

DATABASE AND MYSQL

DATABASE : Database is a recordkeeping system Which holds interrelated data that can be easily accessed, managed and updated.

RDBMS : A relational database management system (RDBMS) is a SOFTWARE that allows you to create, update, and administer a relational database (Data is organized in the form of table which contains rows and columns)

Some advantages of the relational database:

- Easy to categorize and store structured data for faster querying and filtering.
- Easy to scale and aren't reliant on physical organization
- Mature and well-understood database model.
- High level of security

We can have mainly two types of database.

- SQL – Examples: Oracle, MySQL, Microsoft SQL Server, PostgreSQL
- NoSQL – Examples: MongoDB, Redis, Casandra

1. What is SQL? Structured Query Language (SQL) is a Non Procedural computer language for relational database management and data manipulation. SQL is used to query, insert, update and modify data.



A query is a user-request to retrieve data or information with a certain condition

2. WHAT IS NOSQL DATABASE? NoSQL is a non-relational DMS that does not require a fixed schema, avoids joins, and is easy to scale.

- NoSQL database is used for distributed data stores with humongous data storage needs.
- NoSQL is used for Big data and real-time web apps. For example, companies like Twitter, Facebook, Google that collect terabytes of user data every single day.
- NoSQL database stands for "Not Only SQL" or "Not SQL."

NOSQL DATABASE TYPES

- **Document Databases** – These Db usually pair each key with a complex data structure which is called a document. Documents can contain key-array pairs or key-value pairs or even nested documents.
- **Key-value stores** – Every single item is stored as a Key-value pair. Key-value stores are the most *simple* among NoSQL Databases.

- **Wide-column stores** – These types of Databases are optimized for queries over large datasets, and instead of rows, they store columns of data together.
- **Graph stores** – These store information about graphs, networks, such as social connections.

Traditional RDBMS uses SQL syntax to store and retrieve data for further insights. Instead, a NoSQL database system encompasses a wide range of database technologies that can store structured, semi-

ADVANTAGES OF USING DBMS

- **Reducing data redundancy** – This ensures a consistent database and efficient use of storage space.
- **Aiding multiple views of the data:** Database management system enables different users to have a view of the data from their own perspective
- **Implementation of data integrity** The term data integrity refers to the accuracy and consistency of data. When creating databases, attention needs to be given to data integrity and how to maintain it.
- **Data sharing and security** – DBMS limits the accessibility of the data to the authorized users only. This makes data sharing safer. Moreover shared data makes it possible to fulfill data requirements with minimal modifications.
- **Centralized data** ensures a more effective backup and recovery process than the conventional file processing systems.
- **Program & Data independence:** Programs and the data are not mutually dependent. If a programmer wants to change the structure of the database then he/she can do so without affecting the application programs that are using the database.
- **Data Abstraction:** It ensures that the users get an abstract view of the data without showing implementation details and the physical storage of the data.

IMPORTANT TERMINOLOGIES used in RDBMS:

- **Relation /Tables** – In relational data model, relations are Tables. A table has rows and columns,
- **Tuple** – A single row of a table, which contains a single record for that relation is called a tuple.
- **Attribute:** A column in a Table is known as attribute.
- **Data item/Value:** An individual value in a table (at intersection point of row and column) is known as Data Value/item
- **Degree of a Table:** Total no. of Columns in a table is known as Degree of table.
- **Cardinality of a Table :** Total no. of Rows in a table is known as Cardinality of table.
- **Relation key** – Each row has one or more attributes, known as relation key, which can identify the row in the relation (table) uniquely

Why Key ???



- Helps to identify a record uniquely.
- Allows you to establish a relationship between and identify the relation between tables
- Help you to enforce identity and integrity in the relationship.

Types of Keys:

- Primary Key :** A column or group of columns in a table which helps us to identify unique tuple (Record) in a table.
- Candidate Keys :** All the keys which may be selected as Primary Key are Candidate keys
- Alternate Keys :** All the Candidate keys which are not selected as primary key are called an alternate key
- Foreign Key :** A foreign key is a column which is added to create a relationship with another table. Foreign keys help us to maintain data integrity and also allow navigation between two different instances of an entity.

LET'S UNDERSTND ABOVE MENTIONED TERMINOLOGIES THROUGH FOLLOWING EXAMPLE

Here we have taken 2 tables Flights and Fare . Both are interrelated tables.

- ✓ **Flight Table** is showing details of flights available from Source to Destination
- ✓ **Fare Table** is showing Fare details of all the flights along with its Airline Name

Table : FLIGHTS

FNO	SOURCE	DEST	NO_OF_FL	NO_OF_STOP
IC301	MUMBAI	BANGALORE	3	2
IC799	BANGALORE	KOLKATA	8	3
MC101	DELHI	VARANASI	6	0
IC302	MUMBAI	KOCHI	1	4
AM812	LUCKNOW	DELHI	4	0
MU499	DELHI	CHENNAI	3	3

Table : FARES

FNO	AIRLINES	FARE	TAX_percentage
IC301	Indian Airlines	9425	5
IC799	Spice Jet	8846	10
MC101	Deccan Airlines	4210	7
IC302	Jet Airways	13894	5
AM812	Indian Airlines	4500	6
MU499	Sahara	12000	4

In above tables

Relation	FLIGHT AND FARES both are Relation/Table					
Tuple	Every row/record in th tables are Tuple like <table border="1" style="margin: 5px auto;"> <tr> <td>IC799</td> <td>BANGALORE</td> <td>KOLKATA</td> <td>8</td> <td>3</td> </tr> </table> This is the record of flight no. IC799	IC799	BANGALORE	KOLKATA	8	3
IC799	BANGALORE	KOLKATA	8	3		
Attribute	All the column headings like FNO, Source, Dest etc are Attributes					
Degree	Degree of Flight table is 5 and Fare table is 4 (no.of columns)					
Cardinality	Cardinality of Flight table is 6 and Fare table is as both tables are having 6 records(rows)					
Primary Key	FNO is the Primary Key of FLIGHT table as it is the unique value and cannot be left blank IN Fare table FNO is Foreign key as its value is referenced with FNO of FLIGHT table					