CODEBASICS RESUME PROJECT CHALLENGE #4

Provide Insights to Management in Consumer Goods Domain

By: Akash Kumar

AD HOC INSIGHTS

For AtliQ Hardware

Problem Statement

AtliQ Hardware, a major computer hardware manufacturer in India, needed better insights to make data-driven decisions. The company sells products through various channels like Amazon, Flipkart, Walmart, and direct stores. They faced challenges due to a lack of a dedicated data analytics team and needed quick, actionable insights.

Objectives

- Deliver quick, actionable data insights.
- Analyze sales across channels and regions.
- Support strategic planning with ad-hoc analysis.

Project Execution

- Data Import: Imported and cleaned data in MySQL Workbench.
- EDA: Explored and analyzed data for key insights.
- SQL Queries: Created queries for sales performance, product analysis.

Ad hoc -1

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

```
FROM dim_customer
WHERE
    customer LIKE 'Atliq Exclusive'
    AND region = 'APAC';
```

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

customer_code	customer	platform	channel	market	sub_zone	region
70002017	Atliq Exclusive	Brick & Mortar	Direct	India	India	APAC
70003181	70003181 sive	Brick & Mortar	Direct	Indonesia	ROA	APAC
70004069	Atliq Exclusive	Brick & Mortar	Direct	Japan	ROA	APAC
70006157	Atliq Exclusive	Brick & Mortar	Direct	Philiphines	ROA	APAC
70007198	Atliq Exclusive	Brick & Mortar	Direct	South Korea	ROA	APAC
70008169	Atliq Exclusive	Brick & Mortar	Direct	Australia	ANZ	APAC
70009133	Atliq Exclusive	Brick & Mortar	Direct	Newzealand	ANZ	APAC

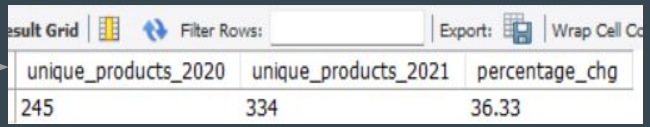
Ad hoc - 2

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.

```sql
create temporary table p20(
select
    count(distinct(product_code)) as unique_products_2020
from fact_sales_monthly
where fiscal_year = 2020);


create temporary table p21(
select
    count(distinct(product_code)) as unique_products_2021
from fact_sales_monthly
where fiscal_year = 2021);

select
    *,
    round((unique_products_2021-unique_products_2020)*100/unique_products_2020,2) as percentage_chg
from p20
join p21
```

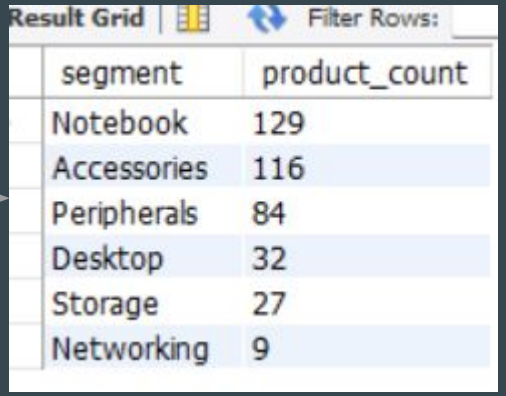


unique_products_2020	unique_products_2021	percentage_chg
245	334	36.33

Ad hoc - 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.

```sql
SELECT
    segment, COUNT(DISTINCT(product_code)) AS product_count
FROM
    dim_product
GROUP BY segment
ORDER BY product_count DESC;
```

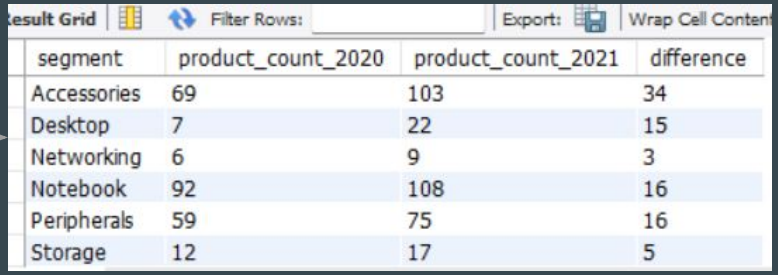


segment	product_count
Notebook	129
Accessories	116
Peripherals	84
Desktop	32
Storage	27
Networking	9

Ad hoc - 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.

```sql
create temporary table t20 (
select
    distinct(segment),
    count(distinct(s.product_code)) as product_count_2020
from fact_sales_monthly s
join dim_product p
using (product_code)
where fiscal_year = 2020
group by segment)
;
create temporary table t21 (
select
    distinct(segment),
    count(distinct(s.product_code)) as product_count_2021
from fact_sales_monthly s
join dim_product p
using (product_code)
where fiscal_year = 2021
group by segment)
;
select
    *,
    (product_count_2021 - product_count_2020) as difference
from t20 a
join t21 b
using (segment)
```

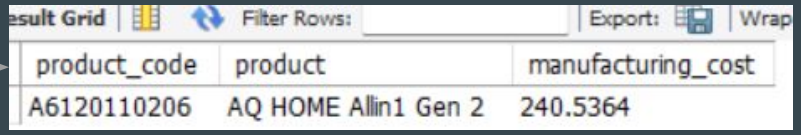


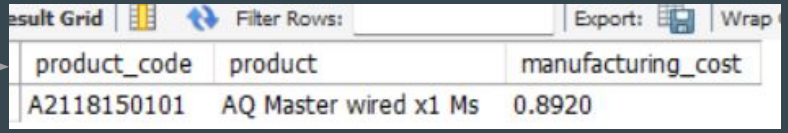
segment	product_count_2020	product_count_2021	difference
Accessories	69	103	34
Desktop	7	22	15
Networking	6	9	3
Notebook	92	108	16
Peripherals	59	75	16
Storage	12	17	5

Ad hoc - 5

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

```sql
-- Highest manufacturing cost
select
    m.product_code,
    p.product,
    m.manufacturing_cost
from fact_manufacturing_cost m
join dim_product p
using (product_code)
where manufacturing_cost = (
        select max(manufacturing_cost)
        from fact_manufacturing_cost)
;
-- Lowest manufacturing cost
select
    m.product_code,
    p.product,
    m.manufacturing_cost
from fact_manufacturing_cost m
join dim_product p
using (product_code)
where manufacturing_cost = (
        select min(manufacturing_cost)
        from fact_manufacturing_cost)
;
```





Ad hoc - 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.

```sql
SELECT
    p.customer_code,
    GROUP_CONCAT(c.customer) AS customer,
    ROUND(AVG(pre_invoice_discount_pct), 2) AS average_discount_percentage
FROM
    fact_pre_invoice_deductions p
        JOIN
    dim_customer c USING (customer_code)
WHERE
    fiscal_year = 2021 AND market = 'India'
GROUP BY p.customer_code
ORDER BY average_discount_percentage DESC
LIMIT 5;
```

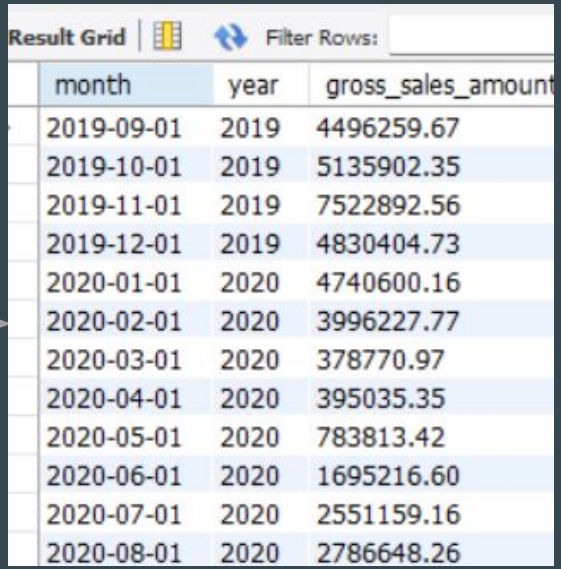
Result Grid | Filter Rows: | Export:

customer_code	customer	average_discount_percentage
90002009	Flipkart	0.31
90002002	Croma	0.30
90002003	Ezone	0.30
90002006	Viveks	0.30
90002016	Amazon	0.29

Ad hoc - 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.

```sql
select
    distinct(s.date) as month,
    year(s.date) as year,
    round(sum(sold_quantity*gross_price),2) as gross_sales_amount
from fact_sales_monthly s
join fact_gross_price g
using (product_code, fiscal_year)
join dim_customer c
using (customer_code)
where customer like "%Atliq Exclusive%"
group by s.date
order by year asc;
```



month	year	gross_sales_amount
2019-09-01	2019	4496259.67
2019-10-01	2019	5135902.35
2019-11-01	2019	7522892.56
2019-12-01	2019	4830404.73
2020-01-01	2020	4740600.16
2020-02-01	2020	3996227.77
2020-03-01	2020	378770.97
2020-04-01	2020	395035.35
2020-05-01	2020	783813.42
2020-06-01	2020	1695216.60
2020-07-01	2020	2551159.16
2020-08-01	2020	2786648.26

Ad hoc - 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.

```sql
with cte1 as (
select
    *,
    month(date_add(date, interval 4 month)) as fiscal_month
from fact_sales_monthly
where fiscal_year = 2020)
,
cte2 as (
select
    *,
    case
        when fiscal_month in (1,2,3) then "Q1"
        when fiscal_month in (4,5,6) then "Q2"
        when fiscal_month in (7,8,9) then "Q3"
        else "Q4"
        end as qtr
from cte1)
select
    qtr,
    sum(sold_quantity) as total_sold_quantity
from cte2
group by qtr
order by total_sold_quantity desc
```

Result Grid | Filter Rows:

qtr	total_sold_quantity
Q1	7005619
Q2	6649642
Q4	5042541
Q3	2075087

Ad hoc - 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.

```sql
with cte1 as (
select
    channel,
    round(sum((sold_quantity*gross_price)/1000000),0) as gross_sales_mln
from fact_sales_monthly s
join fact_gross_price g
using (product_code, fiscal_year)
join dim_customer c
using (customer_code)
where fiscal_year = 2021
group by channel)

select
    *,
    gross_sales_mln*100/sum(gross_sales_mln) over() as pct_share
from cte1
```

Result Grid | Filter Rows: | Expo

channel	gross_sales_mln	pct_share
Direct	258	15.4955
Retailer	1219	73.2132
Distributor	188	11.2913

Ad hoc - 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.

```sql
with cte1 as (
select
    division,
    p.product_code,
    product,
    sum(sold_quantity) as total_sold_quantity
from fact_sales_monthly s
join dim_product p
using (product_code)
where fiscal_year = 2021
group by division, p.product_code, product),
cte2 as (
select
    *,
    dense_rank() over(partition by division order by total_sold_quantity desc) as rank_order
from cte1)

select * from cte2
where rank_order <=3
```

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

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

THANK YOU