**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.**

1. Begin a transaction:

```
START TRANSACTION;
```

2. Perform a series of INSERTs into 'orders', setting a SAVEPOINT after each:

- Assuming we are inserting three orders and setting a savepoint after each insertion.

```
-- First INSERT
    INSERT INTO orders (order_id, product_id, customer_id, quantity)
    VALUES (1001, 101, 2001, 5);
    SAVEPOINT savepoint1;

-- Second INSERT
    INSERT INTO orders (order_id, product_id, customer_id, quantity)
    VALUES (1002, 102, 2002, 3);
    SAVEPOINT savepoint2;

-- Third INSERT
    INSERT INTO orders (order_id, product_id, customer_id, quantity)
    VALUES (1003, 103, 2003, 7);
```

3. Rollback to the second SAVEPOINT:

```
ROLLBACK TO savepoint2;
```

4. Commit the overall transaction:

```
COMMIT;
```

- These SQL statements will begin a transaction, perform a series of INSERTs into the 'orders' table, setting a SAVEPOINT after each insertion, rollback to the second SAVEPOINT, and finally commit the overall transaction.