**1. Atomicity: "All or Nothing"**

2. consistency-  Suppose the sum of all balances in a bank system should always be constant. Before a transfer, the total balance is **$700**. After the transaction, the total balance should remain $700. If the transaction fails in the middle (like updating one account but not the other), the system should maintain its consistency by rolling back the transaction.

**Total before T occurs** = 500 + 200 = 700 .  
**Total after T occurs**= 400 + 300 = 700 .

3.isolation- This property ensures that **multiple transactions** can occur concurrently without leading to the **inconsistency**of the database state.

### 4. Durability: Persisting Changes

After successfully transferring money from Account A to Account B, the changes are stored on disk. Even if there is a crash immediately after the commit, the transfer details will still be intact when the system recovers, ensuring durability.