

Übungsfragen

ACID-Prinzipien

Wofür steht das „A“ in ACID?

- a) Access
- b) Atomicity
- c) Allocation
- d) Architecture

☒ **Lösung:** b) Atomicity

Welche Eigenschaft sorgt dafür, dass Transaktionen keine ungültigen Daten hinterlassen?

- a) Isolation
- b) Atomicity
- c) Durability
- d) Consistency

☒ **Lösung:** d) Consistency

Was verhindert das „Isolation“-Prinzip?

- a) Stromausfälle
- b) Unvollständige Transaktionen
- c) Beeinflussung paralleler Transaktionen
- d) Datenverlust nach dem Commit

☒ **Lösung:** c) Beeinflussung paralleler Transaktionen

Wann gilt eine Transaktion als dauerhaft gespeichert?

- a) Nach dem ersten SQL-Befehl
- b) Sobald sie im RAM liegt
- c) Nach einem Rollback
- d) Nach einem erfolgreichen Commit

☒ **Lösung:** d) Nach einem erfolgreichen Commit

Erkläre in einem Satz, was „Atomicity“ bedeutet.

☒ **Lösung:** Eine Transaktion wird vollständig ausgeführt oder gar nicht – es gibt keine Teilergebnisse.

Was versteht man unter einem „Dirty Read“ und wie hilft ACID, das zu vermeiden?

☑ **Lösung:** Ein „Dirty Read“ ist das Lesen von Daten aus einer nicht abgeschlossenen Transaktion. Das ACID-Prinzip „Isolation“ verhindert solche Probleme, indem es Transaktionen voneinander abschirmt..

Nenne ein Beispiel aus dem Alltag, bei dem das ACID-Prinzip wichtig ist.

☒ **Lösung:** Beim Online-Banking – z. B. beim Überweisen eines Betrags zwischen zwei Konten müssen beide Operationen (Abbuchung und Gutschrift) als atomare Transaktion erfolgen.

Wie unterstützt Write-Ahead Logging das „Durability“-Prinzip?

☒ **Lösung:** WAL sorgt dafür, dass alle Änderungen vorab protokolliert werden, sodass sie auch nach einem Systemabsturz wiederhergestellt werden können.