

Transaction/Concurrency are key concepts which are used to ensure data integrity and consistency when multiple users or processes are accessing and modifying the data.

Transaction -- in sql server transaction is a sequence of one or more sql statement that are executed by a single user or unit of work.

ACID PROPERTIES:

1. Atomicity -- all operations in the transaction are treated as a single unit. If one operation fails the entire transaction fails.
2. Consistency -- Transactions take the database from one consistent state to another maintain the integrity.
3. Isolation -- Transactions are isolated from one another, it means changes made by one transaction are not visible to other until the transaction is completed.
4. Durability -- Once a transaction is committed the changes are permanently stored in the database even if the system crashes

Syntax of Transaction ---

Begin Transaction

Update Employees set salary = salary +1000 where employee id=1

Commit Transaction

BEGIN TRANSACTION -- START A TRANSACTION

COMMIT -- TO SAVE ALL THE CHANGES MADE BY THE TRANSACTION TO THE DATABASE

If something goes wrong you can use the rollback command to undo the changes

Concurrency -- It is a situation where multiple users or processes are accessing or modifying the data at the same time. This is common in multi user environments. Sql server uses mechanism locking and transaction isolation to ensure the data consistency while allowing multiple transactions to occur concurrently.

Shared lock -- two user are trying to read the data from the same table then they will get the shared lock and shared lock is shareable with the other users.

Exclusive lock

11:23 user a
11:23:10 user B

