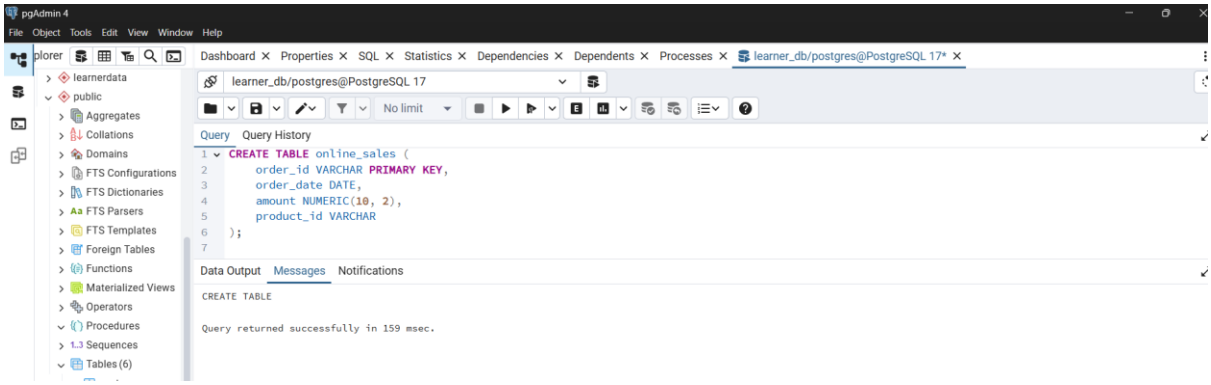
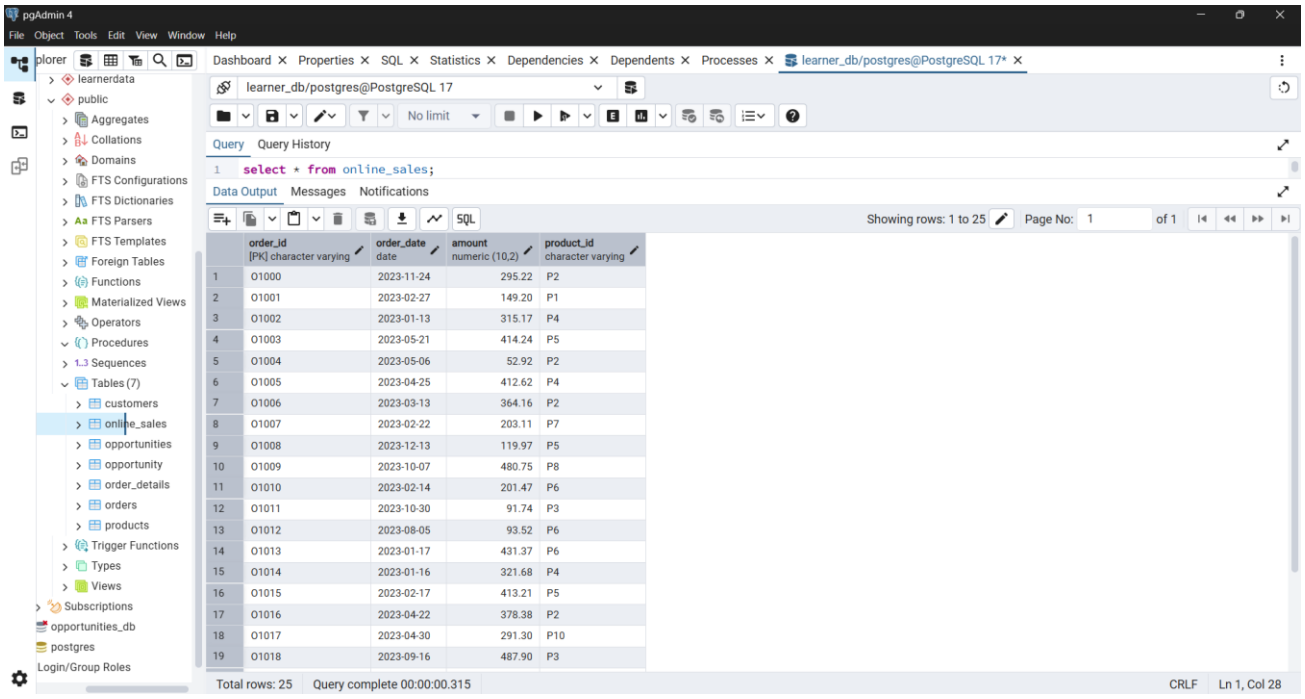
✅ PostgreSQL Table Creation

```
CREATE TABLE online_sales (
    order_id VARCHAR PRIMARY KEY,
    order_date DATE,
    amount NUMERIC(10, 2),
    product_id VARCHAR
);
```



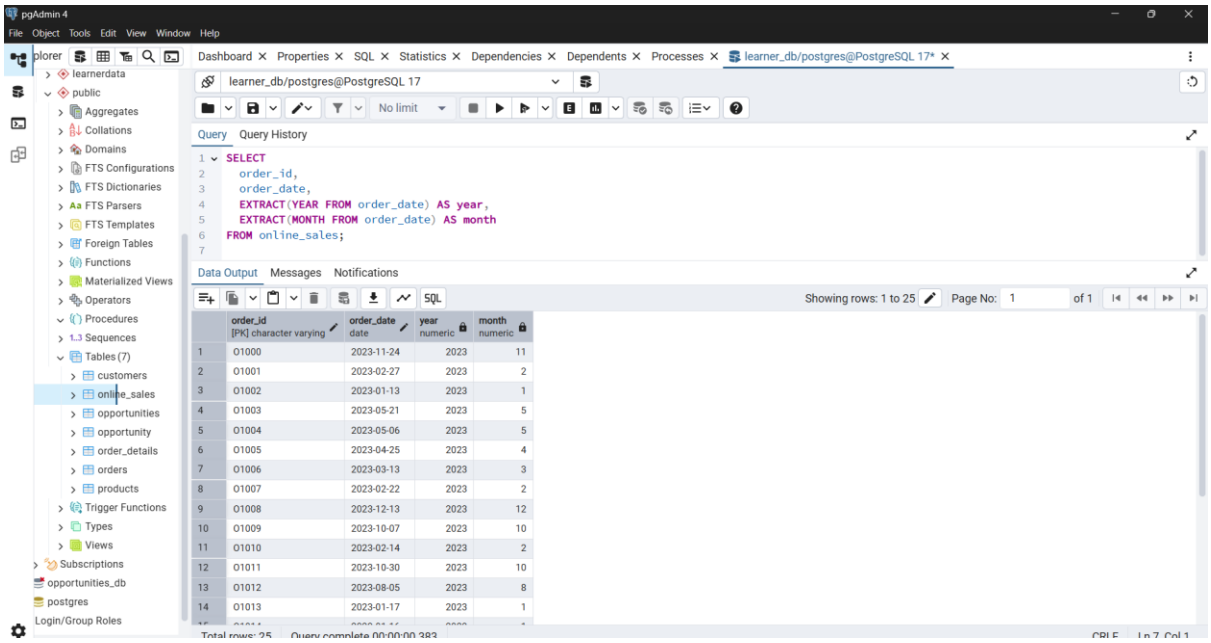
1. View the Dataset

```
SELECT * FROM online_sales;
```



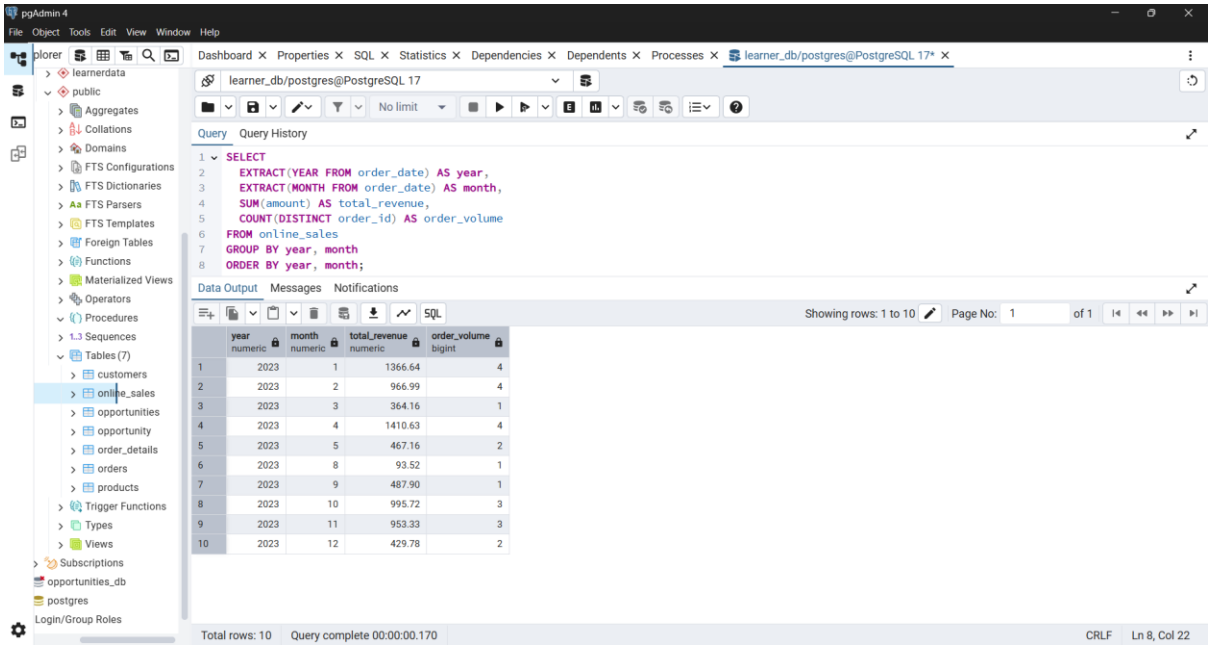
2. Extract Year & Month

```
SELECT
  order_id,
  order_date,
  EXTRACT(YEAR FROM order_date) AS year,
  EXTRACT(MONTH FROM order_date) AS month
FROM online_sales;
```



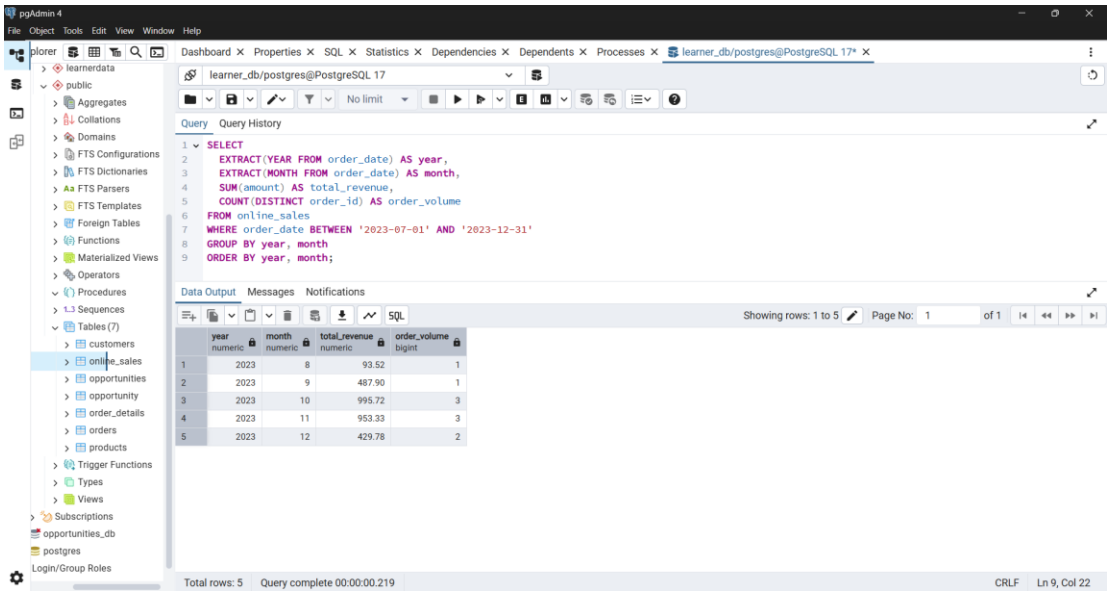
3. Monthly Revenue and Order Volume

```
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
GROUP BY year, month
ORDER BY year, month;
```



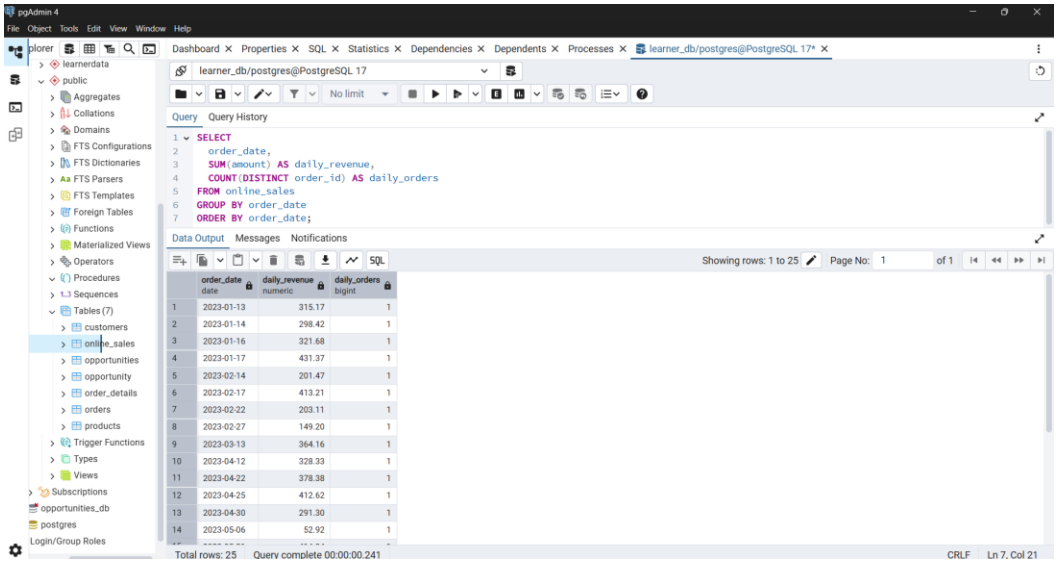
4. Filtered by Date Range (e.g., July–December 2023)

```
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 order_date BETWEEN '2023-07-01' AND '2023-12-31'
GROUP BY year, month
ORDER BY year, month;
```



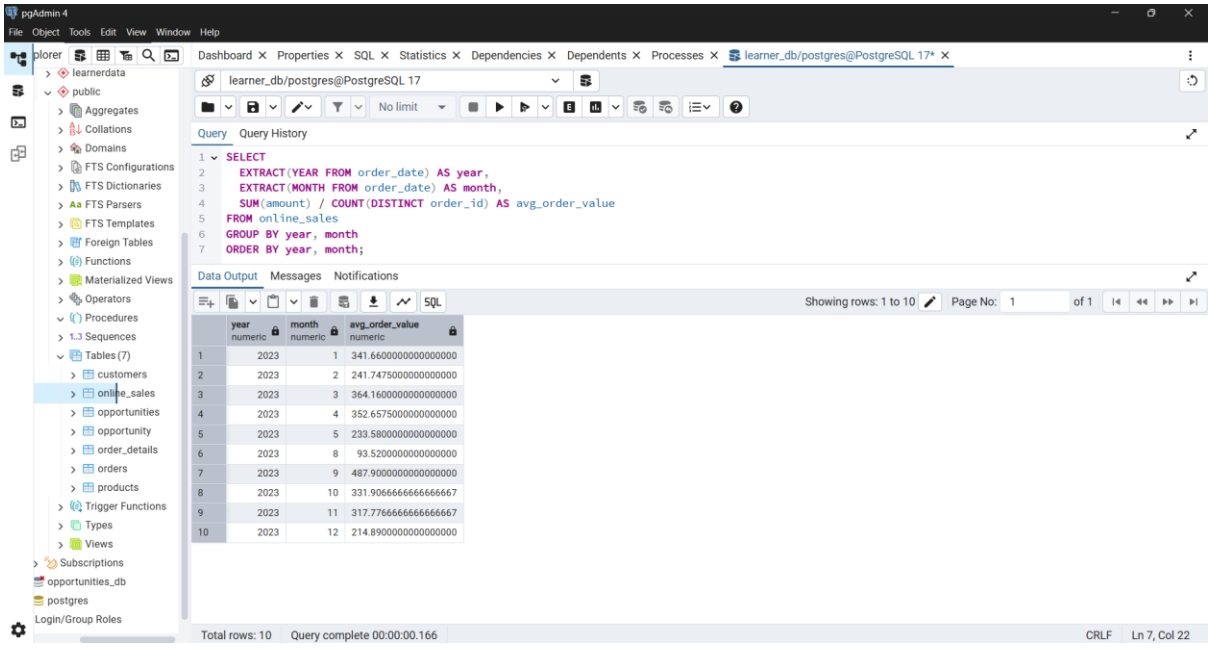
5. Daily Revenue Trend (Optional Granularity)

```
SELECT
  order_date,
  SUM(amount) AS daily_revenue,
  COUNT(DISTINCT order_id) AS daily_orders
FROM online_sales
GROUP BY order_date
ORDER BY order_date;
```



6. Monthly Average Order Value (AOV)

```
SELECT
  EXTRACT(YEAR FROM order_date) AS year,
  EXTRACT(MONTH FROM order_date) AS month,
  SUM(amount) / COUNT(DISTINCT order_id) AS avg_order_value
FROM online_sales
GROUP BY year, month
ORDER BY year, month;
```



7. Top 5 Months by Revenue

SELECT

EXTRACT(YEAR FROM order_date) AS year,

EXTRACT(MONTH FROM order_date) AS month,

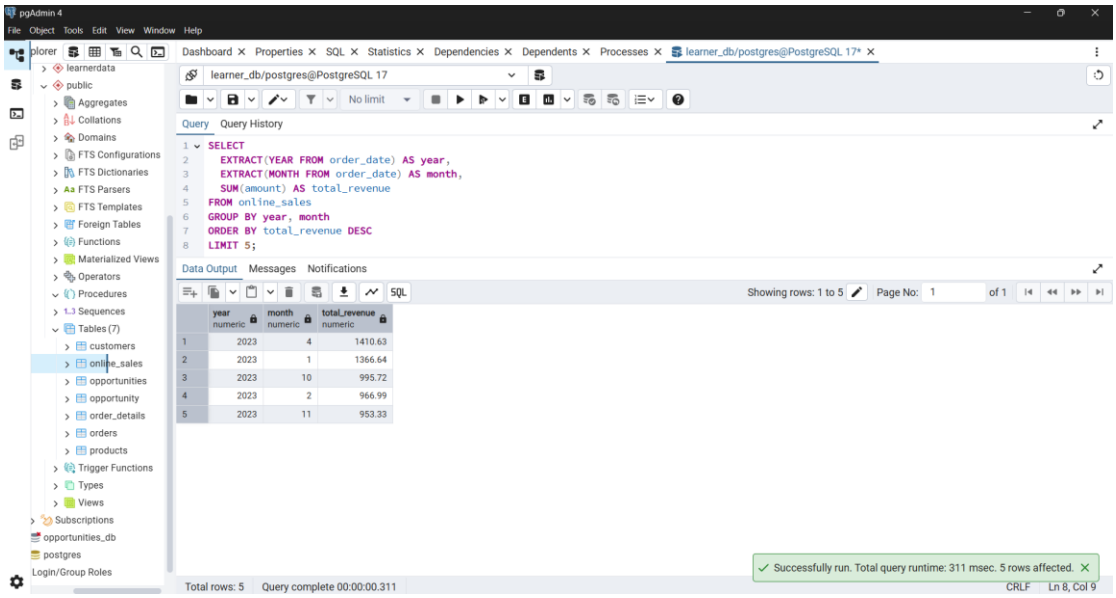
SUM(amount) AS total_revenue

FROM online_sales

GROUP BY year, month

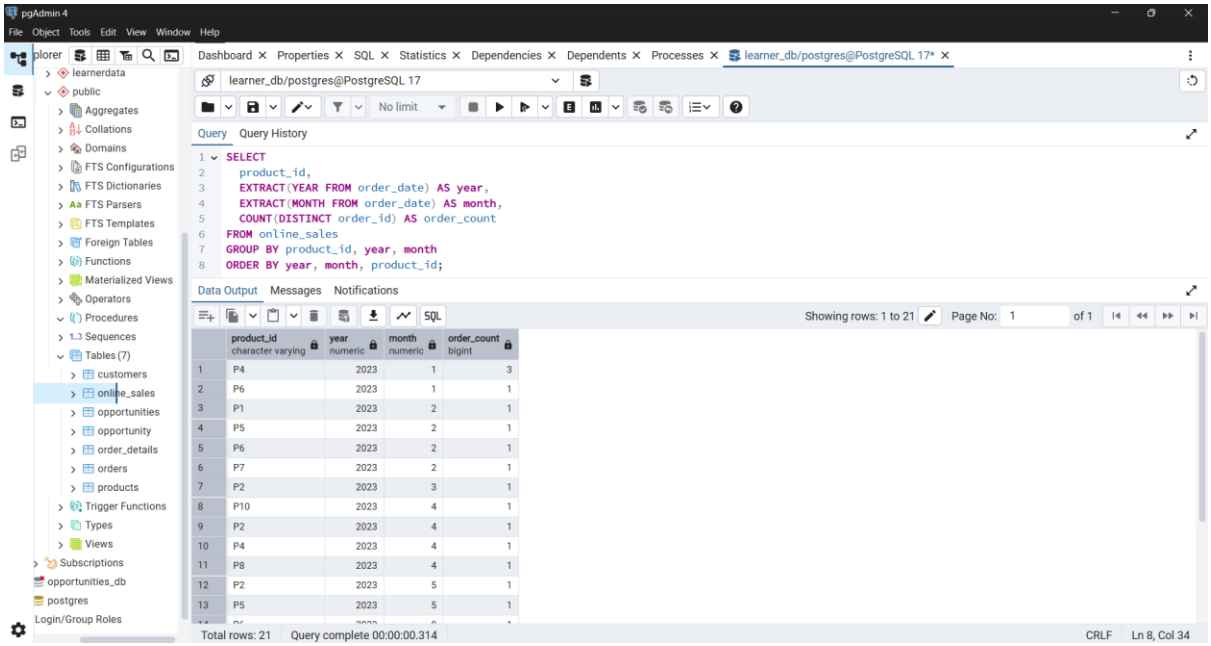
ORDER BY total_revenue DESC

LIMIT 5;



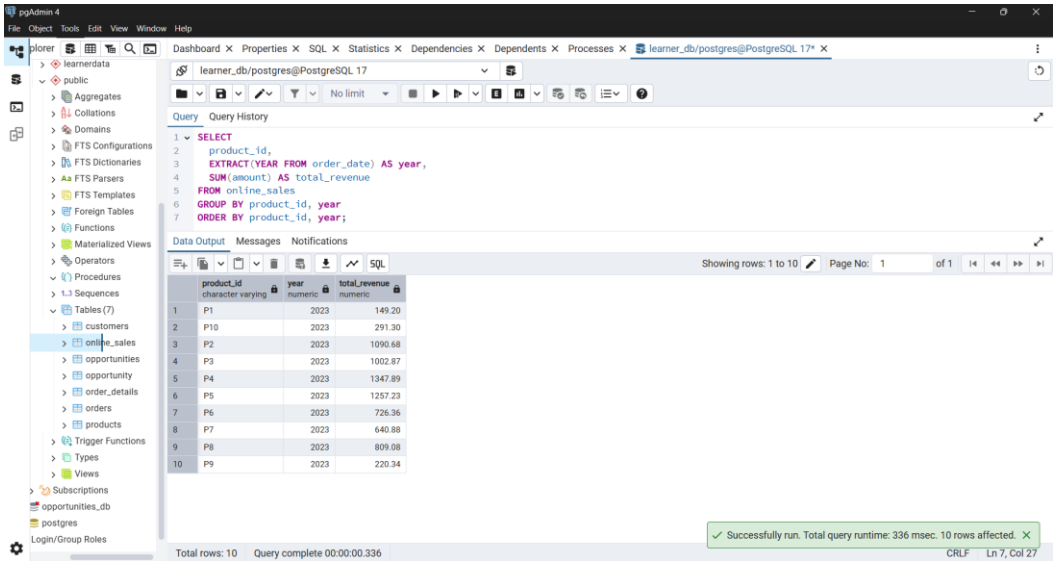
8. Number of Orders Per Product Per Month

```
SELECT
  product_id,
  EXTRACT(YEAR FROM order_date) AS year,
  EXTRACT(MONTH FROM order_date) AS month,
  COUNT(DISTINCT order_id) AS order_count
FROM online_sales
GROUP BY product_id, year, month
ORDER BY year, month, product_id;
```



9. Total Revenue Per Product (Yearly Aggregation)

```
SELECT
  product_id,
  EXTRACT(YEAR FROM order_date) AS year,
  SUM(amount) AS total_revenue
FROM online_sales
GROUP BY product_id, year
ORDER BY product_id, year;
```



10. Revenue Summary by Quarter

```
SELECT
  EXTRACT(YEAR FROM order_date) AS year,
  CEIL(EXTRACT(MONTH FROM order_date) / 3.0) AS quarter,
  SUM(amount) AS total_revenue,
  COUNT(DISTINCT order_id) AS order_volume
FROM online_sales
GROUP BY year, quarter
ORDER BY year, quarter;
```