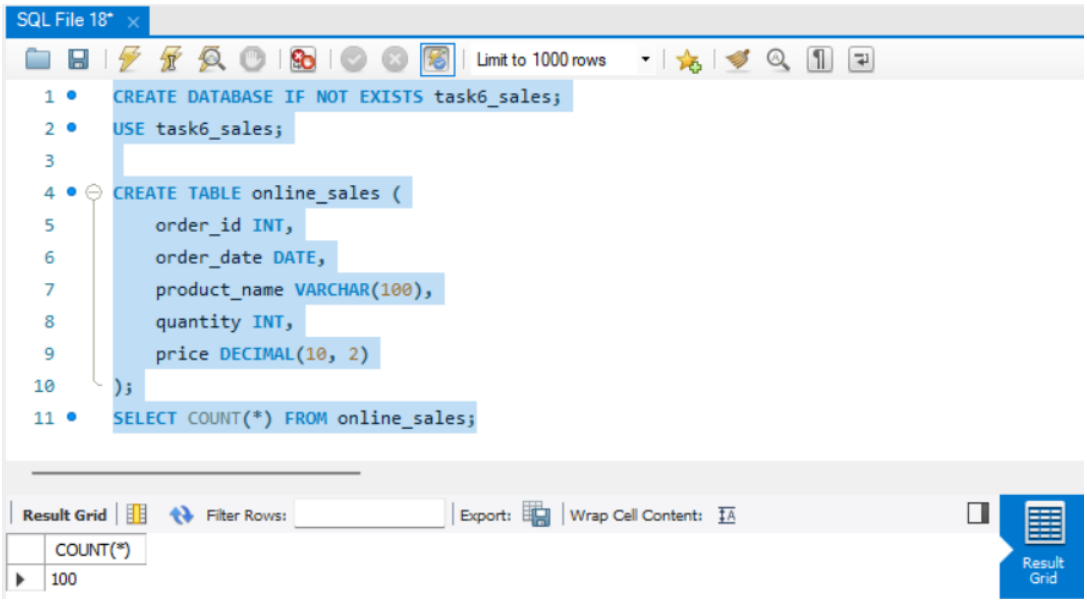
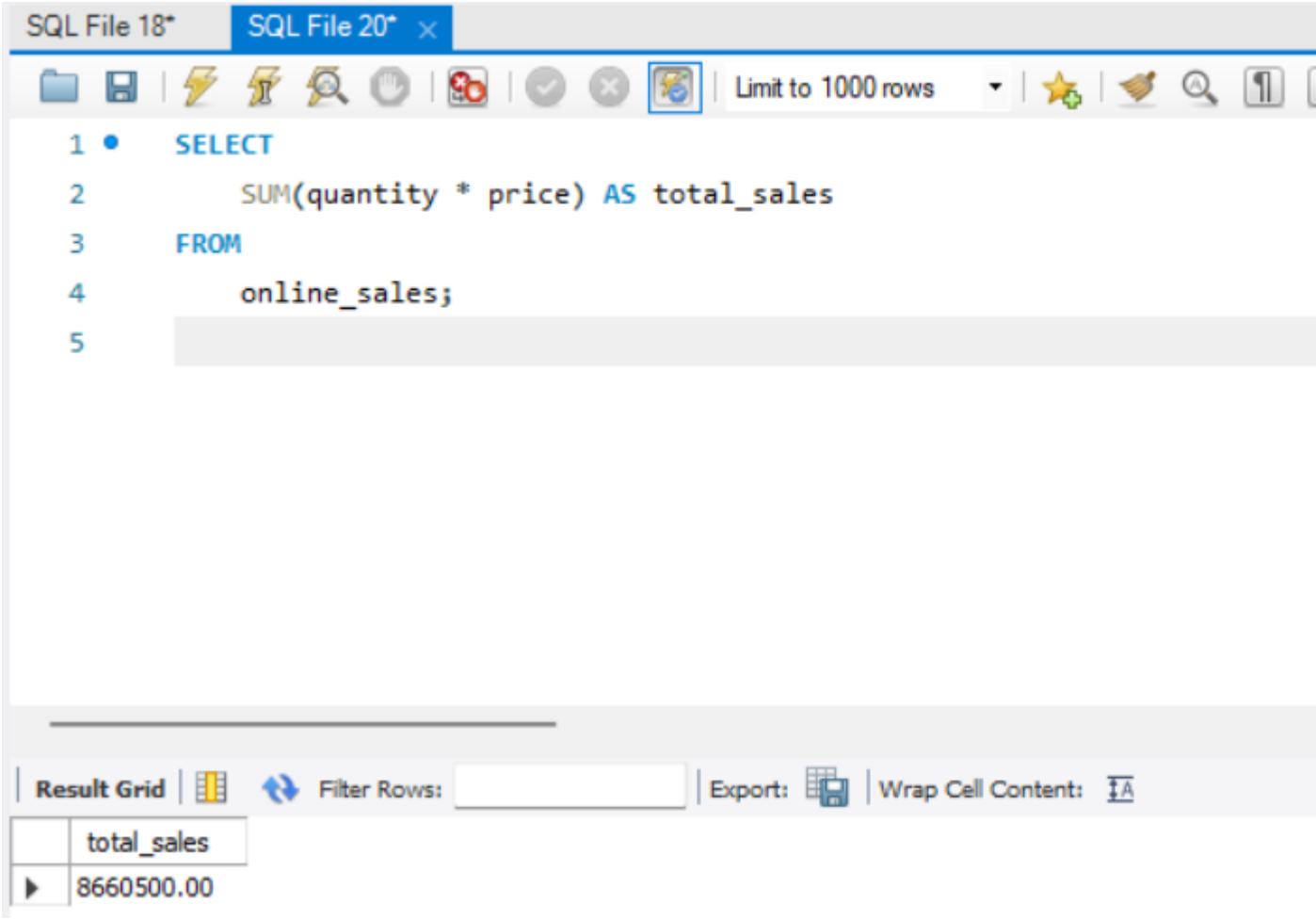
Database Creation

```
CREATE DATABASE IF NOT EXISTS task6_sales;

USE task6_sales;


CREATE TABLE online_sales (
    order_id INT,
    order_date DATE,
    product_name VARCHAR(100),
    quantity INT,
    price DECIMAL(10, 2)
);

SELECT COUNT(*) FROM online_sales;
```



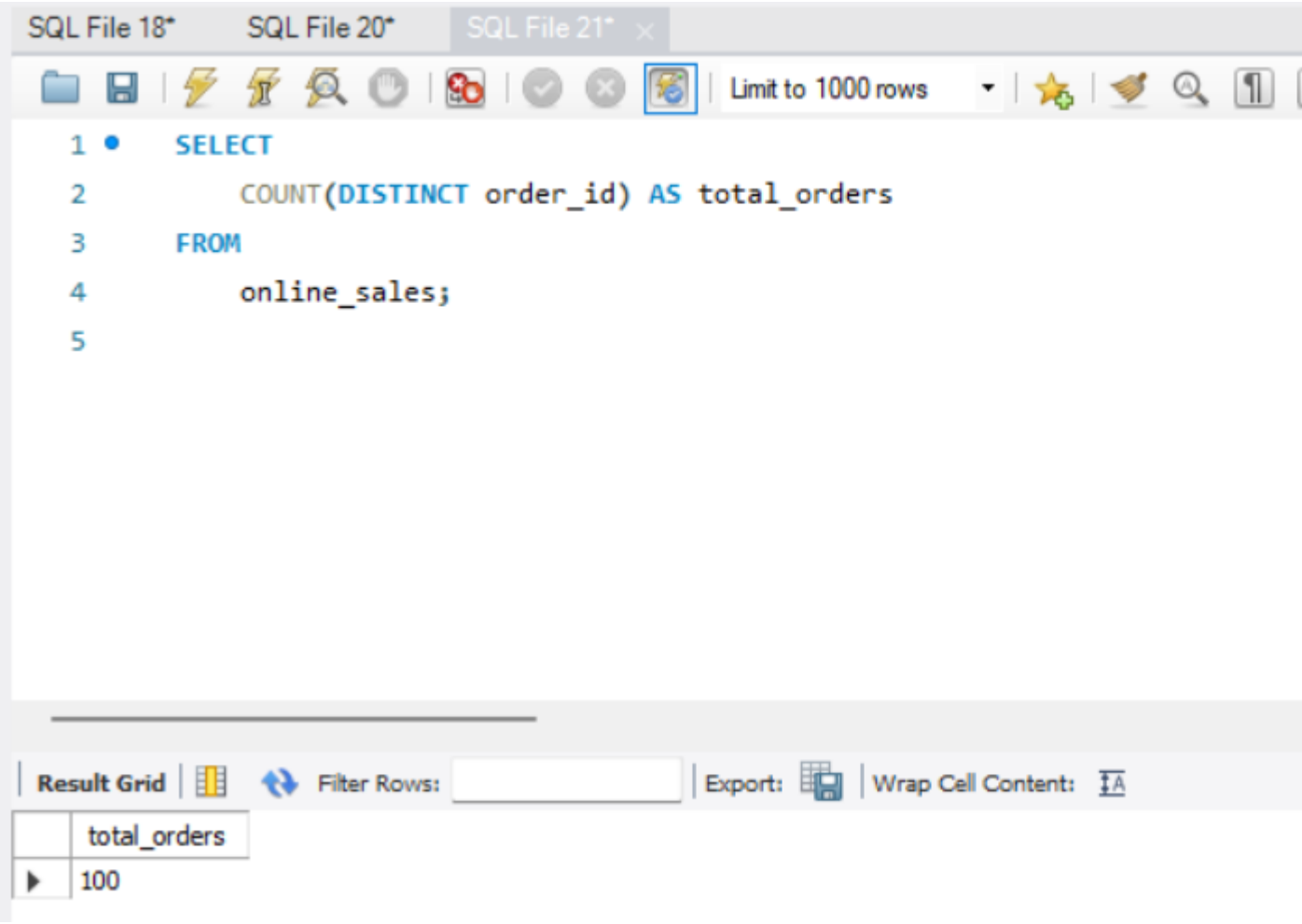
✅ Query 1: Total Sales Amount




```
SELECT
   SUM(quantity * price) AS total_sales
FROM
   online_sales;
```

📌 **Insight:** This query calculates the total revenue generated from all online sales. It multiplies quantity with price for each record and sums them up.

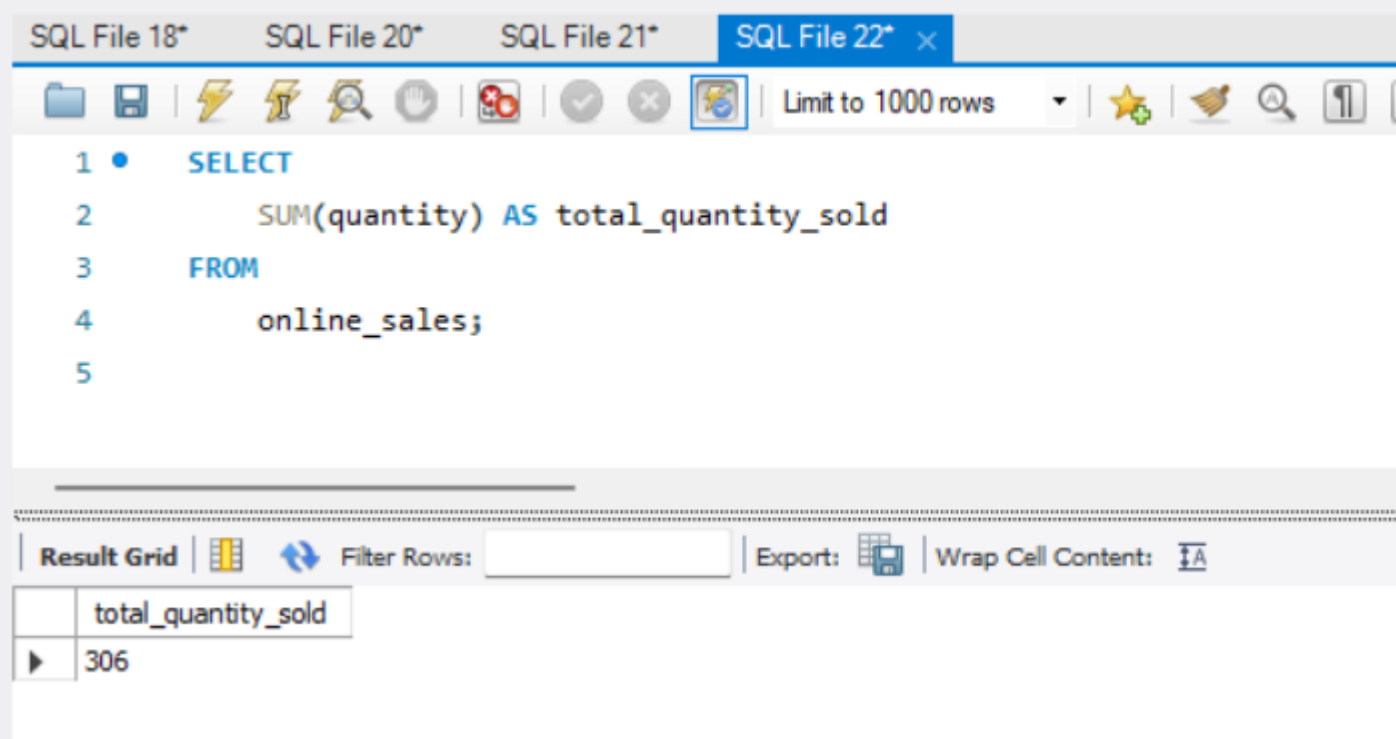
✅ Query 2: Total Orders




```
SELECT
    COUNT(DISTINCT order_id) AS total_orders
FROM
    online_sales;
```

📌 **Insight:** This gives the number of unique orders in the dataset. Helps understand the volume of transactions.

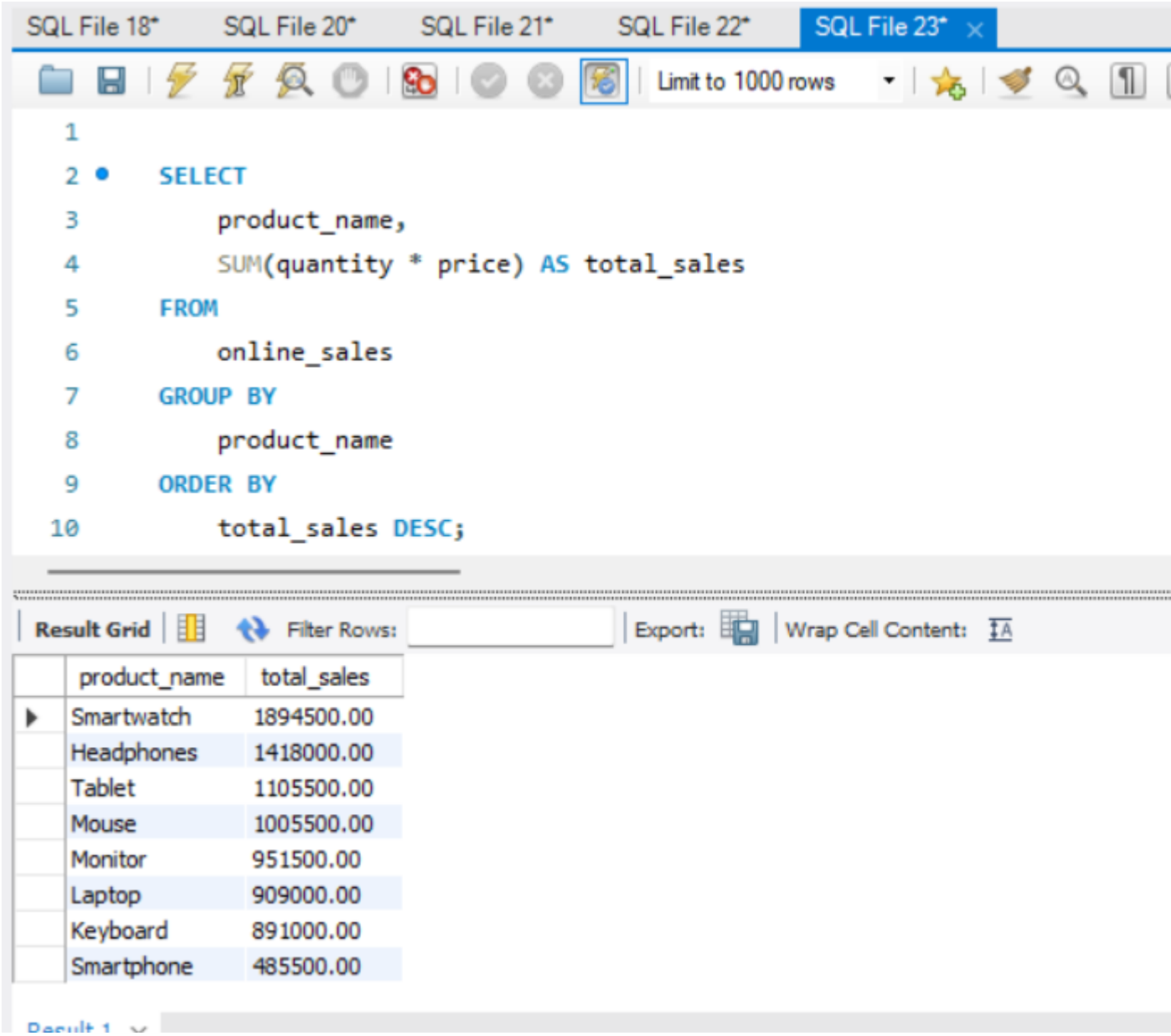
✅ Query 3: Total Quantity Sold




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

📌 **Insight:** This query shows how many total items were sold. Useful for inventory or sales volume analysis.

✅ Query 4: Sales by Product

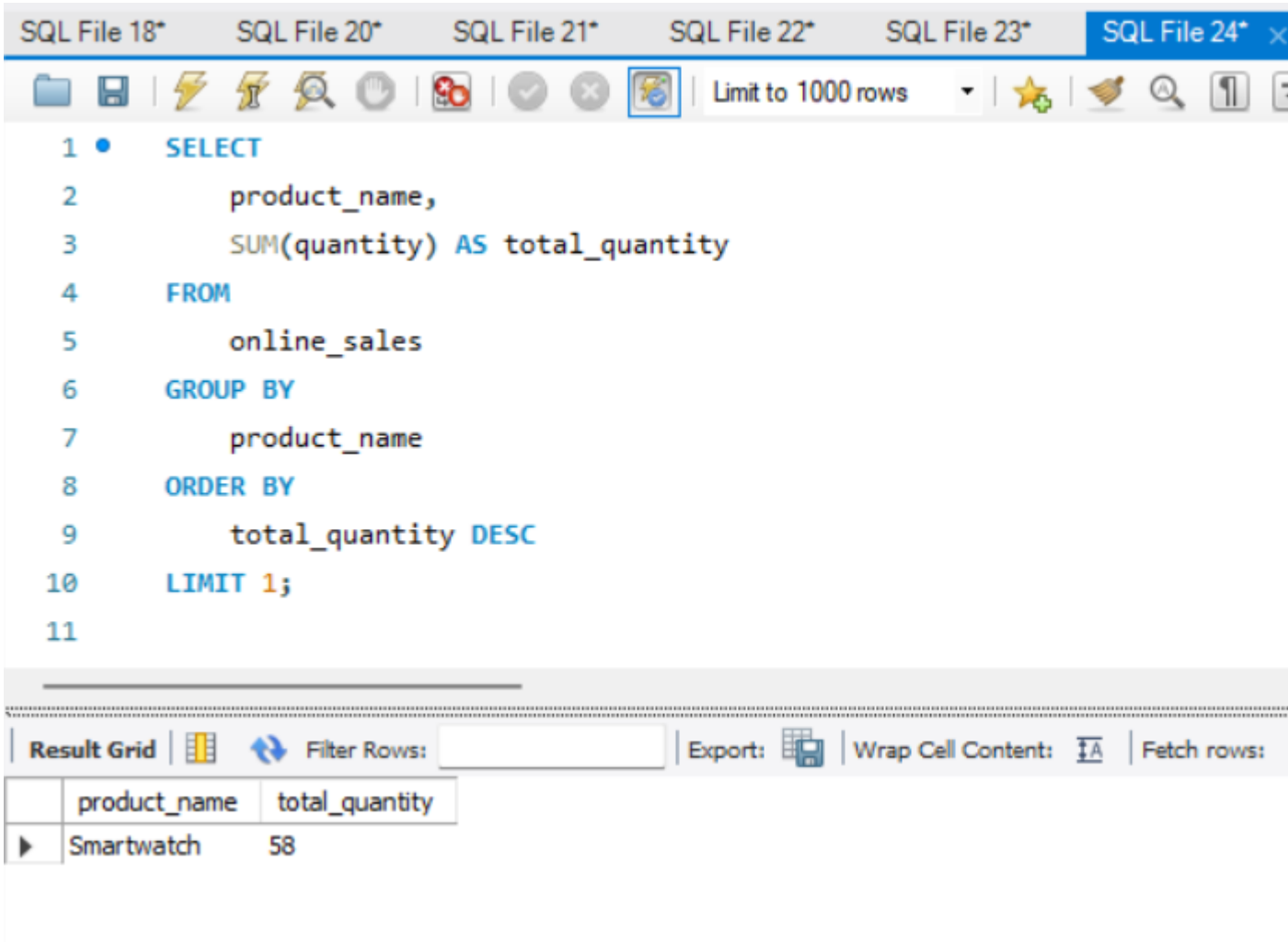



```
SELECT
    product_name,
    SUM(quantity * price) AS total_sales
FROM
    online_sales
GROUP BY
    product_name
ORDER BY
```

total_sales DESC;

📌 **Insight:** This shows which products generated the most revenue. Use it to identify top performers.

✅ Query 5: Most Sold Product (by Quantity)




```
SELECT
    product_name,
    SUM(quantity) AS total_quantity
FROM
    online_sales
GROUP BY
    product_name
```

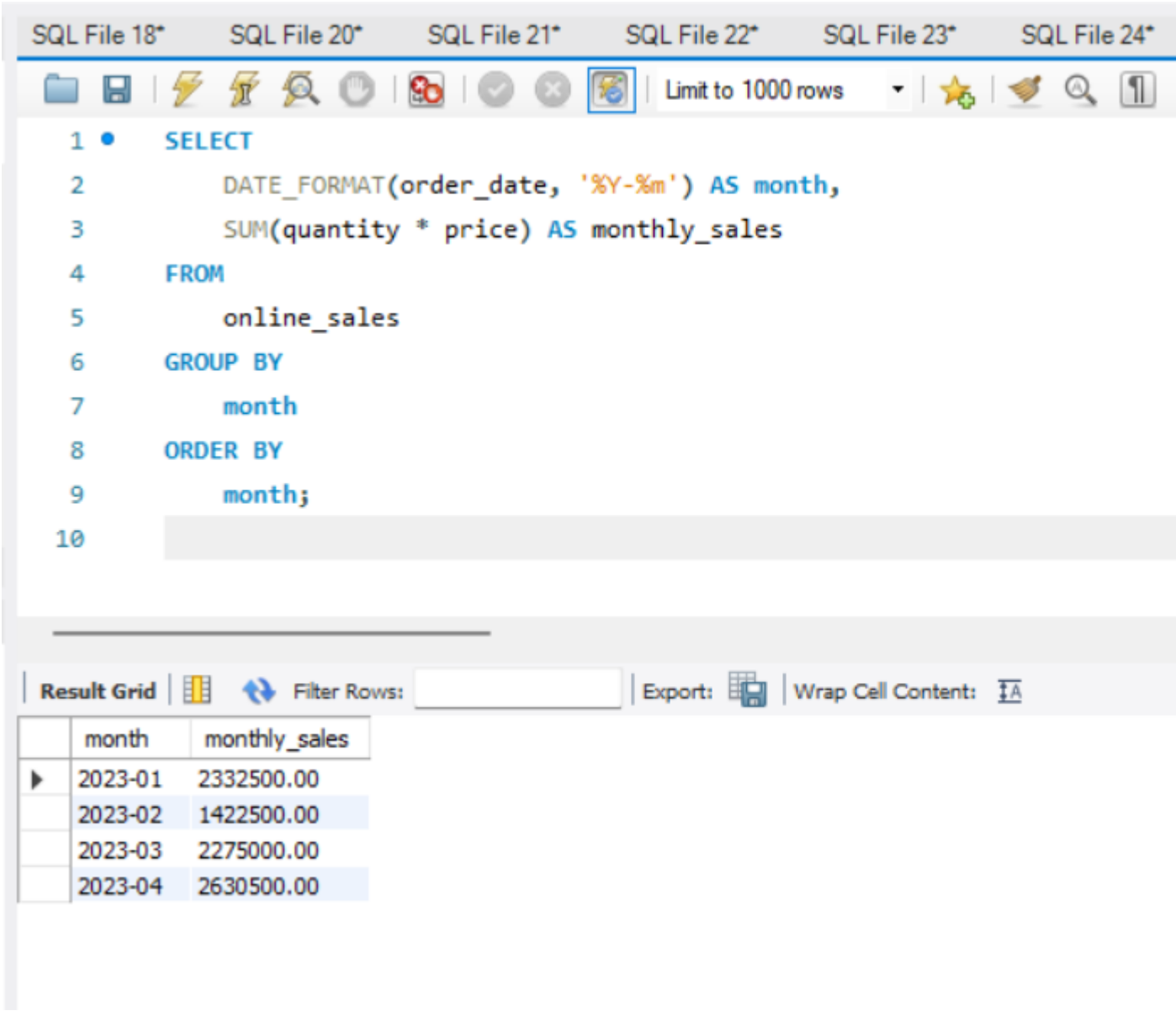
ORDER BY

total_quantity DESC

LIMIT 1;

📌 **Insight:** Finds the most sold product based on quantity. It might not be the most profitable but is in high demand.

✅ Query 6: Monthly Sales Trend



SELECT

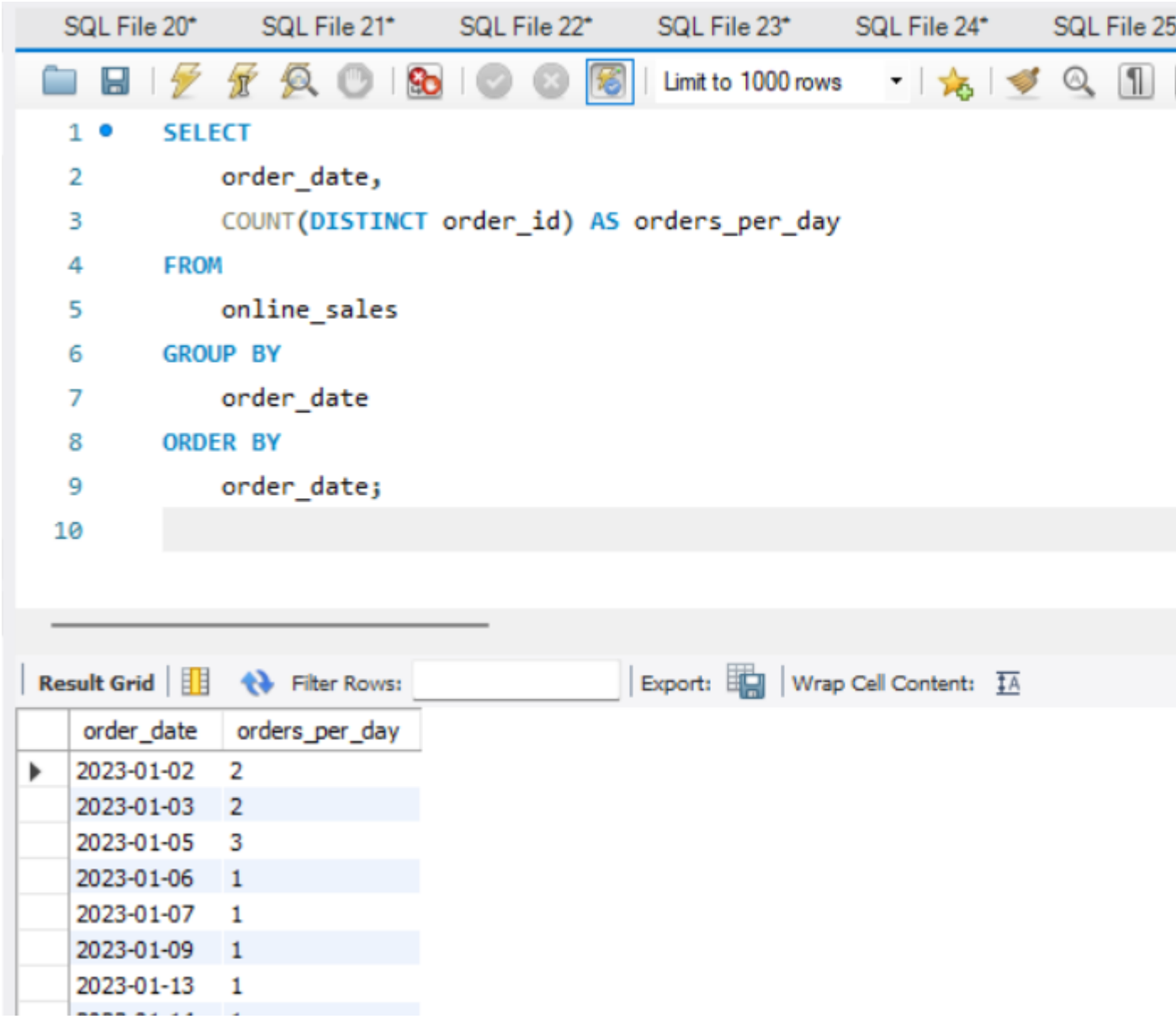
DATE_FORMAT(order_date, '%Y-%m') AS month,

SUM(quantity * price) AS monthly_sales

```
FROM
    online_sales
GROUP BY
    month
ORDER BY
    month;
```

📌 **Insight:** Shows month-wise total sales, useful to identify sales trends over time.

✅ Query 7: Daily Orders

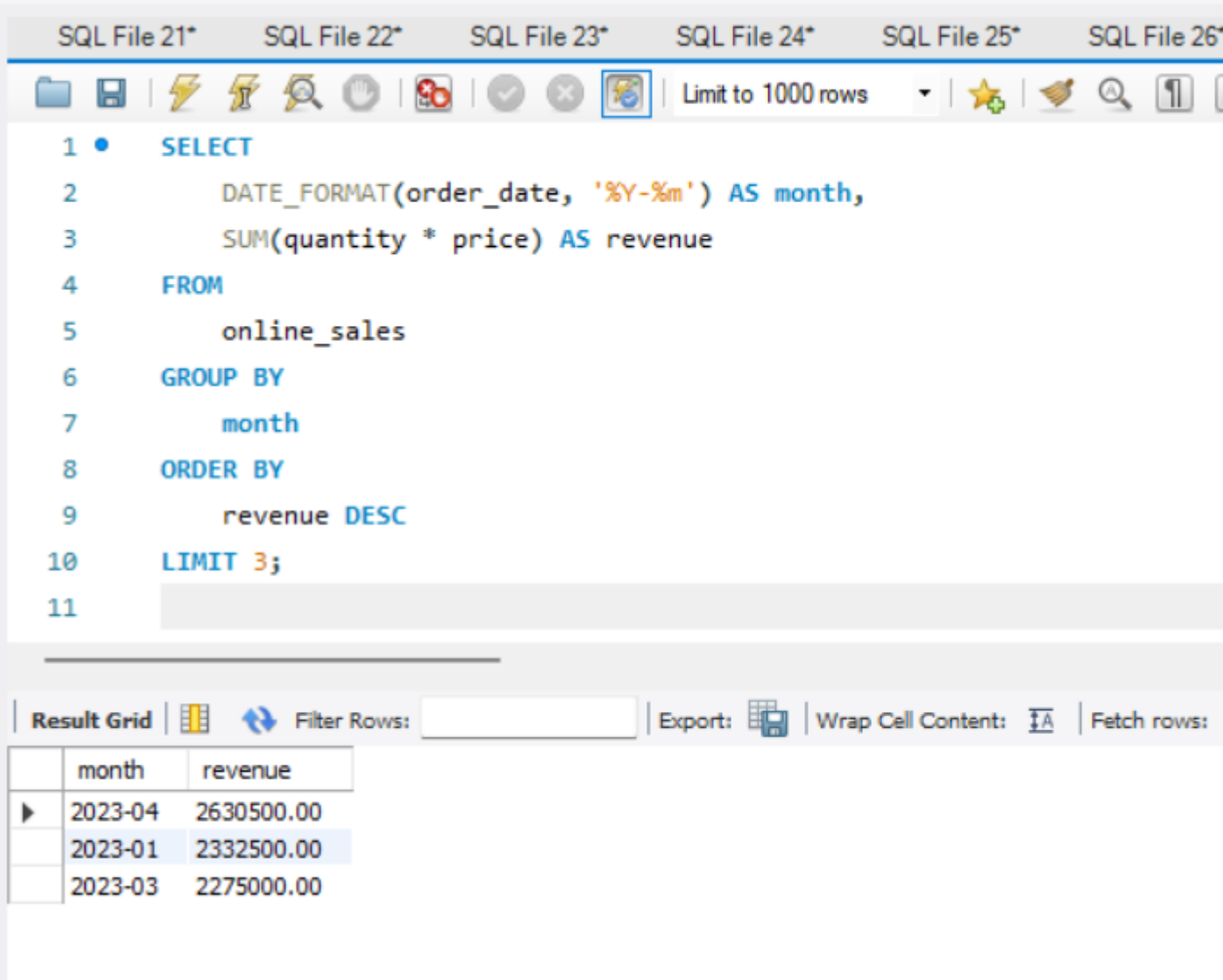



```
SELECT
    order_date,
    COUNT(DISTINCT order_id) AS orders_per_day
FROM
    online_sales
GROUP BY
    order_date
ORDER BY
```

order_date;

📌 **Insight:** This helps to analyze daily activity — can spot busy vs. slow days.

Query 8: Top 3 Months by Revenue




```
SELECT
    DATE_FORMAT(order_date, '%Y-%m') AS month,
    SUM(quantity * price) AS revenue
FROM
    online_sales
```

GROUP BY

month

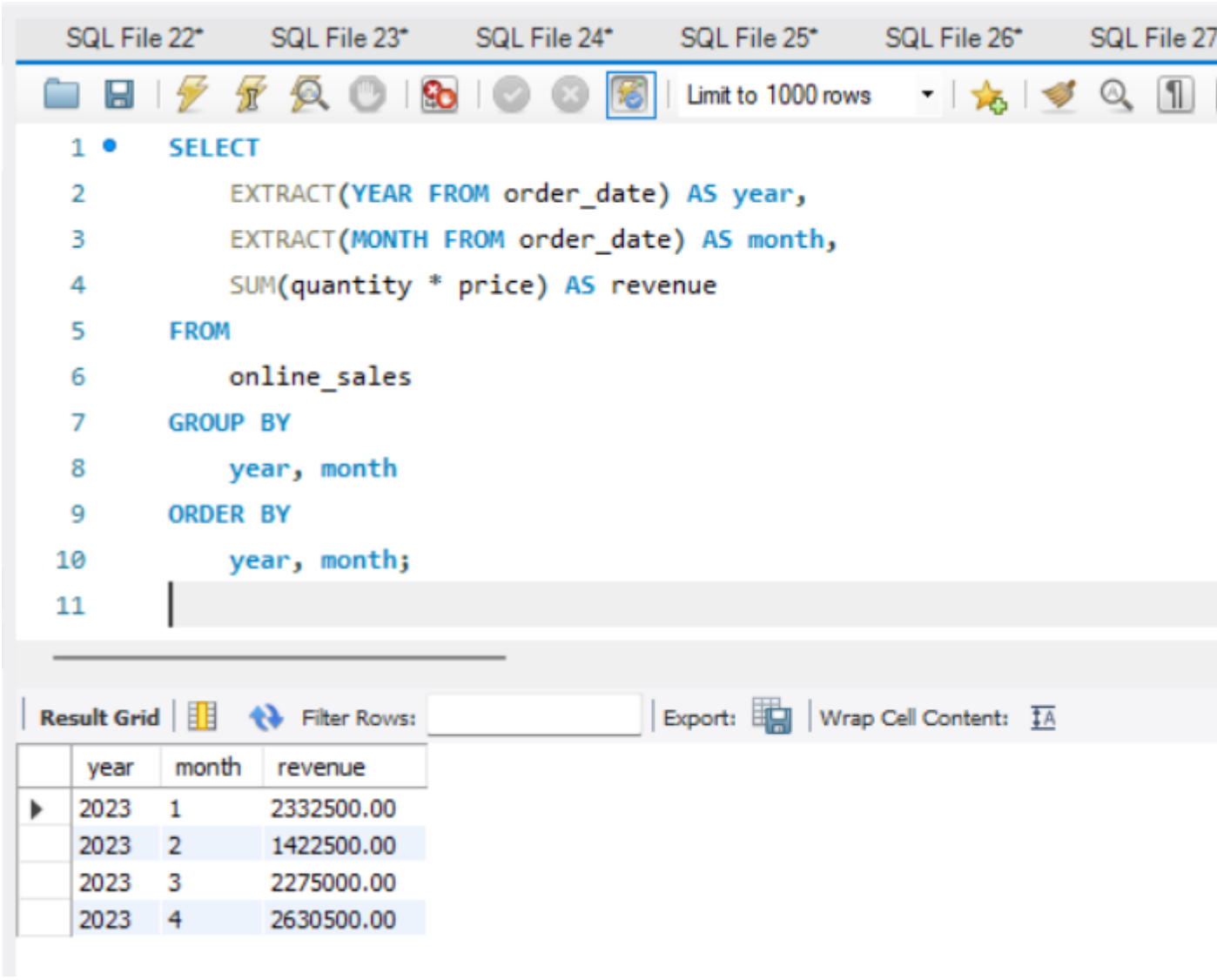
ORDER BY

revenue DESC

LIMIT 3;

📌 **Insight:** This shows the top 3 months where sales were highest. Very helpful for identifying peak business periods.

Query 9: Using EXTRACT



```
SELECT
    EXTRACT(YEAR FROM order_date) AS year,
    EXTRACT(MONTH FROM order_date) AS month,
    SUM(quantity * price) AS revenue
FROM
    online_sales
GROUP BY
    year, month
ORDER BY
    year, month;
```

📌 **Insight:** This query breaks sales down by year and month using SQL's EXTRACT() function.