Assignment 3: Explain the ACID properties of a transaction in your own words. Write SQL statements to simulate a transaction that includes locking and demonstrate different isolation levels to show concurrency control.

SOLUTION :

ACID PROPERTIES:

1. **Atomicity**: Atomicity ensures that either all operations within a transaction are successfully completed, or none of them are. It's like an "all-or-nothing" principle. If any part of the transaction fails, the entire transaction is rolled back, leaving the database in its original state.
2. **Consistency**: Consistency ensures that the database remains in a consistent state before and after the transaction. In other words, transactions should bring the database from one consistent state to another consistent state, preserving integrity constraints, foreign key relationships, etc.
3. **Isolation**: Isolation ensures that the execution of transactions concurrently does not result in any interference or inconsistency. Each transaction should appear to be executing in isolation, regardless of other concurrent transactions. Different isolation levels control the degree of isolation and trade-offs between concurrency and consistency.
4. **Durability**: Durability ensures that once a transaction is committed, its changes are permanent and will not be lost, even in the event of a system failure. This property ensures that the changes made by committed transactions persist even after a crash or restart.

CREATE TABLE BankAccount (

AccountID INT PRIMARY KEY,

Balance DECIMAL(10,2)

);

INSERT INTO BankAccount (AccountID, Balance)

VALUES (1, 5000), (2, 6000);

UPDATE BankAccount SET Balance = Balance - 1000

WHERE AccountID = 1;

UPDATE BankAccount SET Balance = Balance + 1000

WHERE AccountID = 2;

SET TRANSACTION ISOLATION LEVEL READ COMMITTED;

SELECT \* FROM BankAccount WHERE AccountID = 1;

COMMIT;

SET TRANSACTION ISOLATION LEVEL SERIALIZABLE;

UPDATE BankAccount SET Balance = Balance + 100 WHERE AccountID = 1;

COMMIT;

OUTPUT:





