

|  |
| --- |
| **ACID** properties are a set of principles that ensure the **reliability** and **consistency** of data in a database system. ACID stands for **Atomicity**, **Consistency**, **Isolation**, and **Durability**. Here are some real-life examples of ACID properties in database transactions: |
| **Atomicity**: Suppose you want to book a flight and a hotel for your vacation. You use an online travel agency that offers a package deal for both. The transaction involves two operations: reserving a seat on the flight and booking a room in the hotel. Atomicity means that either both operations are completed successfully, or none of them are. If the flight reservation fails, for example, due to unavailability of seats, then the hotel booking is also canceled. This way, you don’t end up paying for a hotel without a flight, or vice versa. |
| **Consistency**: Suppose you have a bank account with a balance of $1000. You withdraw $200 from an ATM and then check your balance from another ATM. Consistency means that the balance shown on both ATMs is $800, reflecting the effect of your withdrawal. The database maintains the **integrity** of the data by enforcing rules and constraints, such as the balance cannot be negative, or the sum of all debits and credits must be zero. |
| **Isolation**: Suppose you and your friend share a Netflix account and want to watch different shows at the same time. You log in from your laptop and your friend logs in from his phone. Isolation means that each of you can access and modify the data on your own device without affecting the other’s view or experience. The database ensures that concurrent transactions do not interfere with each other by using locking mechanisms or other techniques. |
| **Durability**: Suppose you order a pizza online and pay with your credit card. The transaction involves updating the inventory of the pizza store, charging your credit card, and sending you a confirmation email. Durability means that once the transaction is completed successfully, the changes made to the database are permanent and will not be lost or undone even if there is a power outage, a system crash, or a network failure. The database achieves durability by writing the changes to a log file or a backup device. |