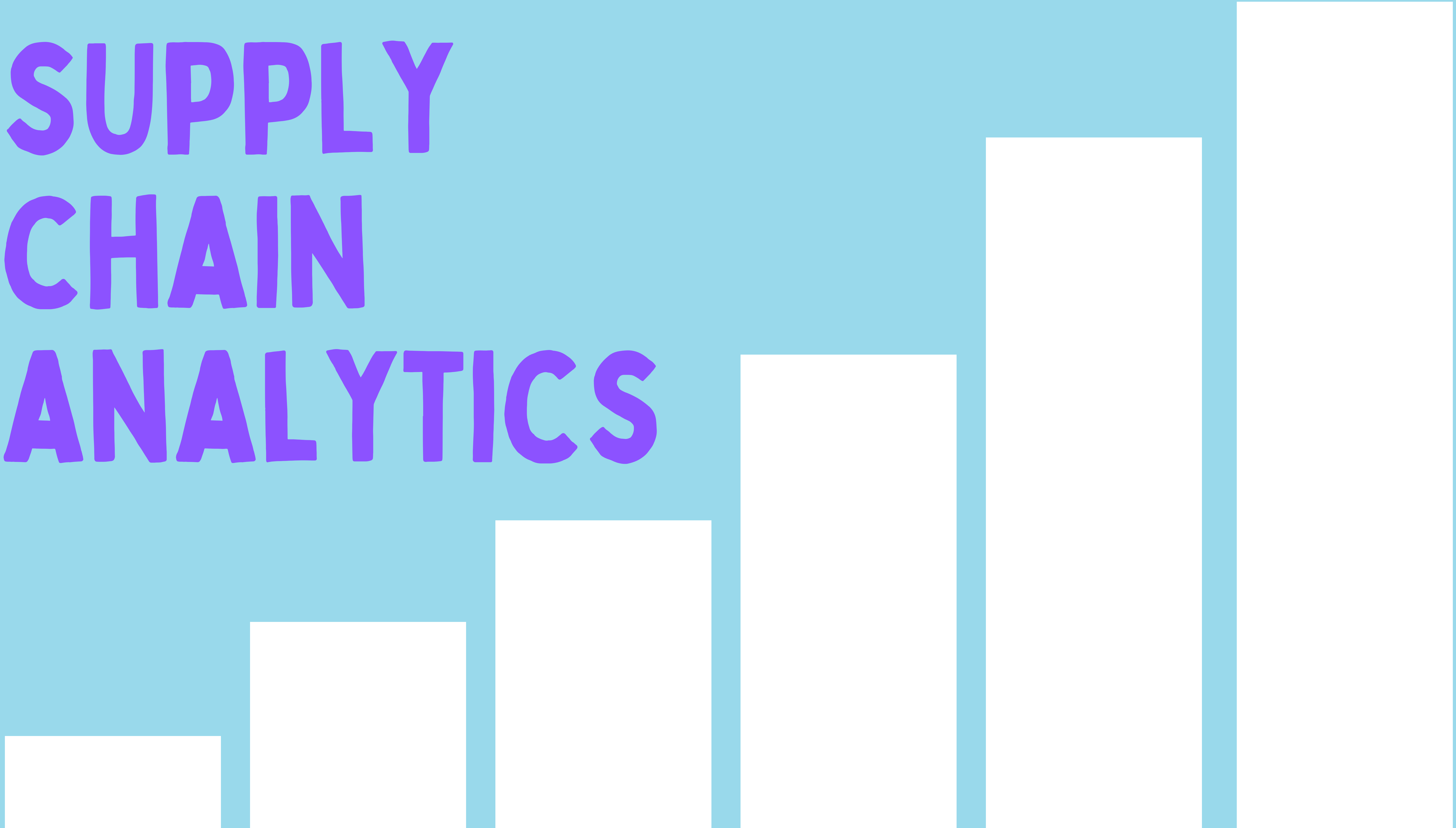
SUPPLY CHAIN ANALYTICS

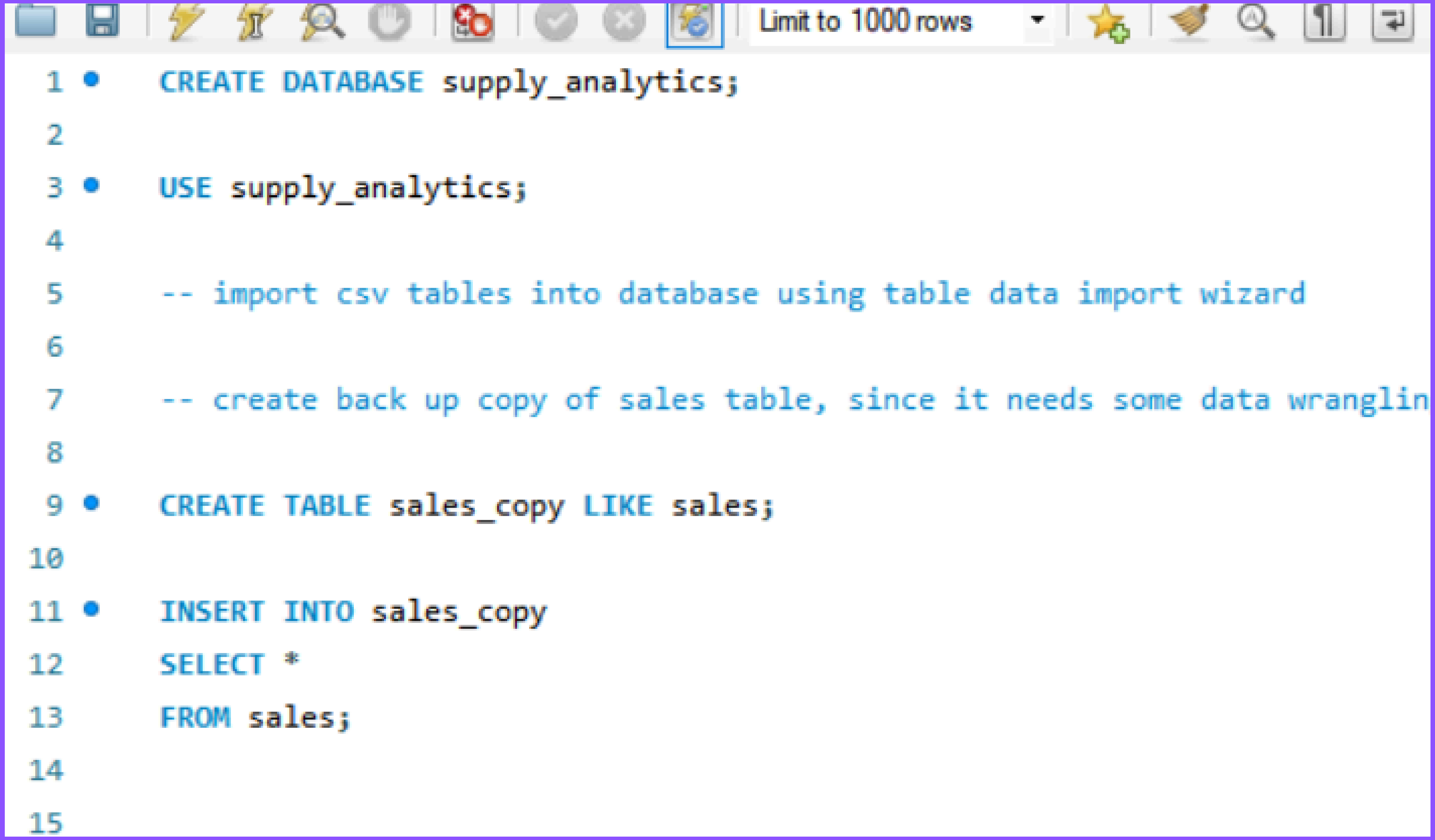


PROJECT SUMMARY

I conducted an analysis of sales data, extracted meaningful insights from raw data and visualized trends. The tools used are:

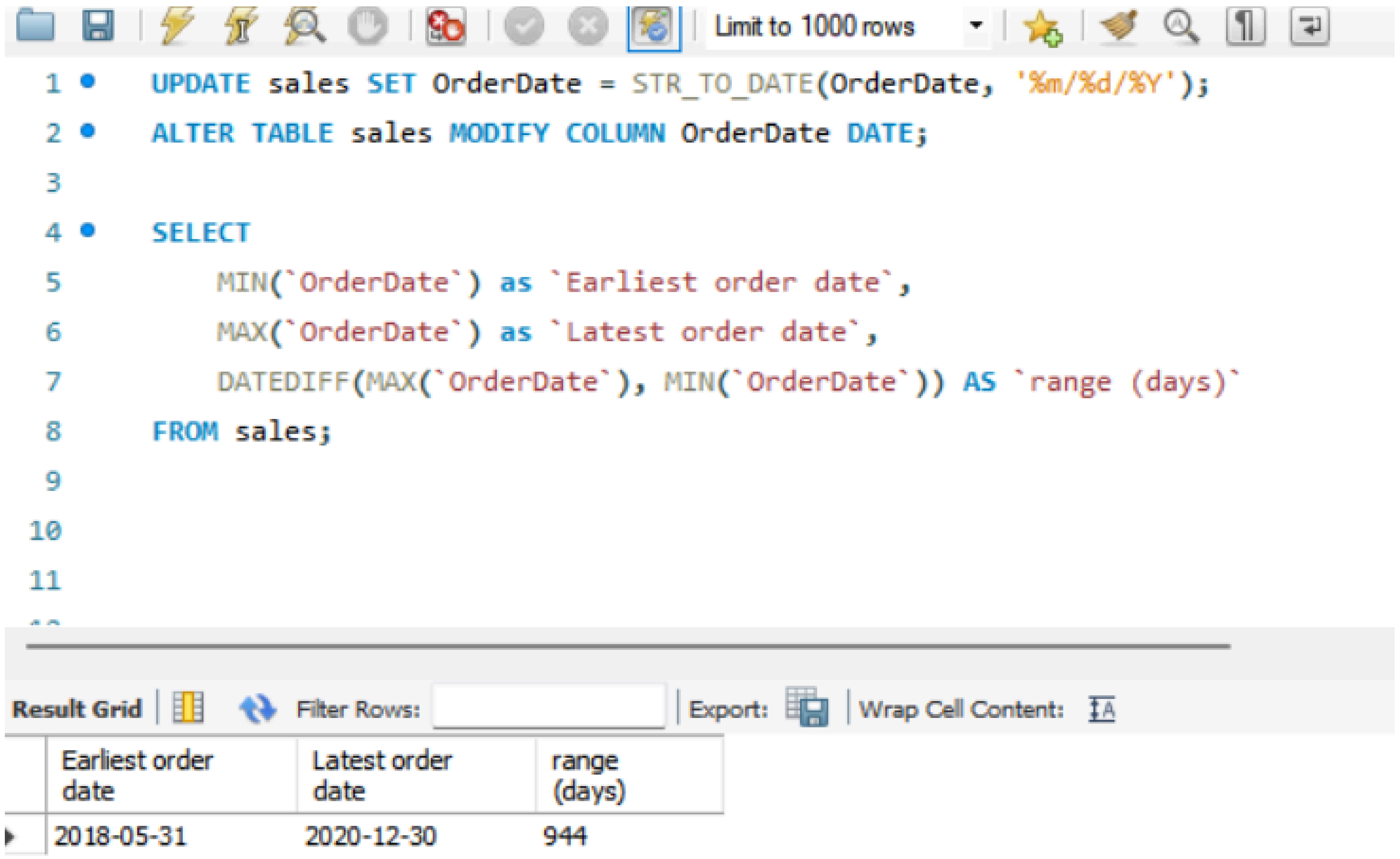
1. Ms Excel - Acquired data in csv files and cleaned it using Ms Excel
2. SQL - Developed SQL queries including joins, subqueries, and aggregate functions to gather comprehensive data from databaes
3. Power BI - Formulated some measures using DAX and visualized data on the designed dashboard

DATA ACQUISITION



```sql
CREATE DATABASE supply_analytics;

USE supply_analytics;

-- import csv tables into database using table data import wizard

-- create back up copy of sales table, since it needs some data wranglin

CREATE TABLE sales_copy LIKE sales;

INSERT INTO sales_copy
SELECT *
FROM sales;
```

Q1: ANALYSE THE ORDERS RECORDING PERIOD

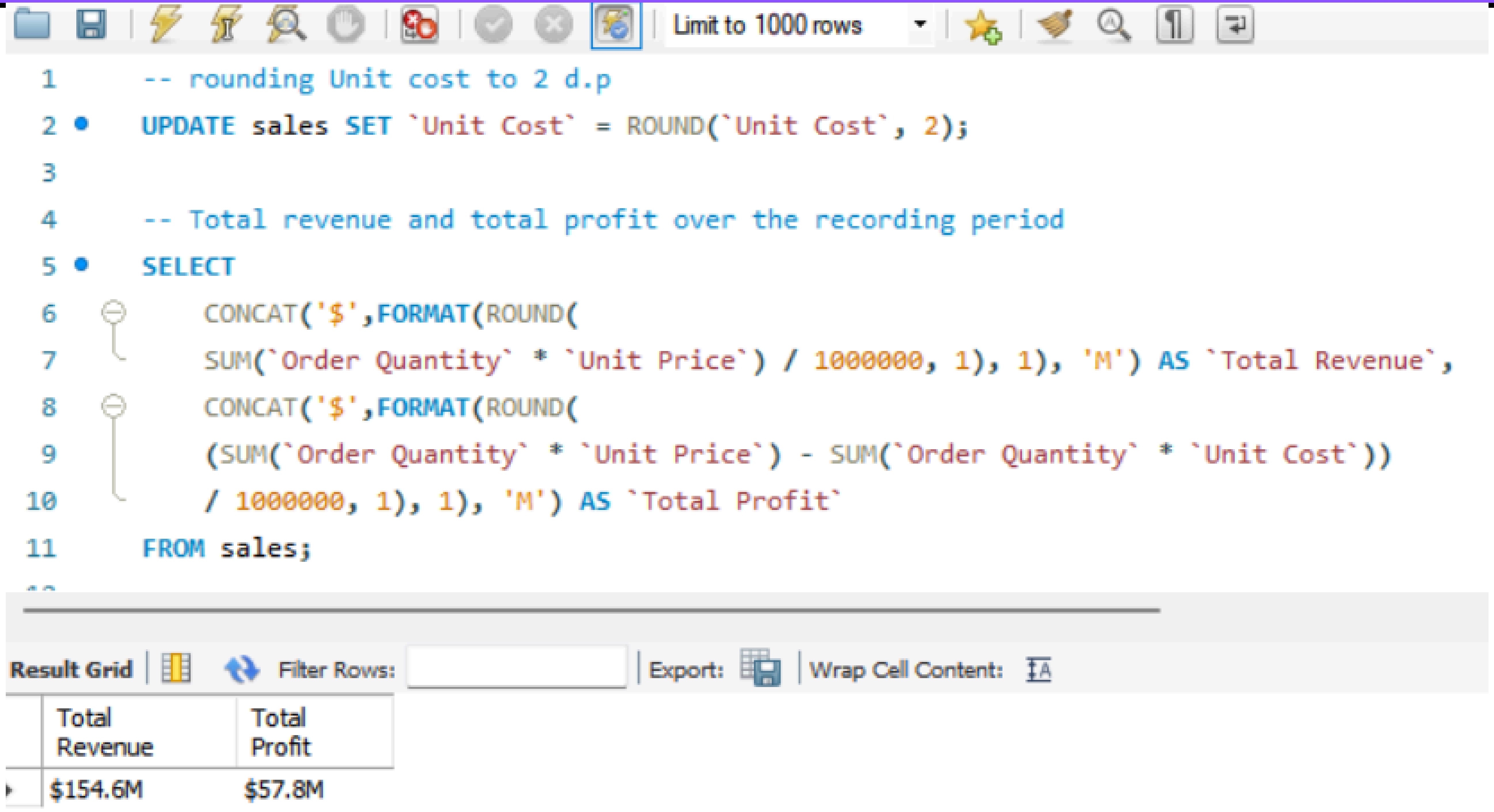


```sql
UPDATE sales SET OrderDate = STR_TO_DATE(OrderDate, '%m/%d/%Y');
ALTER TABLE sales MODIFY COLUMN OrderDate DATE;

SELECT
    MIN(`OrderDate`) as `Earliest order date`,
    MAX(`OrderDate`) as `Latest order date`,
    DATEDIFF(MAX(`OrderDate`), MIN(`OrderDate`)) AS `range (days)`
FROM sales;
```

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

Earliest order date	Latest order date	range (days)
2018-05-31	2020-12-30	944

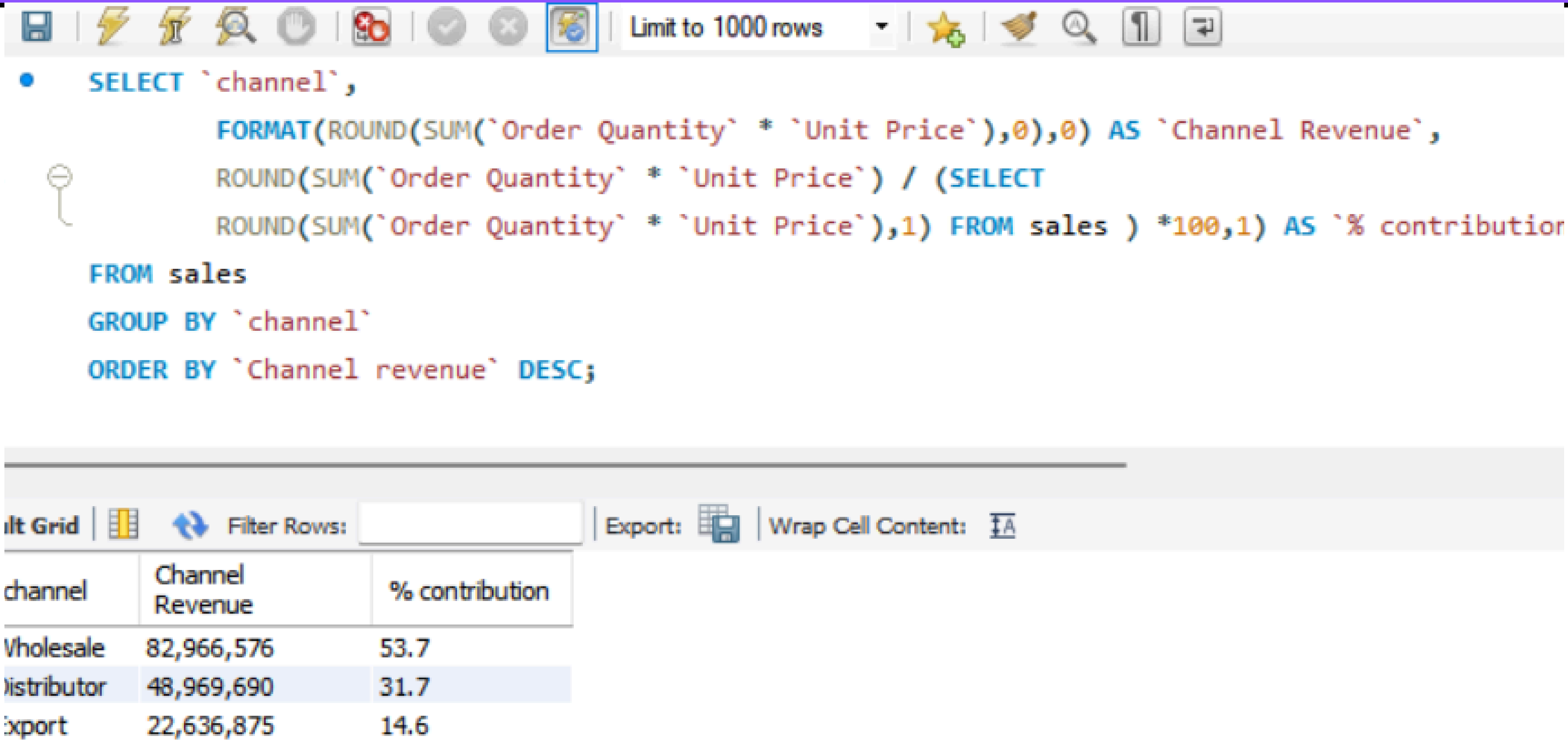
Q2: CALCULATE THE REVENUE AND PROFIT GENERATED OVER THE RECORDING PERIOD



```sql
-- rounding Unit cost to 2 d.p
UPDATE sales SET `Unit Cost` = ROUND(`Unit Cost`, 2);

-- Total revenue and total profit over the recording period
SELECT
    CONCAT('$',FORMAT(ROUND(
    SUM(`Order Quantity` * `Unit Price`) / 1000000, 1), 1), 'M') AS `Total Revenue`,
    CONCAT('$',FORMAT(ROUND(
    (SUM(`Order Quantity` * `Unit Price`) - SUM(`Order Quantity` * `Unit Cost`))
    / 1000000, 1), 1), 'M') AS `Total Profit`
FROM sales;
```

Total Revenue	Total Profit
$154.6M	$57.8M

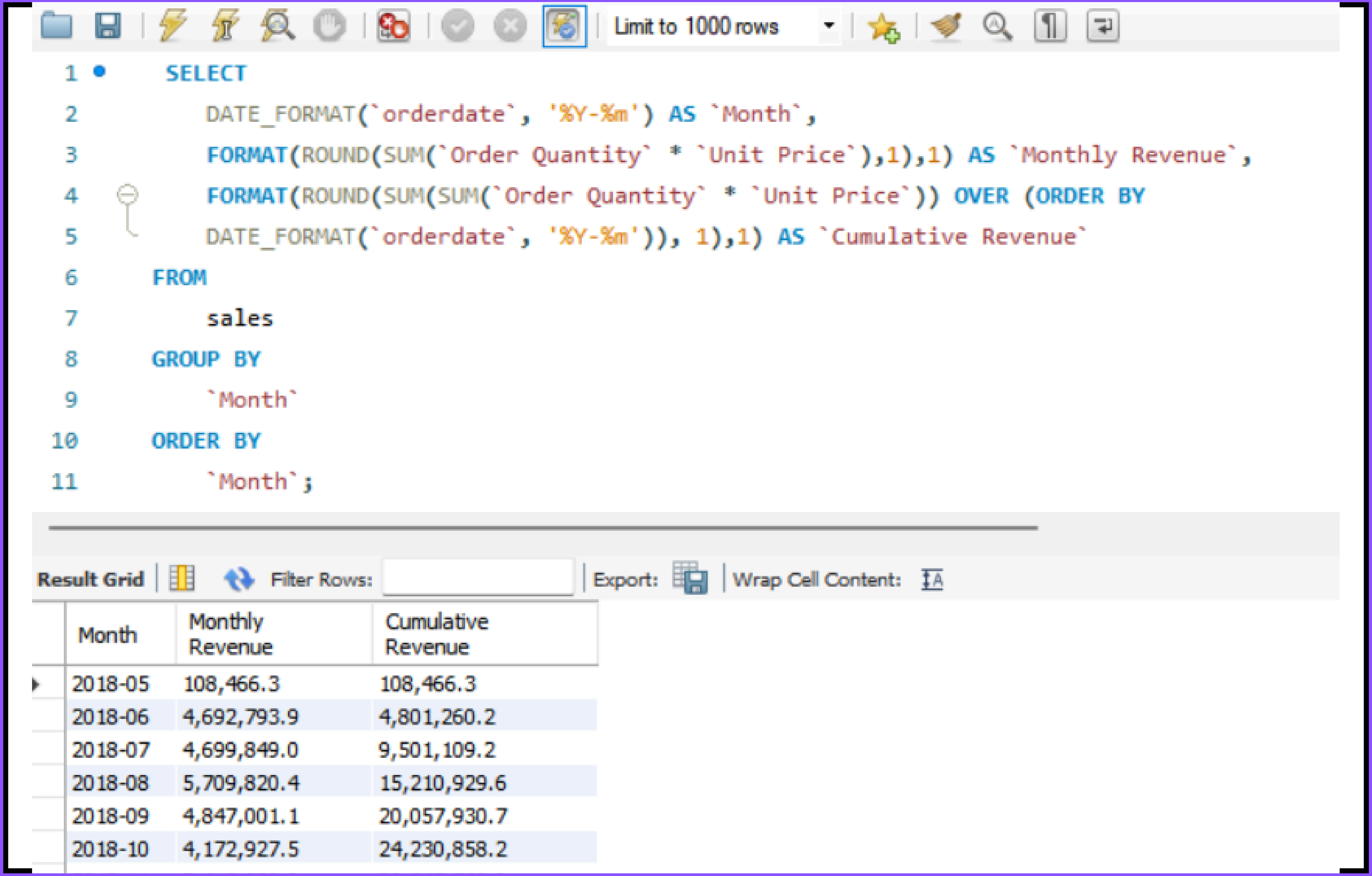
HOW MUCH DOES EACH CHANNEL CONTRIBUTE TO SALES?



```
SELECT `channel`,
        FORMAT(ROUND(SUM(`Order Quantity` * `Unit Price`),0),0) AS `Channel Revenue`,
        ROUND(SUM(`Order Quantity` * `Unit Price`) / (SELECT
        ROUND(SUM(`Order Quantity` * `Unit Price`),1) FROM sales ) *100,1) AS `% contributio
FROM sales
GROUP BY `channel`
ORDER BY `Channel revenue` DESC;
```

channel	Channel Revenue	% contribution
Vholesale	82,966,576	53.7
)istributor	48,969,690	31.7
:xport	22,636,875	14.6

CALCULATE THE TOTAL REVENUE GENERATED PER MONTH AND THE CUMULATIVE REVENUE



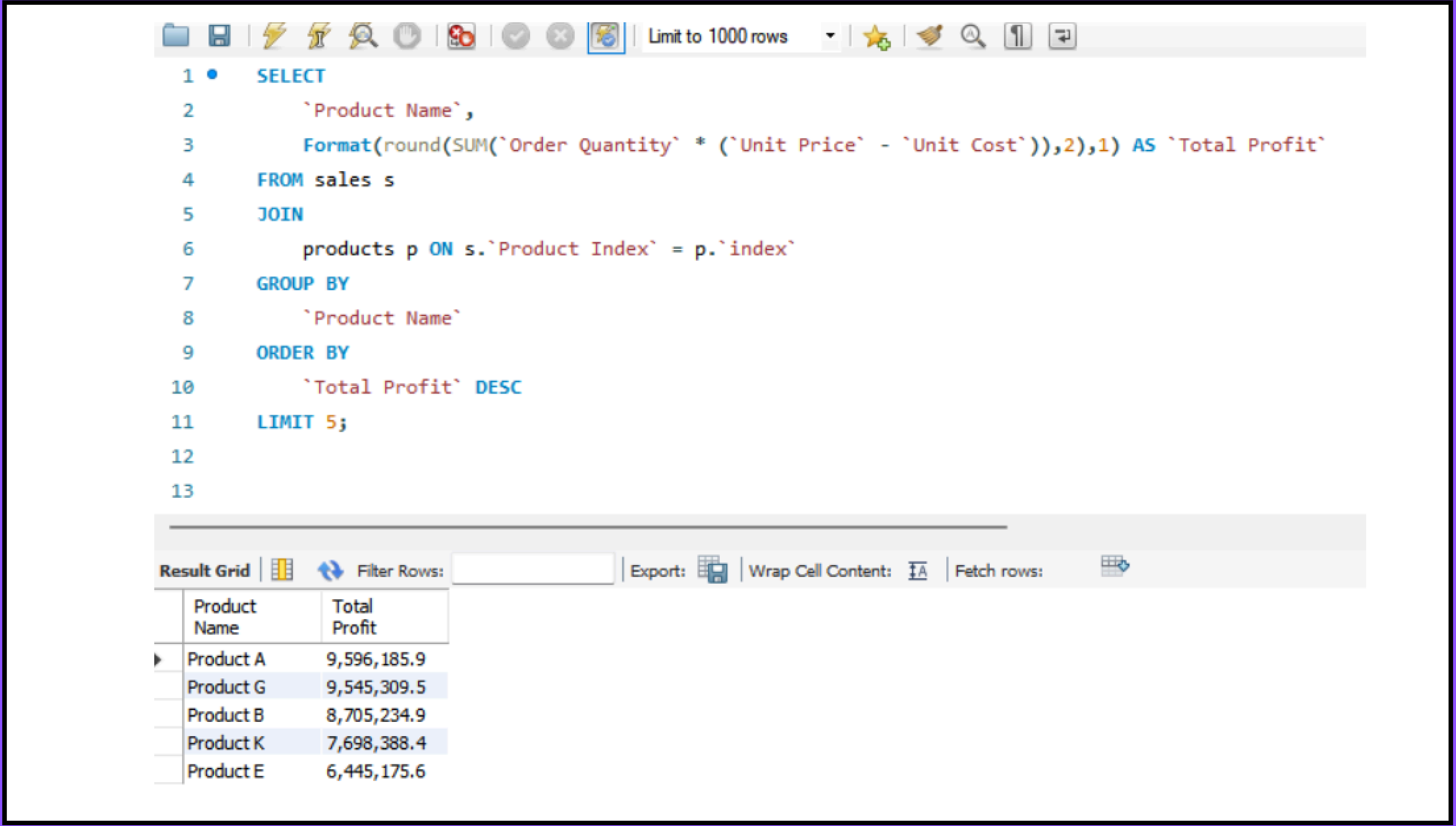
Limit to 1000 rows

```sql
SELECT
    DATE_FORMAT(`orderdate`, '%Y-%m') AS `Month`,
    FORMAT(ROUND(SUM(`Order Quantity` * `Unit Price`),1),1) AS `Monthly Revenue`,
    FORMAT(ROUND(SUM(SUM(`Order Quantity` * `Unit Price`)) OVER (ORDER BY
    DATE_FORMAT(`orderdate`, '%Y-%m')), 1),1) AS `Cumulative Revenue`
FROM
    sales
GROUP BY
    `Month`
ORDER BY
    `Month`;
```

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

Month	Monthly Revenue	Cumulative Revenue
2018-05	108,466.3	108,466.3
2018-06	4,692,793.9	4,801,260.2
2018-07	4,699,849.0	9,501,109.2
2018-08	5,709,820.4	15,210,929.6
2018-09	4,847,001.1	20,057,930.7
2018-10	4,172,927.5	24,230,858.2

5 MOST PROFITABLE PRODUCTS



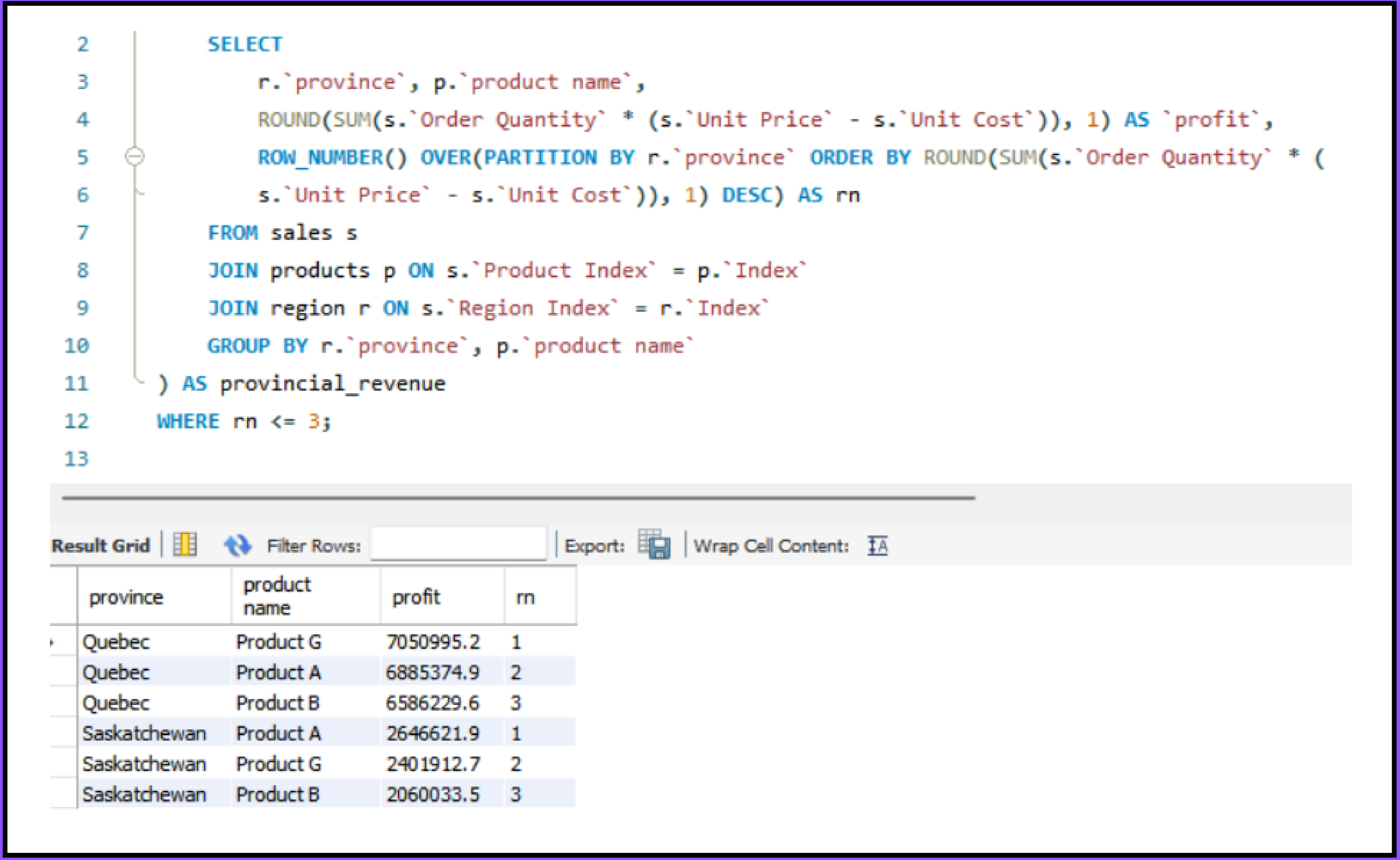
Limit to 1000 rows

```
SELECT
    `Product Name`,
    Format(round(SUM(`Order Quantity` * (`Unit Price` - `Unit Cost`)),2),1) AS `Total Profit`
FROM sales s
JOIN
    products p ON s.`Product Index` = p.`index`
GROUP BY
    `Product Name`
ORDER BY
    `Total Profit` DESC
LIMIT 5;
```

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

Product Name	Total Profit
Product A	9,596,185.9
Product G	9,545,309.5
Product B	8,705,234.9
Product K	7,698,388.4
Product E	6,445,175.6

TOP 3 MOST PROFITABLE PRODUCTS IN EACH PROVINCE

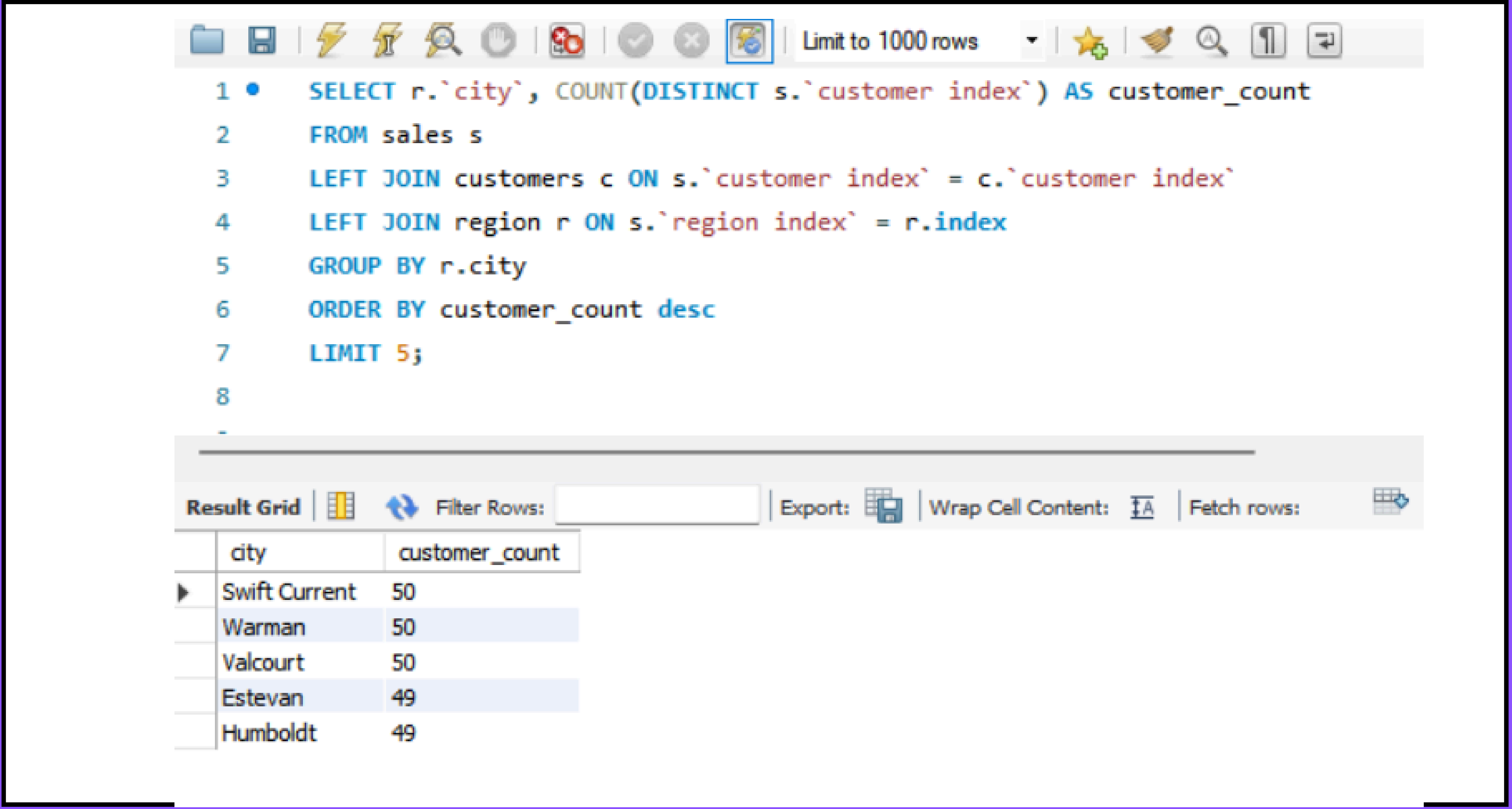


```
    SELECT
        r.`province`, p.`product name`,
        ROUND(SUM(s.`Order Quantity` * (s.`Unit Price` - s.`Unit Cost`)), 1) AS `profit`,
        ROW_NUMBER() OVER(PARTITION BY r.`province` ORDER BY ROUND(SUM(s.`Order Quantity` * (
        s.`Unit Price` - s.`Unit Cost`)), 1) DESC) AS rn
    FROM sales s
    JOIN products p ON s.`Product Index` = p.`Index`
    JOIN region r ON s.`Region Index` = r.`Index`
    GROUP BY r.`province`, p.`product name`
) AS provincial_revenue
WHERE rn <= 3;
```

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

province	product name	profit	rn
Quebec	Product G	7050995.2	1
Quebec	Product A	6885374.9	2
Quebec	Product B	6586229.6	3
Saskatchewan	Product A	2646621.9	1
Saskatchewan	Product G	2401912.7	2
Saskatchewan	Product B	2060033.5	3

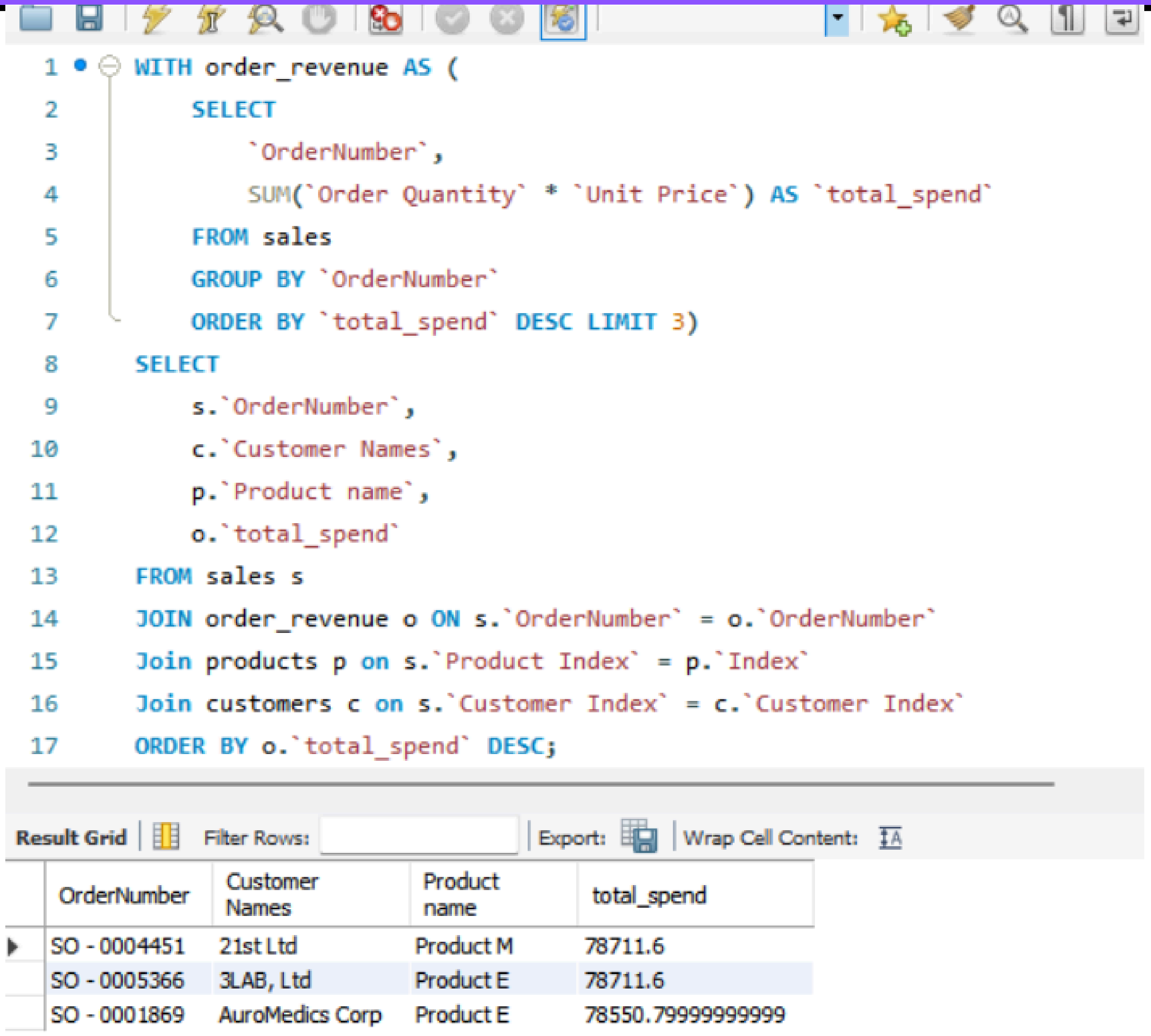
WHICH CITIES HAVE THE MOST CUSTOMERS?



```sql
SELECT r.`city`, COUNT(DISTINCT s.`customer index`) AS customer_count
FROM sales s
LEFT JOIN customers c ON s.`customer index` = c.`customer index`
LEFT JOIN region r ON s.`region index` = r.index
GROUP BY r.city
ORDER BY customer_count desc
LIMIT 5;
```

city	customer_count
Swift Current	50
Warman	50
Valcourt	50
Estevan	49
Humboldt	49

RETRIEVE THE ORDER NUMBER, CUSTOMER NAME, PRODUCT NAME, TOTAL SPENT OF THE TOP 3 ORDERS WITH THE MAXIMUM REVENUE

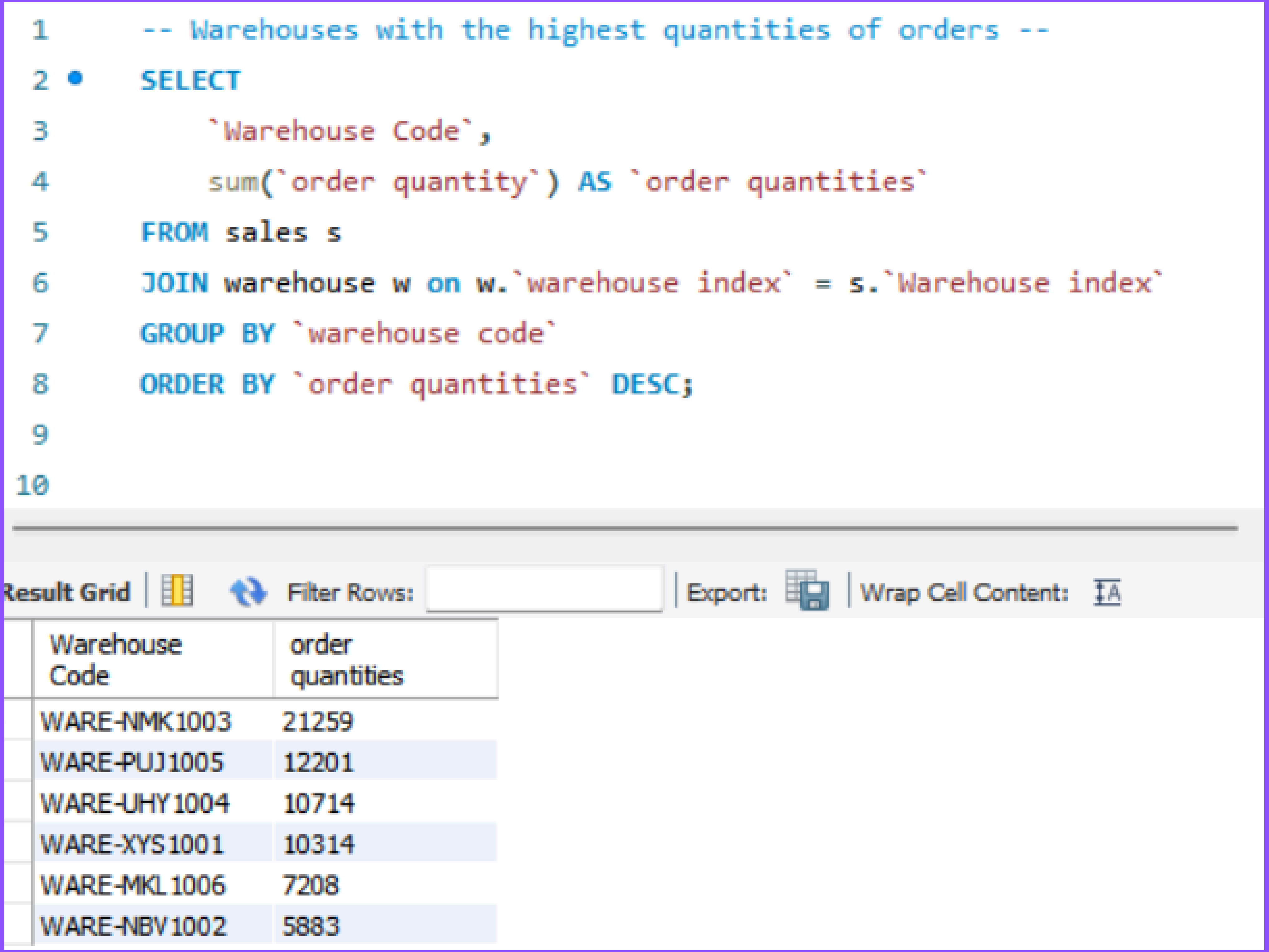


```sql
WITH order_revenue AS (
    SELECT
        `OrderNumber`,
        SUM(`Order Quantity` * `Unit Price`) AS `total_spend`
    FROM sales
    GROUP BY `OrderNumber`
    ORDER BY `total_spend` DESC LIMIT 3)
SELECT
    s.`OrderNumber`,
    c.`Customer Names`,
    p.`Product name`,
    o.`total_spend`
FROM sales s
JOIN order_revenue o ON s.`OrderNumber` = o.`OrderNumber`
Join products p on s.`Product Index` = p.`Index`
Join customers c on s.`Customer Index` = c.`Customer Index`
ORDER BY o.`total_spend` DESC;
```

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

OrderNumber	Customer Names	Product name	total_spend
SO - 0004451	21st Ltd	Product M	78711.6
SO - 0005366	3LAB, Ltd	Product E	78711.6
SO - 0001869	AuroMedics Corp	Product E	78550.79999999999

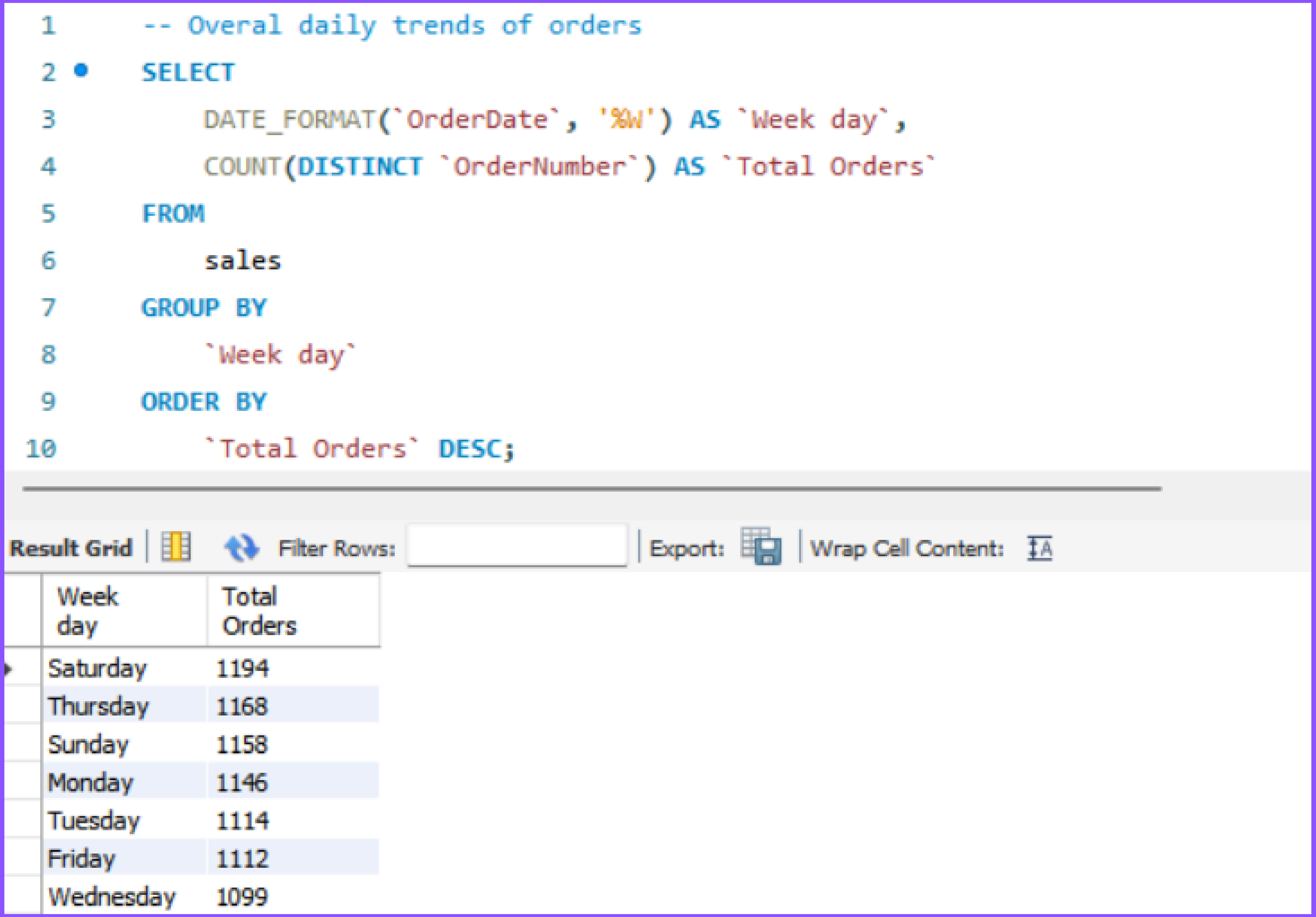
LIST THE WAREHOUSE FROM THE BUSIEST TO THE ONE WHICH PROCESS LEAST ORDERS



```sql
-- Warehouses with the highest quantities of orders --
SELECT
    `Warehouse Code`,
    sum(`order quantity`) AS `order quantities`
FROM sales s
JOIN warehouse w on w.`warehouse index` = s.`Warehouse index`
GROUP BY `warehouse code`
ORDER BY `order quantities` DESC;
```

Warehouse Code	order quantities
WARE-NMK1003	21259
WARE-PUJ1005	12201
WARE-UHY1004	10714
WARE-XYS1001	10314
WARE-MKL1006	7208
WARE-NBV1002	5883

OVERALL DAILY TRENDS OF SALES

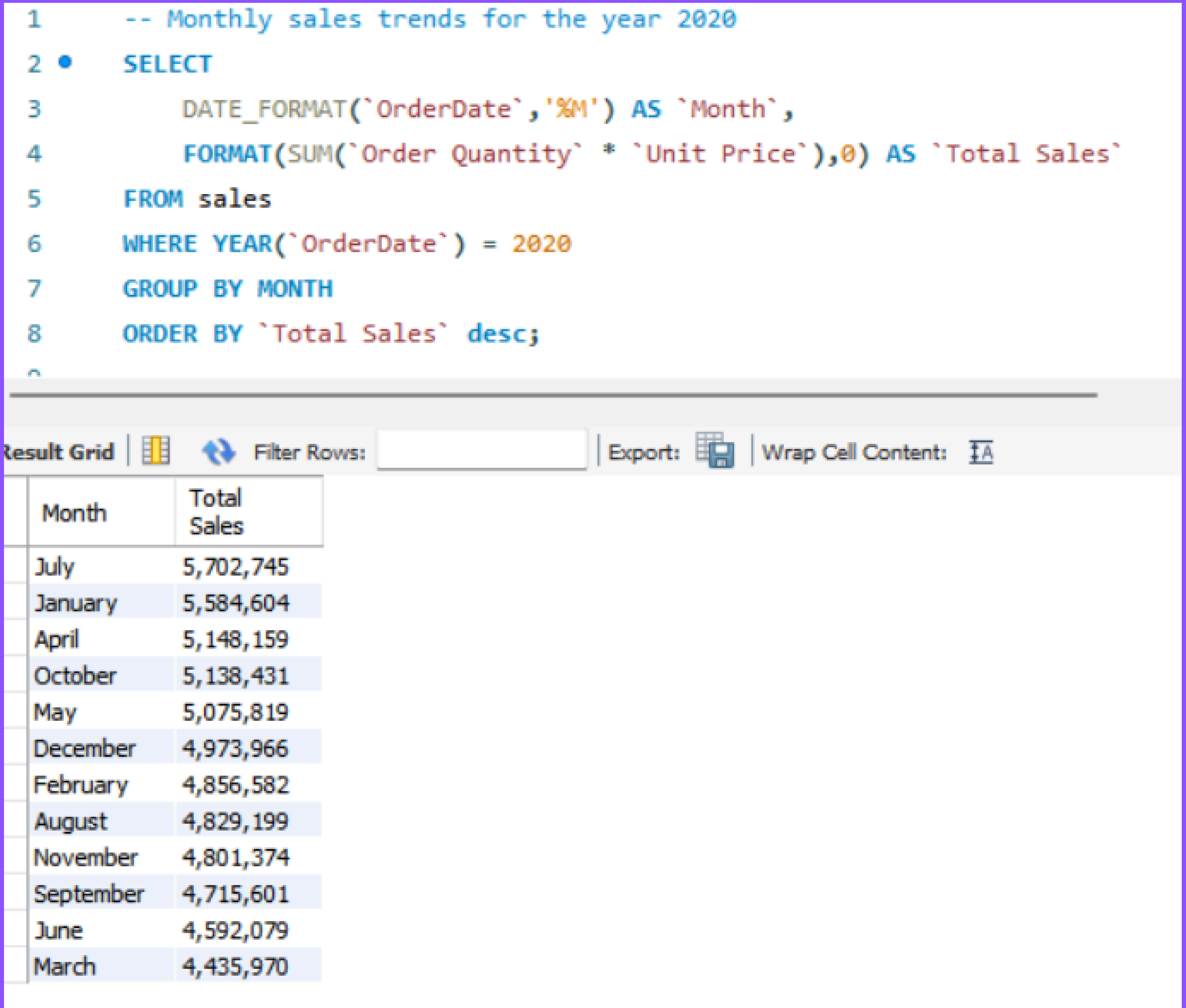


```sql
-- Overal daily trends of orders
SELECT
    DATE_FORMAT(`OrderDate`, '%W') AS `Week day`,
    COUNT(DISTINCT `OrderNumber`) AS `Total Orders`
FROM
    sales
GROUP BY
    `Week day`
ORDER BY
    `Total Orders` DESC;
```

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

Week day	Total Orders
Saturday	1194
Thursday	1168
Sunday	1158
Monday	1146
Tuesday	1114
Friday	1112
Wednesday	1099

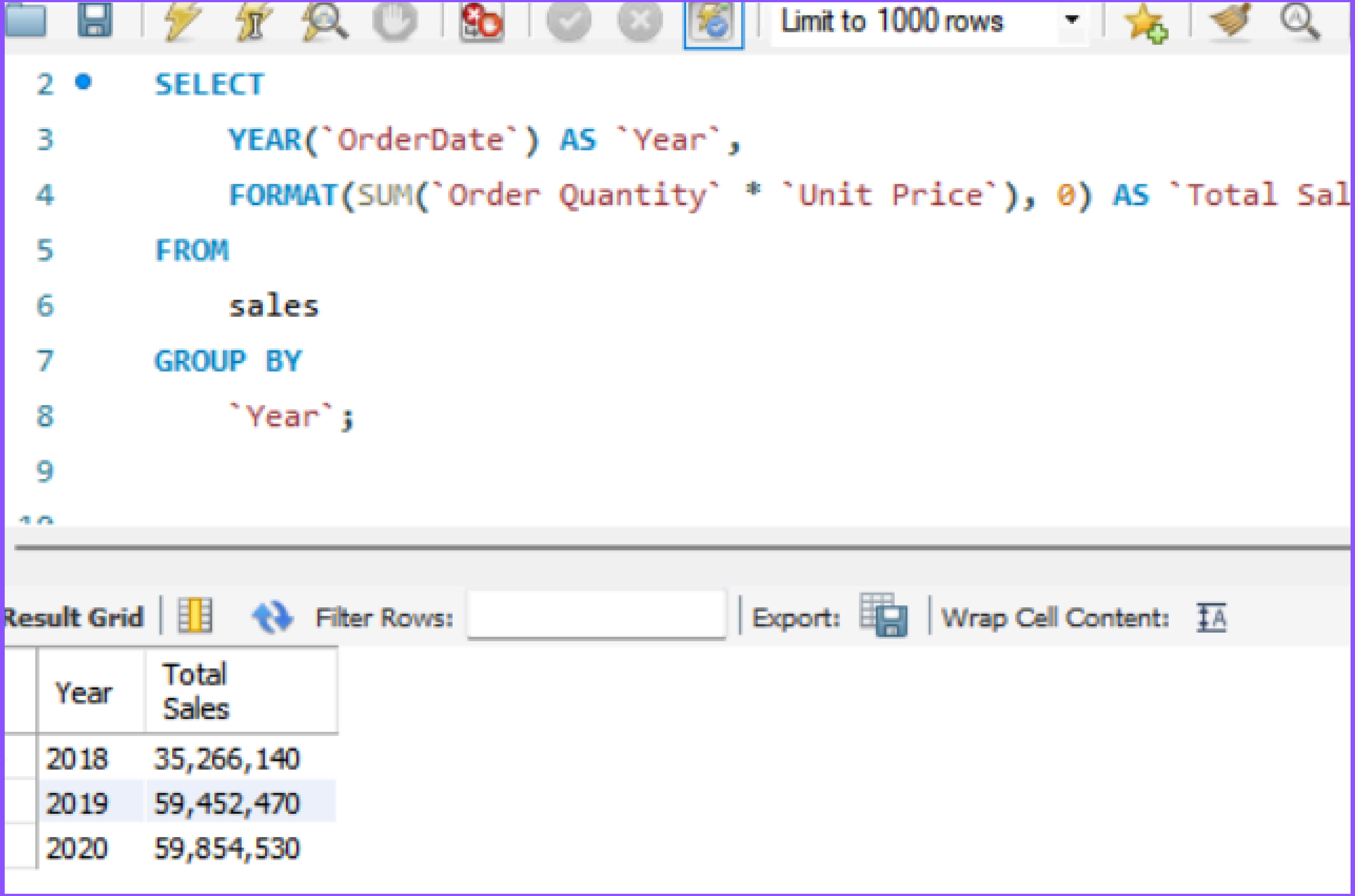
OVERALL MONTHLY TRENDS OF SALES FOR THE YEAR 2022



```sql
-- Monthly sales trends for the year 2020
SELECT
    DATE_FORMAT(`OrderDate`,'%M') AS `Month`,
    FORMAT(SUM(`Order Quantity` * `Unit Price`),0) AS `Total Sales`
FROM sales
WHERE YEAR(`OrderDate`) = 2020
GROUP BY MONTH
ORDER BY `Total Sales` desc;
```

Month	Total Sales
July	5,702,745
January	5,584,604
April	5,148,159
October	5,138,431
May	5,075,819
December	4,973,966
February	4,856,582
August	4,829,199
November	4,801,374
September	4,715,601
June	4,592,079
March	4,435,970

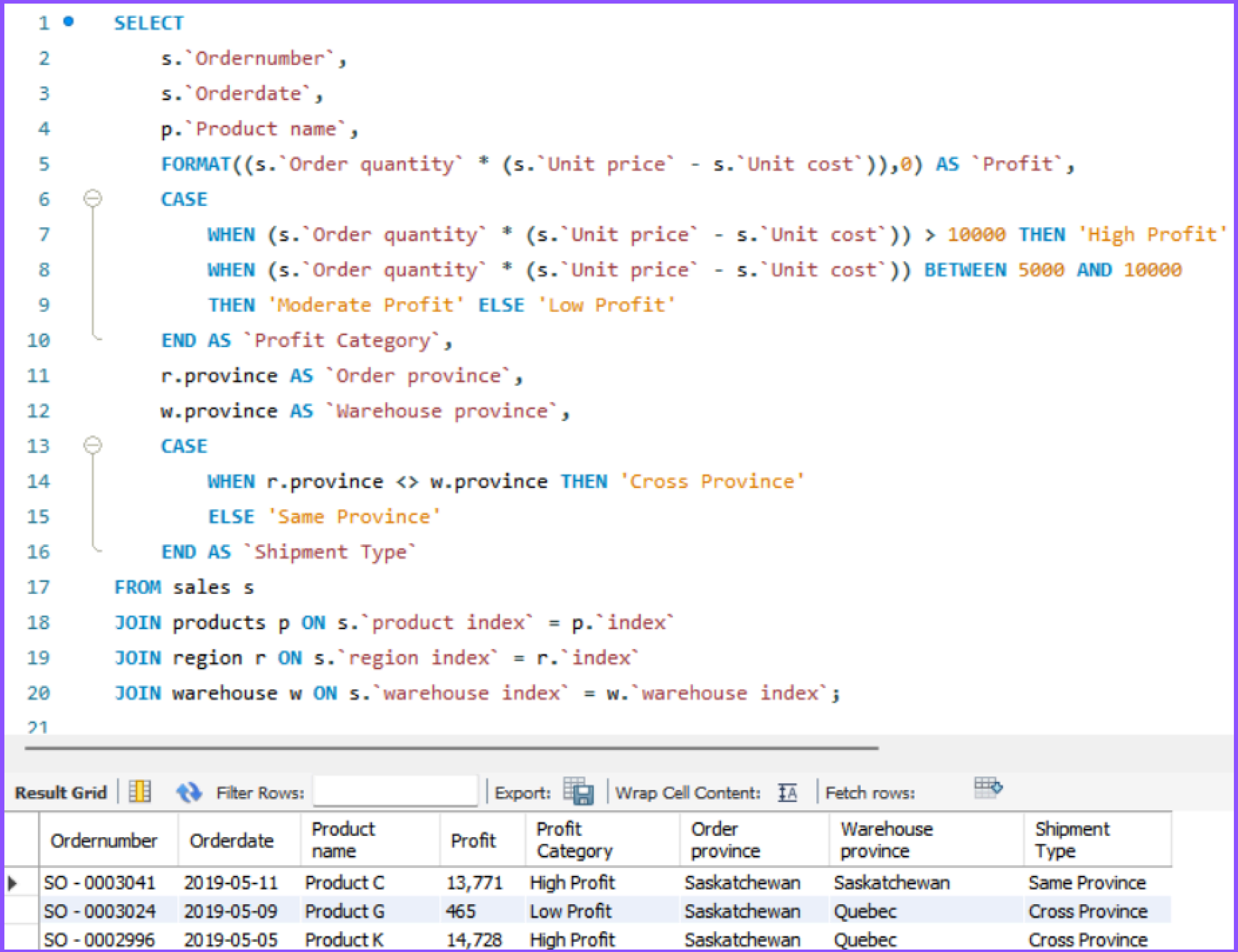
YEARLY SALES



```
SELECT
    YEAR(`OrderDate`) AS `Year`,
    FORMAT(SUM(`Order Quantity` * `Unit Price`), 0) AS `Total Sal
FROM
    sales
GROUP BY
    `Year`;
```

Year	Total Sales
2018	35,266,140
2019	59,452,470
2020	59,854,530

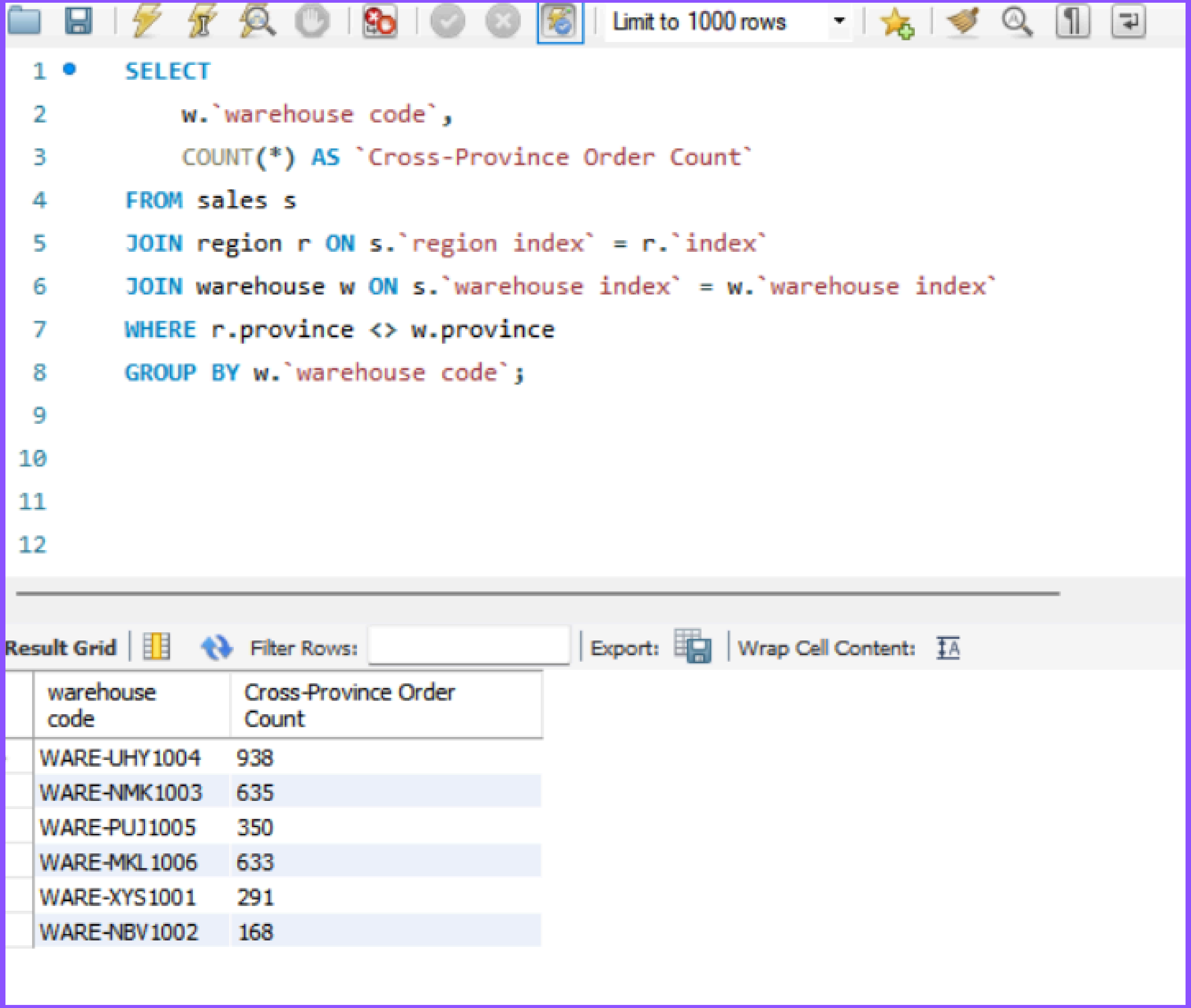
DETERMINING THE PROFITABILITY OF EACH ORDER AND IF THE SHIPMENTS WAS FROM THE SAME PROVINCE



```sql
SELECT
    s.`Ordernumber`,
    s.`Orderdate`,
    p.`Product name`,
    FORMAT((s.`Order quantity` * (s.`Unit price` - s.`Unit cost`)),0) AS `Profit`,
    CASE
        WHEN (s.`Order quantity` * (s.`Unit price` - s.`Unit cost`)) > 10000 THEN 'High Profit'
        WHEN (s.`Order quantity` * (s.`Unit price` - s.`Unit cost`)) BETWEEN 5000 AND 10000
        THEN 'Moderate Profit' ELSE 'Low Profit'
    END AS `Profit Category`,
    r.province AS `Order province`,
    w.province AS `Warehouse province`,
    CASE
        WHEN r.province <> w.province THEN 'Cross Province'
        ELSE 'Same Province'
    END AS `Shipment Type`
FROM sales s
JOIN products p ON s.`product index` = p.`index`
JOIN region r ON s.`region index` = r.`index`
JOIN warehouse w ON s.`warehouse index` = w.`warehouse index`;
```

Ordernumber	Orderdate	Product name	Profit	Profit Category	Order province	Warehouse province	Shipment Type
SO - 0003041	2019-05-11	Product C	13,771	High Profit	Saskatchewan	Saskatchewan	Same Province
SO - 0003024	2019-05-09	Product G	465	Low Profit	Saskatchewan	Quebec	Cross Province
SO - 0002996	2019-05-05	Product K	14,728	High Profit	Saskatchewan	Quebec	Cross Province

IDENTIFY THE WAREHOUSES WHICH SHIPPED THE MOST ORDERS TO OTHER PROVINCES

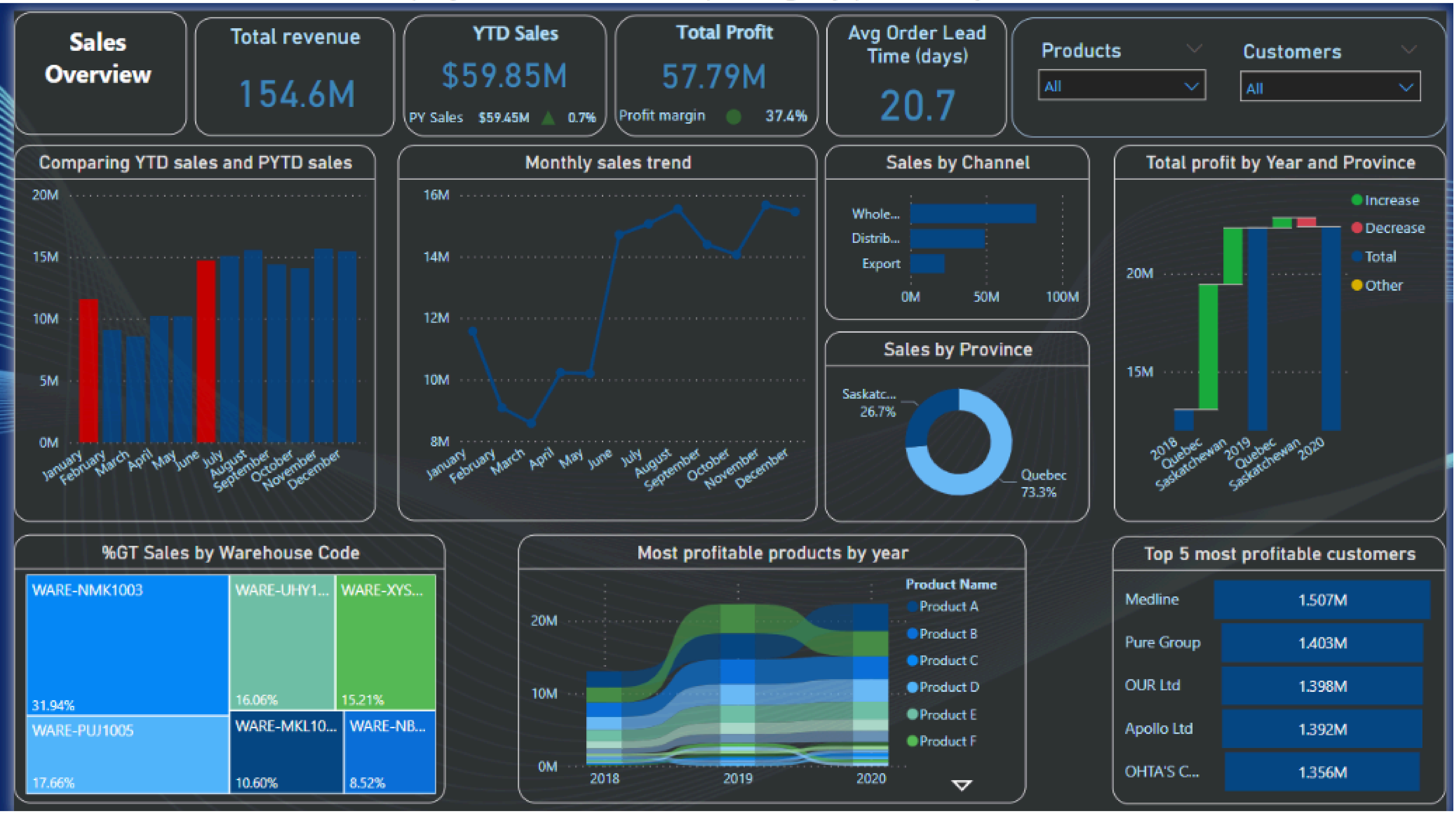


Limit to 1000 rows

```sql
SELECT
    w.`warehouse code`,
    COUNT(*) AS `Cross-Province Order Count`
FROM sales s
JOIN region r ON s.`region index` = r.`index`
JOIN warehouse w ON s.`warehouse index` = w.`warehouse index`
WHERE r.province <> w.province
GROUP BY w.`warehouse code`;
```

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

warehouse code	Cross-Province Order Count
WARE-UHY1004	938
WARE-NMK1003	635
WARE-PUJ1005	350
WARE-MKL1006	633
WARE-XYS1001	291
WARE-NBV1002	168



Sales Overview

Total revenue
154.6M

YTD Sales
$59.85M
PY Sales $59.45M ▲ 0.7%

Total Profit
57.79M
Profit margin ● 37.4%

Avg Order Lead Time (days)
20.7

Products: All

Customers: All