SQL

DBMS

A database management system (DBMS) is a software package designed to define, manipulate, retrieve and manage data in a database. A DBMS generally manipulates the data itself, the data format, field names, record structure and file structure. It also defines rules to validate and manipulate this data.

Database management systems are set up on specific data handling concepts, as the practice of administrating a database evolves. The earliest databases only handled individual single pieces of specially formatted data. Today's more evolved systems can handle different kinds of less formatted data and tie them together in more elaborate ways.

The earliest types of database management systems consisted mainly of hierarchy and network models.

- The **hierarchy model** is one where each node or component has a child/parent relationship with one other node or component.
- In the **network model**, the difference is that a single component can have multiple relationships – think of this as a single node being able to "multicast" connections.

However, over time, these models became overtaken by something called a relational database. In the relational database model, individual components have attributes that are linked to their identities through a database table design. The rows and columns of an individual database table include those identities and attributes in such a way that traditional structured query language or SQL can be used to pull various kinds of information on these relational models.

Since then, an even newer concept has emerged called NoSQL. Experts suggest that the best way to understand NoSQL is to translate it to mean "not only SQL," or in other words, using NoSQL broadly to describe systems that are beyond the traditional SQL and relational database models.

NoSQL is also called the object-oriented database model.

Other type of DBMS models are

- Entity-relationship
- Relational

RDBMS

RDBMS stands for *Relational Database Management System.*

All modern database management systems like SQL, MS SQL Server, IBM DB2, ORACLE, My-SQL, and Microsoft Access are based on RDBMS.

It is called Relational Database Management System (RDBMS) because it is based on the relational model introduced by E.F. Codd.

Data is represented in terms of tuples (rows) in RDBMS.

A relational database is the most commonly used database. It contains several tables, and each table has its primary key.

Due to a collection of an organized set of tables, data can be accessed easily in RDBMS.

Table / Relation

Everything in a relational database is stored in the form of relations. The RDBMS database uses tables to store data. A table is a collection of related data entries and contains rows and columns to store data. Each table represents some real-world objects such as person, place, or event about which information is collected. The organized collection of data into a relational table is known as the logical view of the database.

Properties of a Relation:

- Each relation has a unique name by which it is identified in the database.
- Relation does not contain duplicate tuples.
- The tuples of a relation have no specific order.
- All attributes in a relation are atomic, i.e., each cell of a relation contains exactly one value.

Row / Record

A row of a table is also called a record or tuple. It contains the specific information of each entry in the table. It is a horizontal entity in the table. For example, The above table contains 5 records.

Properties of a row:

- No two tuples are identical to each other in all their entries.
- All tuples of the relation have the same format and the same number of entries.
- The order of the tuple is irrelevant. They are identified by their content, not by their position.

Column / Attribute

A column is a vertical entity in the table which contains all information associated with a specific field in a table. For example, "name" is a column in the above table which contains all information about a student's name.

Properties of an Attribute:

- Every attribute of a relation must have a name.
- Null values are permitted for the attributes.
- Default values can be specified for an attribute automatically inserted if no other value is specified for an attribute.
- Attributes that uniquely identify each tuple of a relation are the primary key.

Data Items / Cell

The smallest unit of data in the table is the individual data item. It is stored at the intersection of tuples and attributes.

Properties of data items:

- Data items are atomic.
- The data items for an attribute should be drawn from the same domain.

Cardinality

The total number of tuples at any one time in a relation is known as the table's cardinality. The relation whose cardinality is 0 is called an empty table.

The cardinality for a table with 6 rows is 6

Schema

a schema is a list of logical structures of data. A database user owns the schema, which has the same name as the database manager. As of SQL Server 2005, a schema is an individual entity (container of objects) distinct from the user who constructs the object. In other words, schemas are similar to separate namespaces or containers used to handle database files. Schemas may be assigned security permissions, making them an effective method for distinguishing and defending database objects based on user access privileges. It increases the database's stability for security-related management.

Advantages of Using Schema

Following are some of the main advantages of using a schema in SQL:

- A SQL schema can be easily transferred to another user.
- A schema may be shared by several users.
- It enables you to transfer database objects between schemas.
- We gain greater power over the access and protection of database objects.

- A user can be removed without removing the database items that are connected with the user.
- Database objects can be grouped into logical groups using schemas. This is advantageous when several people are collaborating on the same database program and the architecture team needs to keep the database tables' credibility.
- Since a schema allows for the logical aggregation of database objects, it can assist us in cases where the database object name is the same but falls into a separate logical category.