DAY 10 :

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:

Below are the SQL statements to:

1. Begin a transaction.

2. Perform a series of INSERTs into the `orders` table.

3. Set a SAVEPOINT after each INSERT.

4. Rollback to the second SAVEPOINT.

5. Commit the overall transaction.

SQL Statements

-- Begin the transaction

```
BEGIN TRANSACTION;
```

-- Insert first record and set the first savepoint

```
INSERT INTO orders (order_id, customer_id, order_date, total_amount)
 VALUES (101, 1, '2024-05-15', 250.00);
SAVEPOINT savepoint1;
```

-- Insert second record and set the second savepoint

```
 INSERT INTO orders (order_id, customer_id, order_date, total_amount)
VALUES (102, 2, '2024-05-16', 150.00);
SAVEPOINT savepoint2;
```

-- Insert third record and set the third savepoint

```
INSERT INTO orders (order_id, customer_id, order_date, total_amount)
VALUES (103, 3, '2024-05-17', 300.00);
SAVEPOINT savepoint3;
```

-- Insert fourth record and set the fourth savepoint

INSERT INTO orders (order_id, customer_id, order_date, total_amount)

VALUES (104, 4, '2024-05-18', 200.00);

SAVEPOINT savepoint4;

-- Rollback to the second savepoint

ROLLBACK TO SAVEPOINT savepoint2;

-- Commit the overall transaction

COMMIT;

Explanation 1.

Begin the transaction:

BEGIN TRANSACTION;

This starts a new transaction.

2. Insert the first record and set the first savepoint:

INSERT INTO orders (order_id, customer_id, order_date, total_amount)

VALUES (101, 1, '2024-05-15', 250.00);

SAVEPOINT savepoint1;

This inserts a new record into the `orders` table and sets the first savepoint.

3. Insert the second record and set the second savepoint:

INSERT INTO orders (order_id, customer_id, order_date, total_amount)

VALUES (102, 2, '2024-05-16', 150.00);

SAVEPOINT savepoint2;

This inserts a second new record and sets the second savepoint.

4. Insert the third record and set the third savepoint:

INSERT INTO orders (order_id, customer_id, order_date, total_amount)

VALUES (103, 3, '2024-05-17', 300.00);

SAVEPOINT savepoint3;

This inserts a third new record and sets the third savepoint.

5. Insert the fourth record and set the fourth savepoint:

INSERT INTO orders (order_id, customer_id, order_date, total_amount) VALUES (104, 4, '2024-05-18', 200.00);

SAVEPOINT savepoint4;

This inserts a fourth new record and sets the fourth savepoint.

6. Rollback to the second savepoint:

ROLLBACK TO SAVEPOINT savepoint2;

This rolls back the transaction to the state after the second savepoint, undoing the third and fourth inserts.

7. Commit the overall transaction:

COMMIT;

This commits all changes made in the transaction up to the second savepoint, making the first and second inserts permanent and discarding the third and fourth inserts.