

SQL → 30% advantage in the

SQL → Structured Query Language

Database → If is the collection of data
in structured form

Types of Databases



Transactional Databases

- ↳ Vertically Scalable
- ↳ ACID properties
- ↳ Uses cases where data consistency is the primary factor
- Faster Read / Write Operation
- Not used for heavy data analytical queries
- Row format

ACID properties

- ① A (Atomicity) → Each Query / Action will be broken down as atomic transaction which will have a complete life cycle.
- ② C (Consistency)
- ③ I (Isolation)
- ④ D (Durability)

Data Consistency

Send 200 from A to B

Transactions

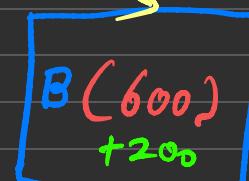
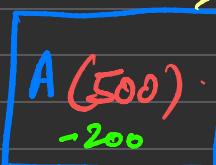
→ Deduct 200 from A

→ Update Value of A to 300

→ Add 200 to account B

→ Update Value of B

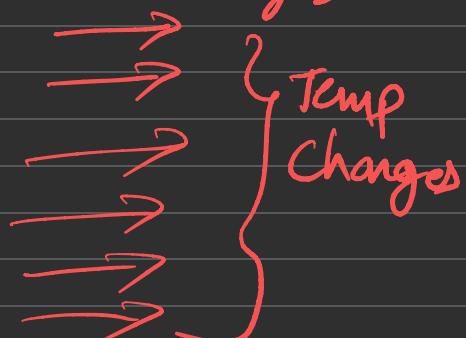
→ Commit these changes



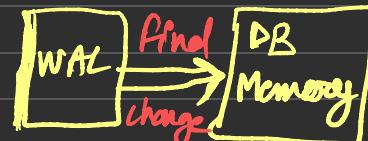
Operations on Database

- Insert
- Update
- Delete

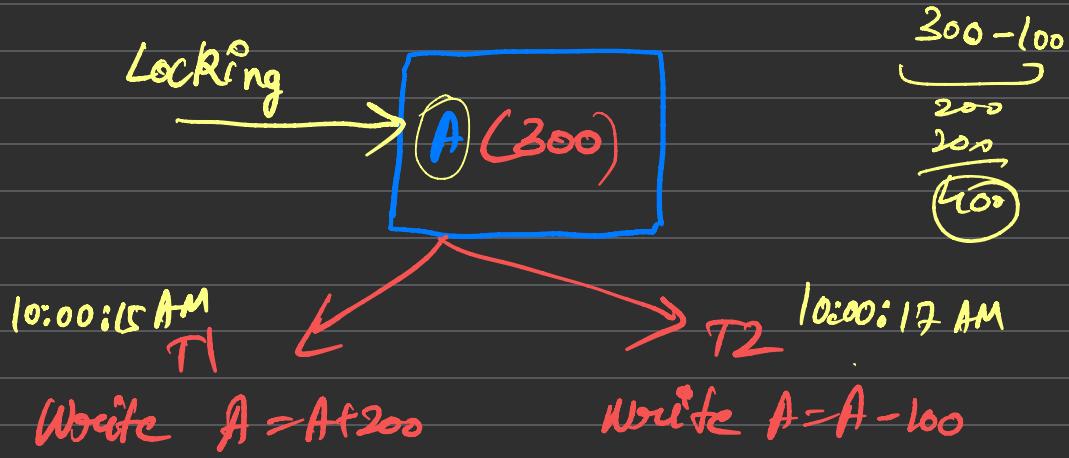
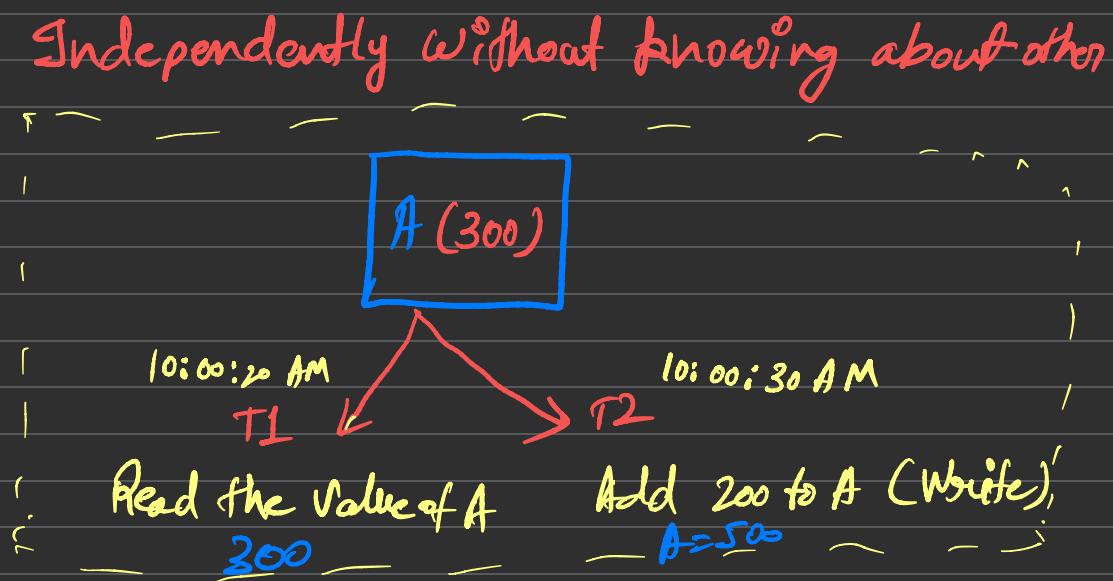
WAL (Write Ahead Log)



Commit



③ Isolation (I) → Each transaction runs
independently without knowing about other



X

④ Durability: Changes will be kept in WAL until committed or rolled back. It will be kept in volatile memory, hence helpful in recovery in case of crash.



DBMS → Database Management System

These are software applications which helps to store & manipulate the data.

RDBMS → Relational Database Management System

Software applications which helps to store data & manipulate. Databases will have relations with each other, and stored data in structured form.

(CSV, Table)

- ↳ Structured → Row-Columnar format
- ↳ Semi Structured → Flexible Schema
 - JSON
 - XML
- ↳ Unstructured → No specific format
 - Audio, Video, pdf etc.



RDBMS

MySQL
PostgreSQL
MSSQL
Oracle

MySQL ⇒ Relation (Table)

Employee

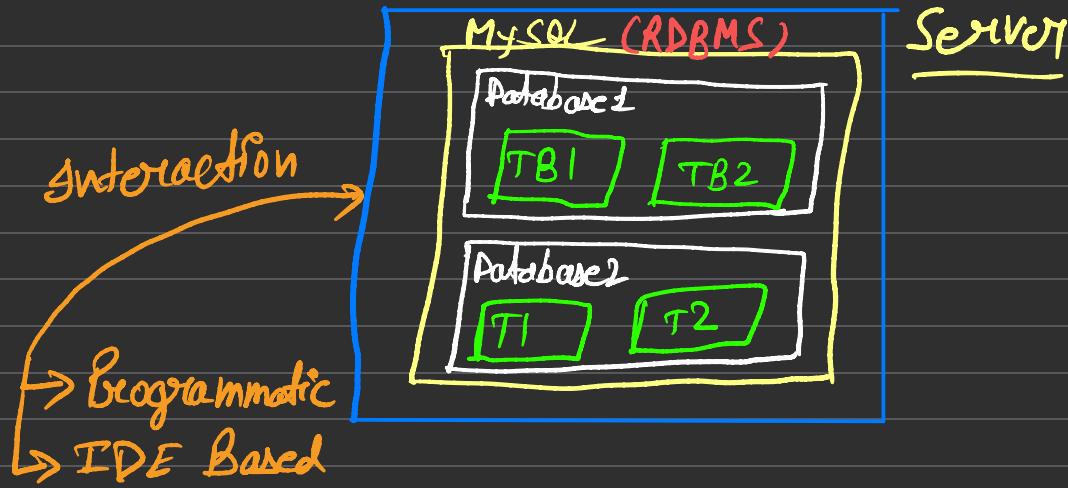
Row1

Row2

name	age	salary	team
A	20	10000	X
B	21	2000	Y

Column (Attribute)

Table



- ① Programmatic → Any Python / Java etc program, which need some library to connect and work with MySQL.
- ② IDE / Workbench → Running Queries from UI Interface.

