Shubham P. Kumbhar

Wipro Assignments

SQL Language

Assignment 1 To 7

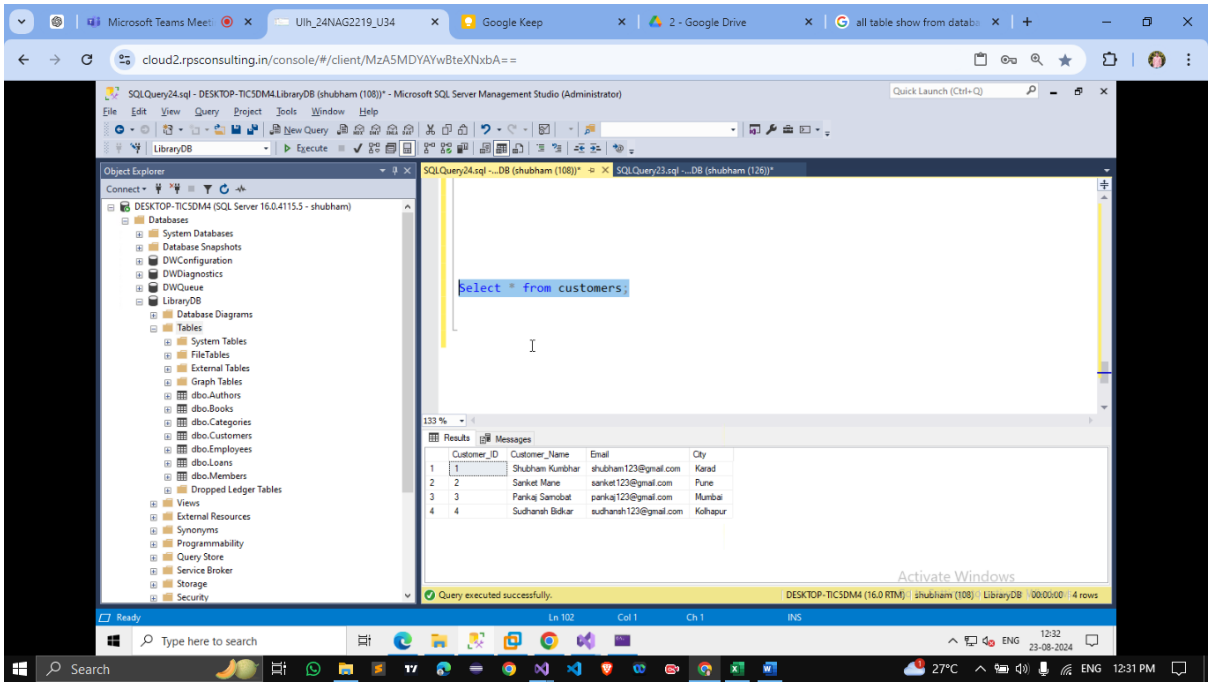
Assignment 1:

Write a SELECT query to retrieve all columns from a 'customers' table, and modify it to return only the customer name and email address for customers in a specific city.

Answer:

1. Retrieve All Columns from the customers Table

SELECT * FROM customer;

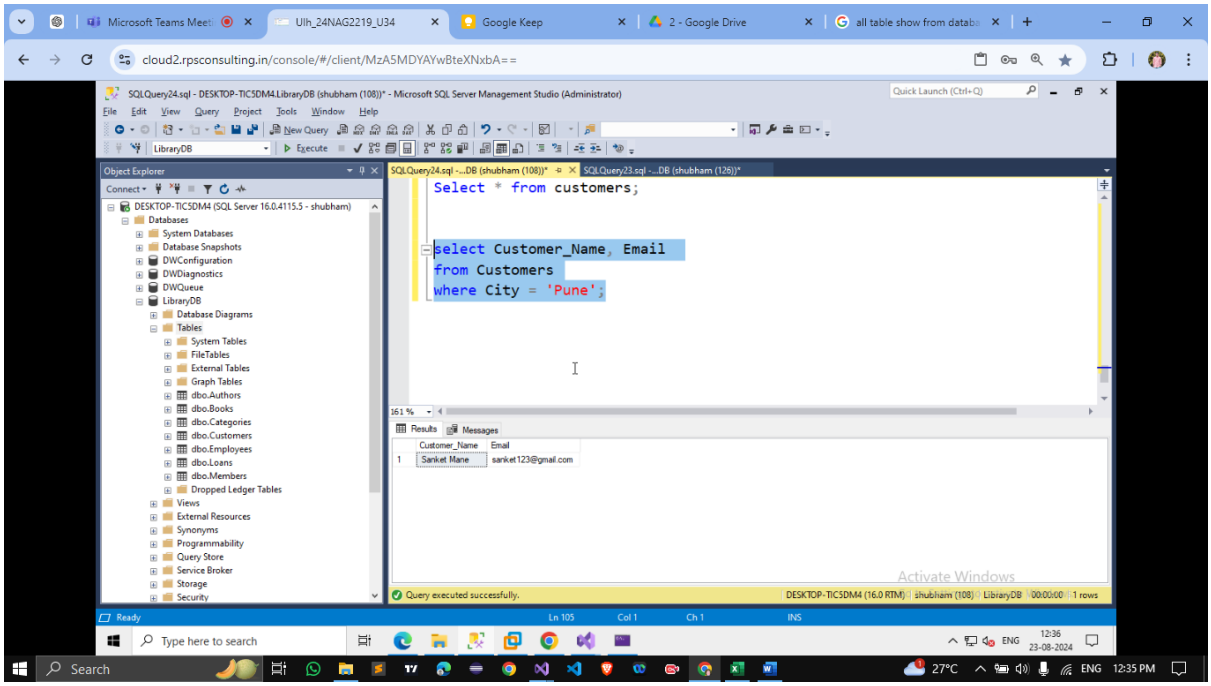


2. Modify the Query to Return Only the Customer Name and Email Address for Customers in a Specific City

SELECT Customer_Name , Email

FROM Customers

WHERE City = ‘Pune';



Assignment 2:

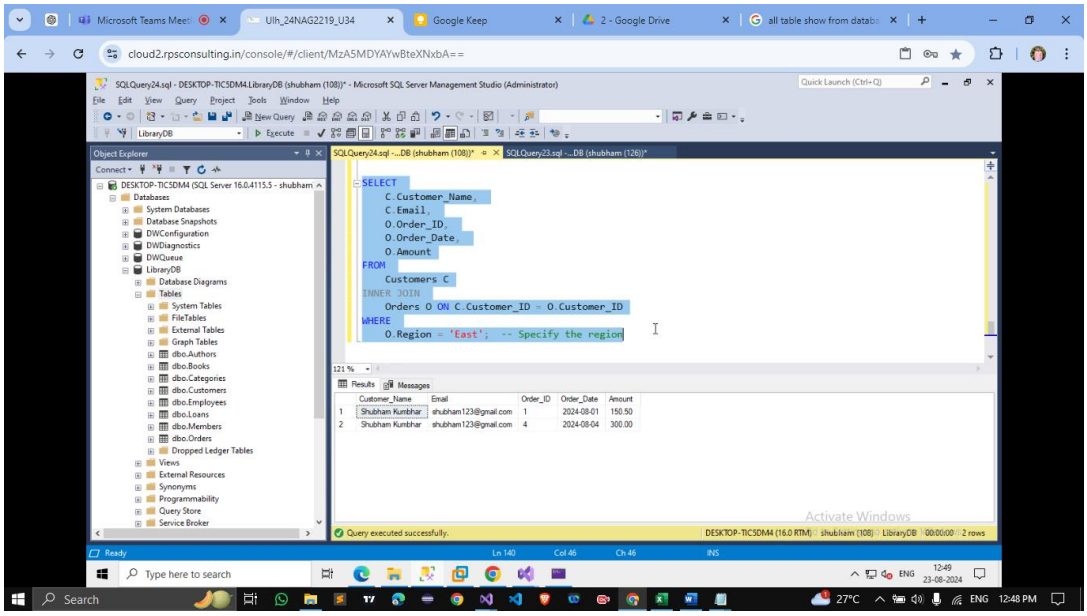
Craft a query using an INNER JOIN to combine 'orders' and 'customers' tables for customers in a specified region, and a LEFT JOIN to display all customers including those without orders.

Answers :

Craft Queries Using JOINs

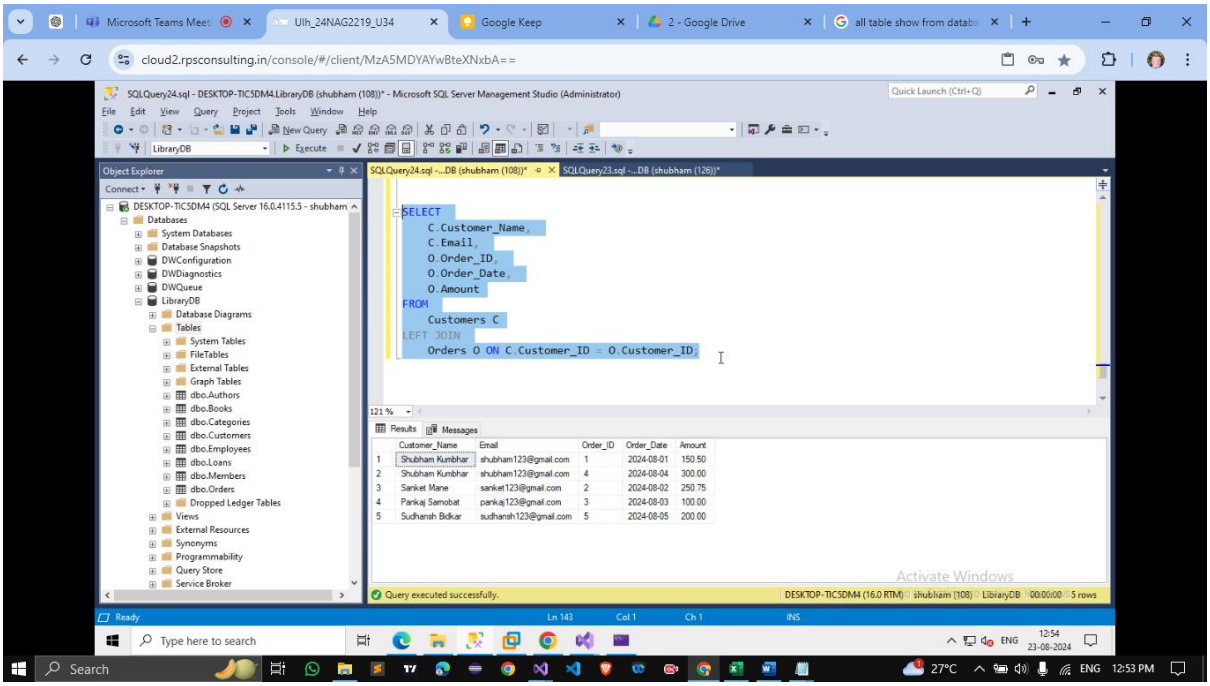
a. **INNER JOIN to combine orders and customers for customers in a specified region**

```
SELECT
    C.Customer_Name,
    C.Email,
    O.Order_ID,
    O.Order_Date,
    O.Amount
FROM
    Customers C
INNER JOIN
    Orders O  ON C.Customer_ID = O.Customer_ID
WHERE
    O.Region = 'East';  -- Specify the region
```



b. LEFT JOIN to display all customers including those without orders

```
SELECT
   C.Customer_Name,
   C.Email,
   O.Order_ID,
   O.Order_Date,
   O.Amount
FROM
   Customers C
LEFT JOIN
   Orders O ON C.Customer_ID = O.Customer_ID;
```



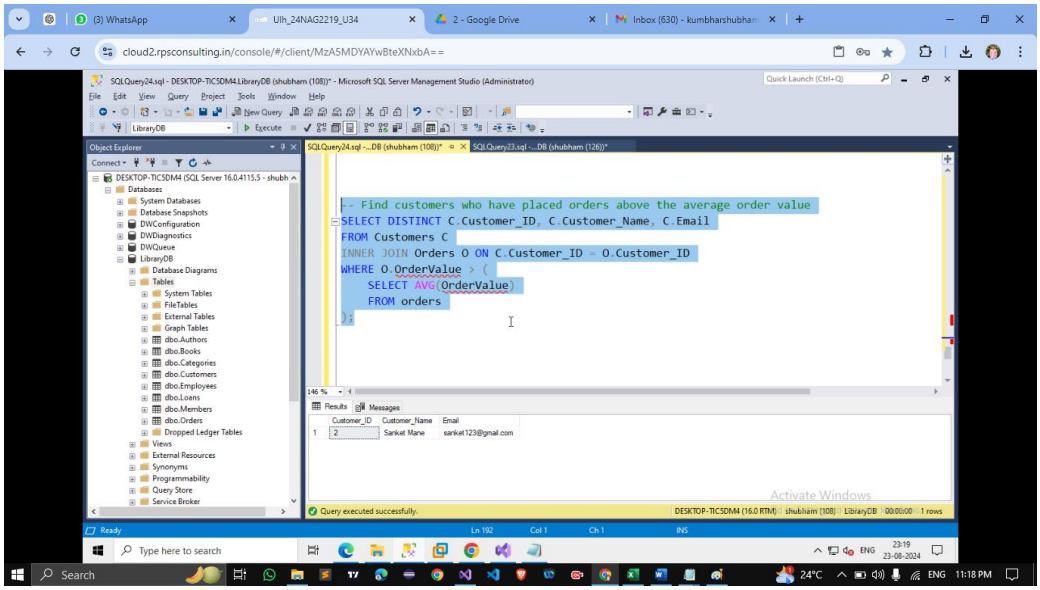
Assignment 3:

Utilize a subquery to find customers who have placed orders above the average order value, and write a UNION query to combine two SELECT statements with the same number of columns.

Answer : →

1. Subquery to Find Customers Who Have Placed Orders Above the Average Order Value

```
-- Find customers who have placed orders above the average order value
SELECT DISTINCT C.Customer_ID, C.Customer_Name
FROM Customers C
JOIN Orders O ON C.CustomerID = O.CustomerID
WHERE O.OrderValue > (SELECT AVG(OrderValue) FROM Orders);
```

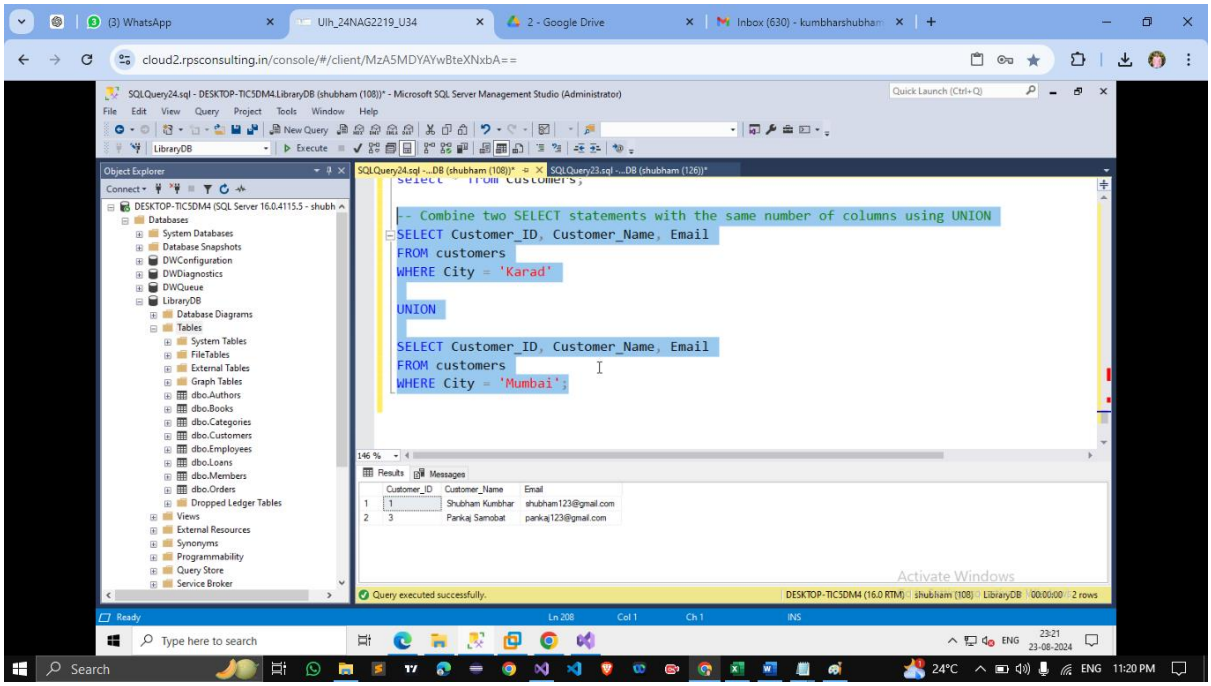


2. UNION Query to Combine Two SELECT Statements with the Same Number of Columns

```
-- Combine results from two different queries with the same number of columns
SELECT Customer_ID, Customer_Name, Email
FROM Customers
WHERE City = 'Karad'

UNION

SELECT Customer_ID, Customer_Name, Email
FROM Customers
WHERE City = 'Mumbai';
```



Assignment 4:

Compose SQL statements to BEGIN a transaction, INSERT a new record into the 'orders' table, COMMIT the transaction, then UPDATE the 'products' table, and ROLLBACK the transaction.

Answer : →

```
-- Begin the transaction

BEGIN TRANSACTION;


-- Insert a new record into the 'orders' table

INSERT INTO orders (customer_id, order_date, order_value)

VALUES (1, '2024-08-24', 250.00);


-- Commit the transaction to save the new order

COMMIT;


-- Begin a new transaction to update the 'products' table

BEGIN TRANSACTION;


-- Update the 'products' table
```

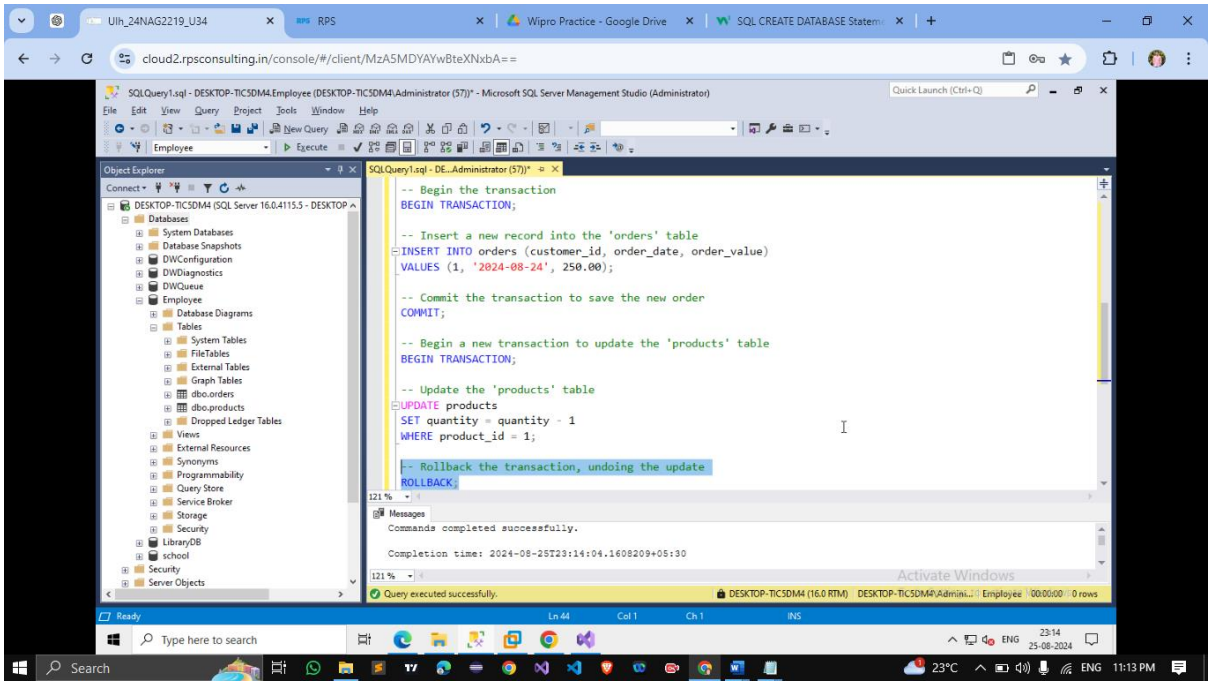
UPDATE products

SET quantity = quantity - 1

WHERE product_id = 1;

-- Rollback the transaction, undoing the update

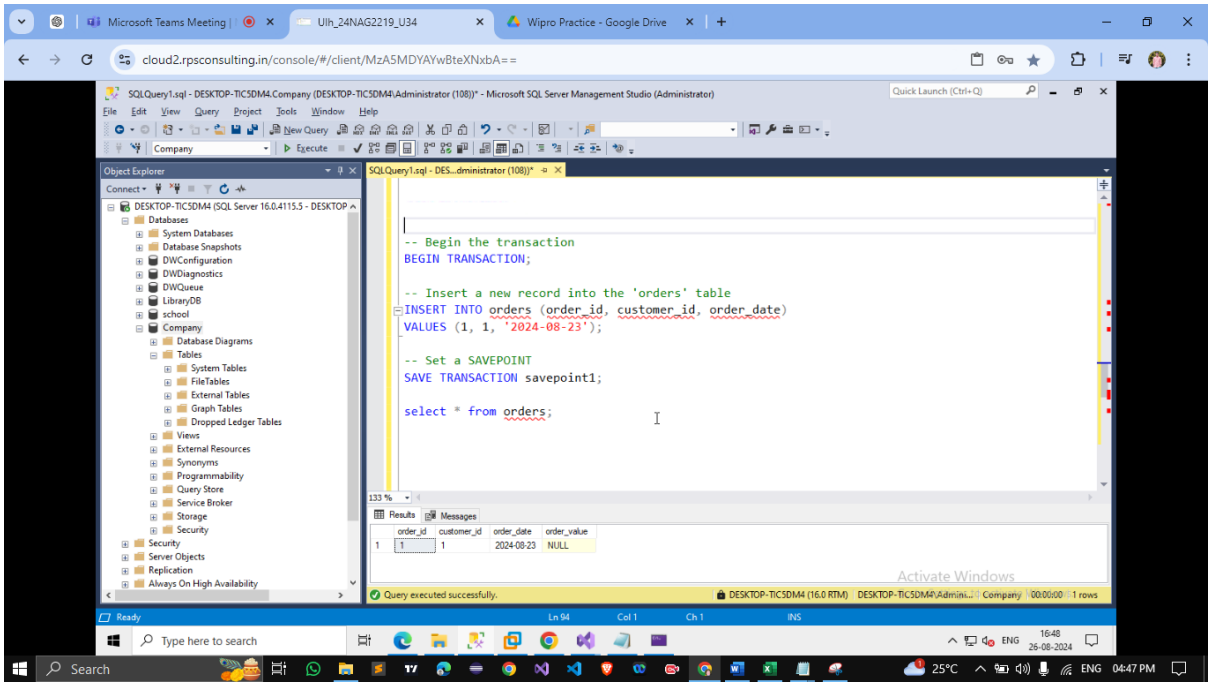
ROLLBACK;

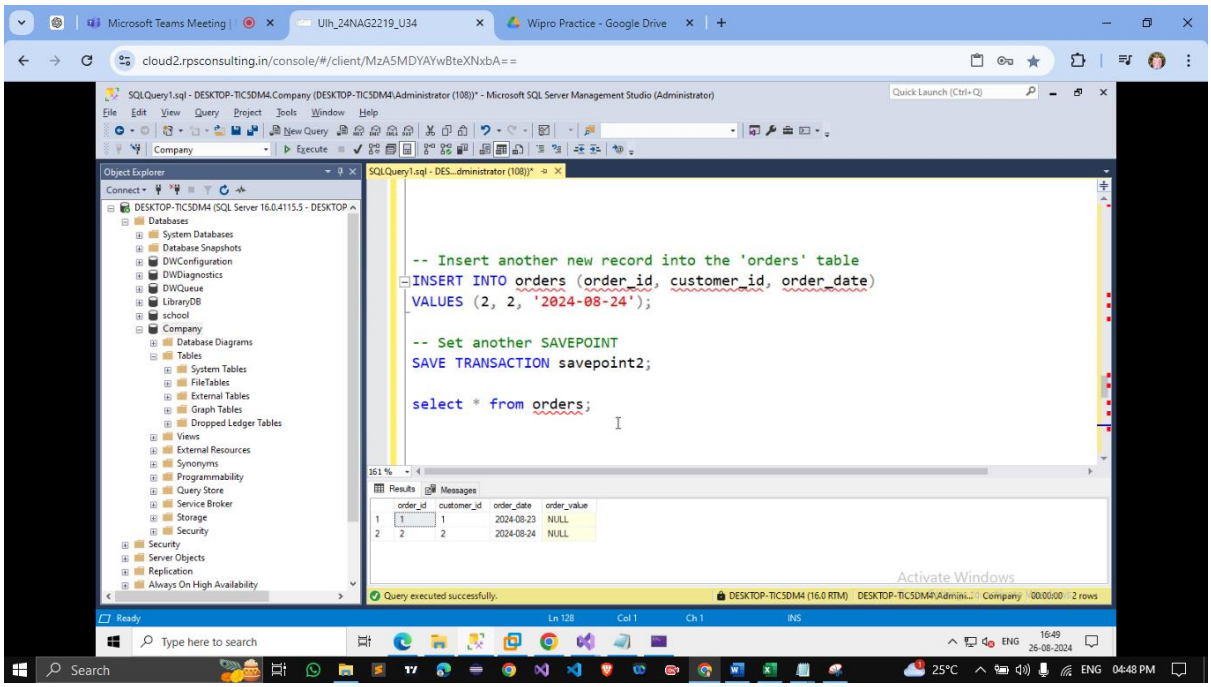


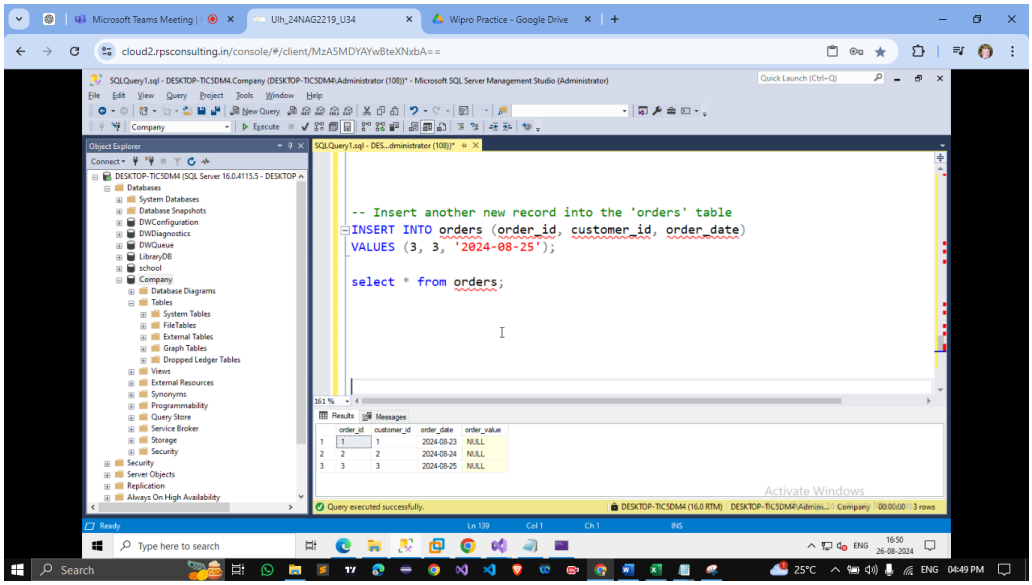
Assignment 5:

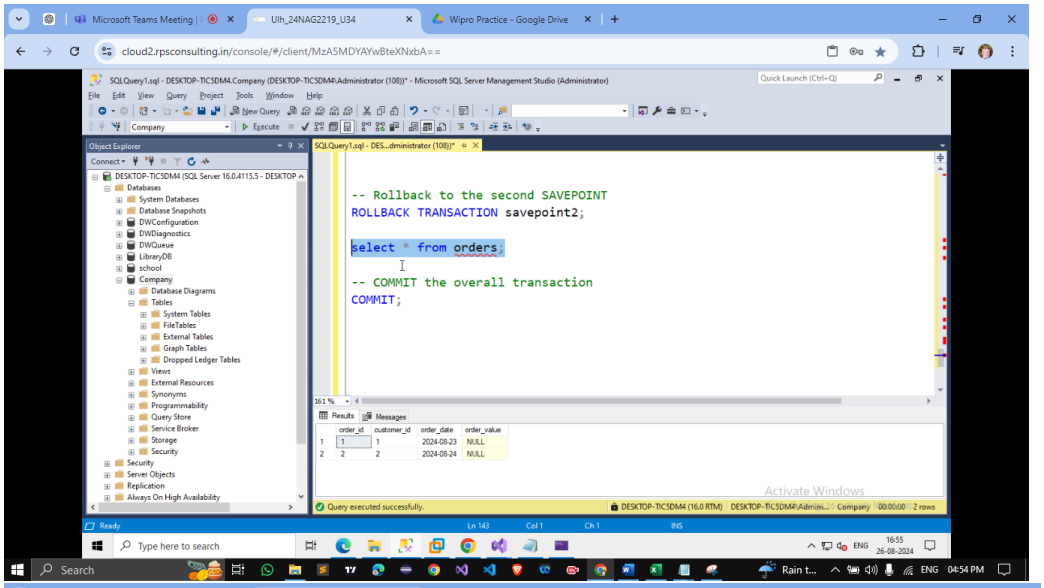
Begin a transaction, perform a series of INSERTs into 'orders', setting a SAVEPOINT after each, rollback to the second SAVEPOINT, and COMMIT the overall transaction.

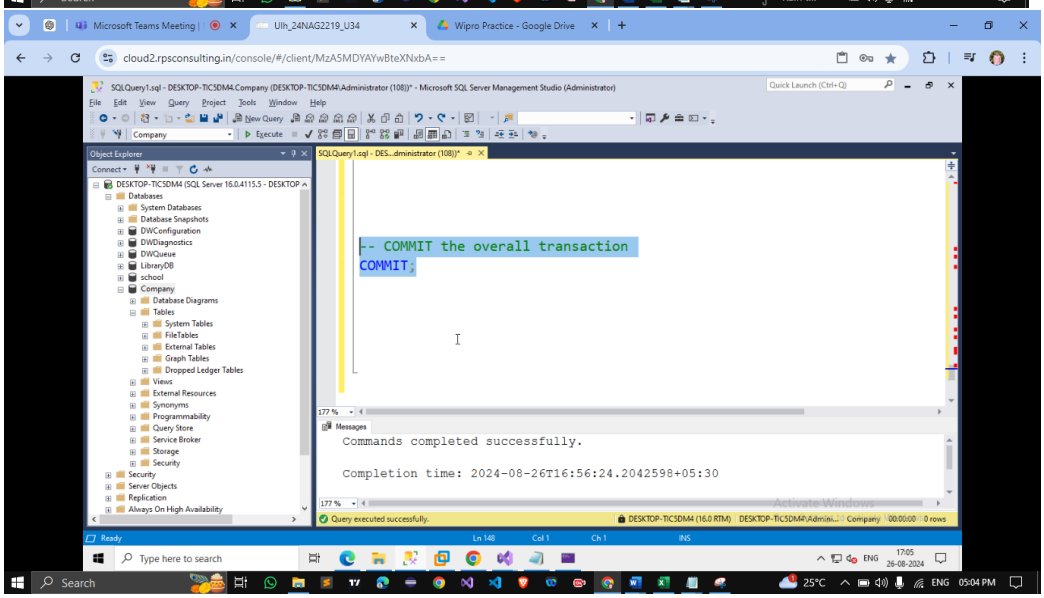
Answer : →

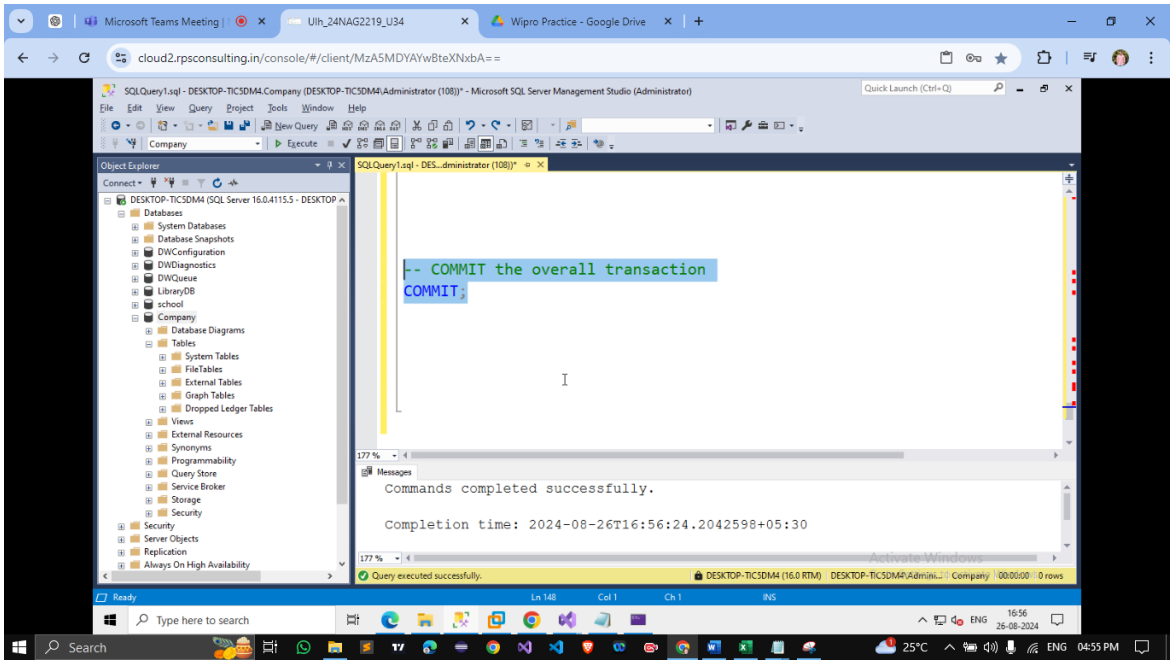





-- Insert another new record into the 'orders' table
INSERT INTO orders (order_id, customer_id, order_date)
VALUES (3, 3, '2024-08-25');
select * from orders;


-- Rollback to the second SAVEPOINT
ROLLBACK TRANSACTION savepoint2;
select * from orders;
-- COMMIT the overall transaction
COMMIT;


-- COMMIT the overall transaction
COMMIT;
Commands completed successfully.
Completion time: 2024-08-26T16:56:24.2042598+05:30



Assignment 6:

Draft a brief report on the use of transaction logs for data recovery and create a hypothetical scenario where a transaction log is instrumental in data recovery after an unexpected shutdown.

Answer : →

Report on Transaction Logs for Data Recovery

Introduction

Transaction logs are a critical component in modern database management systems, playing a crucial role in data recovery and ensuring data integrity. These logs record all changes made to the database, allowing for the restoration of data in the event of system failures or unexpected shutdowns. This report outlines the use of transaction logs for data recovery and provides a hypothetical scenario to illustrate their importance.

Use of Transaction Logs for Data Recovery

1. **Purpose and Functionality**

- **Transaction Tracking**: Transaction logs track every operation performed on the database, including inserts, updates, deletes, and schema changes. Each entry in the log provides a detailed record of these changes.

- **Recovery Operations**: In the event of a failure, transaction logs enable database systems to recover to a consistent state by reapplying committed transactions and undoing incomplete or uncommitted transactions.

2. **Types of Recovery**

- **Crash Recovery**: If a database crashes unexpectedly, transaction logs are used to redo transactions that were committed before the crash and undo transactions that were in progress.
- **Point-in-Time Recovery**: This recovery method uses transaction logs to restore the database to a specific point in time, useful for scenarios where data corruption or loss occurs after a certain date.

3. **Implementation**

- **Logging**: Every transaction is logged with a unique identifier and timestamps. This log is typically stored in a secure and separate location from the database to prevent data loss.
- **Backup Integration**: Transaction logs are often used in conjunction with database backups. Backups provide a base point, while logs capture incremental changes, allowing for a more precise recovery.

Hypothetical Scenario: Data Recovery After Unexpected Shutdown

Scenario:

A retail company, XYZ Corp, relies on a database to manage its inventory, customer orders, and sales transactions. The company's database server experiences an unexpected shutdown due to a power outage during peak business hours. The transaction log plays a vital role in recovering the database to a consistent state.

Steps Involved in Data Recovery:

1. **Initial Assessment**:
    - Upon restarting the database server, the database administrator discovers that the last transaction before the shutdown is incomplete. There is a need to determine the extent of data loss.
2. **Analyze Transaction Log**:
    - The administrator reviews the transaction log, which contains detailed entries for every operation performed up to the point of the shutdown. This includes all committed transactions and transactions that were in progress.
3. **Redo Committed Transactions**:
    - Transactions that were committed before the shutdown are reapplied to the database. For example, a customer order placed just before the outage is reprocessed, ensuring that the inventory reflects the new order.
4. **Undo Uncommitted Transactions**:

- Any transactions that were in progress at the time of the shutdown are rolled back. For instance, if an inventory adjustment was being made but not completed, it is undone to prevent partial updates.

5. **Point-in-Time Restoration**:
    - If necessary, the administrator can restore the database to a specific point in time using the transaction log. This is useful if data corruption occurred just before the shutdown.
6. **Verification and Testing**:
    - After recovery, the database is thoroughly tested to ensure that all data is accurate and that the system is functioning correctly. Any discrepancies are addressed based on the transaction log records.

Outcome:

Thanks to the transaction log, XYZ Corp successfully recovers from the unexpected shutdown with minimal data loss and disruption. The integrity of the database is maintained, and normal operations resume with confidence.

Conclusion

Transaction logs are essential for data recovery, providing a detailed record of changes and ensuring that databases can be restored to a consistent state following failures. The ability to redo committed transactions and undo incomplete ones highlights the importance of transaction logs in maintaining data integrity and continuity of business operations.
