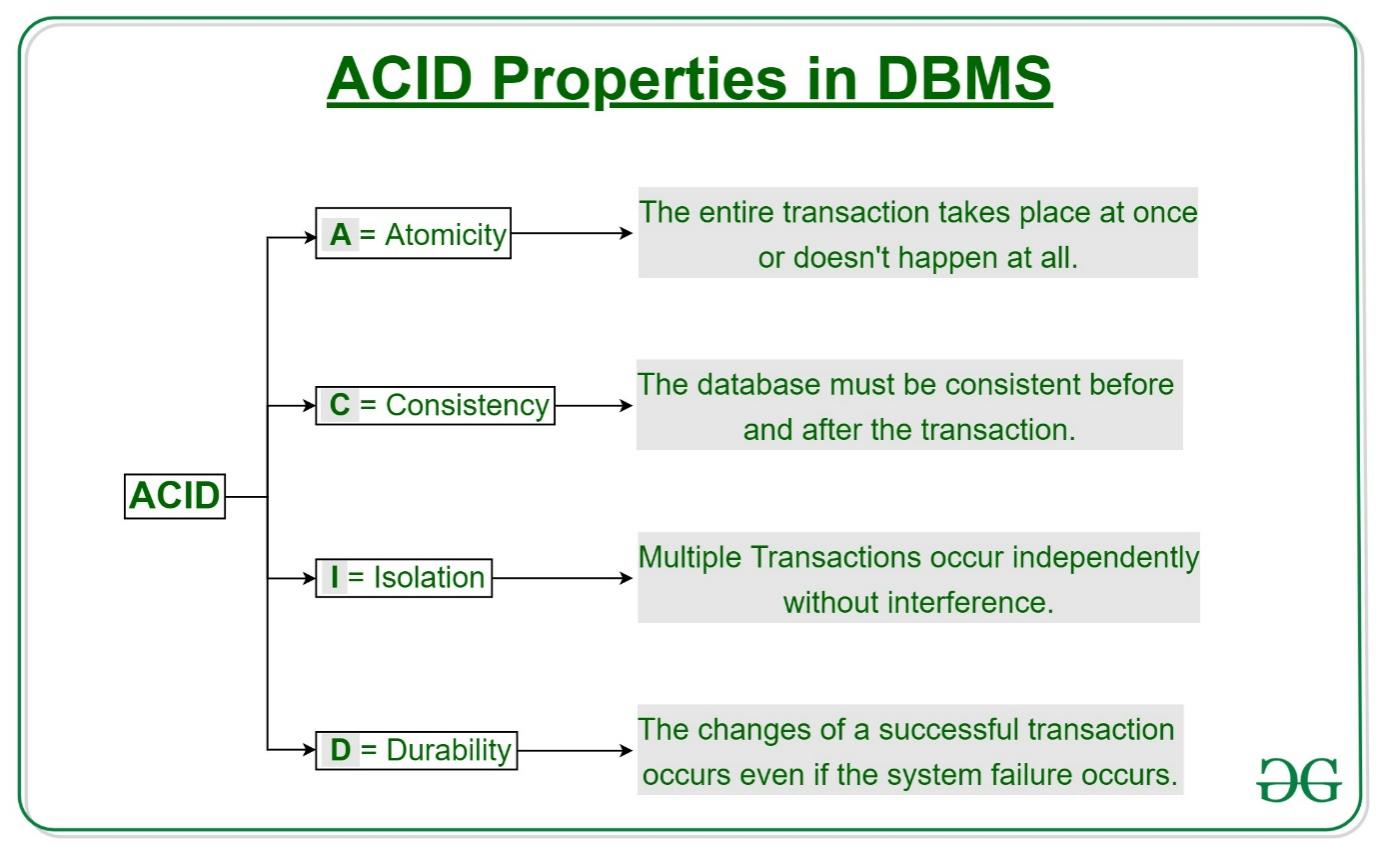
A **[transaction](https://www.geeksforgeeks.org/sql-transactions/)** is a **single logical unit of work** which accesses and possibly modifies the contents of a database. Transactions access data using read and write operations.   
In order to maintain consistency in a database, before and after the transaction, certain properties are followed. These are called **ACID** properties.



Atomicity   
By this, we mean that **either the entire transaction takes place at once or doesn’t happen at all**. There is **no midway** i.e., transactions do **not occur partially**. Each transaction is considered as one unit and either runs to completion or is not executed at all. It involves the following two operations.  
—**Abort**: If a transaction aborts, changes made to database are not visible.  
—**Commit**: If a transaction commits, changes made are visible.  
Atomicity is also known as the ‘**All or nothing rule**’.

Consistency   
This means that **integrity constraints must be maintained** so that the **database is consistent before and after the transaction**. It refers to the **correctness of a database.**

**Example**   
The total amount before and after the transaction must be maintained.   
Total before T occurs = 900 - 200 = 700  
Total after T occurs = 400 + 200 = 600  
total of 1300 is consistent before and after T  
Therefore, database is consistent. Inconsistency occurs in case T1 completes but T2 fails. As a result, T is incomplete.

Isolation   
This property ensures that **multiple transactions can occur concurrently without leading to the inconsistency of database state. Transactions occur independently without interference**. Changes occurring in a particular transaction will not be visible to any other transaction until that particular change in that transaction is written to memory or has been committed. This property ensures that the execution of transactions concurrently will result in a state that is equivalent to a state achieved these were executed serially in some order.

**Example**

Take T1, T2, T3 … Tn as transitions.

No dependencies of T1 on T2, T3, Tn and vice versa. (No knowledge of each other)

Durability  
This property ensures that once the transaction has completed execution, the updates and modifications to the database are stored in and written to disk and they persist even if a system failure occurs. These updates now become permanent and are stored in non-volatile memory. The effects of the transaction, thus, are never lost.