**TASK 6: Sales Trend Analysis Using Aggregations**

***Create Database:***

CREATE DATABASE online\_sales;

***Create Table:***

CREATE TABLE orders (

order\_id SERIAL PRIMARY KEY,

order\_date DATE NOT NULL,

amount DECIMAL(10, 2) NOT NULL,

product\_id INT NOT NULL

);

***Insert sample data for 2023 and 2024:***

INSERT INTO orders(order\_date, amount, product\_id) VALUES

('2023-01-15', 100.50, 1),

('2023-01-20', 200.75, 2),

('2023-02-10', 150.25, 1),

('2023-02-15', 300.00, 3),

('2023-03-05', 50.00, 2),

('2023-03-25', 75.30, 1),

('2024-01-10', 120.00, 2),

('2024-01-25', 180.50, 3),

('2024-02-12', 90.75, 1),

('2024-03-18', 250.00, 2);

***Analyze monthly revenue and order volume:***

SELECT

EXTRACT(YEAR FROM order\_date) AS year,

EXTRACT(MONTH FROM order\_date) AS month,

SUM(amount) AS total\_revenue,

COUNT(DISTINCT order\_id) AS order\_volume

FROM

online\_sales

WHERE

'2022-01-01' AND '2022-12-31'

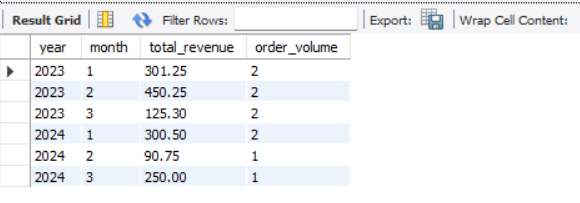
GROUP BY

year, month

ORDER BY

year, month;

**OUTPUT:**



**SQL Query Structure**

* **Extract Month and Year:**
  + Use **EXTRACT(MONTH FROM order\_date)** to get the month.
  + Use **EXTRACT(YEAR FROM order\_date)** to get the year.
* **Group By:**
  + Group the results by year and month.
* **Aggregate Functions:**
  + Use **SUM(amount)** to calculate total revenue.
  + Use **COUNT(DISTINCT order\_id)** to calculate the order volume.
* **Sorting:**
  + Use **ORDER BY** to sort the results.
* **Limiting Results:**
  + You can limit the results to specific time periods using a **WHERE** clause.