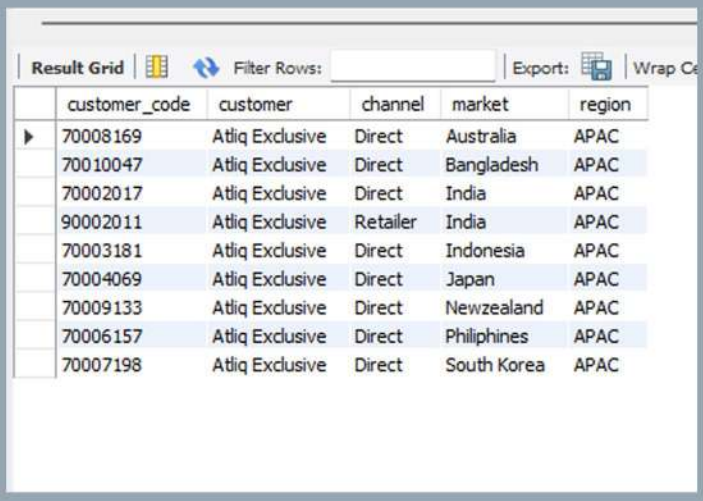
1. Provide the list of markets in which customer "Atliq Exclusive" operates its business in the APAC region.

Code:

```
select  customer_code,
        customer,
        channel,
        market,
        region from dim_customer
        where customer = 'Atliq Exclusive' and
        region ='APAC'
        order by market
```

Result:



Result Grid | Filter Rows: | Export: | Wrap Ce

customer_code	customer	channel	market	region
70008169	Atliq Exclusive	Direct	Australia	APAC
70010047	Atliq Exclusive	Direct	Bangladesh	APAC
70002017	Atliq Exclusive	Direct	India	APAC
90002011	Atliq Exclusive	Retailer	India	APAC
70003181	Atliq Exclusive	Direct	Indonesia	APAC
70004069	Atliq Exclusive	Direct	Japan	APAC
70009133	Atliq Exclusive	Direct	Newzealand	APAC
70006157	Atliq Exclusive	Direct	Philiphines	APAC
70007198	Atliq Exclusive	Direct	South Korea	APAC

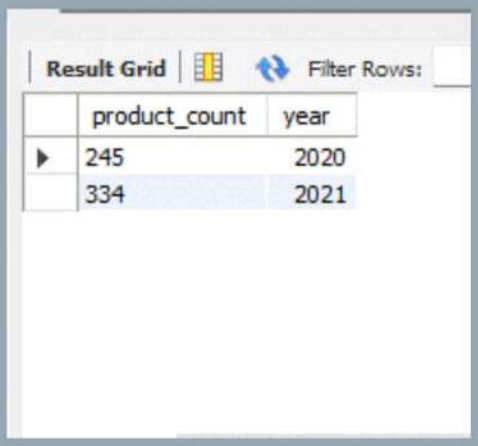
Submitted by M. Vamsi Kumar

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.

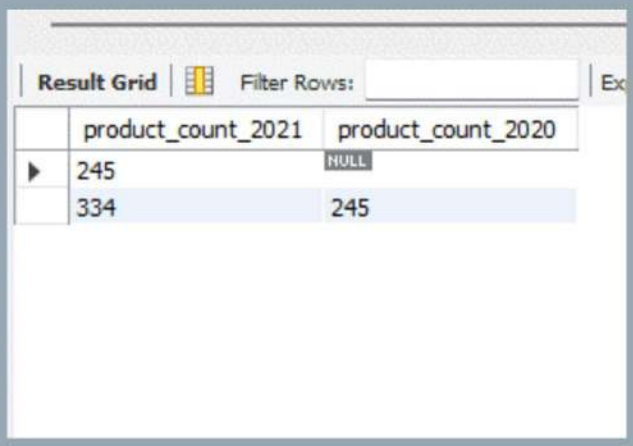
Code:

```
with cte2 as
(
with cte1 as
(
select count(product_code) as product_count, cost_year as year from
fact_manufacturing_cost
group by year
)
select product_count as product_count_2021, lag(product_count) over (order by year) as
product_count_2020 from cte1
)
select product_count_2021,product_count_2020,((product_count_2021-
product_count_2020)/product_count_2020)*100 as percentage_change from cte2
where product_count_2020 is not null
```

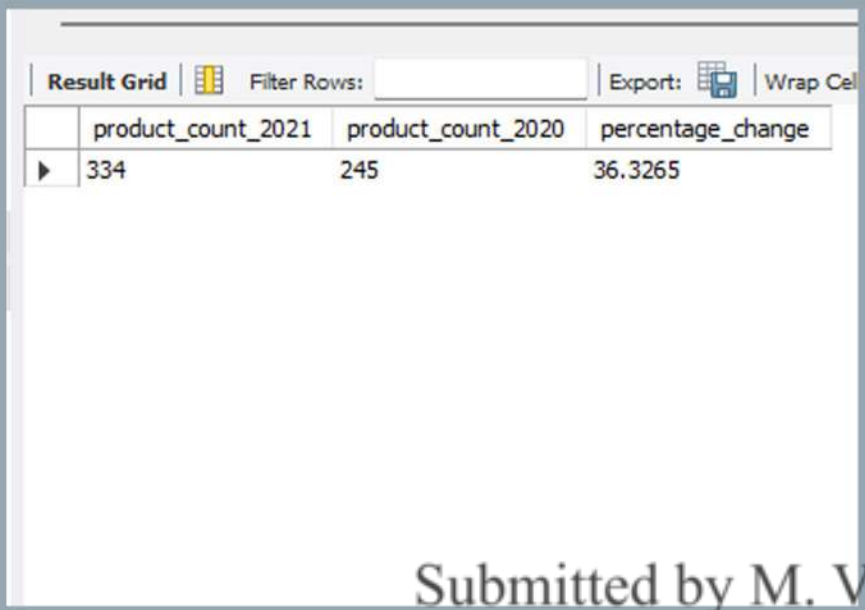
Result:



product_count	year
245	2020
334	2021



product_count_2021	product_count_2020
245	NULL
334	245



product_count_2021	product_count_2020	percentage_change
334	245	36.3265

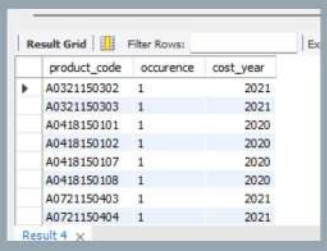
Submitted by M. Vamsi Kumar

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.

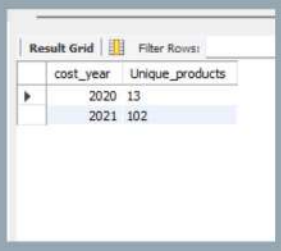
Code:

```
with cte4 as
(
with cte3 as
(
with cte2 as
(
with cte1 as
(
select product_code,count(product_code) as occurence,cost_year from
fact_manufacturing_cost group by product_code
)
select * from cte1 where occurence = '1'
)
select distinct(cost_year),count(product_code) over (partition by cost_year order by
cost_year) as Unique_products from cte2
)
select cost_year as year, Unique_products as Unique_products_2021,
lag(Unique_products) over (order by cost_year) as Unique_products_2020 from cte3
)
select
Unique_products_2021,Unique_products_2020,concat(round((((Unique_products_2021-
Unique_products_2020)/Unique_products_2020)*100),2),'%') as percentage_chg from
cte4
where Unique_products_2020 is not null
```

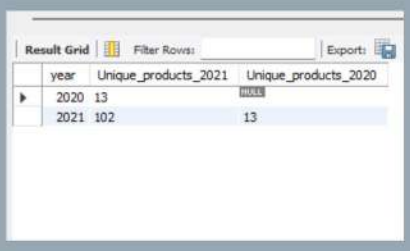
Result:



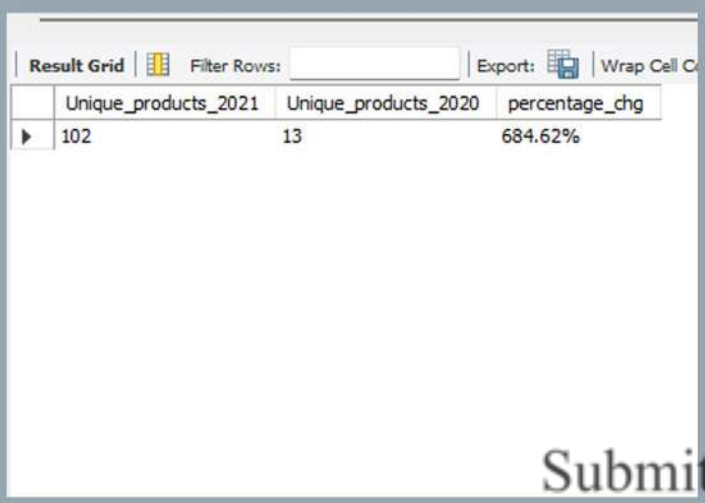
product_code	occurence	cost_year
A0321150302	1	2021
A0321150303	1	2021
A0418150101	1	2020
A0418150102	1	2020
A0418150107	1	2020
A0418150108	1	2020
A0721150403	1	2021
A0721150404	1	2021



cost_year	Unique_products
2020	13
2021	102



year	Unique_products_2021	Unique_products_2020
2020	13	NULL
2021	102	13



Unique_products_2021	Unique_products_2020	percentage_chg
102	13	684.62%

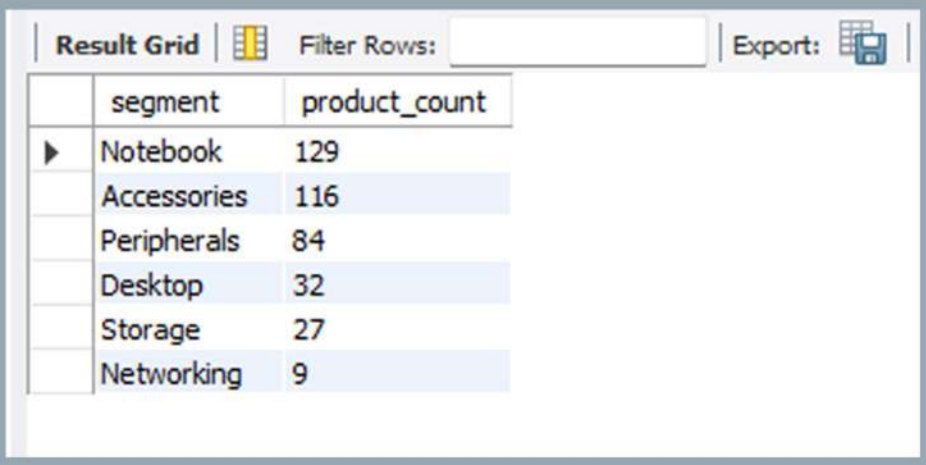
Submitted by M. Vamsi Kumar

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.

Code:

```
with cte1 as
(
select distinct(segment), count(product_code) over (partition by segment order by
segment) as product_count from c4db.dim_product
)
select * from cte1 order by product_count desc
```

Result:



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

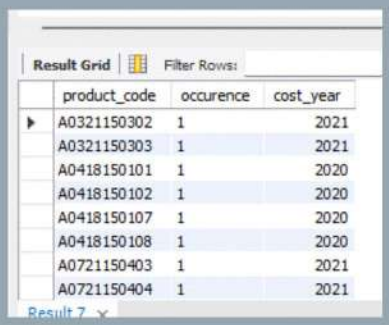
Submitted by M. Vamsi Kumar

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.

Code:

```
with cte3 as
(
with cte2 as
(
with cte1 as
(
select product_code,count(product_code) as occurence,cost_year
        from fact_manufacturing_cost
        group by product_code
)
select * from cte1 where occurence = '1'
)
select B.segment,A.product_code from cte2 as A
left join dim_product as B
on A.product_code = B.product_code
)
select distinct(segment),count(product_code) as No_of_Unique_products  from cte3
group by segment
order by No_of_Unique_products desc
```

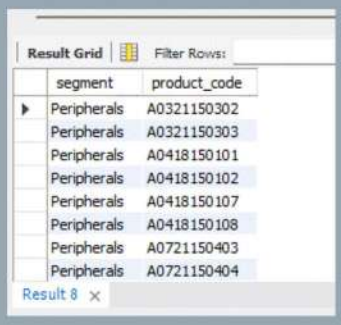
Result:



Result Grid | Filter Rows:

product_code	occurence	cost_year
A0321150302	1	2021
A0321150303	1	2021
A0418150101	1	2020
A0418150102	1	2020
A0418150107	1	2020
A0418150108	1	2020
A0721150403	1	2021
A0721150404	1	2021

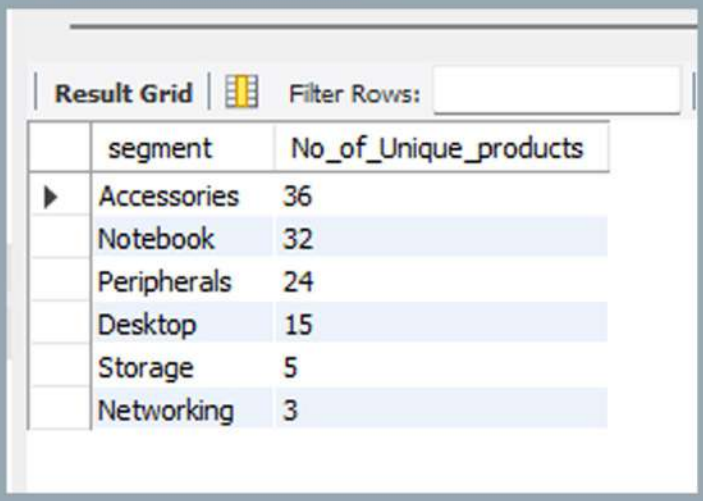
Result 7



Result Grid | Filter Rows:

segment	product_code
Peripherals	A0321150302
Peripherals	A0321150303
Peripherals	A0418150101
Peripherals	A0418150102
Peripherals	A0418150107
Peripherals	A0418150108
Peripherals	A0721150403
Peripherals	A0721150404

Result 8



Result Grid | Filter Rows:

segment	No_of_Unique_products
Accessories	36
Notebook	32
Peripherals	24
Desktop	15
Storage	5
Networking	3

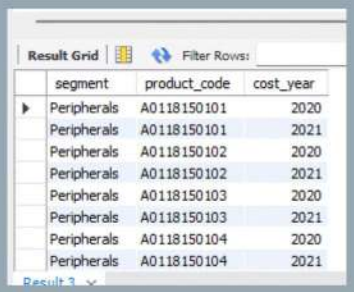
Submitted by M. Vamsi Kumar

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.

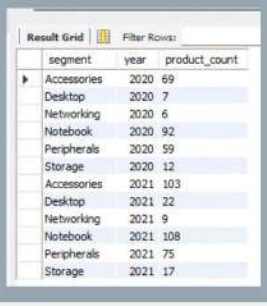
Code:

```
with cte3 as
(
with cte2 as
(
with cte1 as
(
select B.segment,A.product_code,cost_year  from fact_manufacturing_cost as A
left join dim_product as B
on A.product_code = B.product_code
)
select distinct(segment),cost_year as year, count(product_code) over (partition by
cost_year, segment order by cost_year)  as product_count from cte1
)
select segment,year,product_count as product_count_2021,
lag(product_count) over ( partition by segment order by year) as product_count_2020
from cte2
)
select segment,product_count_2021,product_count_2020,(product_count_2021-
product_count_2020) as difference from cte3
where product_count_2020 is not null
order by difference desc
```

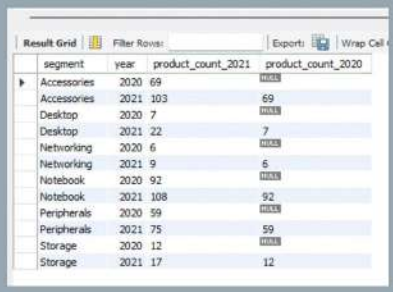
Result:



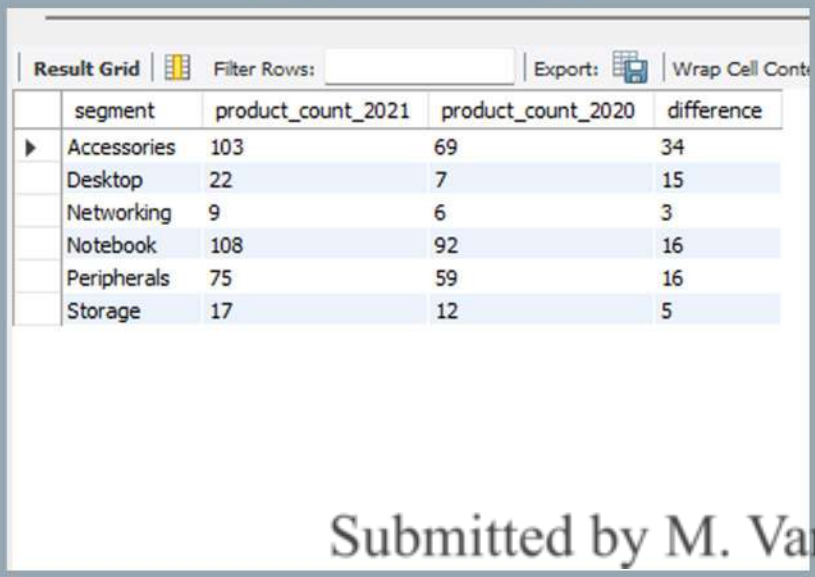
segment	product_code	cost_year
Peripherals	A0118150101	2020
Peripherals	A0118150101	2021
Peripherals	A0118150102	2020
Peripherals	A0118150102	2021
Peripherals	A0118150103	2020
Peripherals	A0118150103	2021
Peripherals	A0118150104	2020
Peripherals	A0118150104	2021



segment	year	product_count
Accessories	2020	69
Desktop	2020	7
Networking	2020	6
Notebook	2020	92
Peripherals	2020	59
Storage	2020	12
Accessories	2021	103
Desktop	2021	22
Networking	2021	9
Notebook	2021	108
Peripherals	2021	75
Storage	2021	17



segment	year	product_count_2021	product_count_2020
Accessories	2020	69	NULL
Accessories	2021	103	69
Desktop	2020	7	NULL
Desktop	2021	22	7
Networking	2020	6	NULL
Networking	2021	9	6
Notebook	2020	92	NULL
Notebook	2021	108	92
Peripherals	2020	59	NULL
Peripherals	2021	75	59
Storage	2020	12	NULL
Storage	2021	17	12



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

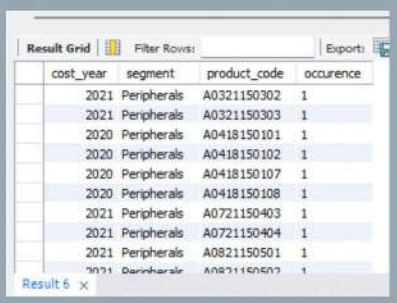
Submitted by M. Vamsi Kumar

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.

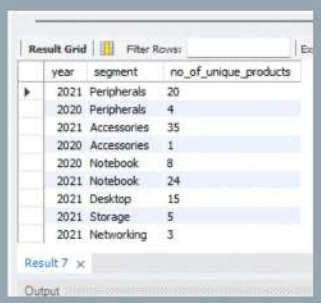
Code:

```
with cte4 as
(
with cte3 as
(
with cte2 as
(
with cte1 as
(
select product_code,count(product_code) as occurence,cost_year from
fact_manufacturing_cost
group by product_code
)
select cost_year,B.segment,A.product_code,occurence from cte1 as A
left join dim_product as B
on A.product_code = B.product_code
where occurence ='1'
)
select cost_year as year, segment,count(product_code) as no_of_unique_products from
cte2
group by cost_year,segment
)
select year,segment,no_of_unique_products as no_of_unique_products_2021,
lag(no_of_unique_products) over (partition by segment order by year) as
no_of_unique_products_2020 from cte3
)
select
segment,no_of_unique_products_2021,coalesce(no_of_unique_products_2020,'0') as
no_of_unique_products_2020,(no_of_unique_products_2021-
no_of_unique_products_2020) as difference from cte4
where year = '2021'
```

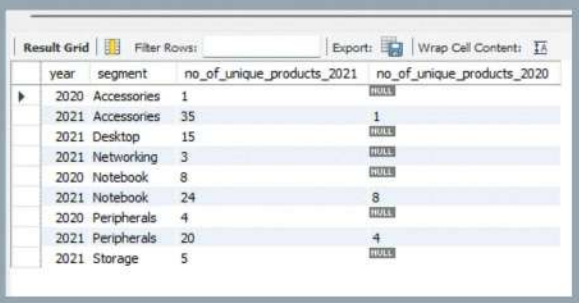
Result:



cost_year	segment	product_code	occurence
2021	Peripherals	A0321150302	1
2021	Peripherals	A0321150303	1
2020	Peripherals	A0418150101	1
2020	Peripherals	A0418150102	1
2020	Peripherals	A0418150107	1
2020	Peripherals	A0418150108	1
2021	Peripherals	A0721150403	1
2021	Peripherals	A0721150404	1
2021	Peripherals	A0821150501	1



year	segment	no_of_unique_products
2021	Peripherals	20
2020	Peripherals	4
2021	Accessories	35
2020	Accessories	1
2020	Notebook	8
2021	Notebook	24
2021	Desktop	15
2021	Storage	5
2021	Networking	3



year	segment	no_of_unique_products_2021	no_of_unique_products_2020
2020	Accessories	1	NULL
2021	Accessories	35	1
2021	Desktop	15	NULL
2021	Networking	3	NULL
2020	Notebook	8	NULL
2021	Notebook	24	8
2020	Peripherals	4	NULL
2021	Peripherals	20	4
2021	Storage	5	NULL

Submitted by M. Vamsi Kumar

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.

Code:

```
with cte4 as
(
with cte3 as
(
with cte2 as
(
with cte1 as
(
select product_code,count(product_code) as occurence,cost_year from
fact_manufacturing_cost
group by product_code
)
select cost_year,B.segment,A.product_code,occurence from cte1 as A
left join dim_product as B
on A.product_code = B.product_code
where occurence ='1'
)
select cost_year as year, segment,count(product_code) as no_of_unique_products from
cte2
group by cost_year,segment
)
select year,segment,no_of_unique_products as no_of_unique_products_2021,
lag(no_of_unique_products) over (partition by segment order by year) as
no_of_unique_products_2020 from cte3
)
select
segment,no_of_unique_products_2021,coalesce(no_of_unique_products_2020,'0') as
no_of_unique_products_2020,(no_of_unique_products_2021-
no_of_unique_products_2020) as difference from cte4
where year = '2021'
```

Final Result:

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

segment	no_of_unique_products_2021	no_of_unique_products_2020	difference
Accessories	35	1	34
Notebook	24	8	16
Peripherals	20	4	16
Desktop	15	0	NULL
Networking	3	0	NULL
Storage	5	0	NULL

Submitted by M. Vamsi Kumar

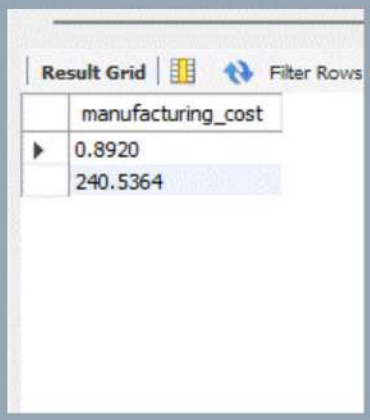
5. Get the products that have the highest and lowest manufacturing costs. The final output should contain these fields:

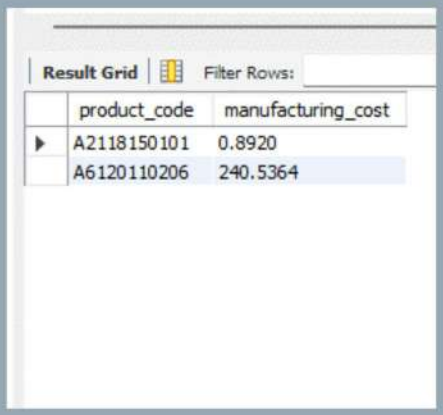
product_code, product, manufacturing_cost.

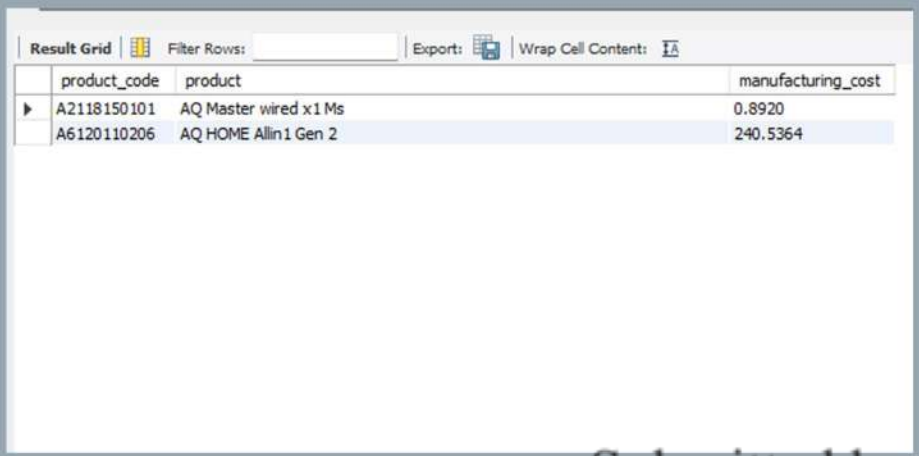
Code:

```
with cte2 as
(
with cte1 as
(
SELECT min(manufacturing_cost) as manufacturing_cost
FROM c4db.fact_manufacturing_cost
union
SELECT max(manufacturing_cost) as manufacturing_cost
FROM c4db.fact_manufacturing_cost
)
select product_code,B.manufacturing_cost from fact_manufacturing_cost as A
inner join cte1 as B
on A.manufacturing_cost = B.manufacturing_cost
)
select A.product_code,B.product,manufacturing_cost from cte2 as A
left join dim_product as B
on A.product_code = B.product_code
```

Result:







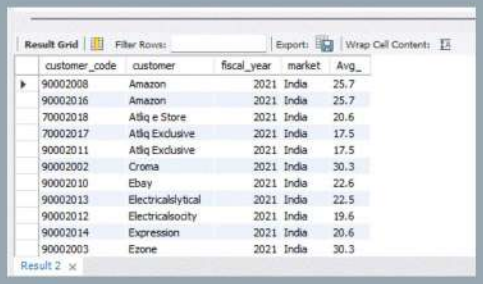
Submitted by M. Vamsi Kumar

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

Code:

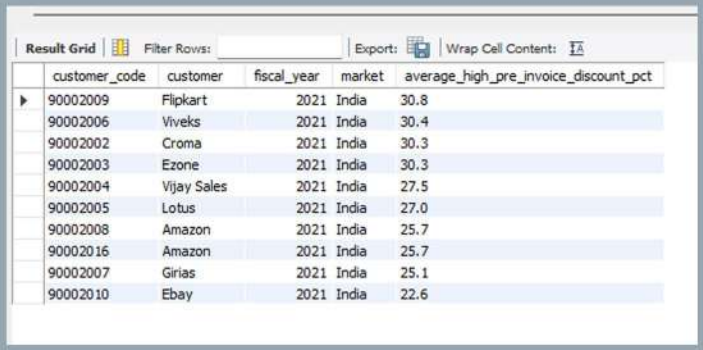
```
with cte3 as
(
with cte2 as
(
with cte1 as
(
select A.customer_code,B.customer,B.market,fiscal_year,pre_invoice_discount_pct
from fact_pre_invoice_deductions as A
left join dim_customer as B
on A.customer_code = B.customer_code
)
select customer_code, customer,
fiscal_year,market,round((avg(pre_invoice_discount_pct) over (partition by
fiscal_year,customer order by fiscal_year))*100,1) as Avg_
from cte1
where fiscal_year = '2021' and market = 'india'
)
select *, dense_rank() over (order by Avg_ desc) as _RANK_ from cte2
order by _RANK_ asc
)
select customer_code, customer,fiscal_year,market,Avg_ as
average_high_pre_invoice_discount_pct
from cte3 limit 10
```

Result:



customer_code	customer	fiscal_year	market	Avg_
90002008	Amazon	2021	India	25.7
90002016	Amazon	2021	India	25.7
70002018	Atliq e Store	2021	India	20.6
70002017	Atliq Exclusive	2021	India	17.5
90002011	Atliq Exclusive	2021	India	17.5
90002002	Croma	2021	India	30.3
90002010	Ebay	2021	India	22.6
90002013	Electricalslytical	2021	India	22.5
90002012	Electricalsocity	2021	India	19.6
90002014	Expression	2021	India	20.6
90002003	Ezone	2021	India	30.3

customer_code	customer	fiscal_year	market	Avg_	_RANK_
90002009	Flipkart	2021	India	30.8	1
90002006	Viveks	2021	India	30.4	2
90002002	Croma	2021	India	30.3	3
90002003	Ezone	2021	India	30.3	3
90002004	Vijay Sales	2021	India	27.5	4
90002005	Lotus	2021	India	27.0	5
90002008	Amazon	2021	India	25.7	6
90002016	Amazon	2021	India	25.7	6
90002007	Girias	2021	India	25.1	7
90002010	Ebay	2021	India	22.6	8
90002013	Electricals...	2021	India	22.5	9



customer_code	customer	fiscal_year	market	average_high_pre_invoice_discount_pct
90002009	Flipkart	2021	India	30.8
90002006	Viveks	2021	India	30.4
90002002	Croma	2021	India	30.3
90002003	Ezone	2021	India	30.3
90002004	Vijay Sales	2021	India	27.5
90002005	Lotus	2021	India	27.0
90002008	Amazon	2021	India	25.7
90002016	Amazon	2021	India	25.7
90002007	Girias	2021	India	25.1
90002010	Ebay	2021	India	22.6

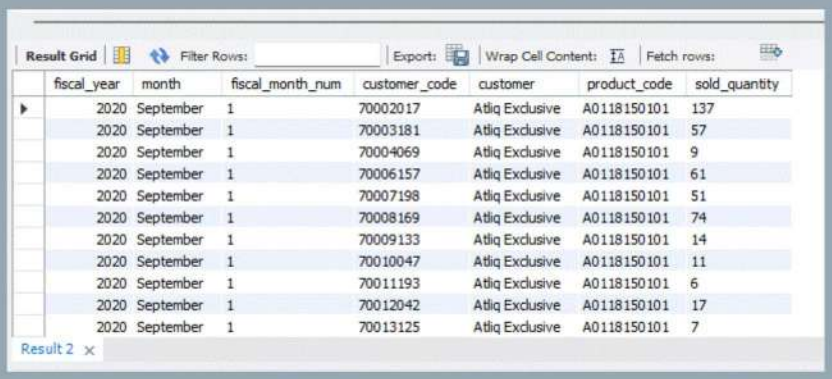
Submitted by M. Vamsi Kumar

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 and Gross sales Amount.

Code:

```
with cte2 as
(
with cte1 as
(
select fiscal_year,monthname(date) as month,month(date_add(date, interval 4 month))
as fiscal_month_num,A.customer_code,B.customer,product_code,sold_quantity
from fact_sales_monthly as A
left join dim_customer as B
on A.customer_code=B.customer_code
where customer = "Atliq Exclusive”
)
select
C.fiscal_year,month,fiscal_month_num,customer,C.product_code,C.sold_quantity,D.gros
s_price,round((sold_quantity*gross_price),2) as gross_sales from cte1 as C
left join fact_gross_price as D
on C.product_code = D.product_code and C.fiscal_year = D.fiscal_year
)
select fiscal_year,month,fiscal_month_num,customer,sum(gross_sales)/1000000 as
gross_sales_in_millions
from cte2
group by fiscal_year,month
```

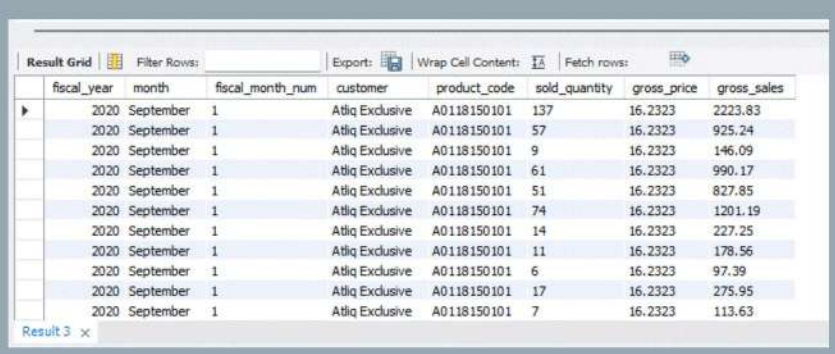
Result:



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

fiscal_year	month	fiscal_month_num	customer_code	customer	product_code	sold_quantity
2020	September	1	70002017	Atliq Exclusive	A0118150101	137
2020	September	1	70003181	Atliq Exclusive	A0118150101	57
2020	September	1	70004069	Atliq Exclusive	A0118150101	9
2020	September	1	70006157	Atliq Exclusive	A0118150101	61
2020	September	1	70007198	Atliq Exclusive	A0118150101	51
2020	September	1	70008169	Atliq Exclusive	A0118150101	74
2020	September	1	70009133	Atliq Exclusive	A0118150101	14
2020	September	1	70010047	Atliq Exclusive	A0118150101	11
2020	September	1	70011193	Atliq Exclusive	A0118150101	6
2020	September	1	70012042	Atliq Exclusive	A0118150101	17
2020	September	1	70013125	Atliq Exclusive	A0118150101	7

Result 2



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

fiscal_year	month	fiscal_month_num	customer	product_code	sold_quantity	gross_price	gross_sales
2020	September	1	Atliq Exclusive	A0118150101	137	16.2323	2223.83
2020	September	1	Atliq Exclusive	A0118150101	57	16.2323	925.24
2020	September	1	Atliq Exclusive	A0118150101	9	16.2323	146.09
2020	September	1	Atliq Exclusive	A0118150101	61	16.2323	990.17
2020	September	1	Atliq Exclusive	A0118150101	51	16.2323	827.85
2020	September	1	Atliq Exclusive	A0118150101	74	16.2323	1201.19
2020	September	1	Atliq Exclusive	A0118150101	14	16.2323	227.25
2020	September	1	Atliq Exclusive	A0118150101	11	16.2323	178.56
2020	September	1	Atliq Exclusive	A0118150101	6	16.2323	97.39
2020	September	1	Atliq Exclusive	A0118150101	17	16.2323	275.95
2020	September	1	Atliq Exclusive	A0118150101	7	16.2323	113.63

Result 3

Continued...Submitted by M. Vamsi Kumar

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 and Gross sales Amount.

Result:

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

fiscal_year	month	fiscal_month_num	customer_code	customer	product_code	sold_quantity
2020	September	1	70002017	Atliq Exclusive	A0118150101	137
2020	September	1	70003181	Atliq Exclusive	A0118150101	57
2020	September	1	70004069	Atliq Exclusive	A0118150101	9
2020	September	1	70006157	Atliq Exclusive	A0118150101	61
2020	September	1	70007198	Atliq Exclusive	A0118150101	51
2020	September	1	70008169	Atliq Exclusive	A0118150101	74
2020	September	1	70009133	Atliq Exclusive	A0118150101	14
2020	September	1	70010047	Atliq Exclusive	A0118150101	11
2020	September	1	70011193	Atliq Exclusive	A0118150101	6
2020	September	1	70012042	Atliq Exclusive	A0118150101	17
2020	September	1	70013125	Atliq Exclusive	A0118150101	7

Result 2

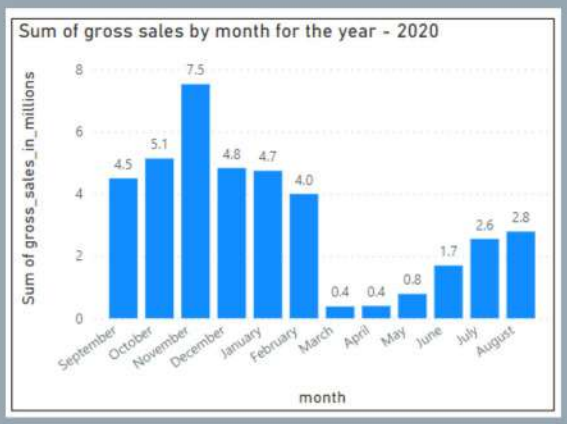
Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

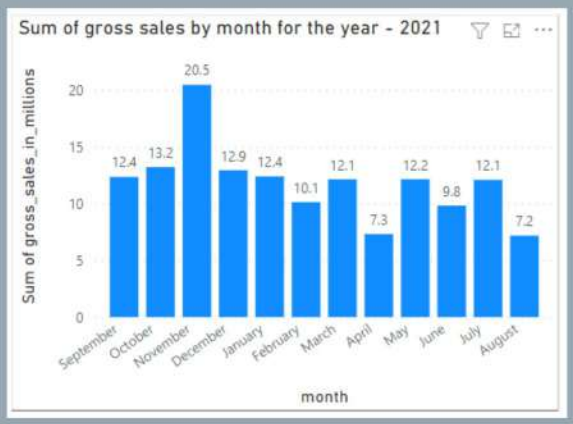
fiscal_year	month	fiscal_month_num	customer	product_code	sold_quantity	gross_price	gross_sales
2020	September	1	Atliq Exclusive	A0118150101	137	16.2323	2223.83
2020	September	1	Atliq Exclusive	A0118150101	57	16.2323	925.24
2020	September	1	Atliq Exclusive	A0118150101	9	16.2323	146.09
2020	September	1	Atliq Exclusive	A0118150101	61	16.2323	990.17
2020	September	1	Atliq Exclusive	A0118150101	51	16.2323	827.85
2020	September	1	Atliq Exclusive	A0118150101	74	16.2323	1201.19
2020	September	1	Atliq Exclusive	A0118150101	14	16.2323	227.25
2020	September	1	Atliq Exclusive	A0118150101	11	16.2323	178.56
2020	September	1	Atliq Exclusive	A0118150101	6	16.2323	97.39
2020	September	1	Atliq Exclusive	A0118150101	17	16.2323	275.95
2020	September	1	Atliq Exclusive	A0118150101	7	16.2323	113.63

Result 3

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

fiscal_year	month	fiscal_month_num	customer	gross_sales_in_millions
2020	September	1	Atliq Exclusive	4.496260
2020	October	2	Atliq Exclusive	5.135903
2020	November	3	Atliq Exclusive	7.522893
2020	December	4	Atliq Exclusive	4.830405
2020	January	5	Atliq Exclusive	4.740600
2020	February	6	Atliq Exclusive	3.996228
2020	March	7	Atliq Exclusive	0.378771
2020	April	8	Atliq Exclusive	0.395035
2020	May	9	Atliq Exclusive	0.783813
2020	June	10	Atliq Exclusive	1.695217
2020	July	11	Atliq Exclusive	2.551159
2020	August	12	Atliq Exclusive	2.786648
2021	September	1	Atliq Exclusive	12.353510
2021	October	2	Atliq Exclusive	13.218637
2021	November	3	Atliq Exclusive	20.465000
2021	December	4	Atliq Exclusive	12.944660
2021	January	5	Atliq Exclusive	12.399394
2021	February	6	Atliq Exclusive	10.129736
2021	March	7	Atliq Exclusive	12.144061
2021	April	8	Atliq Exclusive	7.312000
2021	May	9	Atliq Exclusive	12.150226
2021	June	10	Atliq Exclusive	9.824522
2021	July	11	Atliq Exclusive	12.092347
2021	August	12	Atliq Exclusive	7.178708





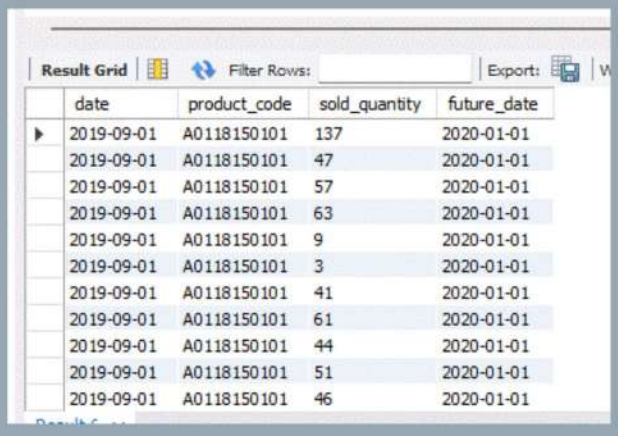
Submitted by M. Vamsi Kumar

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

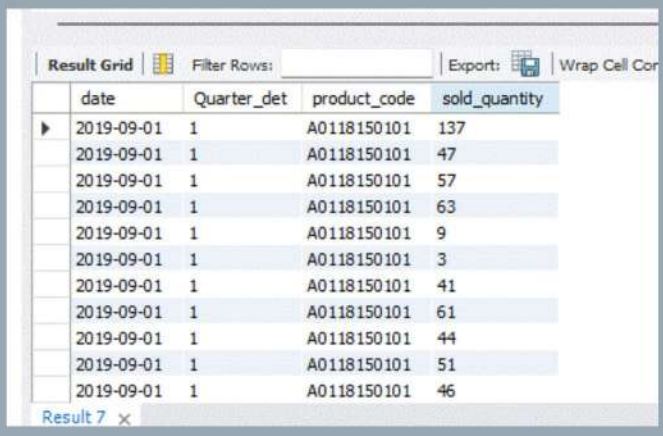
Code:

```
with cte2 as
(
with cte1 as
(
SELECT A.date,A.product_code,A.sold_quantity,date_add(date,INTERVAL 4 month) as
future_date
FROM c4db.fact_sales_monthly as A
where fiscal_year = '2020'
)
select date,ceil(month(future_date)/3) as Quarter_det,product_code,sold_quantity
from cte1
)
select
Case
when Quarter_det ='1' then 'Q1'
when Quarter_det ='2' then 'Q2'
when Quarter_det ='3' then 'Q3'
when Quarter_det ='4' then 'Q4'end as Quarter_num,
sum(sold_quantity) as total_sold_quantity from cte2
group by Quarter_num
order by total_sold_quantity desc
```

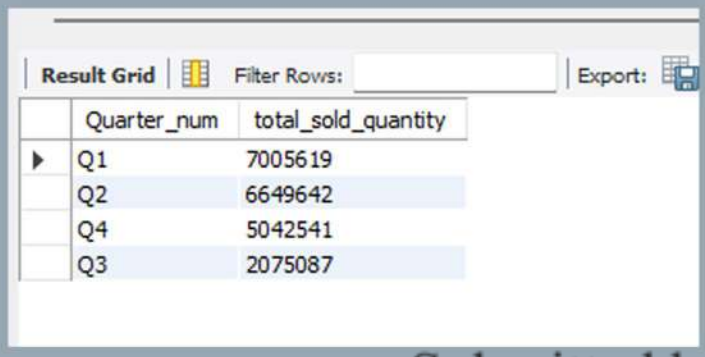
Result:



date	product_code	sold_quantity	future_date
2019-09-01	A0118150101	137	2020-01-01
2019-09-01	A0118150101	47	2020-01-01
2019-09-01	A0118150101	57	2020-01-01
2019-09-01	A0118150101	63	2020-01-01
2019-09-01	A0118150101	9	2020-01-01
2019-09-01	A0118150101	3	2020-01-01
2019-09-01	A0118150101	41	2020-01-01
2019-09-01	A0118150101	61	2020-01-01
2019-09-01	A0118150101	44	2020-01-01
2019-09-01	A0118150101	51	2020-01-01
2019-09-01	A0118150101	46	2020-01-01



date	Quarter_det	product_code	sold_quantity
2019-09-01	1	A0118150101	137
2019-09-01	1	A0118150101	47
2019-09-01	1	A0118150101	57
2019-09-01	1	A0118150101	63
2019-09-01	1	A0118150101	9
2019-09-01	1	A0118150101	3
2019-09-01	1	A0118150101	41
2019-09-01	1	A0118150101	61
2019-09-01	1	A0118150101	44
2019-09-01	1	A0118150101	51
2019-09-01	1	A0118150101	46



Quarter_num	total_sold_quantity
Q1	7005619
Q2	6649642
Q4	5042541
Q3	2075087

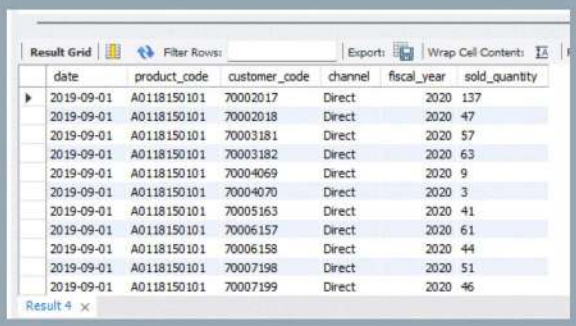
Submitted by M. Vamsi Kumar

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.

Code:

```
with cte4 as
(
with cte3 as
(
with cte2 as
(
with cte1 as
(
SELECT A.date,A.product_code,A.customer_code,B.channel,fiscal_year,sold_quantity
FROM c4db.fact_sales_monthly as A
left join dim_customer as B
on A.customer_code=B.customer_code
)
select
channel,C.product_code,C.fiscal_year,sold_quantity,D.gross_price,(sold_quantity*D.gros
s_price) as Gross_sales from cte1 as C
left join fact_gross_price as D
on C.product_code = D.product_code and C.fiscal_year = D.fiscal_year
)
select channel,sum(Gross_sales) as Total_sales from cte2
where fiscal_year = '2021'
group by channel
)
select channel,Total_sales,sum(Total_sales) over (order by channel rows between
unbounded preceding and unbounded following) as Grand_total
from cte3
)
select channel,Total_sales as
gross_sales,Grand_total,round(((Total_sales/Grand_total)*100),2) as Perc  from cte4
```

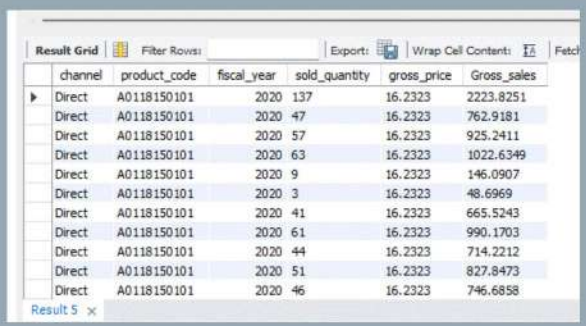
Result:



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

date	product_code	customer_code	channel	fiscal_year	sold_quantity
2019-09-01	A0118150101	70002017	Direct	2020	137
2019-09-01	A0118150101	70002018	Direct	2020	47
2019-09-01	A0118150101	70003181	Direct	2020	57
2019-09-01	A0118150101	70003182	Direct	2020	63
2019-09-01	A0118150101	70004069	Direct	2020	9
2019-09-01	A0118150101	70004070	Direct	2020	3
2019-09-01	A0118150101	70005163	Direct	2020	41
2019-09-01	A0118150101	70006157	Direct	2020	61
2019-09-01	A0118150101	70006158	Direct	2020	44
2019-09-01	A0118150101	70007198	Direct	2020	51
2019-09-01	A0118150101	70007199	Direct	2020	46

Result 4



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

channel	product_code	fiscal_year	sold_quantity	gross_price	Gross_sales
Direct	A0118150101	2020	137	16.2323	2223.8251
Direct	A0118150101	2020	47	16.2323	762.9181
Direct	A0118150101	2020	57	16.2323	925.2411
Direct	A0118150101	2020	63	16.2323	1022.6349
Direct	A0118150101	2020	9	16.2323	146.0907
Direct	A0118150101	2020	3	16.2323	48.6969
Direct	A0118150101	2020	41	16.2323	665.5243
Direct	A0118150101	2020	61	16.2323	990.1703
Direct	A0118150101	2020	44	16.2323	714.2212
Direct	A0118150101	2020	51	16.2323	827.8473
Direct	A0118150101	2020	46	16.2323	746.6858

Result 5

Continued...

Submitted by M. Vamsi Kumar

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.

Result:

channel	Total_sales
Direct	257532002.6536
Distributor	188025630.9348
Retailer	1219081639.9472

Result Grid | Filter Rows: | Export:

channel	Total_sales	Grand_total
Direct	257532002.6536	1664639273.5356
Distributor	188025630.9348	1664639273.5356
Retailer	1219081639.9472	1664639273.5356

Result Grid | Filter Rows: | Export: | Wrap

channel	gross_sales	Grand_total	Perc
Direct	257532002.6536	1664639273.5356	15.47
Distributor	188025630.9348	1664639273.5356	11.30
Retailer	1219081639.9472	1664639273.5356	73.23

Submitted by M. Vamsi Kumar

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.

Code:

```
with cte1 as
(
SELECT A.fiscal_year,B.division,A.product_code,B.product,sum(sold_quantity) as
total_sold_quantity FROM c4db.fact_sales_monthly as A
left join dim_product as B
on A.product_code=B.product_code
group by fiscal_year,product_code
)
select *, dense_rank() over (order by total_sold_quantity desc) as Rank_order from cte1
where fiscal_year ='2021'  limit 5
```

Result:

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

fiscal_year	division	product_code	product	total_sold_quantity
2020	P & A	A0118150101	AQ Dracula HDD – 3.5 Inch SATA 6 Gb/s 5400 RPM 256 MB Cache	62631
2020	P & A	A0118150102	AQ Dracula HDD – 3.5 Inch SATA 6 Gb/s 5400 RPM 256 MB Cache	63902
2020	P & A	A0118150103	AQ Dracula HDD – 3.5 Inch SATA 6 Gb/s 5400 RPM 256 MB Cache	61748
2020	P & A	A0118150104	AQ Dracula HDD – 3.5 Inch SATA 6 Gb/s 5400 RPM 256 MB Cache	60961
2020	P & A	A0219150201	AQ WereWolf NAS Internal Hard Drive HDD – 8.89 cm	62464
2020	P & A	A0219150202	AQ WereWolf NAS Internal Hard Drive HDD – 8.89 cm	62827
2020	P & A	A0220150203	AQ WereWolf NAS Internal Hard Drive HDD – 8.89 cm	63680
2020	P & A	A0320150301	AQ Zion Saga	64502
2020	P & A	A0418150101	AQ Mforce Gen X	45758
2020	P & A	A0418150102	AQ Mforce Gen X	46530
2020	P & A	A0418150103	AQ Mforce Gen X	45581

Result 2

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

fiscal_year	division	product_code	product	total_sold_quantity	Rank_order
2021	N & S	A6720160103	AQ Pen Drive 2 IN 1	701373	1
2021	N & S	A6818160202	AQ Pen Drive DRC	688003	2
2021	N & S	A6819160203	AQ Pen Drive DRC	676245	3
2021	N & S	A6818160201	AQ Pen Drive DRC	670321	4
2021	P & A	A2319150302	AQ Gamers Ms	428498	5

Submitted by M. Vamsi Kumar