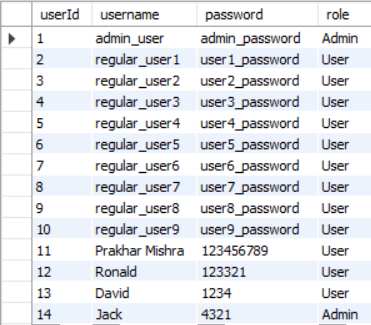
**NAME: PRAKHAR MISHRA**

**DATE: 25/01/2024**

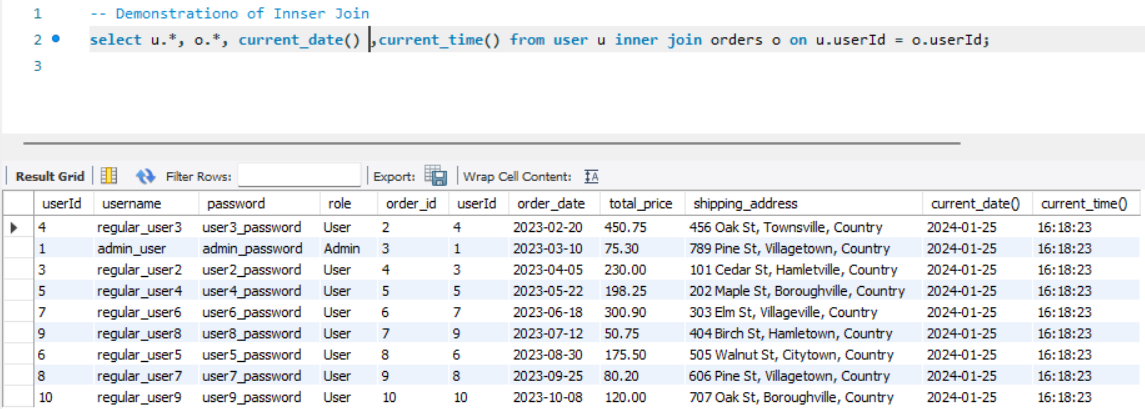
**QUESTIONS**

1. **Demonstration of all the joins with example.**

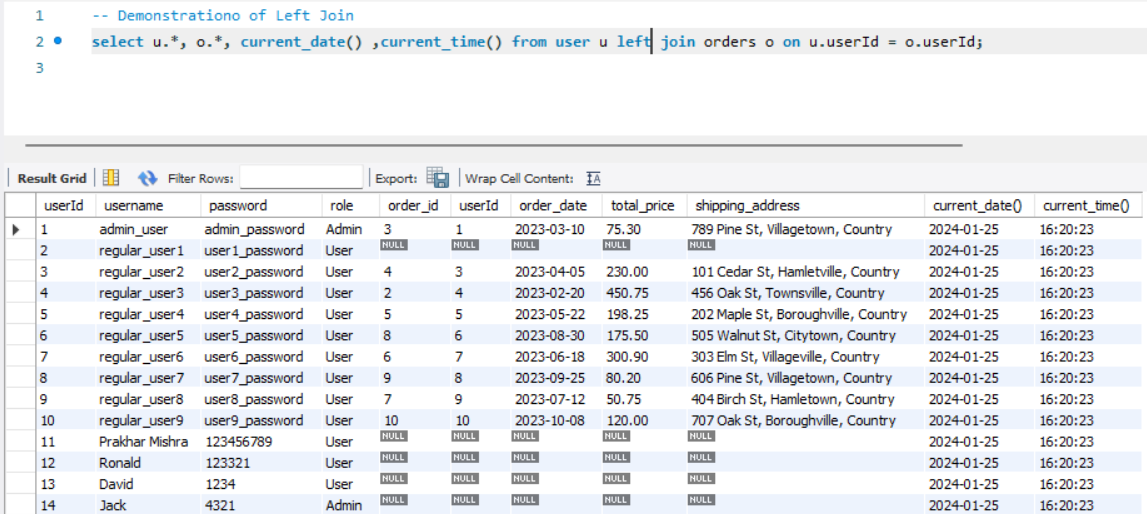
****

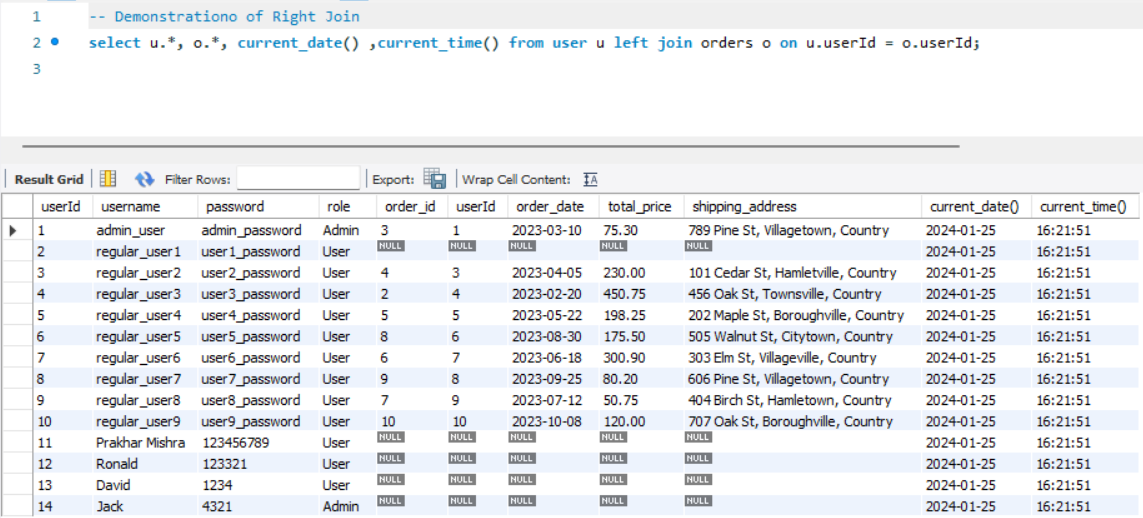
****

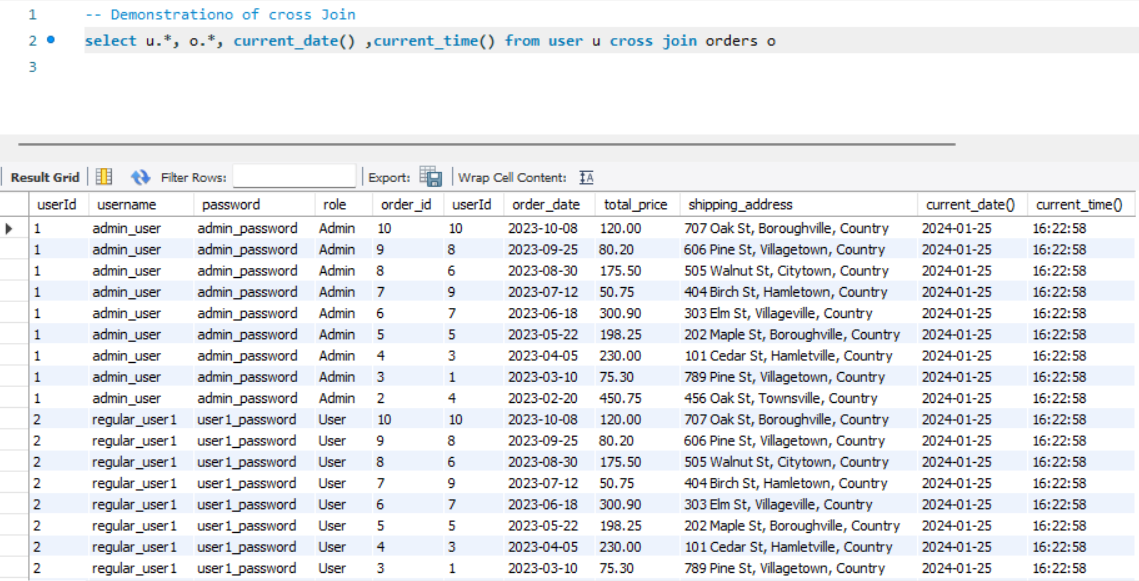
1. **Inner Join – It joins tables on the basis on some common attribute and leaves row if not meets the condition. It is like intersection of set**

****

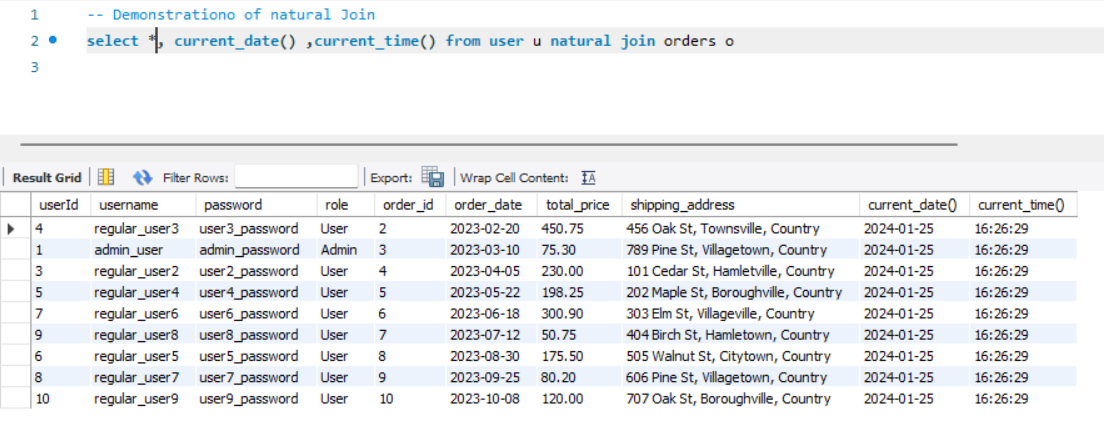
1. **Left join – It is like A-B of sets in which we keep all of A as well and A intersection B.**



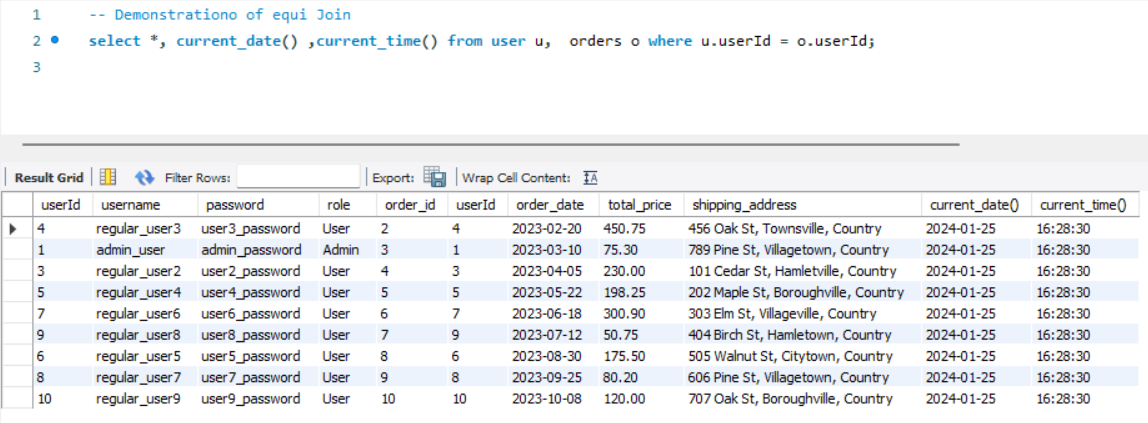
1. **Right join - It is like B-A in which we keep all of B as well A intersection B.**
2. **cross join – It is the cartesion product of two relactions. It is like each row is join with every other row. AXB.**

****

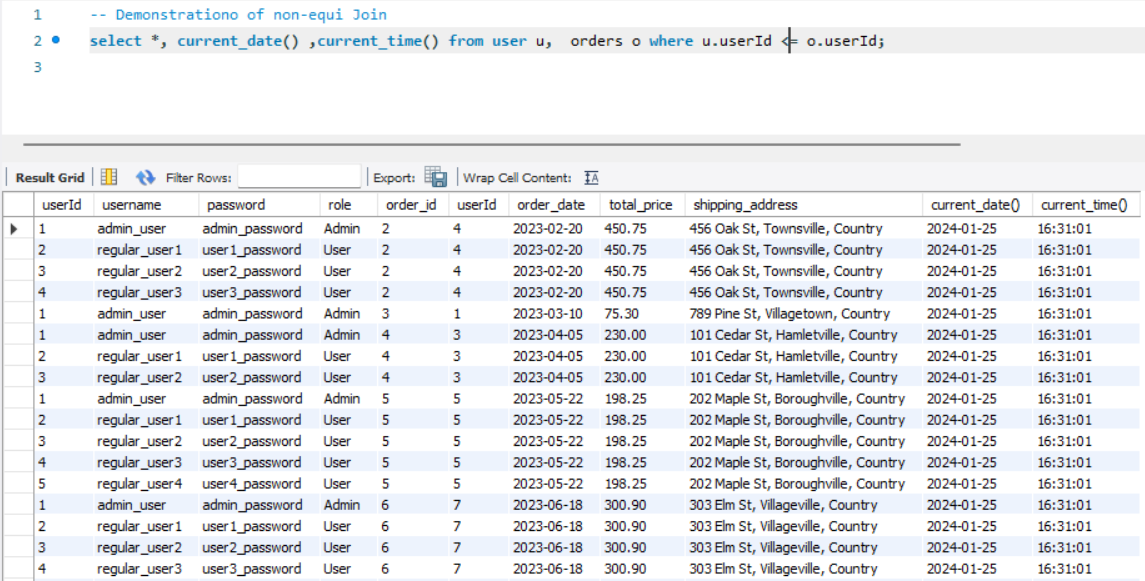
1. **Natural Join – It is also like inner join only but only one copy of connection column is kept in the final table.**

****

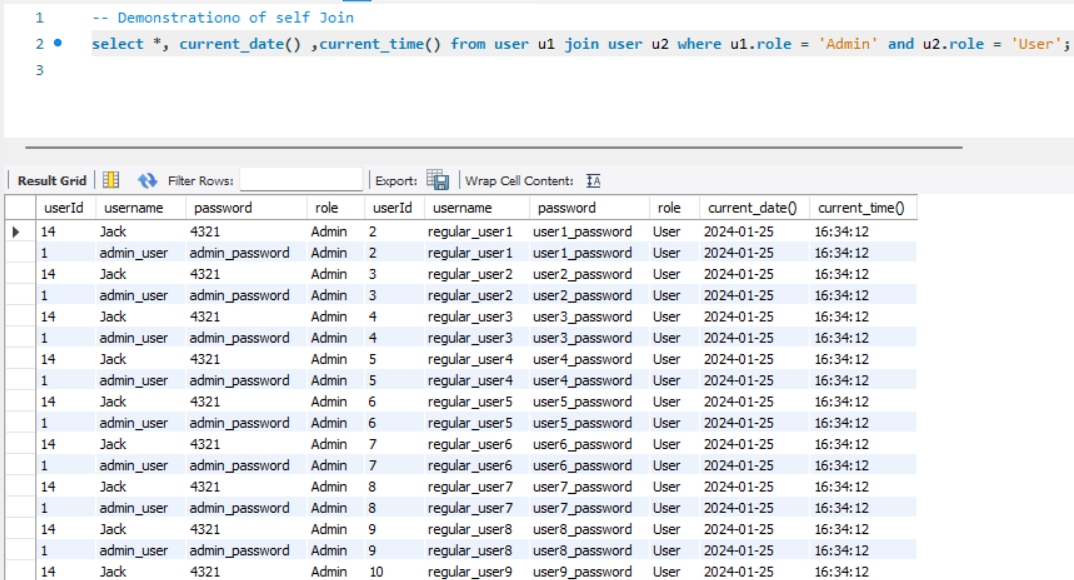
1. **Equi Join – It join tables based on some condition and uses = necessariliy.**



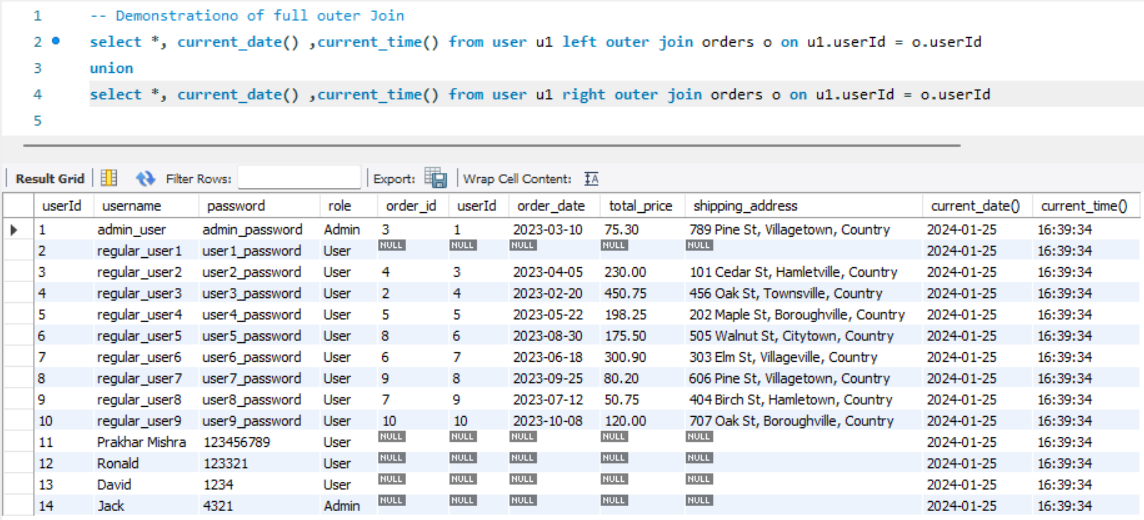
1. **Non-Equi Join – It also joins two tables but use <=, >=, !=, <, > to join tables.**

****

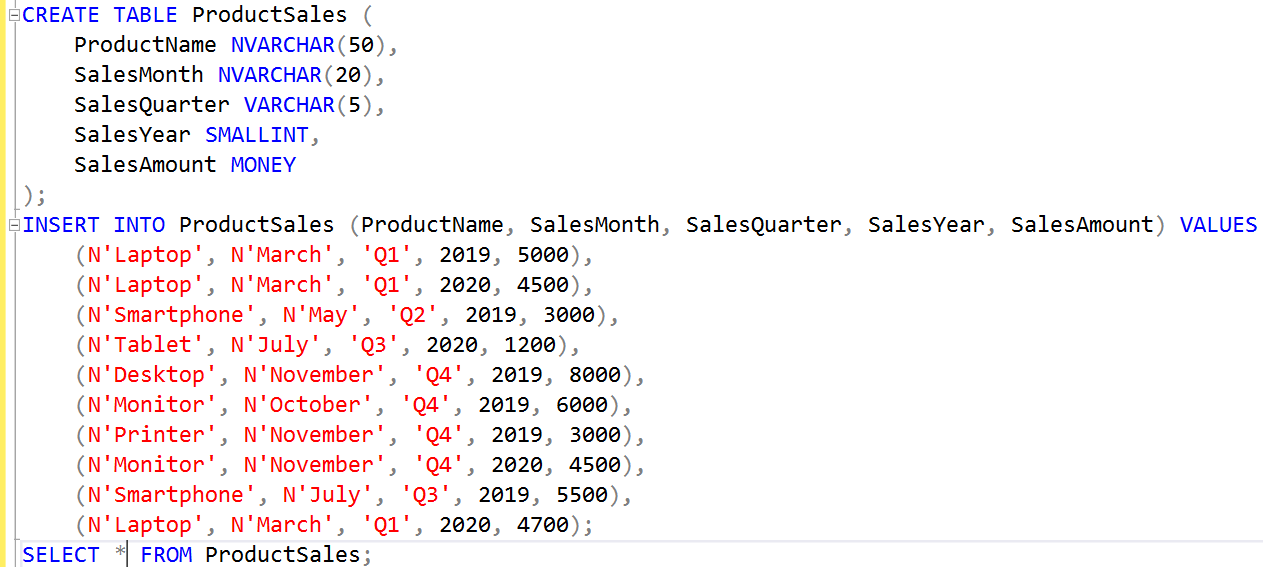
1. **Self-Join - When a tables is joined to itself based upon some condition it is know as self join.**

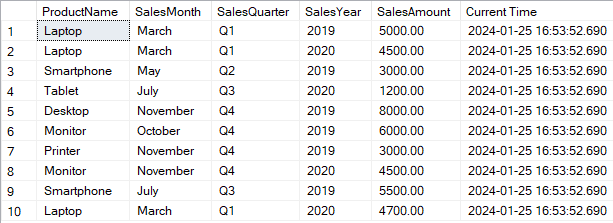
****

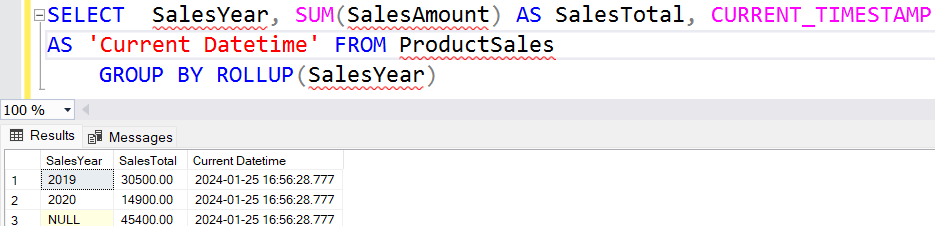
1. **Full Outer Join - This is like combination of left join and right join as well as inner join.**

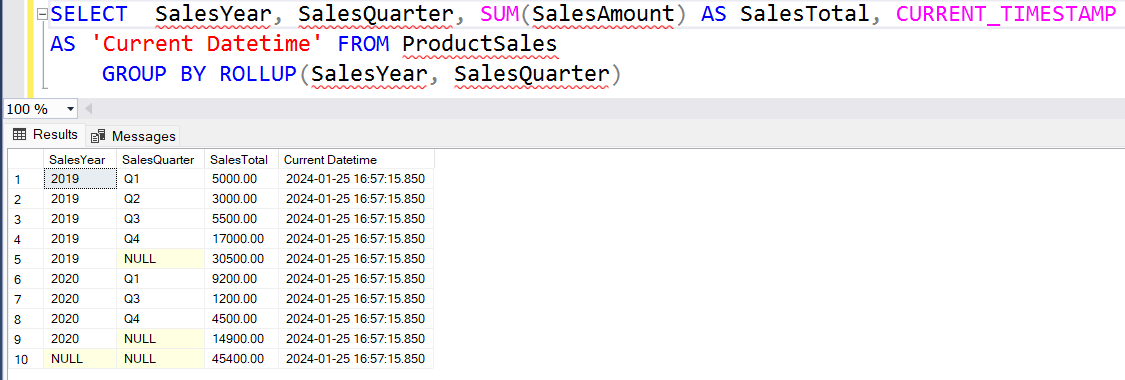
****

1. **Calculate sub-totals and aggregations using SQL queries.**

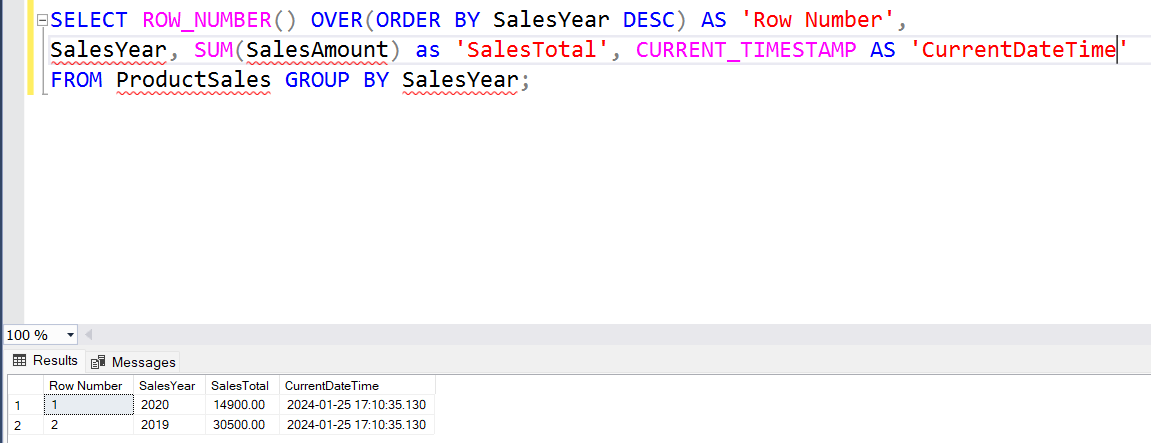
****

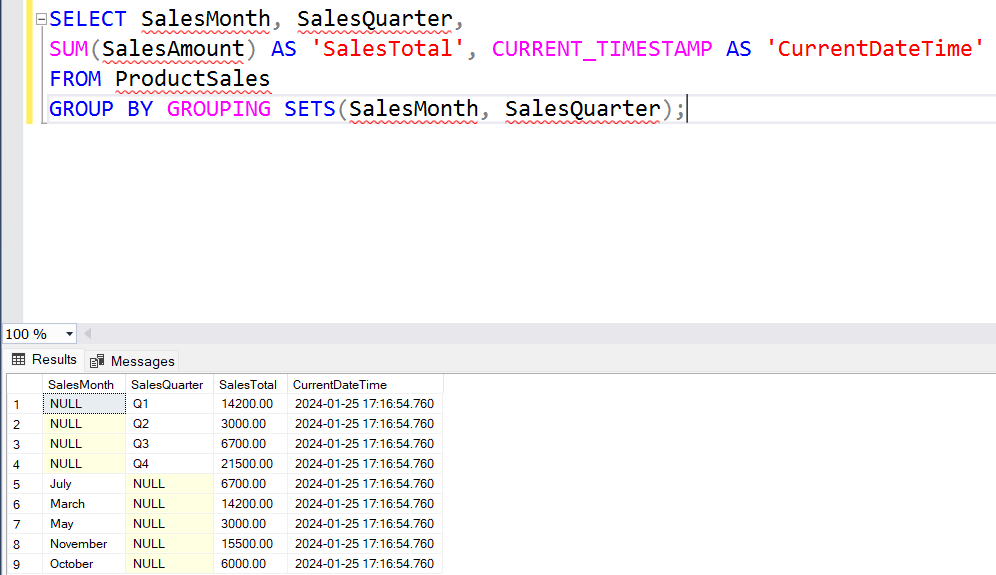
****

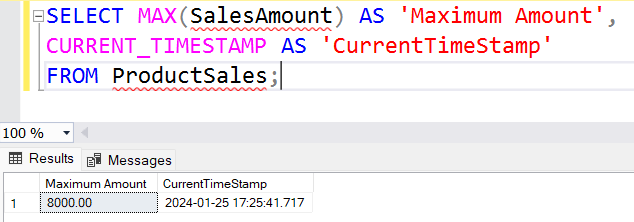
1. **Demonstration of GROUP BY Rollup. – It groups by Sales year and calculates the sum as well as calculates a cumulative sum of both the years.**

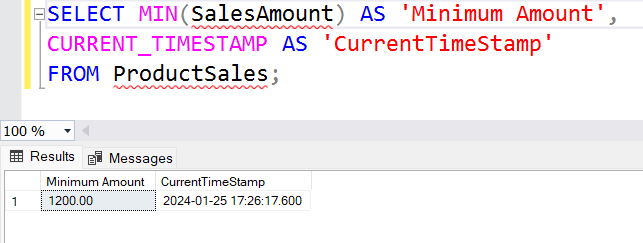
****

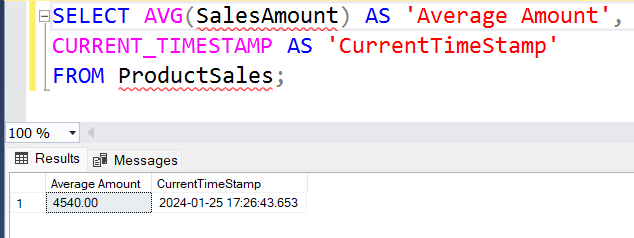
1. **Row\_Number(), Over() demonstration. – It provides a row number to calculate the futher sub-totals.**

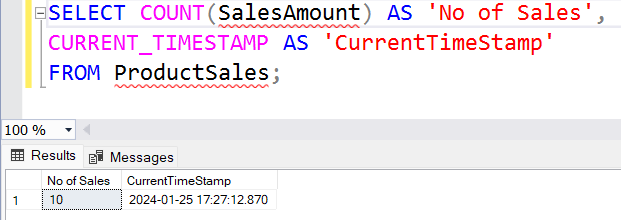
****

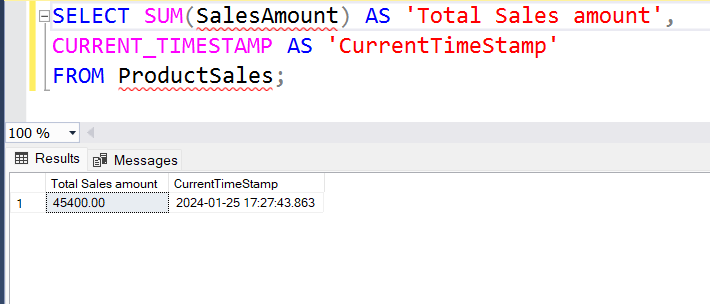
1. **Demonstration of Grouping Sets. – It groups columns to indicate null values which can be further replaced with meaning full description. Eg Subtotal, grand-total.**
2. **Aggregation Queries – These are different aggregation queries like AVG, MAX, MIN, COUNT, SUM which are used upon different column to get insights.**

****

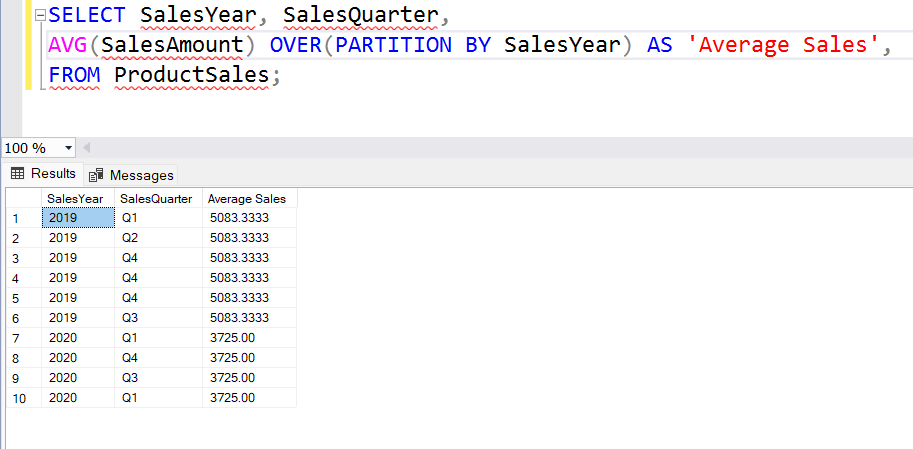
****

****

****

****

1. **Demonstration of PARTITION BY() and OVER(). – It partitions the data based upon some column. It is different from group by because it keeps the rows even after grouping them.**

****