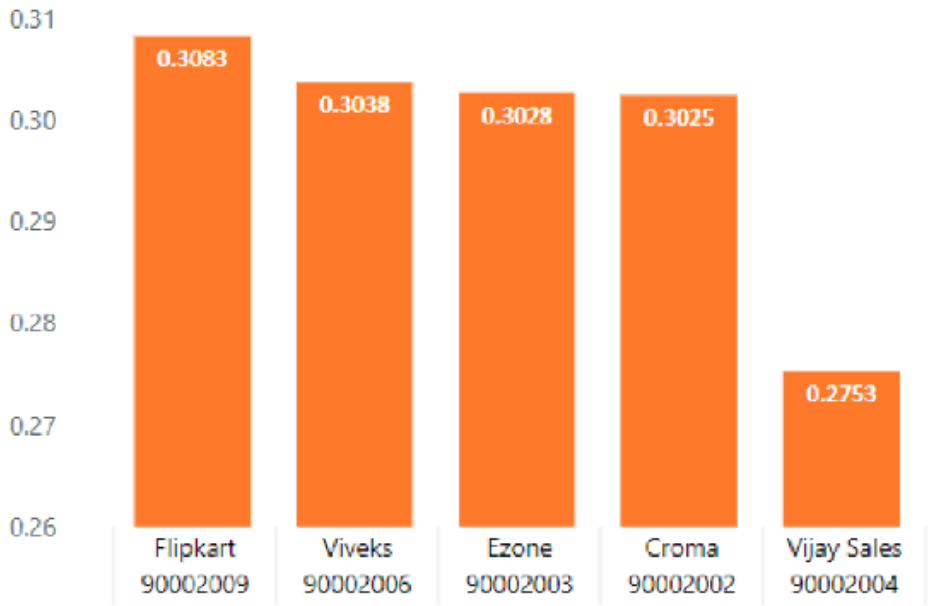
```sql
with unique_prd_20 as(
  select
    d.segment,
    count(distinct d.product_code) as unique_product_20
  from
    dim_product d
    join fact_gross_price as f on d.product_code = f.product_code
  where
    fiscal_year = 2020
  group by
    segment
),
unique_prd_21 as(
  select
    d.segment,
    count(distinct d.product_code) as unique_product_21
  from
    dim_product d
    join fact_gross_price as f on d.product_code = f.product_code
  where
    fiscal_year = 2021
  group by
    segment
)
SELECT
  uq20.segment,
  uq21.unique_product_21,
  uq20.unique_product_20,
  (
    uq21.unique_product_21 - uq20.unique_product_20
  ) AS difference
FROM
  unique_prd_20 uq20
  JOIN unique_prd_21 uq21 ON uq20.segment = uq21.segment
ORDER BY
  difference DESC;
```

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



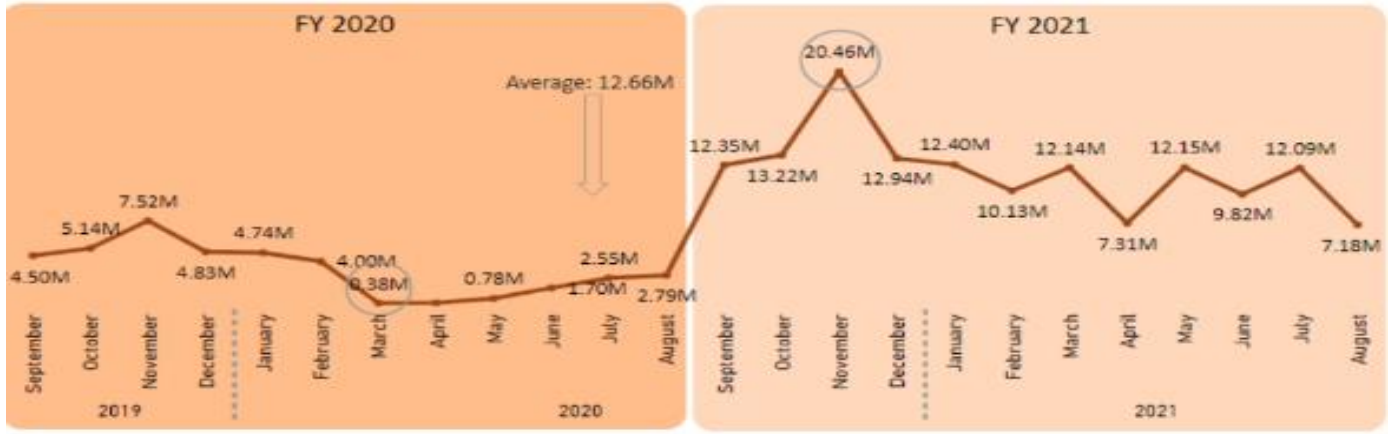

```sql
SELECT
  fmc.product_code,
  dp.product,
  fmc.manufacturing_cost
FROM
  fact_manufacturing_cost fmc
  JOIN dim_product dp ON
dp.product_code = fmc.product_code
WHERE
  fmc.manufacturing_cost IN (
    SELECT
      MAX(manufacturing_cost)
    FROM
      fact_manufacturing_cost
    UNION
    SELECT
      MIN(manufacturing_cost)
    FROM
      fact_manufacturing_cost
  )
ORDER BY
  fmc.manufacturing_cost DESC;
```

6Q.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.




```sql
select
  c.customer_code,
  c.customer,
  avg(f.pre_invoice_discount_pct) as
average_discount_percentage
from
  fact_pre_invoice_deductions as f
  join dim_customer as c on
f.customer_code = c.customer_code
where
  c.market = "India"
  and f.fiscal_year = 2020
group by
  c.customer
ORDER BY
  average_discount_percentage DESC
limit
  5;
```

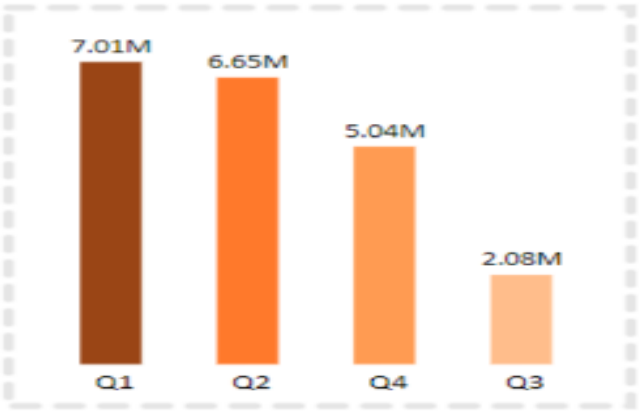
7Q. Get the complete report of the Gross sales amount for the customer “Atliq Exclusive” for each month. This analysis helps to get an idea of low and high-performing months and take strategic decisions. The final report contains these columns:Month,Year,Gross sales Amount.




```sql
select
  DATE_FORMAT(fsm.date, '%M (%Y)') AS
Month,
  fsm.fiscal_year AS Fiscal_Year,
  ROUND(
    SUM(
      (
        fsm.sold_quantity *
fgp.gross_price
      )
    ),
    2
  ) AS Gross_Sales_Amount
from
  dim_customer c
  join fact_sales_monthly as fsm on
c.customer_code = fsm.customer_code
  JOIN fact_gross_price fgp ON
fgp.product_code = fsm.product_code
  AND fgp.fiscal_year = fsm.fiscal_year
where
  c.customer = "Atliq Exclusive"
GROUP BY
  Month,
  Fiscal_Year
ORDER BY
  Fiscal_Year;
```

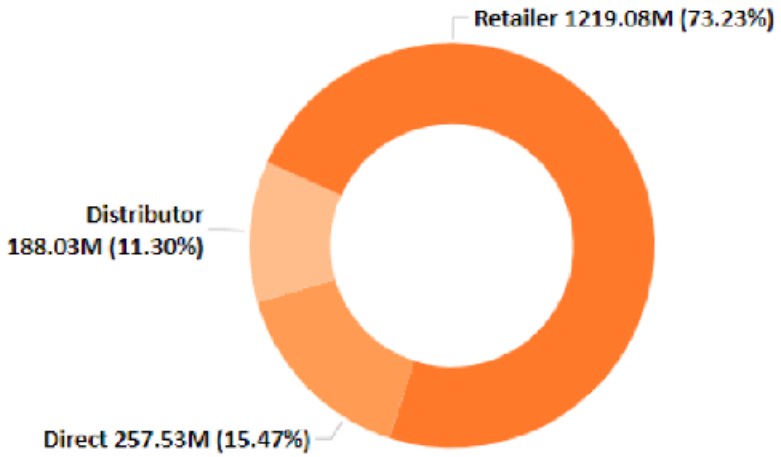
8Q.In which quarter of 2020, got the maximum total_sold_quantity? The final output contains these fields sorted by the total_sold_quantity,Quarter,total_sold_quantity

Month	Quarter	Total Sold Quantity
September	Q1	1.76M
October	Q1	2.19M
November	Q1	3.05M
January	Q2	1.76M
February	Q2	1.70M
December	Q2	3.18M
March	Q3	0.24M
April	Q3	0.82M
May	Q3	1.02M
June	Q4	1.56M
July	Q4	1.69M
August	Q4	1.79M




```sql
SELECT
    CASE
        WHEN MONTH(date) BETWEEN 4 AND 6 THEN 'Q1'
        WHEN MONTH(date) BETWEEN 7 AND 9 THEN 'Q2'
        WHEN MONTH(date) BETWEEN 10 AND 12 THEN 'Q3'
        WHEN MONTH(date) BETWEEN 1 AND 3 THEN '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;
```

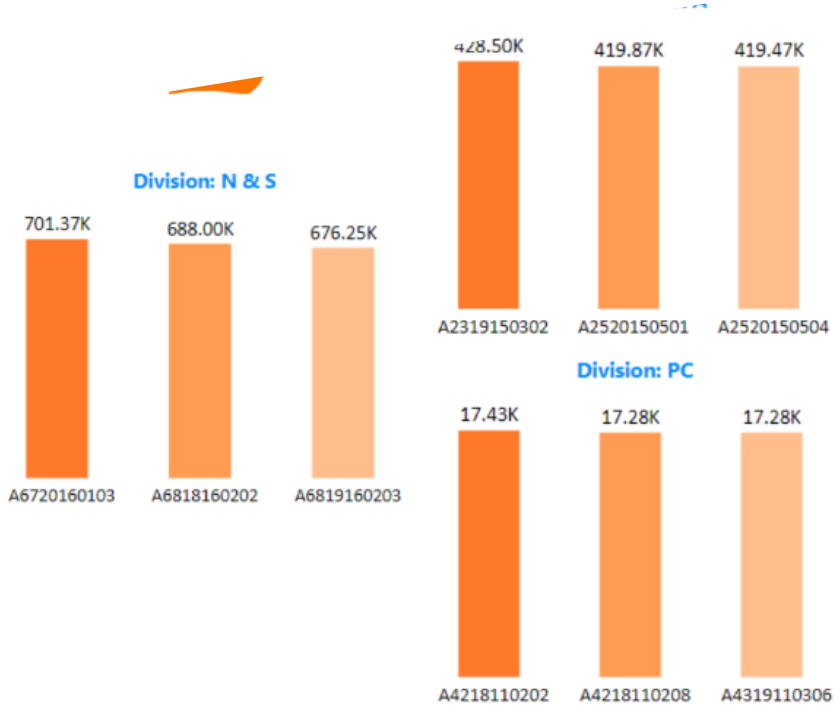
9Q.Which channel helped to bring more gross sales in the fiscal year 2021 and the percentage of contribution? The final output contains these fields,channel, gross_sales_mln,percentage




```sql
with channel_sales_2021 as(
select
c.channel,round(sum(fsm.sold_quantity*fgp.gross_price/1000000),2)as
gross_sales_mln from dim_customer c join fact_sales_monthly as fsm
on c.customer_code=fsm.customer_code
join fact_gross_price as fgp on fsm.product_code=fgp.product_code
where fsm.fiscal_year=2021
group by c.channel
order by  gross_sales_mln  desc),
total_sales_2021 AS(
    SELECT
      SUM(gross_sales_mln) AS total_gross_sales_mln
    FROM
      channel_sales_2021
    )

SELECT
  cs21.channel,
    CONCAT( cs21.gross_sales_mln, 'M' ) AS gross_sales_mln,
    CONCAT( ROUND( ( (cs21.gross_sales_mln * 100) /
ts21.total_gross_sales_mln ), 2 ), '%' ) AS percentage
FROM
  channel_sales_2021 cs21,
    total_sales_2021 ts21;
```

10Q. 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




```sql
with division_sales_2021 as(
select p.division,p.product_code, concat( p.product, ' (', p.variant,
')') AS product, sum(fsm.sold_quantity)as total_sold_quantity from
dim_product  as p  join fact_sales_monthly as fsm on p.product_code =
fsm.product_code
where fsm.fiscal_year=2021
group by p.division,p.product_code,p.product,p.variant),

sales_rank_2021 AS(
        SELECT
      *,
      dense_rank() OVER( PARTITION BY division ORDER BY
total_sold_quantity DESC ) AS rank_order
    FROM
      division_sales_2021
  )
    SELECT
    *
  FROM
    sales_rank_2021
  WHERE
    rank_order <= 3;
```

Thank you

PRESENTED BY : BHAVANI