Sales Performance

SQL+POWER BI

Business Objective(This project will solve business problems)

- How are sales, revenue and profit trending over time?
- Which product, regions, and sales representative drive performance?
- Where are leakeges (Low profit, declining sales)?
- How can management optimize strategies?

Total sales, profit, and profit margin

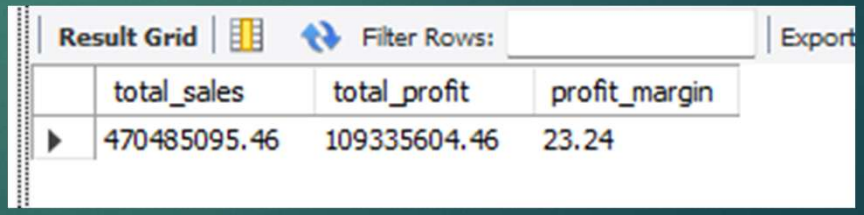
- select

  round(sum(sales_amount),2) as total_sales,

  round(sum(sales_amount-cost_amount),2) as total_profit,

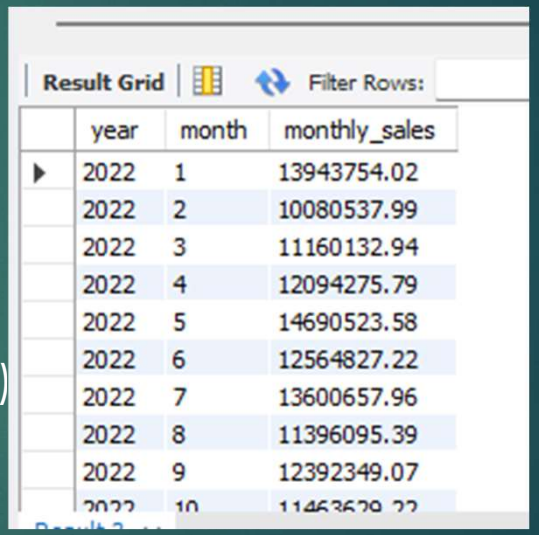
  round(sum(sales_amount-cost_amount)*100/sum(sales_amount),2) as profit_margin

  from ecommerce_sales_dataset;



Monthly sales trend

- select
  Year(order_date) as year,
  month(order_date) as month,
  round(sum(sales_amount),2) as monthly_sales
  from ecommerce_sales_dataset
  group by Year(order_date), month(order_date)
  order by year, month;



year	month	monthly_sales
2022	1	13943754.02
2022	2	10080537.99
2022	3	11160132.94
2022	4	12094275.79
2022	5	14690523.58
2022	6	12564827.22
2022	7	13600657.96
2022	8	11396095.39
2022	9	12392349.07
2022	10	11463629.22

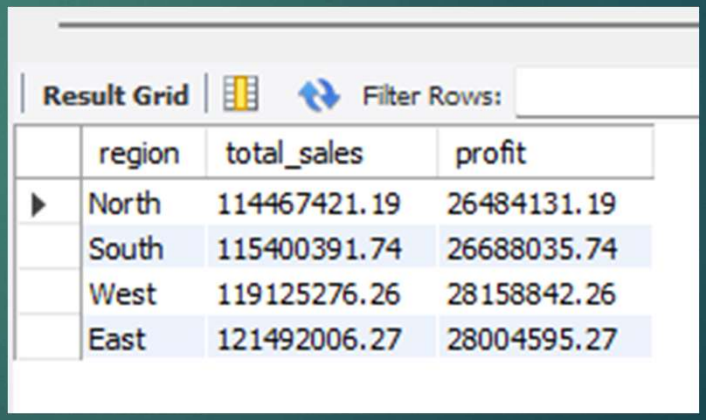
Sales by region

- select region,round(sum(sales_amount),2) as total_sales,
  round(sum(sales_amount-cost_amount),2) as profit

  from ecommerce_sales_dataset

  group by region

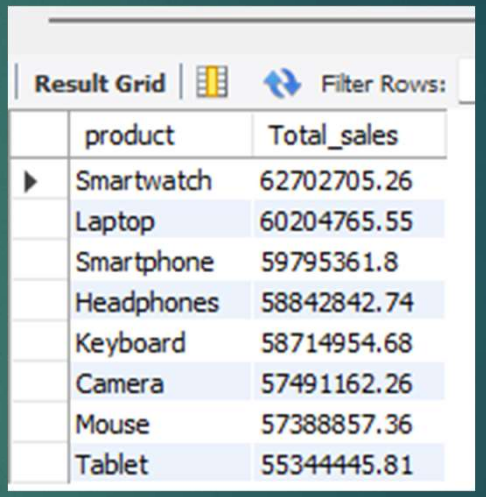
  order by sum(sales_amount);


Result Grid | Filter Rows:

region	total_sales	profit
North	114467421.19	26484131.19
South	115400391.74	26688035.74
West	119125276.26	28158842.26
East	121492006.27	28004595.27

Top 10 product by sales

- select product,
  round(sum(sales_amount),2) as Total_sales
  from ecommerce_sales_dataset
  group by product
  order by sum(sales_amount) desc
  limit 10;



Result Grid | Filter Rows:

product	Total_sales
Smartwatch	62702705.26
Laptop	60204765.55
Smartphone	59795361.8
Headphones	58842842.74
Keyboard	58714954.68
Camera	57491162.26
Mouse	57388857.36
Tablet	55344445.81

Best Sales representatives

- select sales_representative,

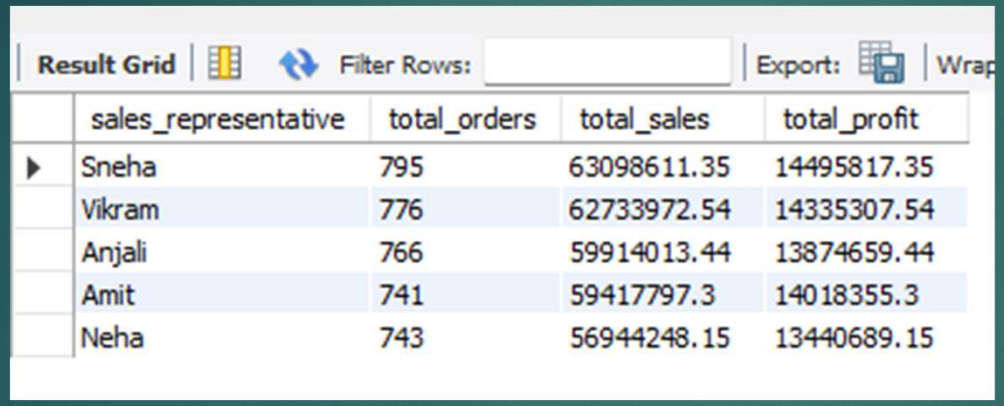
  count(distinct order_id) as total_orders,
round(sum(sales_amount),2) as total_sales,
round(sum(sales_amount-cost_amount),2) as total_profit

  from ecommerce_sales_dataset

  group by sales_representative

  order by sum(sales_amount) desc

  limit 5;


Result Grid | Filter Rows: | Export: | Wrap

sales_representative	total_orders	total_sales	total_profit
Sneha	795	63098611.35	14495817.35
Vikram	776	62733972.54	14335307.54
Anjali	766	59914013.44	13874659.44
Amit	741	59417797.3	14018355.3
Neha	743	56944248.15	13440689.15

YOY Growth

- select

year(order_date) as year,

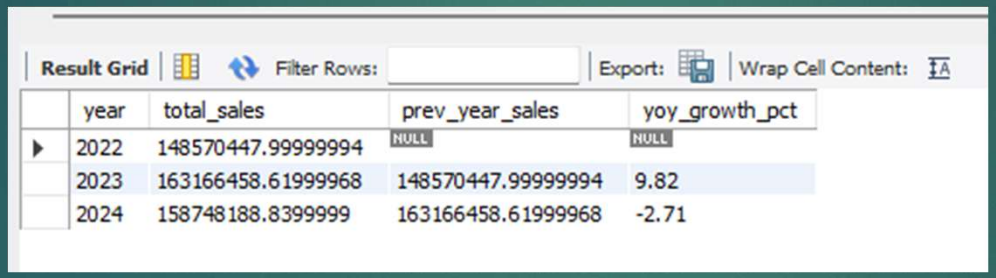
sum(sales_amount) as total_sales,
Lag(sum(sales_amount))over(order by year(order_date)) as prev_year_sales,

round((sum(sales_amount)-Lag(sum(sales_amount))over(order by year(order_date)))/(Lag(sum(sales_amount))over(order by year(order_date)))*100,2) as yoy_growth_pct

from ecommerce_sales_dataset

group by year(order_date)

order by year;



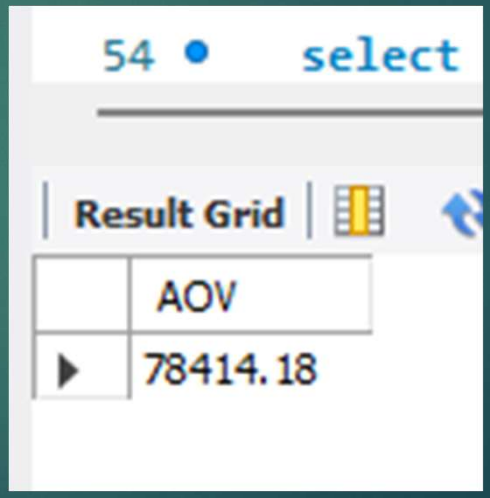
year	total_sales	prev_year_sales	yoy_growth_pct
2022	148570447.99999994	NULL	NULL
2023	163166458.61999968	148570447.99999994	9.82
2024	158748188.8399999	163166458.61999968	-2.71

Average Order values

- select

  round((round(sum(sales_amount),2)) / count(distinct order_id),2) as AOV

  from ecommerce_sales_dataset;



Power BI dashboard design

- DAX for KPI cards
- Total Sales = SUM(orders[sales_amount])
- Total Profit = SUM(orders[sales_amount]) - SUM(orders[cost])
- Profit Margin % = DIVIDE([Total Profit], [Total Sales], 0)
- AOV = DIVIDE([Total Sales], DISTINCTCOUNT(orders[order_id]))
- MoM Growth % =

  DIVIDE([Total Sales] - CALCULATE([Total Sales], PREVIOUSMONTH(orders[order_date])),

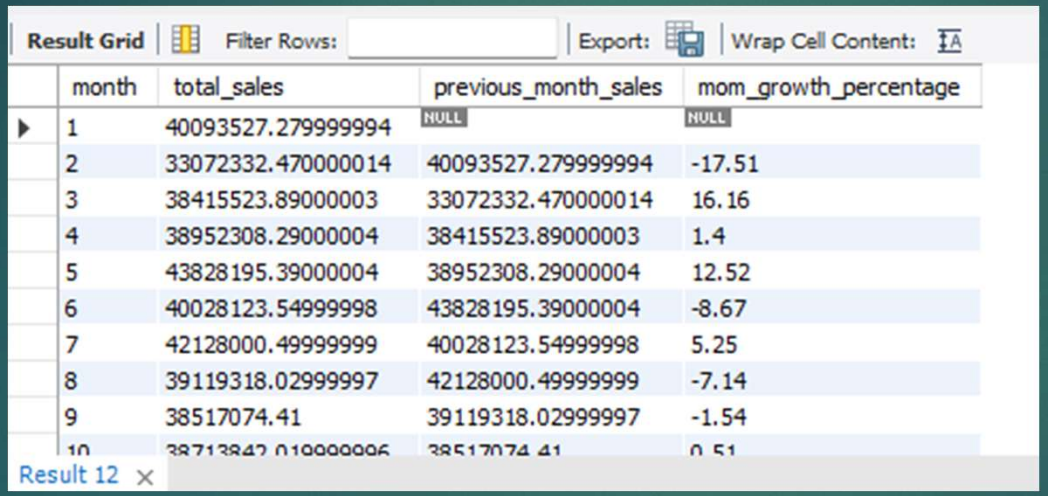
  CALCULATE([Total Sales], PREVIOUSMONTH(orders[order_date])))

Dashboard overview

- KPI cards – Sales, Profit, Profit Margin
- Slicers – Product, Region, and Representative
- Line chart – monthly sales trend
- Bar chart – Sales by region
- Pie chart - Category wise sales
- Donut chart - Sales by Representative
- Column chart – Sales by product type
- Bar chart – Total profit by years

Monthly Sales with MoM Growth (CTE + LAG)

```
with monthly_sales as (    select        month(order_date) as month,
sum(sales_amount) as total_sales
 from ecommerce_sales_dataset
 group by month(order_date))
select    month, total_sales, lag(total_sales)
over (order by month) as previous_month_sales,
 round((total_sales-lag(total_sales)over(order by month))*100.0
/lag(total_sales)over(order by month),2) as mom_growth_percentage
from monthly_sales     order by month;
```



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

month	total_sales	previous_month_sales	mom_growth_percentage
1	40093527.279999994	NULL	NULL
2	33072332.470000014	40093527.279999994	-17.51
3	38415523.89000003	33072332.470000014	16.16
4	38952308.29000004	38415523.89000003	1.4
5	43828195.39000004	38952308.29000004	12.52
6	40028123.54999998	43828195.39000004	-8.67
7	42128000.49999999	40028123.54999998	5.25
8	39119318.02999997	42128000.49999999	-7.14
9	38517074.41	39119318.02999997	-1.54
10	38713842.019999996	38517074.41	0.51

Result 12

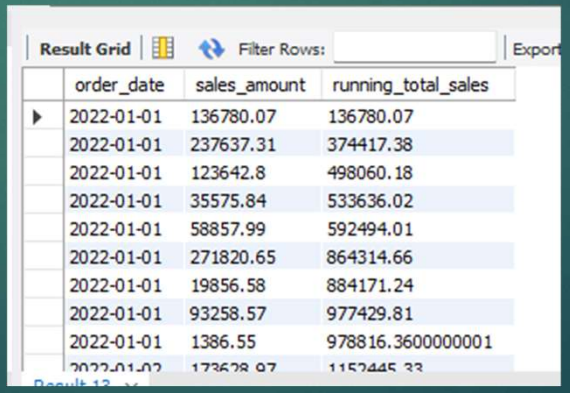
Running Total Sales (Window Function)

- select order_date, sales_amount, sum(sales_amount)over(order by order_date rows between unbounded preceding and current row) as running_total_sales

  from

  ecommerce_sales_dataset

  order by order_date;



order_date	sales_amount	running_total_sales
2022-01-01	136780.07	136780.07
2022-01-01	237637.31	374417.38
2022-01-01	123642.8	498060.18
2022-01-01	35575.84	533636.02
2022-01-01	58857.99	592494.01
2022-01-01	271820.65	864314.66
2022-01-01	19856.58	884171.24
2022-01-01	93258.57	977429.81
2022-01-01	1386.55	978816.3600000001
2022-01-02	173628.97	1152445.33

Top 3 Products per Region (RANK)

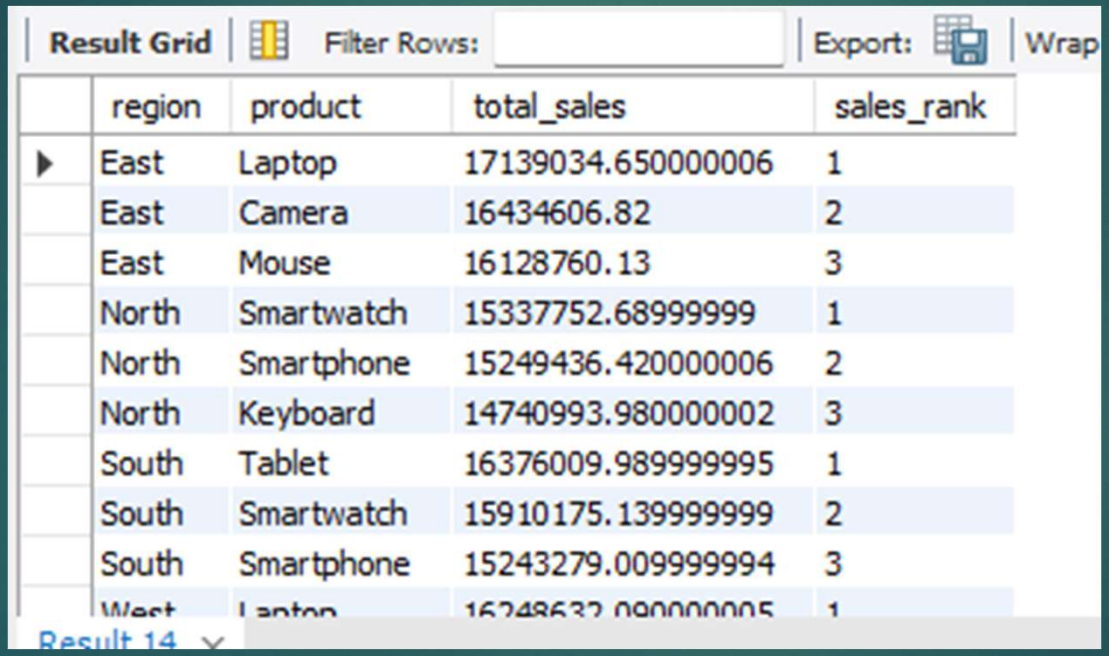
- with product_sales as (select region, product, sum(sales_amount) as total_sales

  from ecommerce_sales_dataset

  group by region, product),ranked_products as (select region, product, total_sales, rank() over (partition by region order by total_sales desc) as sales_rank from product_sales)

  select *from ranked_productswhere sales_rank <= 3

  order by region, sales_rank;


Result Grid | Filter Rows: | Export: | Wrap

region	product	total_sales	sales_rank
East	Laptop	17139034.650000006	1
East	Camera	16434606.82	2
East	Mouse	16128760.13	3
North	Smartwatch	15337752.68999999	1
North	Smartphone	15249436.420000006	2
North	Keyboard	14740993.980000002	3
South	Tablet	16376009.989999995	1
South	Smartwatch	15910175.139999999	2
South	Smartphone	15243279.009999994	3
West	Laptop	16248632.090000005	1

Result 14

Sales Rep Performance vs Regional Average

- with rep_sales as (

  select region, sales_representative, sum(sales_amount)

  as rep_sales from ecommerce_sales_dataset

  group by region, sales_representative)

  select region, sales_representative, rep_sales,

  round(avg(rep_sales)over(partition by region), 2) as regional_avg_sales, rep_sales - avg(rep_sales) over

  (partition by region) as performance_gapfrom rep_salesorder by region, performance_gap desc;

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

region	sales_representative	rep_sales	regional_avg_sales	performance_gap
East	Anjali	17320042.350000005	15186500.78	2133541.566250004
East	Neha	16414832.330000004	15186500.78	1228331.5462500025
East	Sneha	16084286.600000005	15186500.78	897785.8162500039
East	Amit	15281060.330000002	15186500.78	94559.5462500006
East	Karan	14617905.540000003	15186500.78	-568595.2437499985

Result 4

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

region	sales_representative	rep_sales	regional_avg_sales	performance_gap
North	Sneha	17258681.069999997	14308427.65	2950253.4212499987
North	Vikram	15775248.970000004	14308427.65	1466821.3212500066
North	Amit	15634578.959999986	14308427.65	1326151.3112499882
North	Anjali	14260451.709999997	14308427.65	-47975.93875000067
North	Neha	14025921.200000001	14308427.65	-282506.4487499967