In SQL, **ACID** refers to a set of properties that ensure database transactions are processed reliably. Each letter stands for a key principle (ACID). These properties ensure reliable and predictable outcomes in SQL databases, which is crucial for applications where data integrity and stability are essential.

Atomicity: Atomicity means that each transaction is an all-or-nothing operation. If any part of a transaction fails, the entire transaction is rolled back, and no changes are committed to the database. This ensures data integrity by preventing partial updates in cases of failures.

Consistency: Consistency ensures that a transaction takes the database from one valid state to another. Rules and constraints (such as foreign keys, data types, and unique constraints) must be adhered to, so any data written to the database complies with these requirements. If a transaction violates these rules, it is rolled back, maintaining database validity.

Isolation: Isolation ensures that concurrently executed transactions do not affect each other. It keeps transactions separated until they are completed, so they do not see each other's intermediate states. SQL provides different levels of isolation (such as Read Uncommitted, Read Committed, Repeatable Read, and Serializable), allowing fine-grained control over concurrency.

Durability: Durability guarantees that once a transaction is committed, its changes are permanently saved, even in case of a system failure. Databases achieve durability by logging transactions to a file that can be used to reconstruct changes if there's a crash or power loss.

Conclusion

Together, these properties ensure reliable and predictable outcomes in SQL databases, which is crucial for applications where data integrity and stability are essential.