SALES ANALYSIS 2021-2024

🔍 1. View Creation

-- Create a unified view combining two years of sales data with a source identifier

```
CREATE VIEW sales_analysis AS
SELECT *, 'year_21_22' AS source_year FROM sales_2021_2022
UNION ALL
SELECT *, 'year_23_24' AS source_year FROM sales_2023_2024;
```

👥 2. Customer Analysis

-- Total number of customers

```
SELECT COUNT(*) AS total_customers FROM sales_analysis;
```

-- Total customers by gender

```
SELECT gender, COUNT(customer_id) AS total_customers
FROM sales_analysis
GROUP BY gender;
```

-- Total customers by year and gender

```
SELECT YEAR(date) AS year, gender, COUNT(customer_id) AS total_customers
FROM sales_analysis
GROUP BY YEAR(date), gender;
```

-- Total customers by year and month

```
SELECT YEAR(date) AS year, MONTH(date) AS month, COUNT(customer_id) AS total_customers
FROM sales_analysis
GROUP BY YEAR(date), MONTH(date)
ORDER BY month;
```

-- Total customers by product category

```
SELECT product_category, COUNT(customer_id) AS total_customers
FROM sales_analysis
GROUP BY product_category
ORDER BY total_customers DESC;
```

3. Demographics

-- Average age of customers by product category

```
SELECT product_category, AVG(age) AS Avg_Age
FROM sales_analysis
GROUP BY product_category;
```

4. Quantity Sold

-- Total quantity sold

```
SELECT SUM(quantity) AS total_quantity_sold FROM sales_analysis;
```

-- Quantity sold by product category

```
SELECT product_category, SUM(quantity) AS total_quantity_sold
FROM sales_analysis
GROUP BY product_category;
```

-- Average quantity sold by product category

```
SELECT product_category, AVG(quantity) AS avg_quantity_sold
FROM sales_analysis
GROUP BY product_category;
```

-- Quantity sold by year

```
SELECT YEAR(date) AS year, SUM(quantity) AS quantity_sold
FROM sales_analysis
GROUP BY year;
```

-- Min and max quantity sold by product category and year

```
SELECT product_category, year
MIN(quantity) AS min_quantity_sold,
MAX(quantity) AS max_quantity_sold
FROM (
   SELECT product_category, YEAR(date) AS year, quantity
   FROM sales_analysis
) AS sub
GROUP BY product_category, year;
```

💰 5. Price Analysis

-- Average price per unit

```
SELECT ROUND(AVG(price_per_unit), 2) AS avg_priceperunit FROM sales_analysis;
```

-- Min and max price per unit by year

```
SELECT YEAR(date) AS year, MIN(price_per_unit) AS min_price, MAX(price_per_unit) AS max_price
FROM sales_analysis
GROUP BY year;
```

-- Average price per unit by product category

```
SELECT product_category, ROUND(AVG(price_per_unit)) AS avg_priceperunit
FROM sales_analysis
GROUP BY product_category;
```

-- Min and max price per unit by product category and year

```
SELECT product_category, YEAR(date) AS year,
    MIN(price_per_unit) AS minimum_price,
    MAX(price_per_unit) AS maximum_price
FROM sales_analysis
GROUP BY product_category, year
ORDER BY year;
```

🧾 6. Cost Analysis

-- Average cost per unit

```
SELECT ROUND(AVG(cost_per_unit), 2) AS avg_costperunit FROM sales_analysis;
```

-- Min and max cost per unit by year

```
SELECT YEAR(date) AS year, MIN(cost_per_unit) AS min_cost, MAX(cost_per_unit) AS max_cost
FROM sales_analysis
GROUP BY year;
```

-- Average cost per unit by product category

```
SELECT product_category, ROUND(AVG(cost_per_unit)) AS avg_costperunit
FROM sales_analysis
GROUP BY product_category;
```

-- Min and max cost per unit by product category

```
SELECT product_category,
    MIN(cost_per_unit) AS minimum_cost,
    MAX(cost_per_unit) AS maximum_cost
FROM sales_analysis
GROUP BY product_category;
```

☑ 7. Profit Analysis

-- Average profit per unit

```
SELECT ROUND(AVG(profit_per_unit), 2) AS avg_profitperunit FROM sales_analysis;
```

-- Min and max profit per unit by year

```
SELECT YEAR(date) AS year, MIN(profit_per_unit) AS min_profit, MAX(profit_per_unit) AS max_profit
FROM sales_analysis
GROUP BY year;
```

-- Average profit per unit by product category

```
SELECT product_category, ROUND(AVG(profit_per_unit)) AS avg_profitperunit
FROM sales_analysis
GROUP BY product_category;
```

-- Min and max profit per unit by product category

```
SELECT product_category,
   MIN(profit_per_unit) AS minimum_price,
    MAX(profit_per_unit) AS maximum_price
FROM sales_analysis
GROUP BY product_category;
```

🏧 8. Revenue Analysis

-- Total revenue

```
SELECT SUM(total_amount) AS Total_Revenue FROM sales_analysis;
```

-- Total revenue by year

```
SELECT YEAR(date) AS year, SUM(total_amount) AS Total_Revenue
FROM sales_analysis
GROUP BY year;
```

-- Total revenue by product category and year

```
SELECT product_category, SUM(total_amount) AS Total_Revenue, YEAR(date) AS year
FROM sales_analysis
GROUP BY product_category, year;
```

-- Revenue growth rate year-over-year

```
SELECT curr.year AS current_year,
    ROUND(((curr.total_amount - prev.total_amount) / NULLIF(prev.total_amount, 0)) * 100,
2) AS revenue_growth_rate
FROM (
  SELECT YEAR(date) AS year, SUM(total_amount) AS total_amount
  FROM sales_analysis
  GROUP BY YEAR(date)
) AS curr
JOIN (
  SELECT YEAR(date) AS year, SUM(total_amount) AS total_amount
  FROM sales_analysis
  GROUP BY YEAR(date)
) AS prev ON curr.year = prev.year + 1;
```

-- Revenue percentage contribution by year

```
SELECT YEAR(date) AS year,
    SUM(total_amount) AS revenue,
    ROUND((SUM(total_amount) / (SELECT SUM(total_amount) FROM sales_analysis
WHERE YEAR(date) IN (2021,2022,2023, 2024))) * 100, 2) AS revenue_percentage
FROM sales_analysis
WHERE YEAR(date) IN (2021, 2022, 2023, 2024)
GROUP BY YEAR(date)
ORDER BY year;
```

💸 9. Profit Summary

-- Total profit

```
SELECT SUM(profit) AS Total_Profit FROM sales_analysis;
```

-- Total profit by year

```
SELECT YEAR(date) AS year, SUM(profit) AS Total_Profit
FROM sales_analysis
GROUP BY year;
```

-- Total profit by product category and year

```
SELECT product_category, SUM(profit) AS Total_Profit, YEAR(date) AS year
FROM sales_analysis
GROUP BY product_category, year;
```

-- Profit percentage contribution by year

```
SELECT YEAR(date) AS year,
  SUM(profit) AS revenue,
  ROUND((SUM(profit) / (SELECT SUM(profit) FROM sales_analysis WHERE YEAR(date) IN
(2021,2022,2023, 2024))) * 100, 2) AS Profit_Percentage FROM sales_analysis
WHERE YEAR(date) IN (2021, 2022, 2023, 2024)
GROUP BY YEAR(date)
ORDER BY year;
```

-- Profit growth rate year-over-year

```
SELECT curr.year AS current_year,
    ROUND(((curr.profit - prev.profit) / NULLIF(prev.profit, 0)) * 100, 2) AS
profit_growth_rate
FROM (
  SELECT YEAR(date) AS year, SUM(profit) AS profit
  FROM sales_analysis
  GROUP BY YEAR(date)
) AS curr
JOIN (
  SELECT YEAR(date) AS year, SUM(profit) AS profit
  FROM sales_analysis
  GROUP BY YEAR(date)
) AS prev ON curr.year = prev.year + 1;
```

💳 10. Payment Analysis

-- Total customers by payment type

```
SELECT payment, COUNT(customer_id) AS total_customers
FROM sales_analysis
GROUP BY payment
ORDER BY total_customers DESC;
```

-- Payment type by product category

```
SELECT product_category, payment, COUNT(customer_id) AS total_customers
FROM sales_analysis
GROUP BY payment, product_category
ORDER BY payment DESC;
```

-- Percentage usage of each payment method

```
SELECT payment,
    COUNT(*) AS payment_usage,
    ROUND((COUNT(*) / (SELECT COUNT(*) FROM sales_analysis)) * 100, 2) AS
usage_percentage
FROM sales_analysis
GROUP BY payment;
```

👦🤝👦 11. Customer Type Analysis

-- Total customers by type

```
SELECT customer_type, COUNT(customer_id) AS customers
FROM sales_analysis
GROUP BY customer_type;
```

-- Customer type by product category

```
SELECT product_category, customer_type, COUNT(customer_id) AS customers
FROM sales_analysis
GROUP BY customer_type, product_category
ORDER BY customer_type;
```

😊 12. Customer Satisfaction

-- Overall average customer satisfaction

```
SELECT AVG(customer_satisfication) AS avg_customer_satisfication FROM sales_analysis;
```

-- Average satisfaction by product category

```
SELECT product_category, ROUND(AVG(customer_satisfication), 2) AS
avg_customer_satisfication
FROM sales_analysis
GROUP BY product_category;
```

-- Average satisfaction by customer type

```
SELECT customer_type, ROUND(AVG(customer_satisfication), 2) AS
avg_customer_satisfication

FROM sales_analysis

GROUP BY customer_type;
```