Atliq-Hardware Analysis (SQL)

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

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

Output:

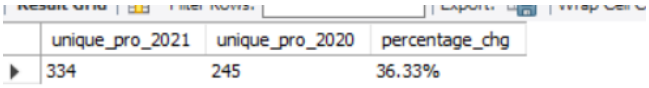
market
India
Indonesia
Japan
Philiphines
South Korea
Australia
Newzealand
Bangladesh

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
   - percentage_chg

```
with unique_2020 as
(
        select
                count(distinct product_code ) as unique_pro_2020
        from
                dim_product as dp
        join
                fact_sales_monthly as fsm
        using
                (product_code)
        where
```

```
                fiscal_year = 2020
),
unique_2021 as
(
                select
                count(distinct product_code ) as unique_pro_2021
        from
                dim_product as dp
        join
                fact_sales_monthly as fsm
        using
                (product_code)
        where
                fiscal_year = 2021
)
```

Output :



unique_pro_2021	unique_pro_2020	percentage_chg
334	245	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

```
select
     distinct segment  ,
    count(*) over(partition by segment )  as product_count
from dim_product
order by product_count 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
    - difference

```
with pro_2020 as (
select
        distinct dm.segment ,
        count(*) over(partition by dm.product_code  ) as product_count_2020
from dim_product as dm
join
        fact_sales_monthly  as fsm
 on
        dm.product_code=fsm.product_code
where
        fiscal_year =2020
),
pro_2021 as
(
select
        distinct dm.segment ,
        count(*) over(partition by dm.product_code ) as product_count_2021
from dim_product as dm
join
        fact_sales_monthly  as fsm
 on
        dm.product_code=fsm.product_code
where
        fiscal_year =2021
)
select distinct p1.segment,p0.product_count_2020,p1.product_count_2021
,(p1.product_count_2021-p0.product_count_2020)as difference
from pro_2020 as p0
join pro_2021 as p1
on
        p0.segment =p1.segment
group by p1.segment
```

Output :

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

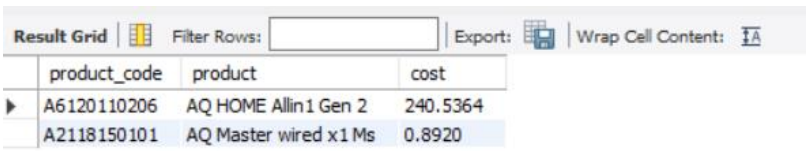
segment	product_count_2020	product_count_2021	difference
Peripherals	1812	1881	69
Accessories	1870	1881	11
Notebook	1494	1852	358
Desktop	289	1826	1537
Storage	1868	1881	13
Networking	1870	1881	11

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

```
with cte as (
select
        p.product_code,
  p.product,
  f.manufacturing_cost
from
        dim_product p
join
        fact_manufacturing_cost f
on
        p.product_code=f.product_code

)
select
        product_code,
  product,
  manufacturing_cost as cost
  from cte
  where manufacturing_cost=(select max(manufacturing_cost) from cte)
union
select
        product_code,
  product,
  manufacturing_cost as cost
  from cte
  where manufacturing_cost=(select min(manufacturing_cost) from cte)
```

Output:

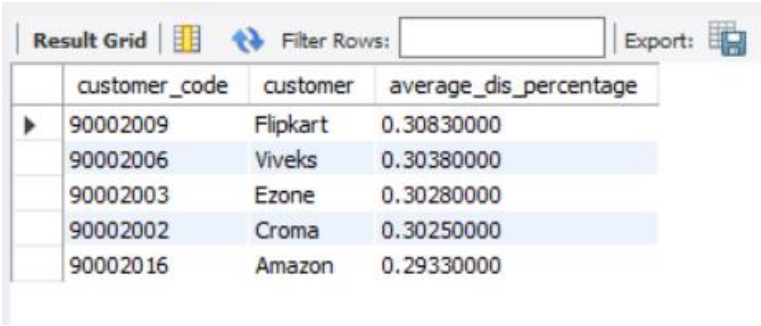


product_code	product	cost
A6120110206	AQ HOME Allin1 Gen 2	240.5364
A2118150101	AQ Master wired x1 Ms	0.8920

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

```
select
        distinct(ded.customer_code),
  cus.customer,
  avg(ded.pre_invoice_discount_pct) over(partition by ded.customer_code ) as
average_dis_percentage
from
        fact_pre_invoice_deductions as ded
join
        dim_customer as  cus
on
        cus.customer_code=ded.customer_code
where
        ded.fiscal_year=2021
  and
  cus.market ='India'
order by average_dis_percentage  desc limit 5;
```

Output:



Result Grid | Filter Rows: | Export:

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

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

```
SELECT
   YEAR(fact_sales_monthly.date) AS Year,
   MONTHNAME(fact_sales_monthly.date) AS Month,
   concat(round(SUM(fact_sales_monthly.sold_quantity* fact_gross_price.gross_price)/100000
,2)," L") AS Gross_sales_Amount
FROM
        dim_customer
JOIN
        fact_sales_monthly
        ON dim_customer.customer_code = fact_sales_monthly.customer_code
JOIN
        fact_gross_price
        ON fact_gross_price.product_code=fact_sales_monthly.product_code
WHERE dim_customer.customer = "Atliq Exclusive"
GROUP BY Year, Month
order by Year, Month asc ;
```

Output:

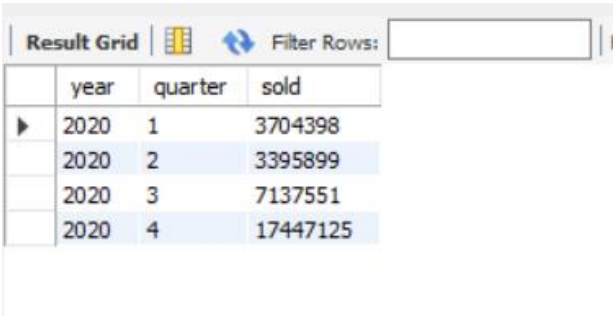
Result Grid | Filter Rows: | Exp

Year	Month	Gross_sales_Amount
2019	December	97.56 L
2019	November	152.32 L
2019	October	103.79 L
2019	September	90.93 L
2020	April	8.00 L
2020	August	56.38 L
2020	December	204.09 L
2020	February	80.84 L
2020	January	95.85 L
2020	July	51.52 L
2020	June	34.30 L
2020	March	7.67 L
2020	May	15.87 L
2020	November	322.47 L
2020	October	210.16 L
2020	September	195.30 L
2021	April	114.84 L
2021	August	113.25 L
2021	February	159.87 L
2021	January	195.71 L
2021	July	190.45 L
2021	June	154.58 L
2021	March	191.50 L
2021	May	192.04 L

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

```
select
        Year(date) as year,
        quarter(date) as quarter,
        sum(sold_quantity) as sold
from
        fact_sales_monthly
where
        year(date)=2020
group by year ,quarter
order by year ,quarter
```

Output:



Result Grid | Filter Rows:

year	quarter	sold
2020	1	3704398
2020	2	3395899
2020	3	7137551
2020	4	17447125

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

```
with channel_sum as
(
        select
                cus.channel ,
                sum(gross.gross_price *sales.sold_quantity) as sold
        from
                fact_gross_price as gross
        join
                fact_sales_monthly as sales
        using (product_code)
        join
                dim_customer as cus
        using (customer_code)
```

```
    where
        sales.fiscal_year=2021
    group by cus.channel

),
        total_sold as
    (
    select

                sum(gross.gross_price *sales.sold_quantity) as total_sold
        from
                fact_gross_price as gross
        join
                fact_sales_monthly as sales
        using (product_code)
    where
        sales.fiscal_year=2021
    )
select
        channel ,
        sold ,
    concat(round((sold/total_sold)*100 ,2),' %') as percentage
from
        channel_sum,
        total_sold
```

Output:

Result Grid | Filter Rows: | Export:

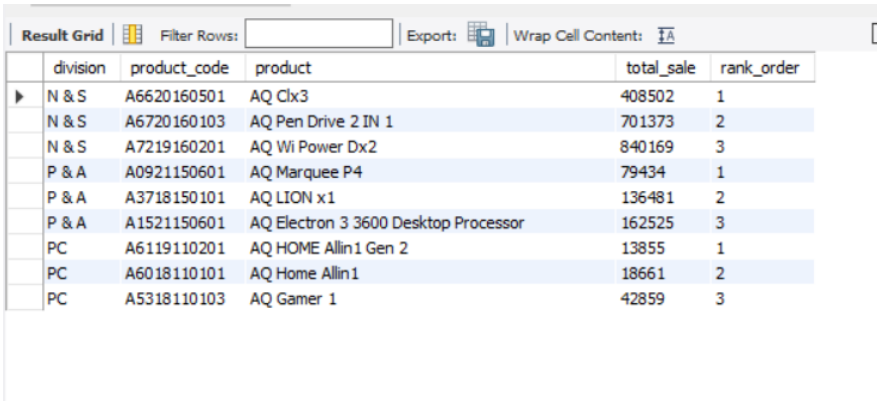
channel	sold	percentage
Direct	406686873.9033	15.47 %
Distributor	297175879.7188	11.31 %
Retailer	1924170397.9096	73.22 %

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

```
SET sql_mode = '';
with hi_cte as(
select
```

```
        division,
        pro.product_code,
        product,
   sum(sold_quantity) as total_sale,
   rank() over (partition by division order by sum(sold_quantity)) as 'rank_order'
from
        dim_product as pro
join
        fact_sales_monthly as sales
using (product_code)
where
        sales.fiscal_year=2021
group by
        pro.product
)
select * from hi_cte where rank_order < 4
```

Output:



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

division	product_code	product	total_sale	rank_order
N & S	A6620160501	AQ Clx3	408502	1
N & S	A6720160103	AQ Pen Drive 2 IN 1	701373	2
N & S	A7219160201	AQ Wi Power Dx2	840169	3
P & A	A0921150601	AQ Marquee P4	79434	1
P & A	A3718150101	AQ LION x1	136481	2
P & A	A1521150601	AQ Electron 3 3600 Desktop Processor	162525	3
PC	A6119110201	AQ HOME Allin1 Gen 2	13855	1
PC	A6018110101	AQ Home Allin1	18661	2
PC	A5318110103	AQ Gamer 1	42859	3