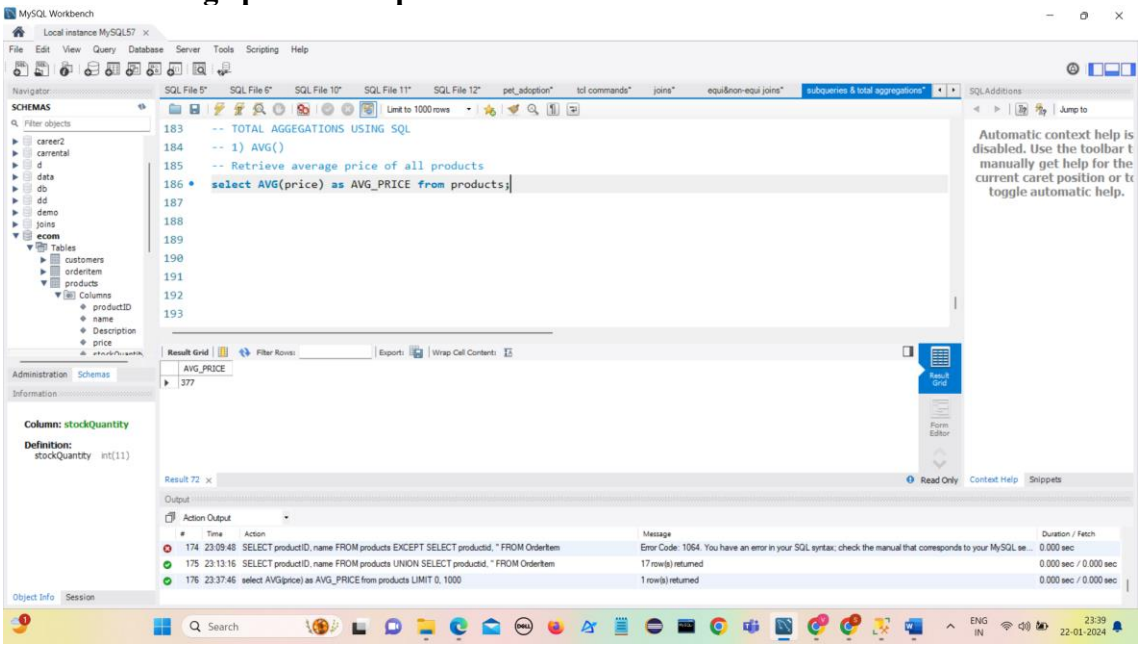
ASSIGNMENT-5

1) Total aggregations using SQL

The SQL aggregate functions — AVG, COUNT, DISTINCT, MAX, MIN, SUM — all return a value computed or derived from one column's values, after discarding any NULL values.

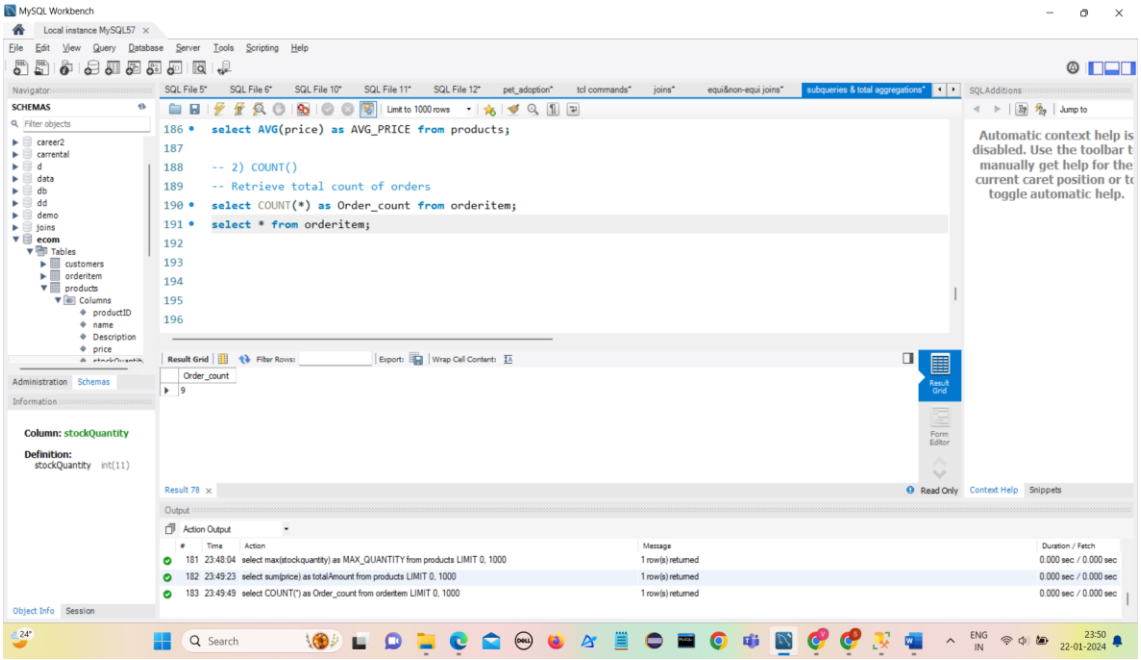
1) AVG(): The AVG() function returns the average value of a numeric column.

Retrieve average price of all products



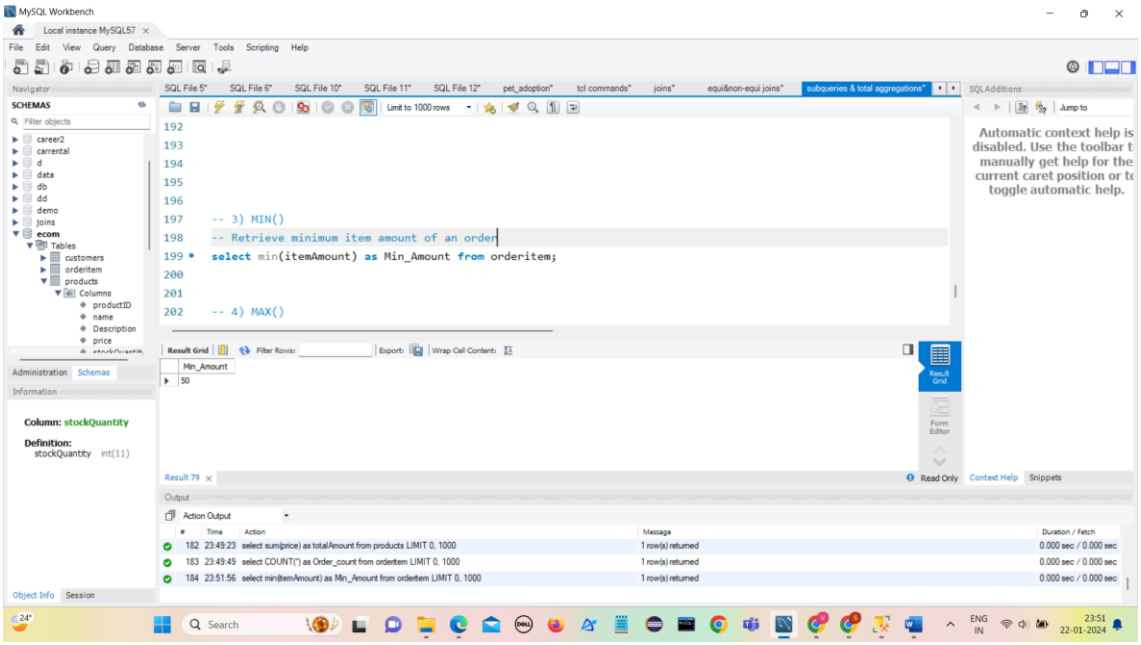
2) COUNT(): The COUNT() function returns the number of rows that matches a specified criterion.

Retrieve total count of orders



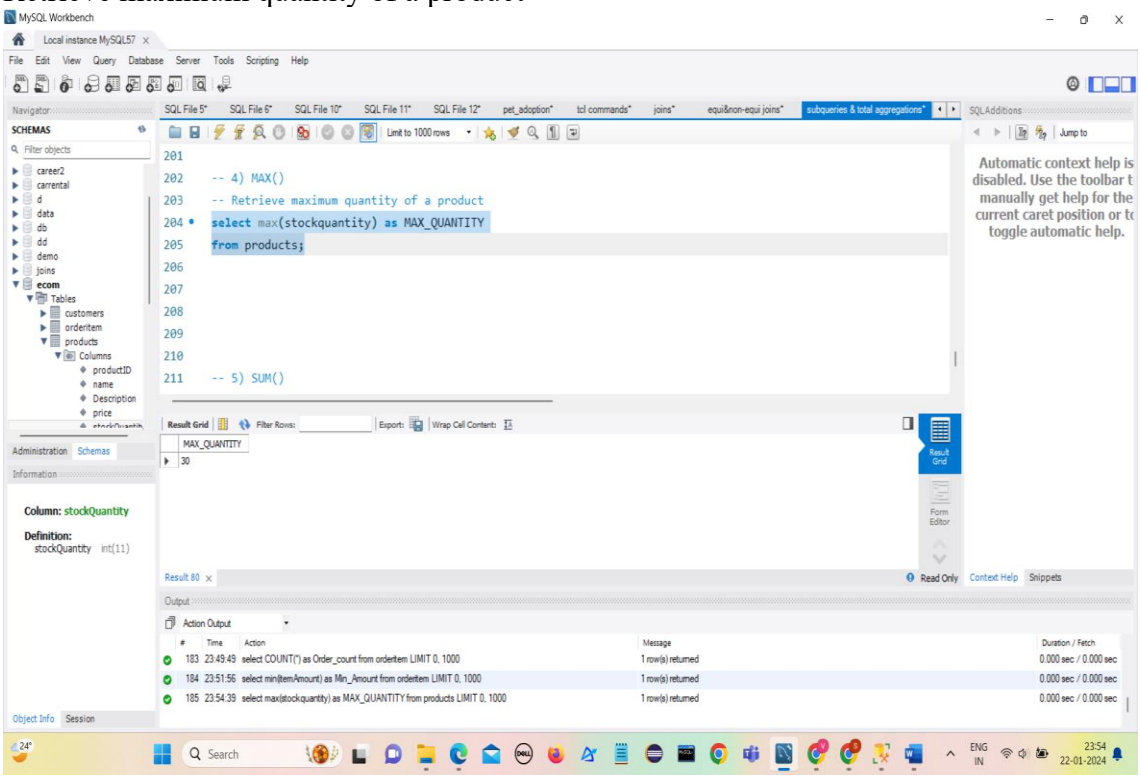
3) MIN(): The MIN() function returns the smallest value of the selected column

Retrieve minimum item amount of an order



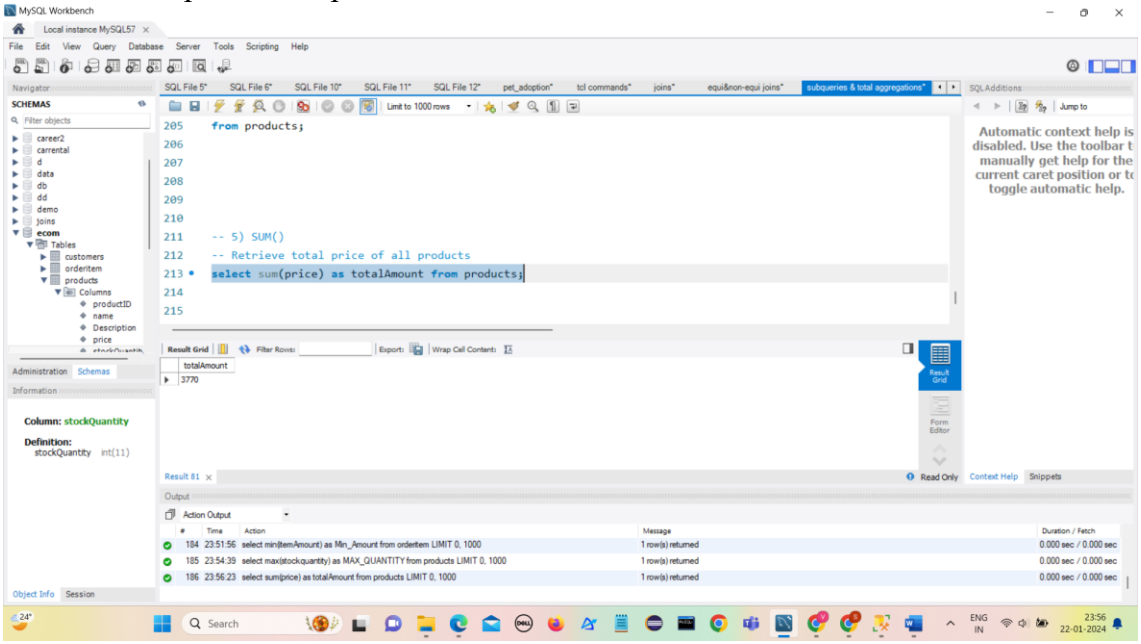
4) MAX(): The MAX() function returns the largest value of the selected column.

Retrieve maximum quantity of a product



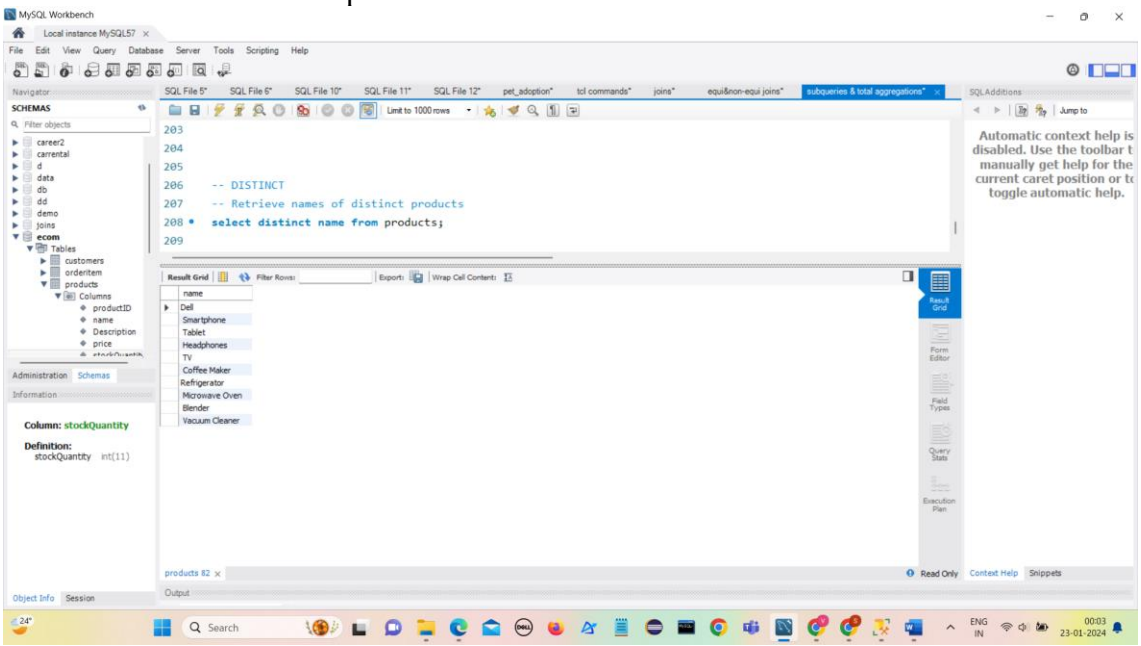
5) **SUM() :** The SUM() function returns the total sum of a numeric column.

Retrieve total price of all products



6) **DISTINCT():**

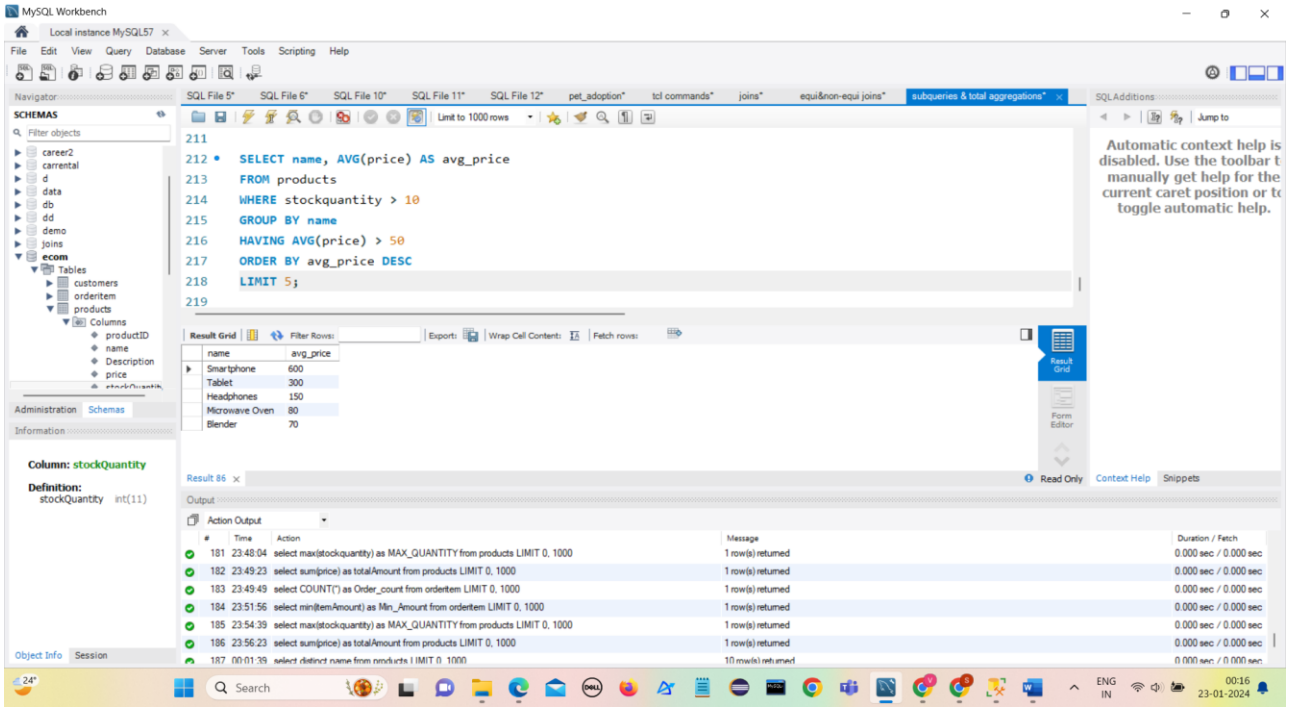
Retrieve names of distinct products.



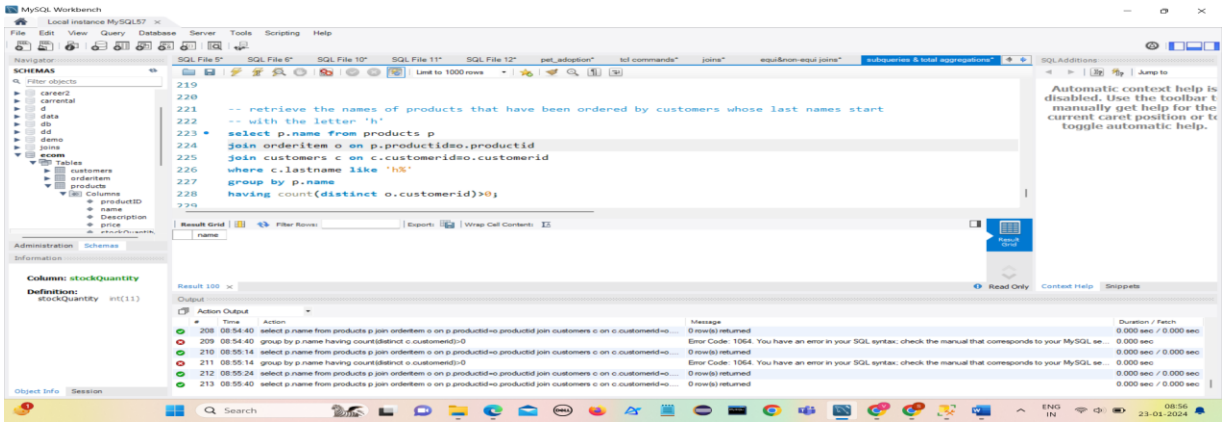
2) Order of Execution of SQL Queries:

- The order of execution determines how the database processes your query and retrieves the desired results.
- SQL order of execution refers to **the sequence in which different clauses and operations within a SQL query are processed** by the database management system.
- **Order:** FROM , WHERE , GROUP BY , HAVING , SELECT , ORDER BY, LIMIT/OFFSET.

1) Example:



2) Retrieve the names of products that have been ordered by customers whose last names start with the letter 'h'

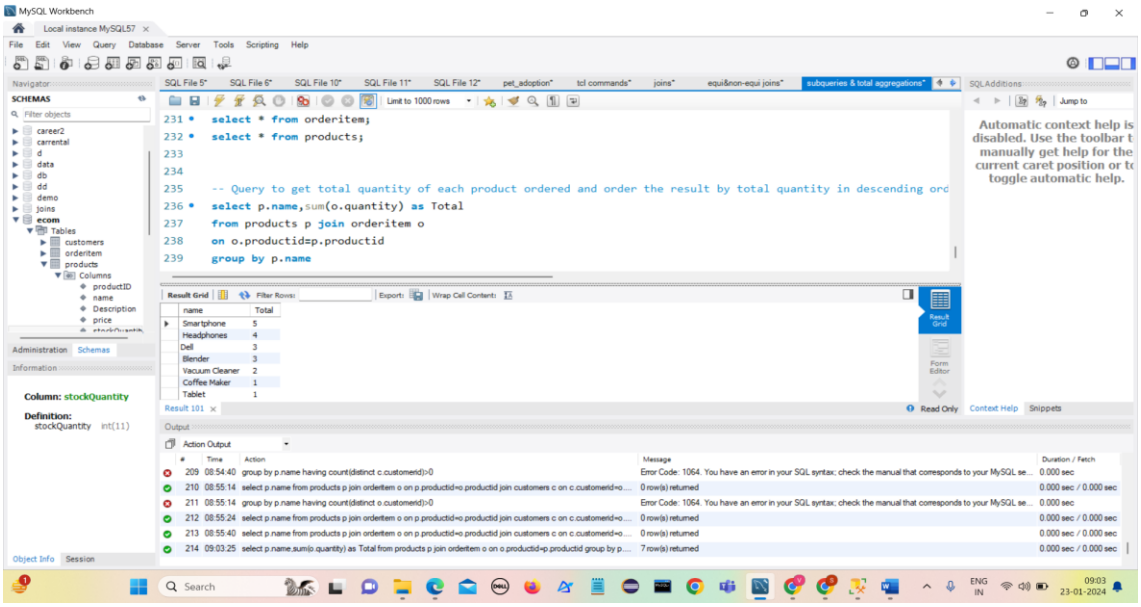


3) Rules and Restrictions to Group and Filter Data in SQL queries

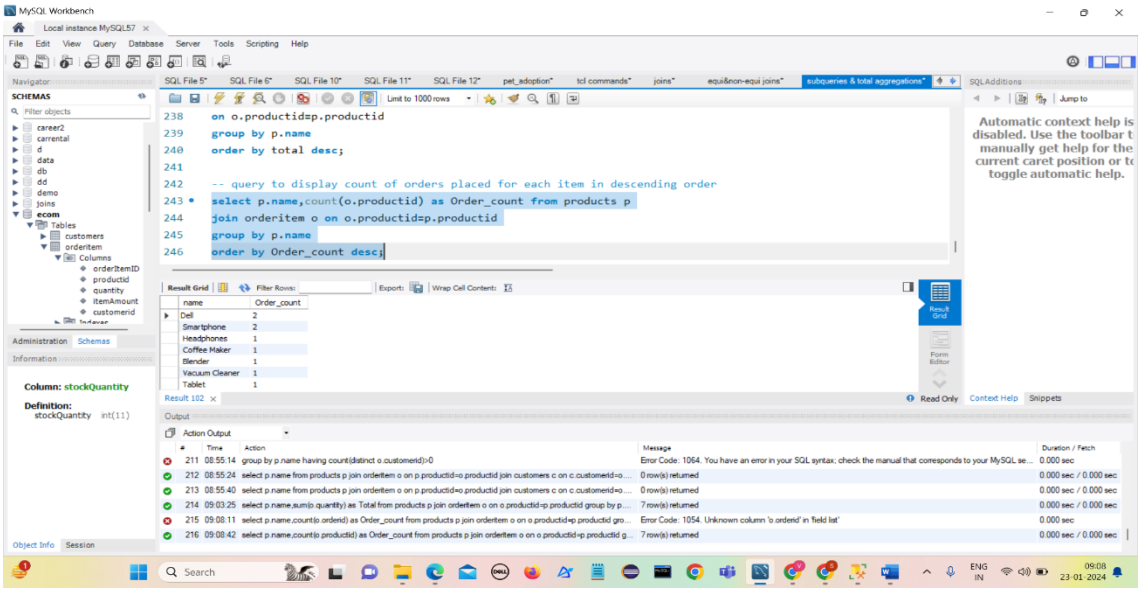
1) GROUP BY

GROUP BY enables you to use aggregate functions on groups of data returned from a query.

Query to get total quantity of each product ordered and order the result by total quantity in descending order.

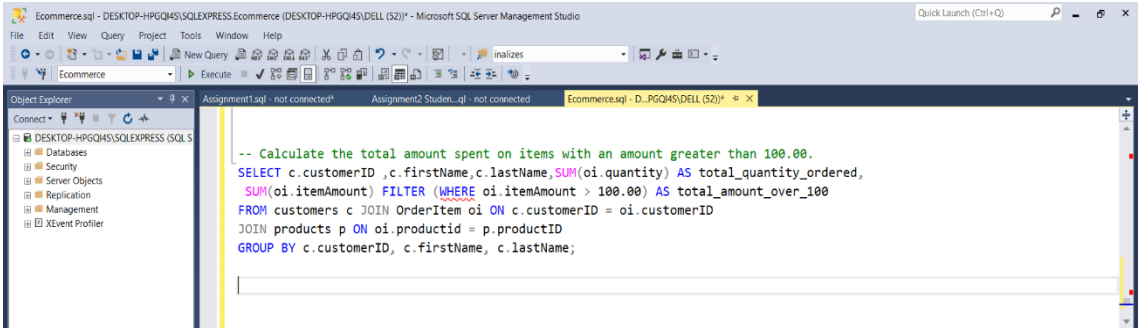


Query to display count of orders placed for each item in descending order



2) FILTER:

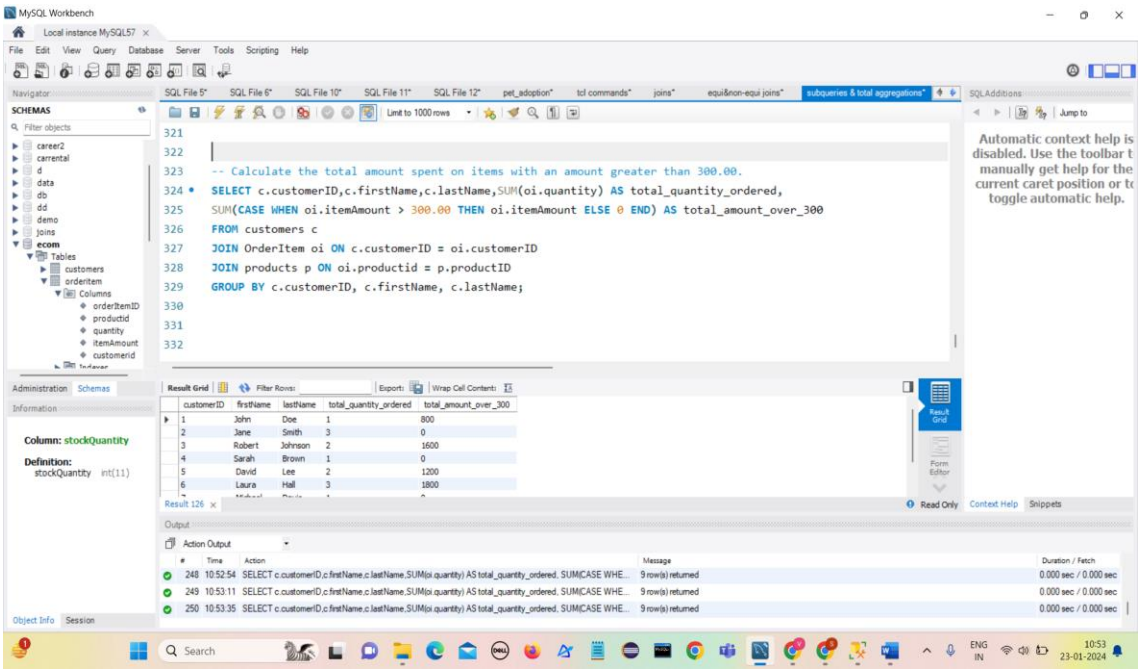
FILTER is a modifier used on an aggregate function to limit the values used in an aggregation.



It is better to use case instead of filter.

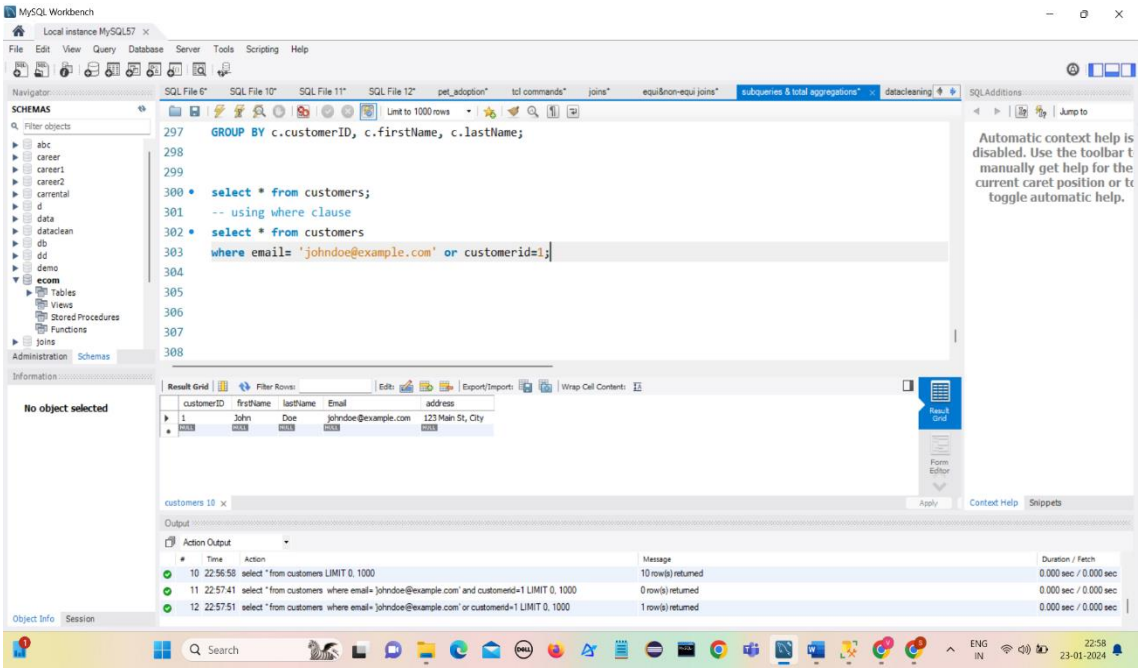
3) CASE statement

Calculate the total amount spent on items with an amount greater than 300.00.

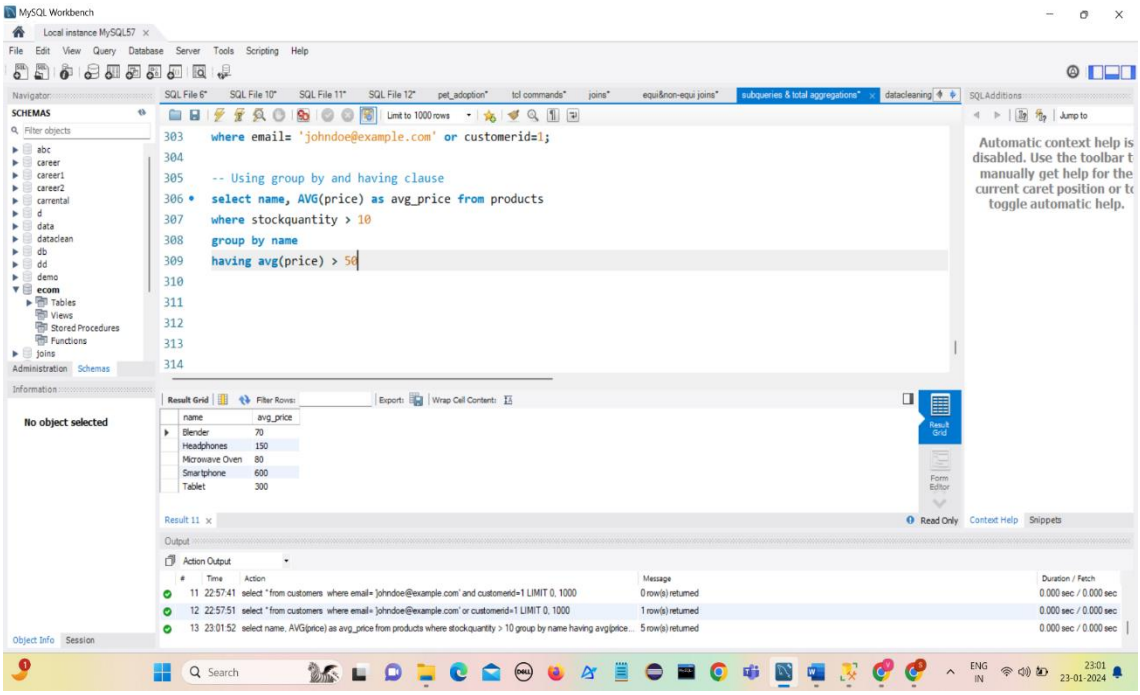


4) FILTERING DATA USING SQL QUERIES

1) Using where clause

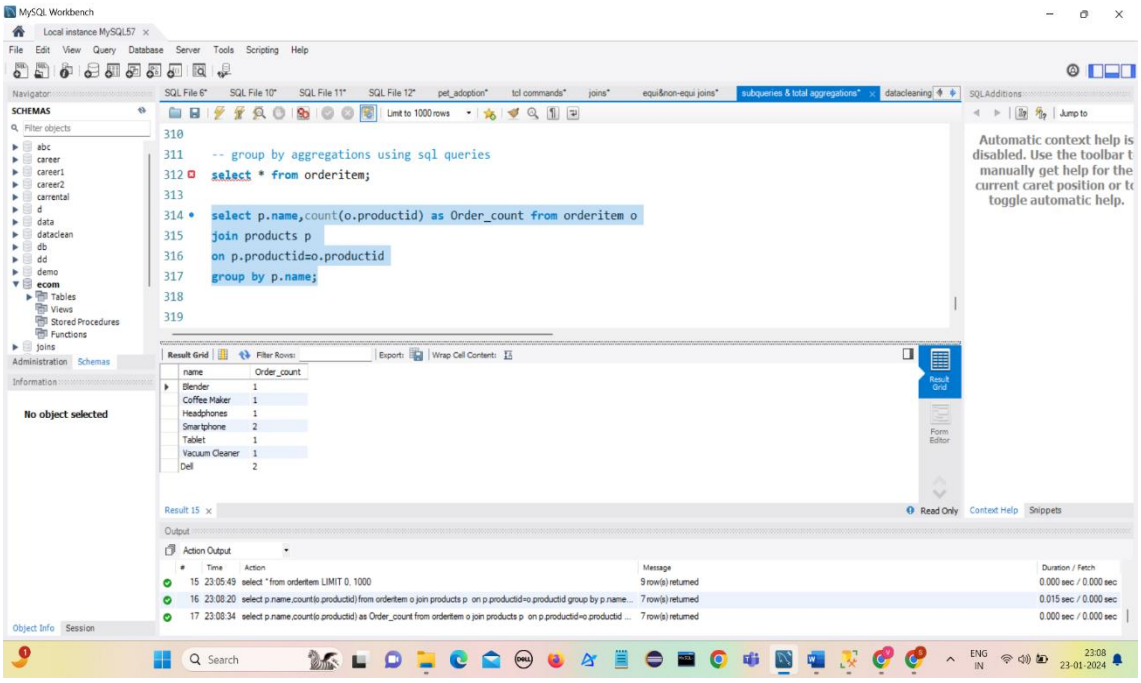


2) Using group by and having clause



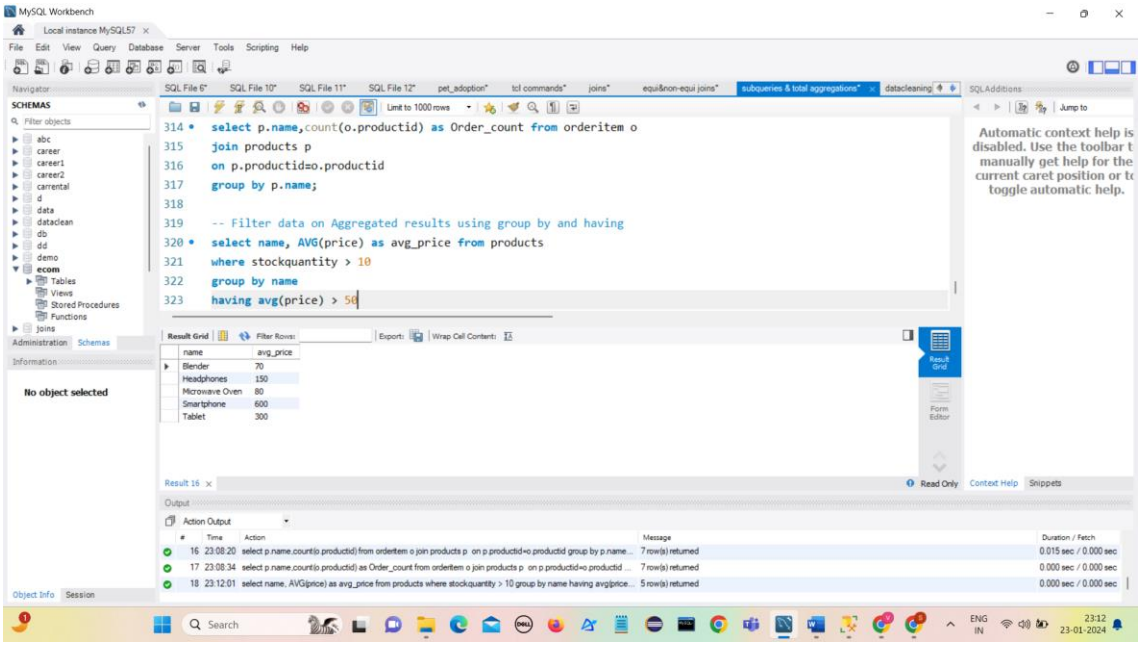
5) Group by aggregations using sql

Count of orders of each product

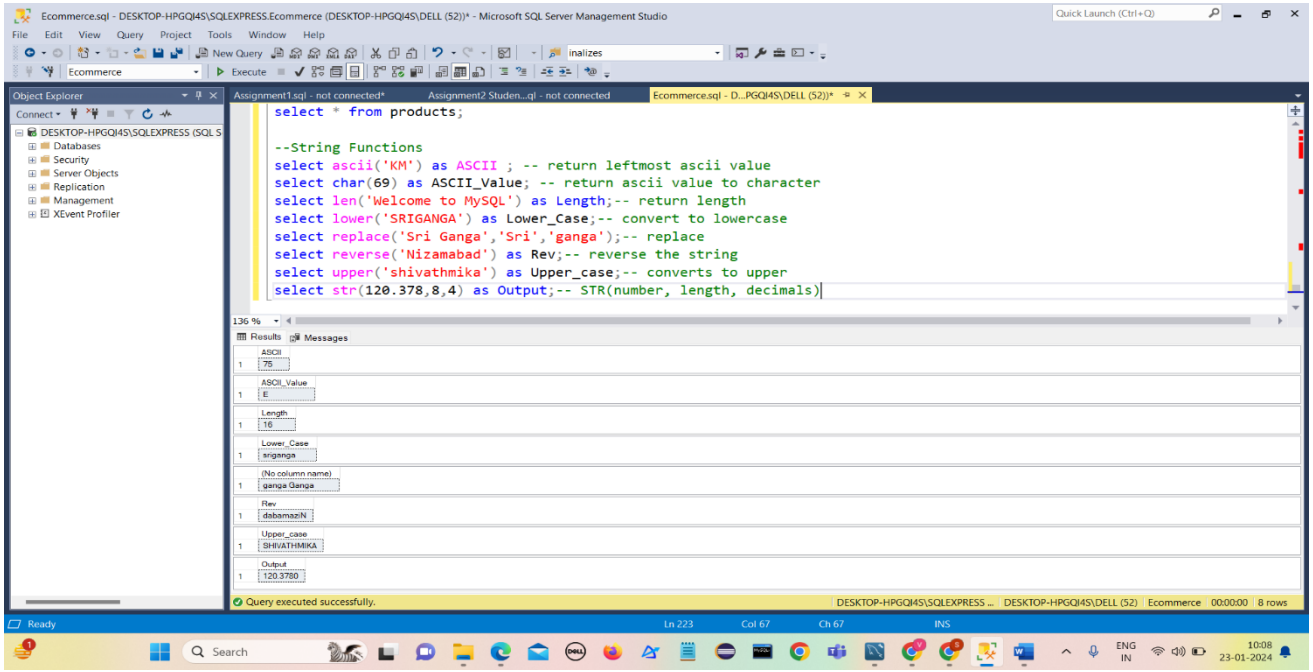


6) Filter data on aggregated results using group by and having

Product names having stock quantity more than 10 and average price greater than 50.



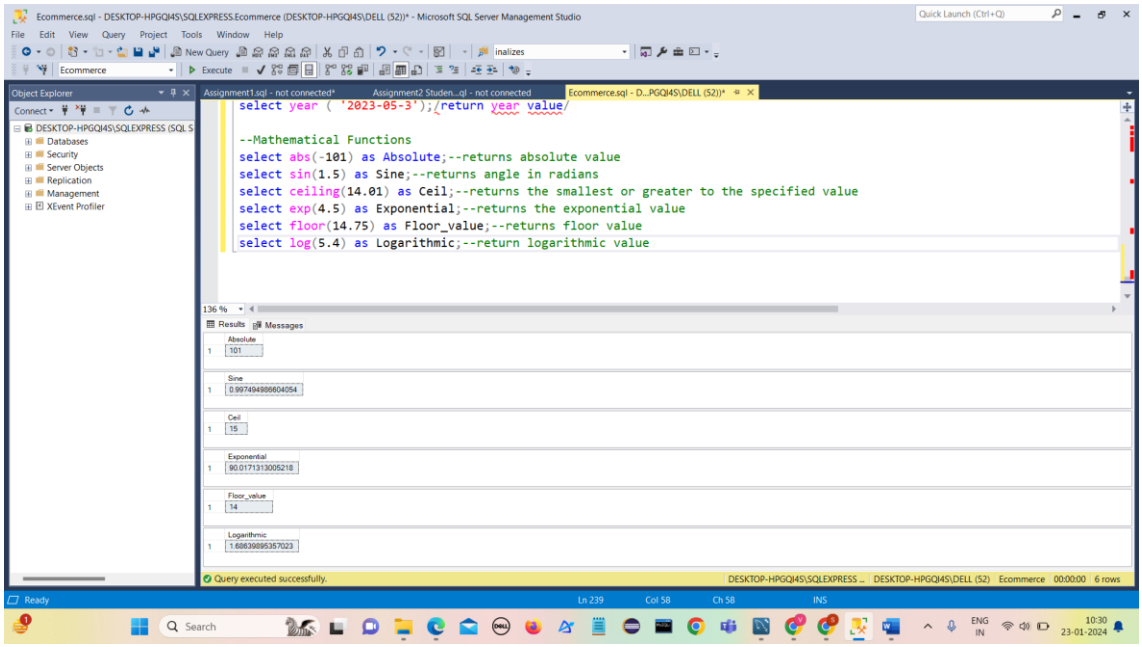
String functions:



1) **Ascii-** Returns the ASCII value for the specific character
2) **char-** Returns the character based on the ASCII code.
3) **len-** Returns the length of a string
4) **lower-** Converts a string to lower-case
5) **replace-** The REPLACE() function replaces all occurrences of a substring within a string, with a new substring.
   REPLACE(*string*, *old_string*, *new_string*)
6) **reverse-** Reverses a string and returns the result
7) **upper-** Converts a string to upper-case
8) **str-** Returns a number as string

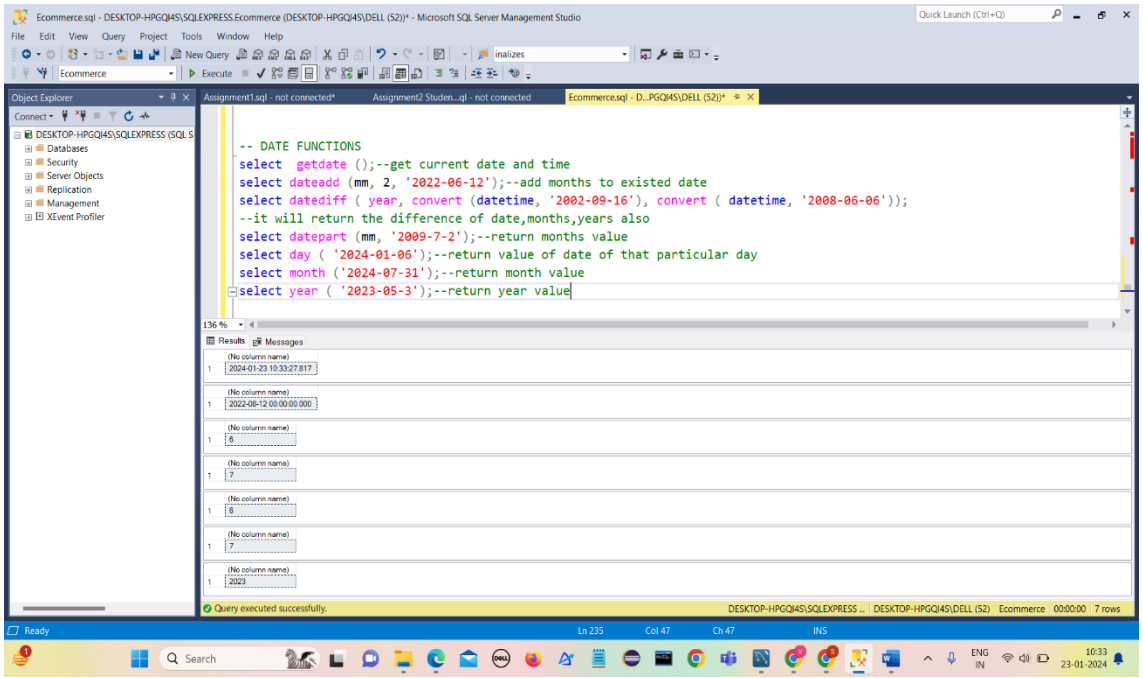
Mathematical Functions:

1) **abs-** returns absolute value
2) **sin-** returns angle in radians
3) **ceiling-** returns the smallest or greater to the specified value
4) **exp-** returns the exponential value
5) **floor-** returns floor value
6) **log-** return logarithmic value



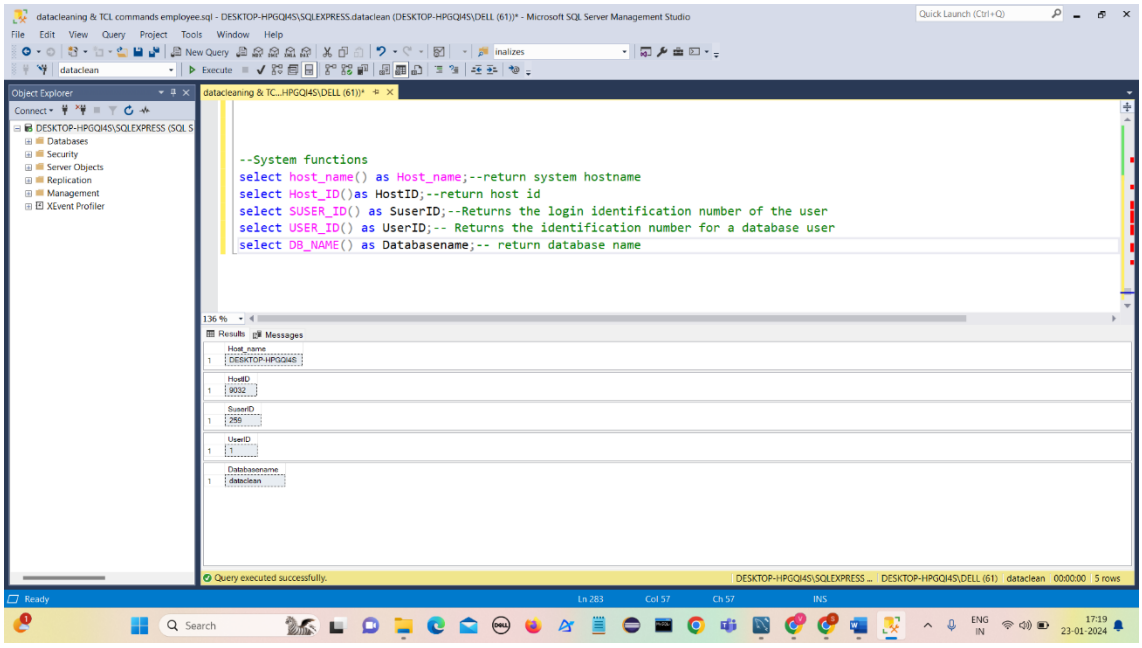
Date Functions:

1) **getdate-** get current date and time
2) **dateadd-** add months to existed date
3) **datediff-** it will return the difference of date, months, years also DATEDIFF(*interval*, *date1*, *date2*)
4) **datepart-** return months value.
5) **day-** return value of date of that particular day
6) **month-** return month value
7) **year-** return year value



System Functions

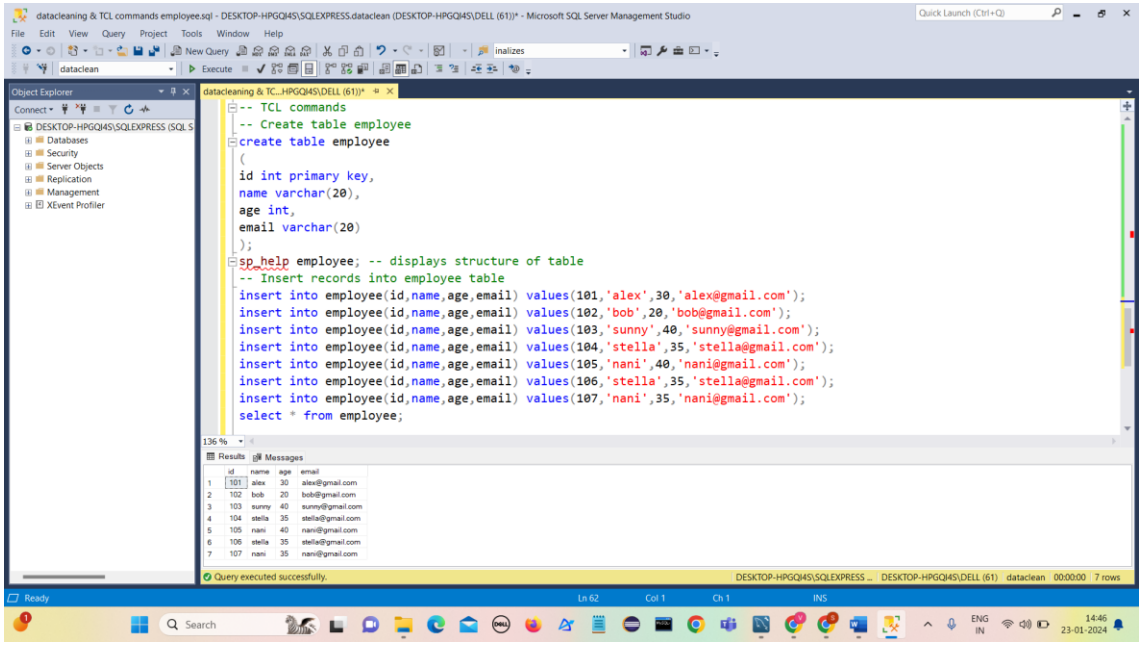
1) **host_name-**return system hostname
2) **Host_ID-**return host id
3) **SUSER_ID-**Returns the login identification number of the user
4) **USER_ID-**Returns the identification number for a database user
5) **DB_NAME** -Returns database name



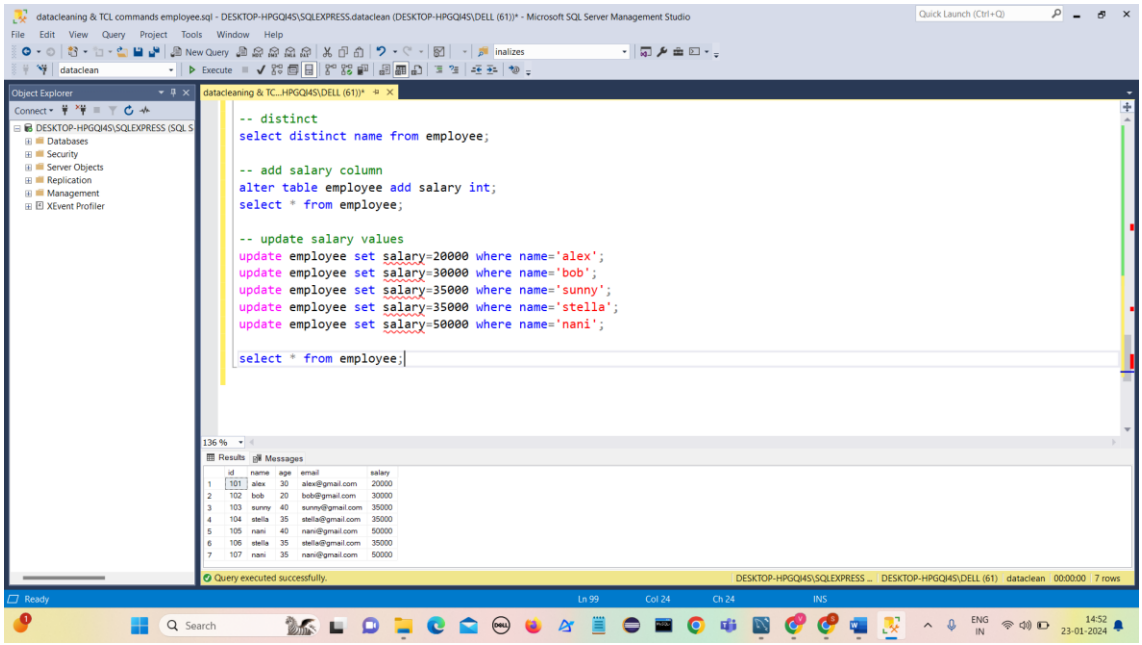
TCL Commands

Create table employee and insert values into it.

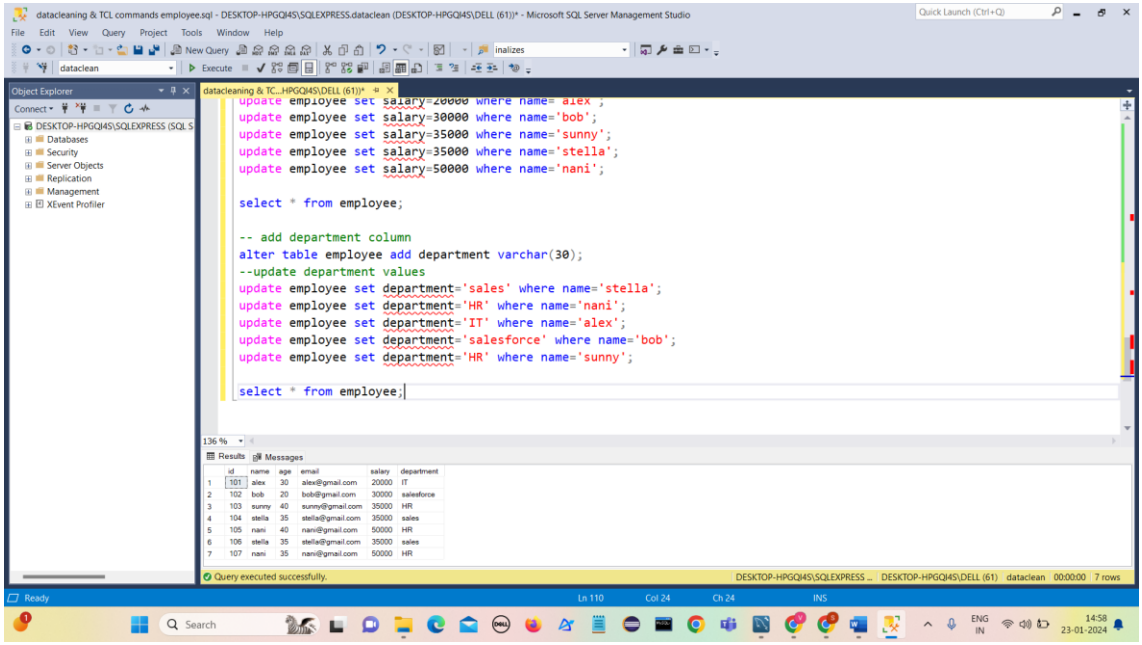
Retrieve the records of employee table.



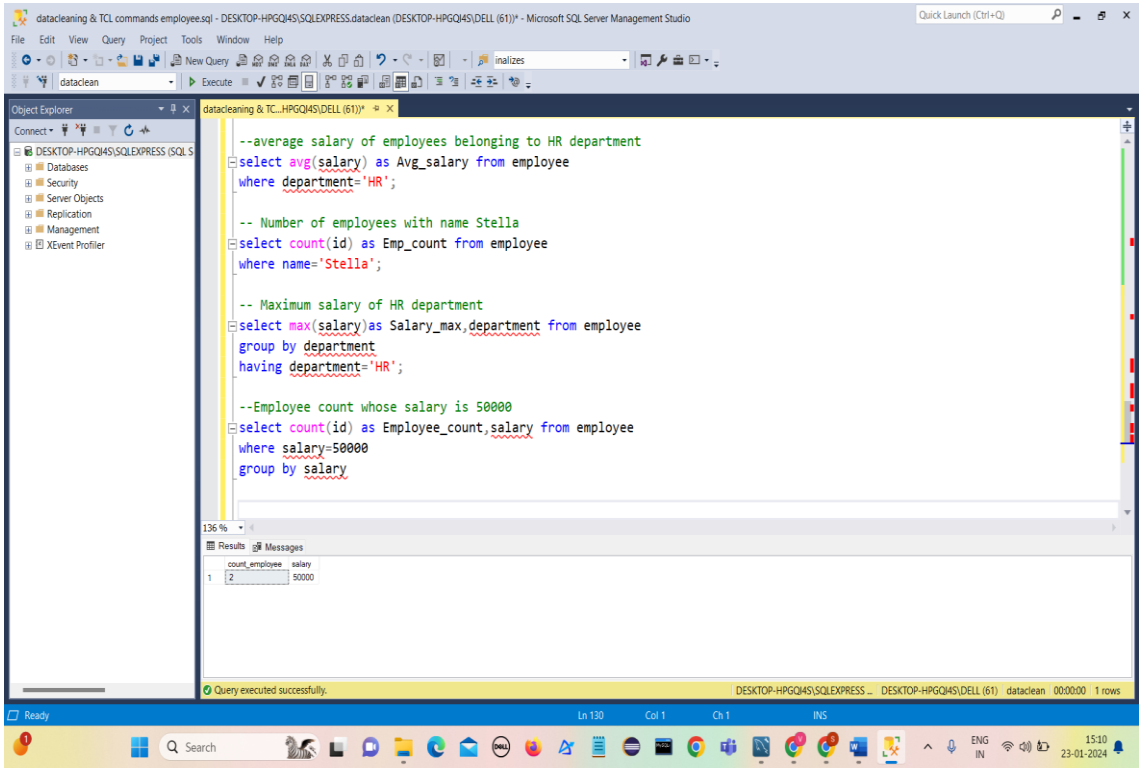
Add column salary and update the values of salary.



Add another column department and update values.

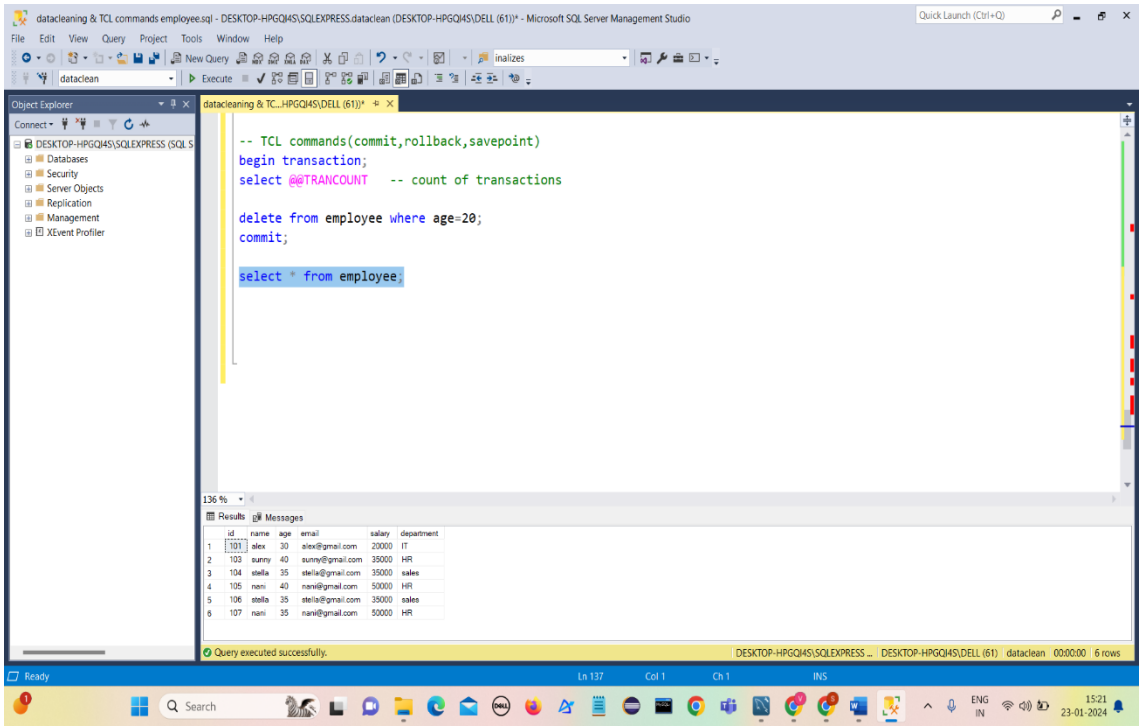


Some operations using aggregations, group by and having clause

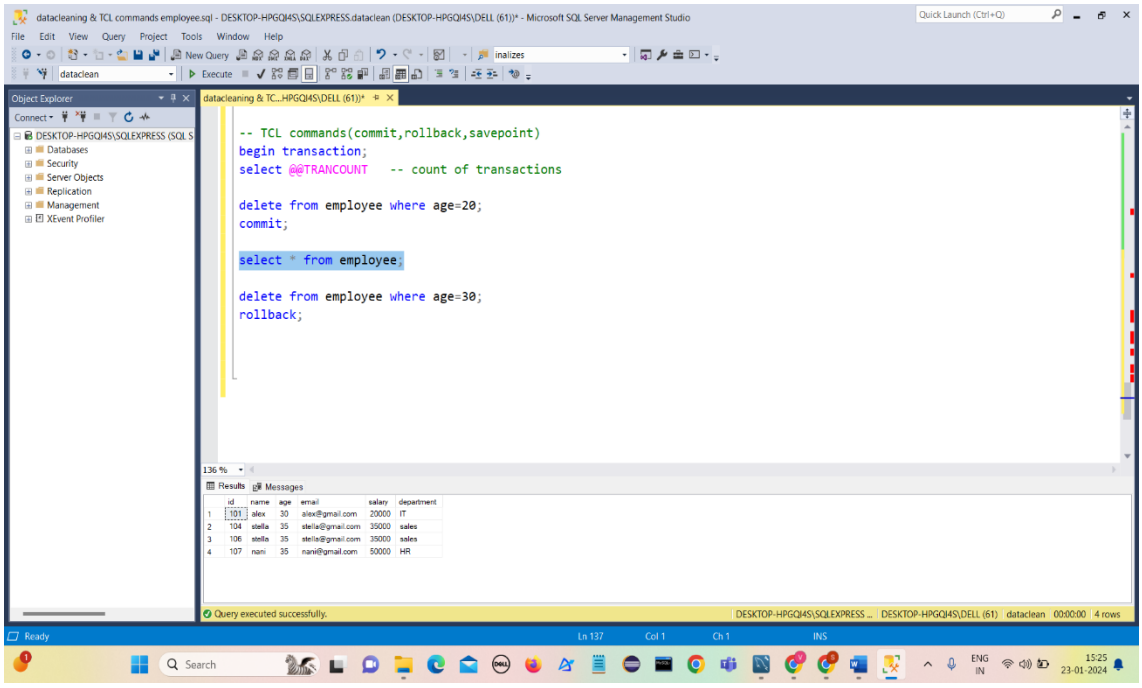


TCL commands

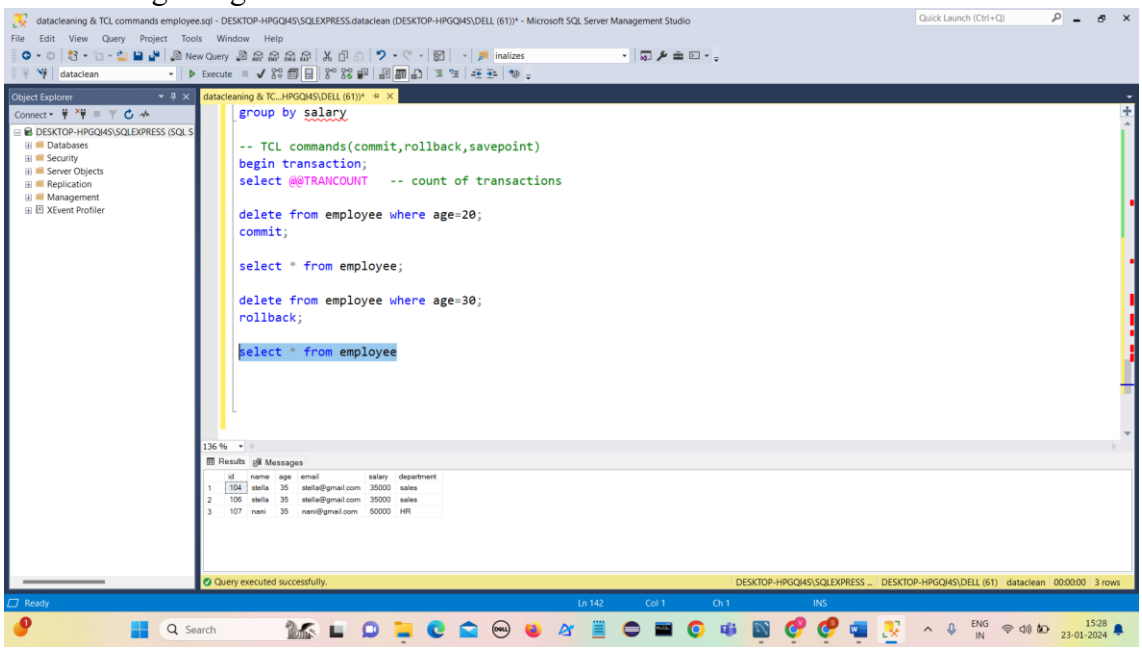
1) Commit



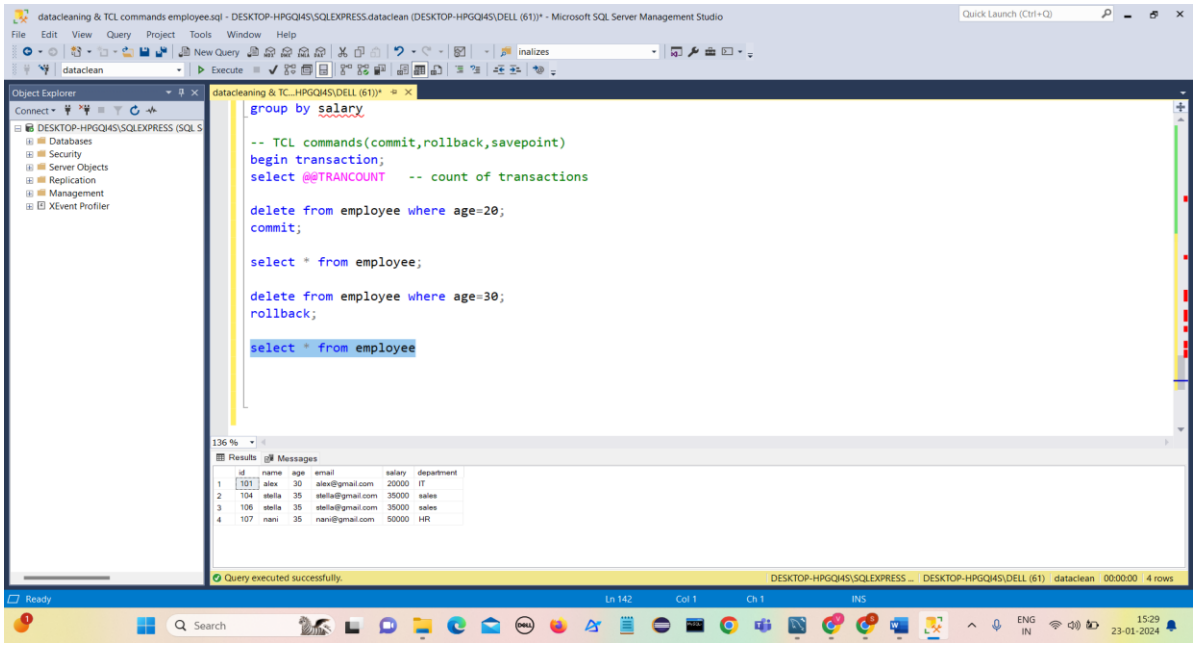
2) Before using rollback after deleting data



Row with age 30 got deleted



Use rollback



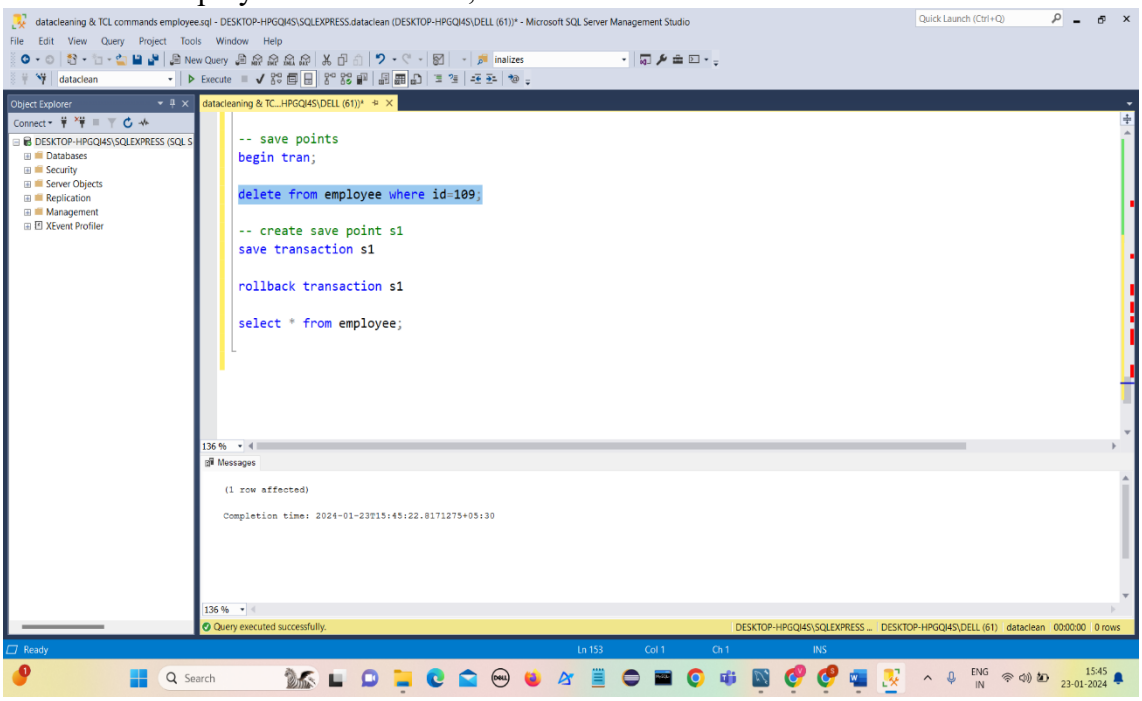
Steps to be followed:

1) Run begin transcation
2) Run the query
3) Verify using select
4) Now run either commit or rollback
5) If commited values can't be modified
6) Using rollback we can get back the values.

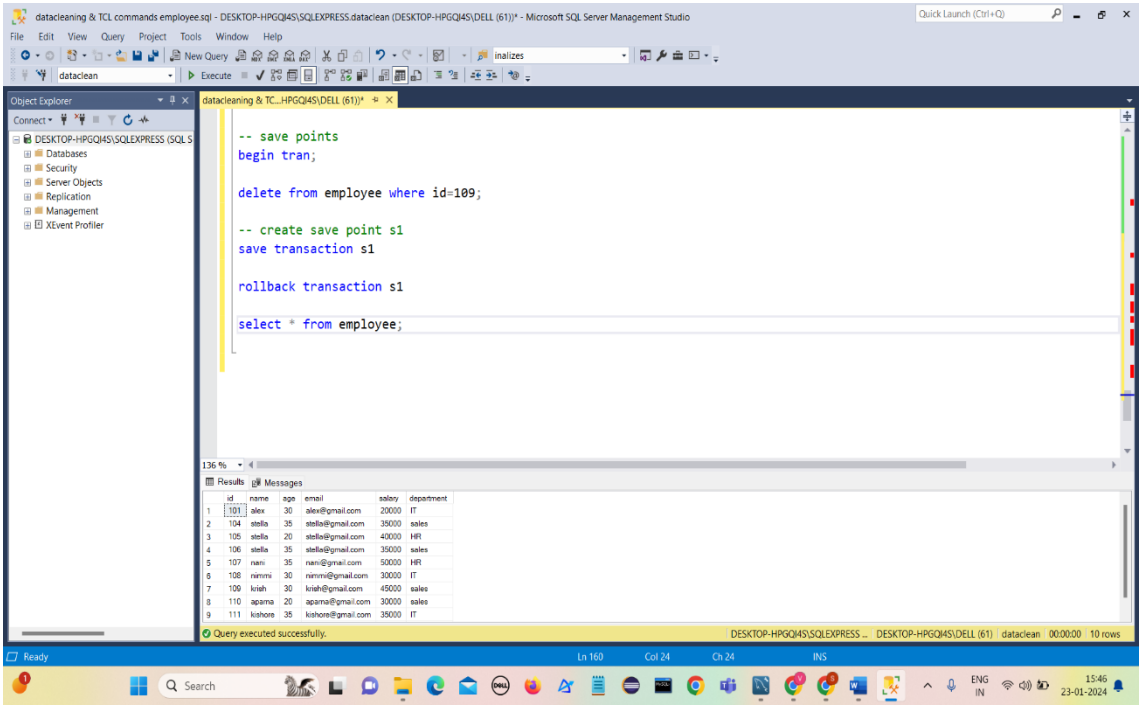
Using savepoints:

begin tran;

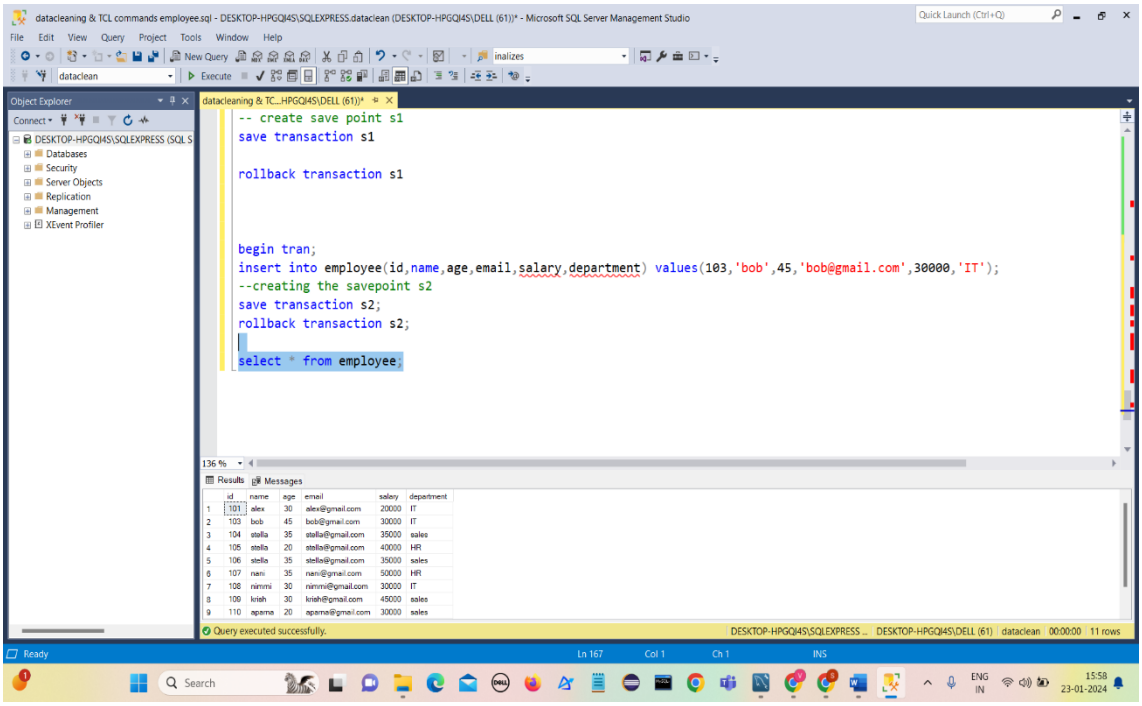
delete from employee where id=112;



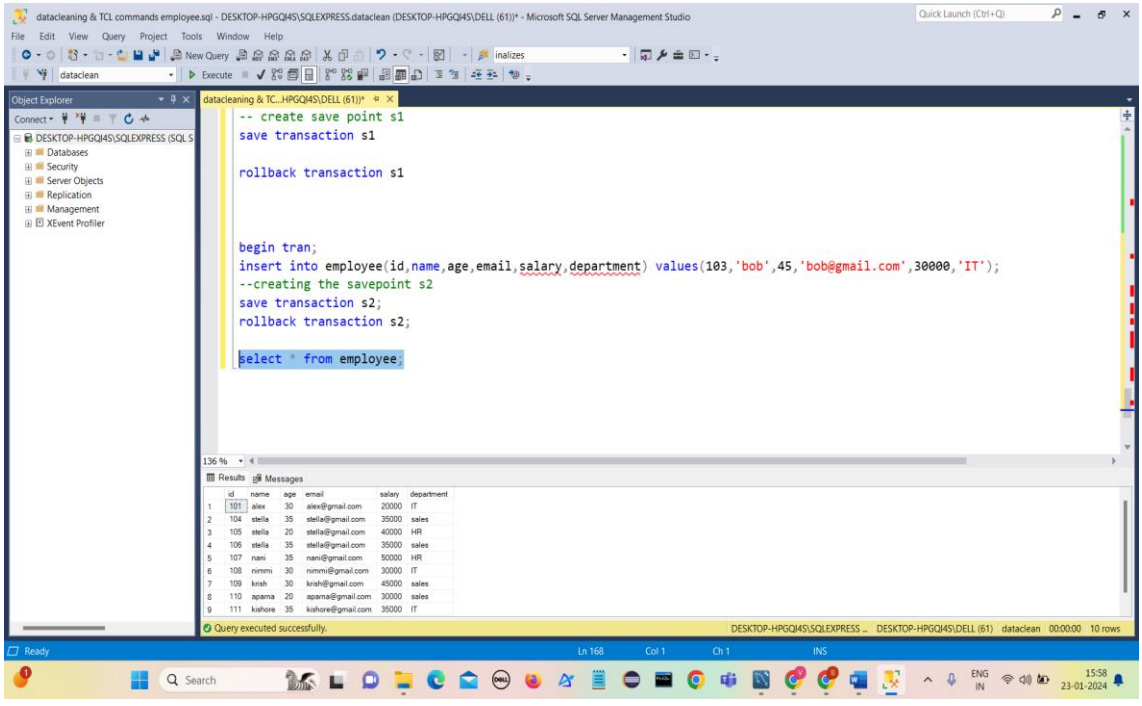
Rollback transcation s1



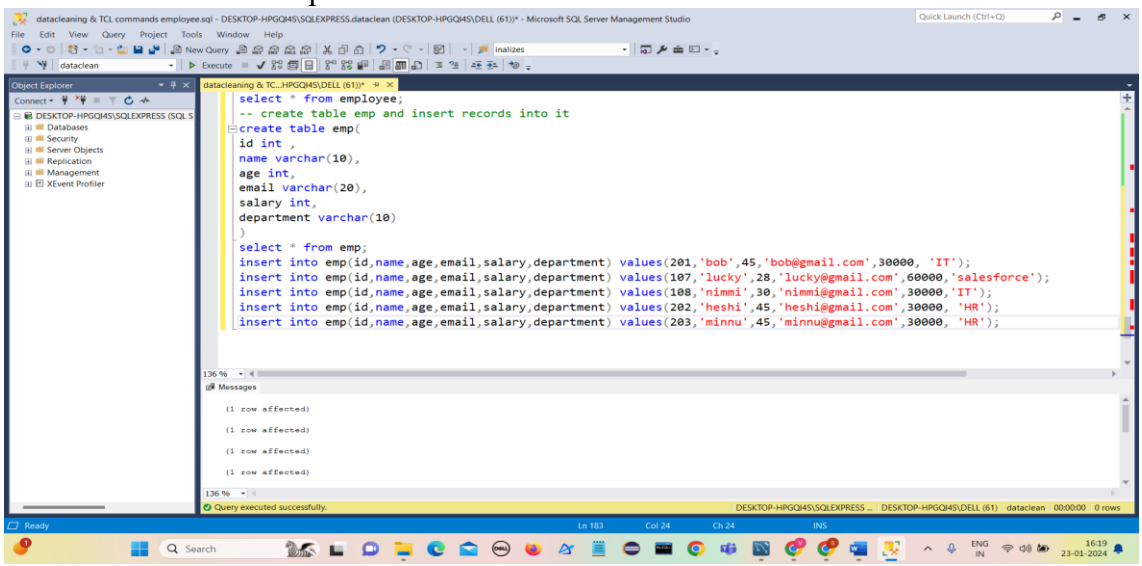
Insert command



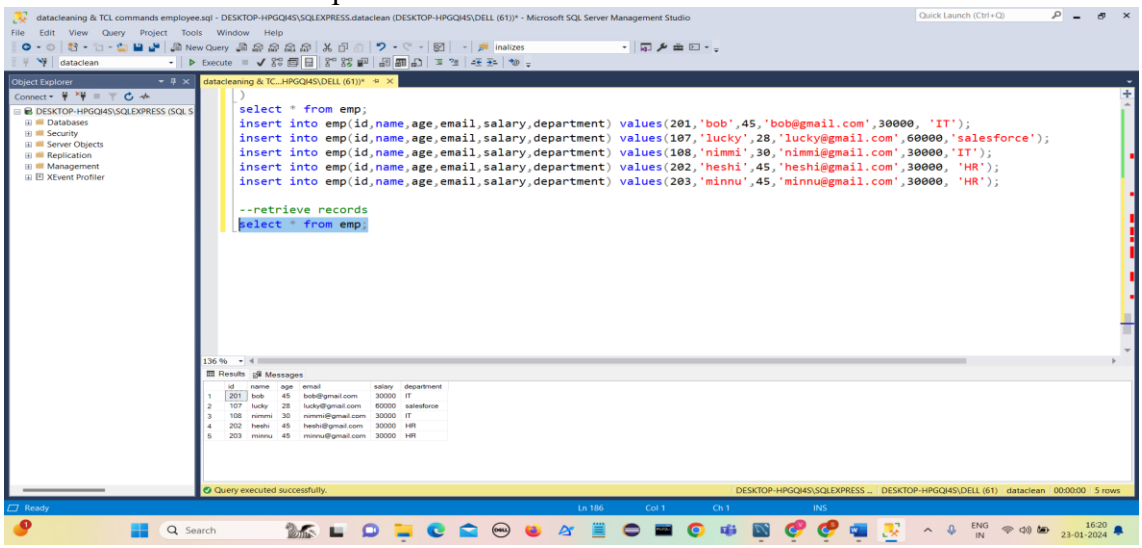
After rollback



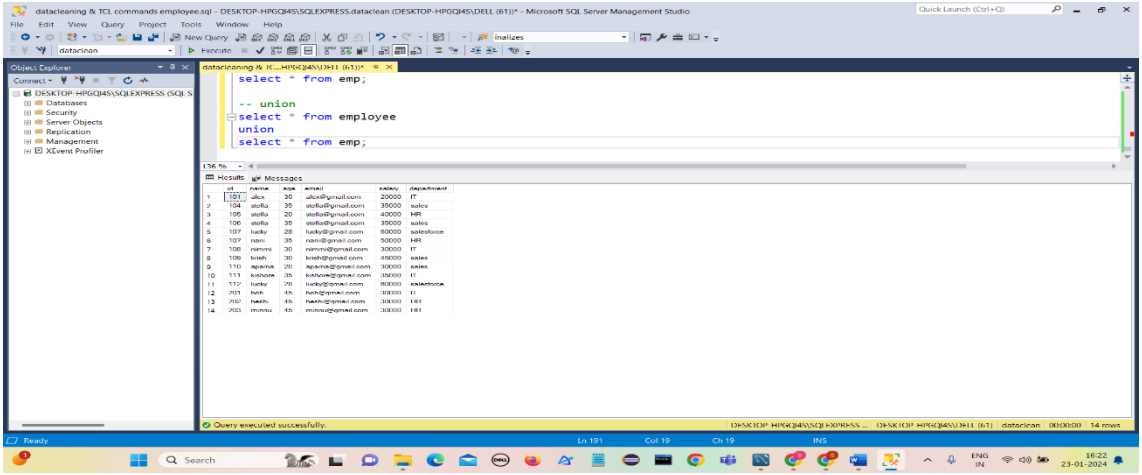
Create another table emp and add records into it



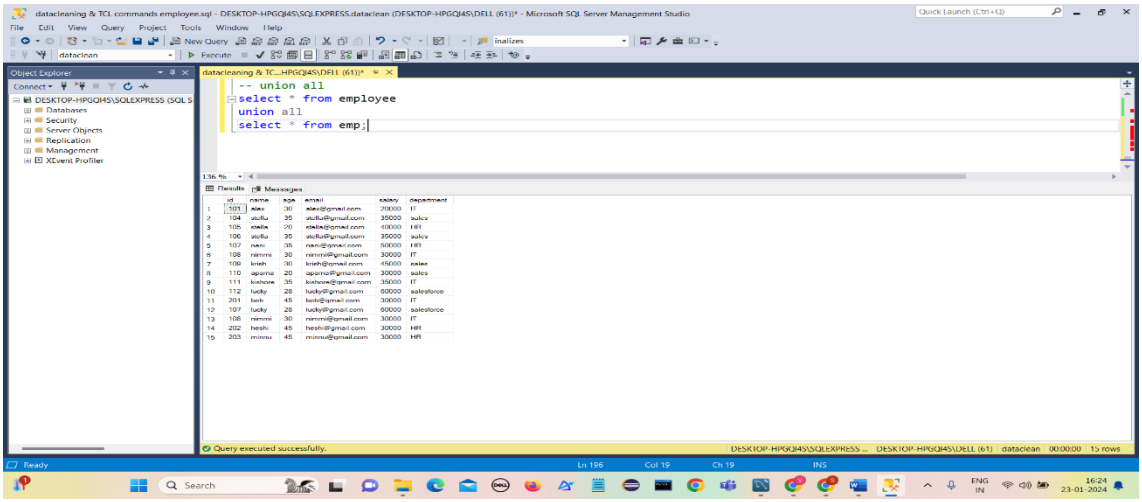
Retrieve records from emp table



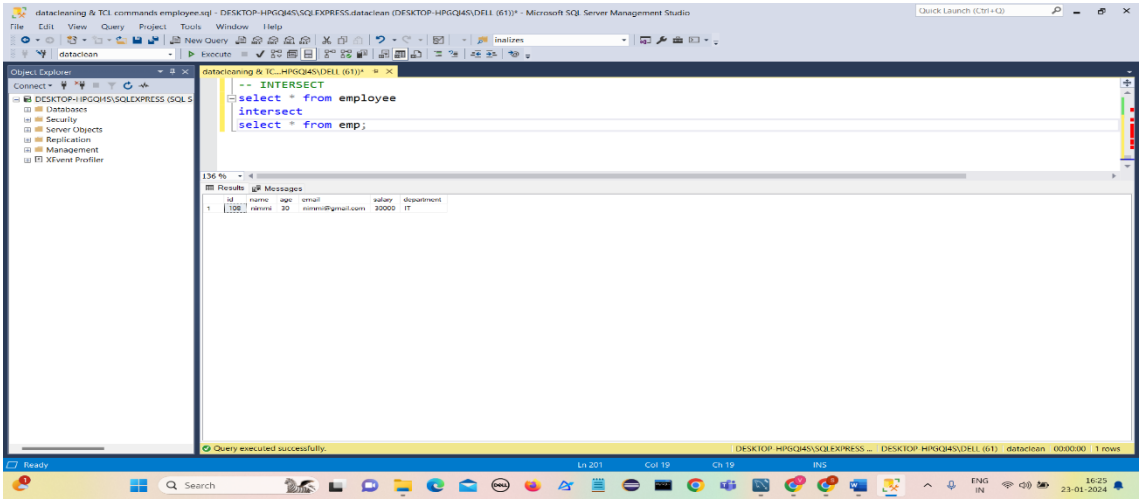
1) UNION



2) UNION ALL



3) INTERSECT



4) EXCEPT