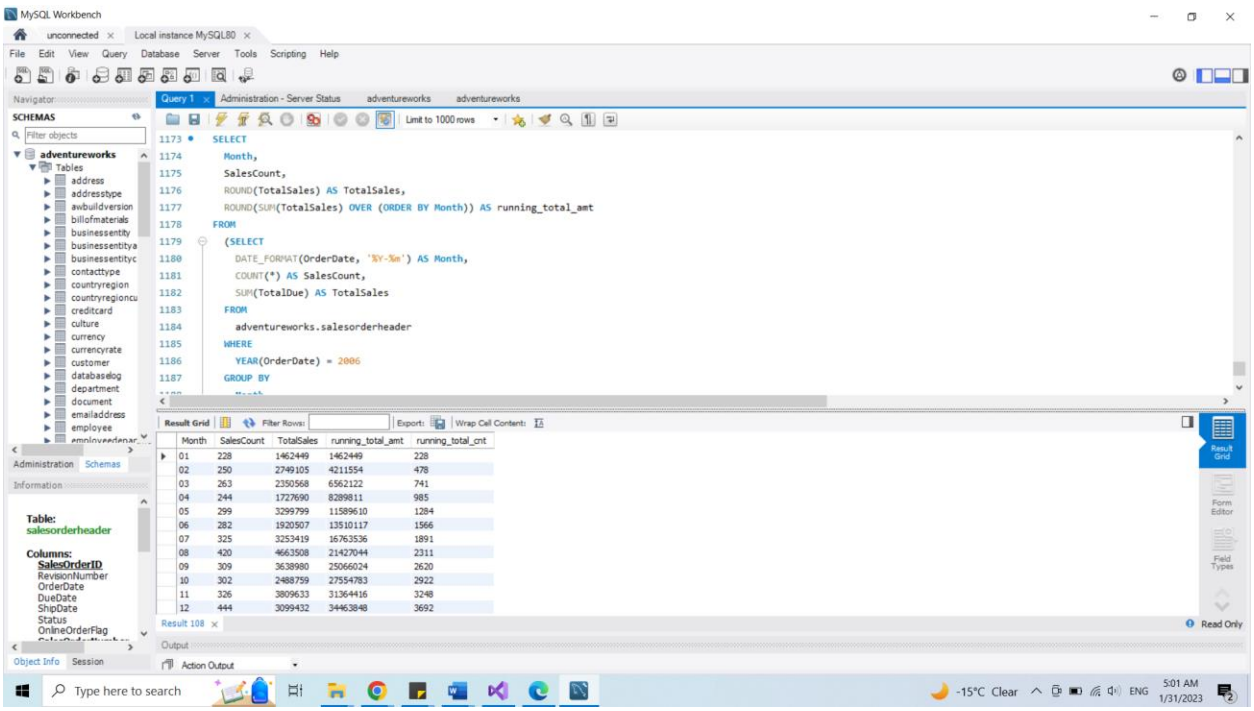


```
SELECT
  Month,
  SalesCount,
  ROUND(TotalSales) AS TotalSales,
  ROUND(SUM(TotalSales) OVER (ORDER BY Month)) AS running_total_amt
FROM
  (SELECT
    DATE_FORMAT(OrderDate, '%Y-%m') AS Month,
    COUNT(*) AS SalesCount,
    SUM(TotalDue) AS TotalSales
  FROM
    adventureworks.salesorderheader
  WHERE
    YEAR(OrderDate) = 2006
  GROUP BY
    Month
```

```
ORDER BY
  Month) AS sales_data;
```

Explain:

To get the query that calculates cumulative sum of **SalesCount** and **TotalSales** ordered by **Month**:

1. Start with the inner query that retrieves the data needed for the report.
    - This query uses the **DATE_FORMAT** function to format the **OrderDate** into the desired format (MM).
    - It then uses the **COUNT** function to count the number of orders for each month, and **SUM** function to calculate the total sales for each month.
    - The data is grouped by **Month**, and ordered by **Month** to ensure the cumulative sum is calculated in the correct order.
2. Wrap the inner query with the outer query to calculate the cumulative sum of **SalesCount** and **TotalSales** using the **SUM** function with the **OVER** clause.
    - The cumulative sum is calculated for each row in the result set, ordered by **Month**.
    - The result set includes columns for **Month**, **SalesCount**, **TotalSales**, cumulative sum of **TotalSales**, and cumulative sum of **SalesCount**.
3. Round the values in the **TotalSales** and cumulative sum columns using the **ROUND** function.