

WHAT IS DATA BASE?

- DATA ⇒ Raw and unprocessed facts.
- INFORMATION ⇒ Processed Data
- DATABASE ⇒ Collection of RELATED DATA OR STRUCTURED INFORMATION / DATA

PURPOSE :-

- ⇒ Store and Organize Data
- ⇒ Provide efficient data retrieval
- ⇒ ENSURE Data integrity and Security
- ⇒ Support Multiple users and Concurrent access

KEY COMPONENTS :-

DATA ⇒ The Actual information stored

META DATA ⇒ The database definition

SCHEMA ⇒ Description / Blueprint of database

DBMS ⇒ Software That manages The database

There are so many types of databases but we only focus on 2 types that cover all the family names of database that satisfies us.

I. RELATIONAL DataBase

- Store data in TABLES with PREDEFINED SCHEMA
- SQL is used for querying and managing data.
- Data Organized in Rows & Columns

ENTITY NAME ←

STUDENT

ID	NAME	DEPARTMENT
1	JACK	MECHANICAL
2	ROBIN	CIVIL
3	MONUKA	COMP. SCIENCE

→ TUPLES / ENTITIES

→ ATTRIBUTES OF ENTITY

- Enforce Data Integrity Through CONSTRAINTS

INTEGRITY ⇒ ACCURACY, CONSISTENCY AND RELIABILITY OF DATA THROUGH ITS (DATA BASE) LIFE CYCLE

SOME INTEGRITY :-

1. ENTITY INTEGRITY ⇒ EACH TABLE HAS TO HAVE ENTITIES WHICH ARE UNIQUE AND DIFFERENT, AND CAN ACHIEVE WITH PRIMARY KEYS EXAMPLE ⇒ ID IN STUDENT TABLE

Special

2. DOMAIN INTEGRITY \Rightarrow DOMAIN ATTRIBUTES ARE 100% OF THE DATA

TYPE ON VALUE STORED IN ATTRIBUTE, AND WE ACHIEVE IT THROUGH DATA TYPES, CONSTRAINTS & VALIDITY.

3. REFERENTIAL INTEGRITY \Rightarrow ALL TABLES ARE RELATED TO EACH OTHER

OVER OWN IDENTIFIERS / FOREIGN KEYS KEPT THROUGH ALL TABLES & RELATION CREATE ENTRIES, THAT IS TO SAY INTEGRITY OF EACH OTHER FOR CHILD AND PARENT TABLE OVER 3RD & 4TH ATTRIBUTE ENTRIES WHICH ARE PART OF RELATION IDENTITIES!

4. USER-DEFINED INTEGRITY \Rightarrow Self explanatory, and implemented through triggers, stored procedures or application logic.

5. CONSISTENCY \Rightarrow ENSURE data remains consistent across the entire database

6. ACCURACY \Rightarrow A database represent some aspect of the real world.

7. COMPLETENESS \Rightarrow All data is present without any missing information.

\rightarrow RDBMS is highly influenced by the Set Theory

\rightarrow EXAMPLE OF RDBMS \rightarrow MySQL, PostgreSQL, Oracle, Microsoft SQL Server.

points

2. NOSQL DATABASE

- NOT Only STRUCTURED QUERY LANGUAGE
- Non-Relational → Don't need Tabular relation
- Schema-flexible → allow storing data in other formats. Non tabular relation, and don't require fixed schema
- Horizontally Scalable → Designed to scale out by distributing data across many servers.
- EVENTUAL CONSISTENCY ⇒ Updates may take time to propagate to all nodes. Different nodes might return slightly different data for a short time after an update

Nosql are optimised and good for distributed system cause it follows the CAP THEOREM
A and P which is AVAILABILITY AND PARTITION TOLERANCE

AVAILABILITY(A) → Every request receives a response, without guarantee that it contains the most recent version of the data.

Special

Partition Tolerance (P) → The system continues to operate despite of network partitions.

→ NoSQL are optimized for specific data models :-

For this different types of NoSQL

(a) DOCUMENT-ORIENTED

- Store data in flexible, JSON-like documents.
- Good for Content management and user profile.
- MongoDB, Couchbase.

(b) KEY-VALUE STORES

- Simple data model of key-value pair
- Highly scalable and fast
- Used for Caching, Session management
- Redis, Amazon Dynamo DB

(c) WIDE-COLUMN STORES

- Store data in column families
- Suitable for time-series data, IoT
- Apache Cassandra, Google BigTable

(d) GRAPH DATABASES :

- Optimized for interconnected data.
- Used for Social networks and recommendation engines.
- Neo4j, Amazon Neptune.