RDBMS

Relational Database Management Systems

What is RDBMS?

- RDBMS stands for Relational Database Management System.
- It's a program that allows users to create, update, and manage relational databases.
- RDBMSs also provide an interface between users and applications and the database, and administrative functions for managing data storage, access, and performance.
- RDBMSs use SQL queries to access data in the database.

DBMS vs RDBMS

DBMS	RDBMS
• DBMS stands for "Database Management System".	• RDBMS stands for "Relational Database Management System".
• DBMS technology stores the data in the form of files.	• RDBMS stores the data in the form of tables.
• DBMS is designed to handle small amounts of data.	• RDBMS is designed to deal with vast amount of data.
• DBMS provides support only for a single user at a time.	• RDBMS provides support for multiple users at a time.

RDBMS Terminologies

Table

- The data in an RDBMS is stored in database objects known as **tables**.
- This table is basically a collection of related data entries and it consists of numerous columns and rows.
- Remember, a table is the most common and simplest form of data storage in a relational database.

Field

- Every table is broken up into smaller entities called fields.
- A field is a column in a table that is designed to maintain specific information about every record in the table.
- For example, The CUSTOMERS table consists of different fields like ID, Name, Age, Salary, City and Country.

Record or Row

- A record is also called as a row of data is each individual entry that exists in a table.

- Following is a single row of data or record in the CUSTOMERS table –

ID	Name	Age	Salary	City	Country
1	Ramesh	32	2000.00	Hyderabad	India

- A record is a horizontal entity in a table.

Column

- A column is a vertical entity in a table that contains all information associated with a specific field in a table.

- For example, The CUSTOMERS table have different columns to represent ID, Name, Age, Salary, City and Country.

NULL Value

- A NULL value in a table is a value in a field that appears to be blank, which means a field with a NULL value is a field with no value.
- It is very important to understand that a NULL value is different than a zero value or a field that contains spaces.
- A field with a NULL value is the one that has been left blank during a record creation.

Constraints

SQL Constraints

- Constraints are the rules enforced on data columns on a table. These are used to limit the type of data that can go into a table.
- This ensures the accuracy and reliability of the data in the database.
- Constraints can either be column level or table level. Column level constraints are applied only to one column whereas, table level constraints are applied to the entire table.

SQL Constraint Types

S.No.	Constraints
1	**NOT NULL Constraint** Ensures that a column cannot have a NULL value.
2	**DEFAULT Constraint** Provides a default value for a column when none is specified.
3	**UNIQUE Key** Ensures that all the values in a column are different.
4	**PRIMARY Key** Uniquely identifies each row/record in a database table.
5	**FOREIGN Key** Uniquely identifies a row/record in any another database table.
6	**CHECK Constraint** Ensures that all values in a column satisfy certain conditions.

Normalization

Database Normalization

- Database normalization is a process that organizes data in a database by creating tables and establishing relationships between them.
- This process is based on rules that aim to protect the data and make the database more flexible.

- Normalization can help to:
  - Eliminate redundancy
  - Improve data integrity
  - Avoid data anomalies
  - Reduce data duplications

First Normal Form

- **First Normal Form (1NF)** sets the basic rules to organize the data in a database. A database is said to be in first normal form if it satisfies the following conditions –

  - **Rule 1 (Atomic Values)** – Every column of a table should contain only atomic values. An atomic value is a value that cannot be divided further.
  - **Rule 2 (No Repeating Groups)** – There are no repeating groups of data. This means a table should not contain repeating columns.

Second Normal Form

- For a database to meet second normal form, or 2NF, it must meet the rules of first normal form, and also "each non-key attribute must be functionally dependent on the primary key".

- This means that any column that is not the primary key needs to depend on the primary key. If it doesn't, we need to solve that.

Third Normal Form

- A design will meet third normal form if it has no "transitive functional dependency".
- A "transitive functional dependency" means that every attribute that is not the primary key is dependent on only the primary key.

- For example, in one table:
  - Column A determines Column B. Column B determines Column C.
  - This is a "transitive functional dependency" because Column C depends on Column B instead of Column A.

ER Diagram

ER Diagram

- Entity Relationship Diagram, also known as ERD, ER Diagram or ER model, is a type of structural diagram for use in database design.

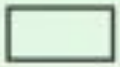
- An ERD contains different symbols and connectors that visualize two important information: **The major entities within the system scope**, and the **inter-relationships among these entities**.

- And that's why it's called "Entity" "Relationship" diagram (ERD).

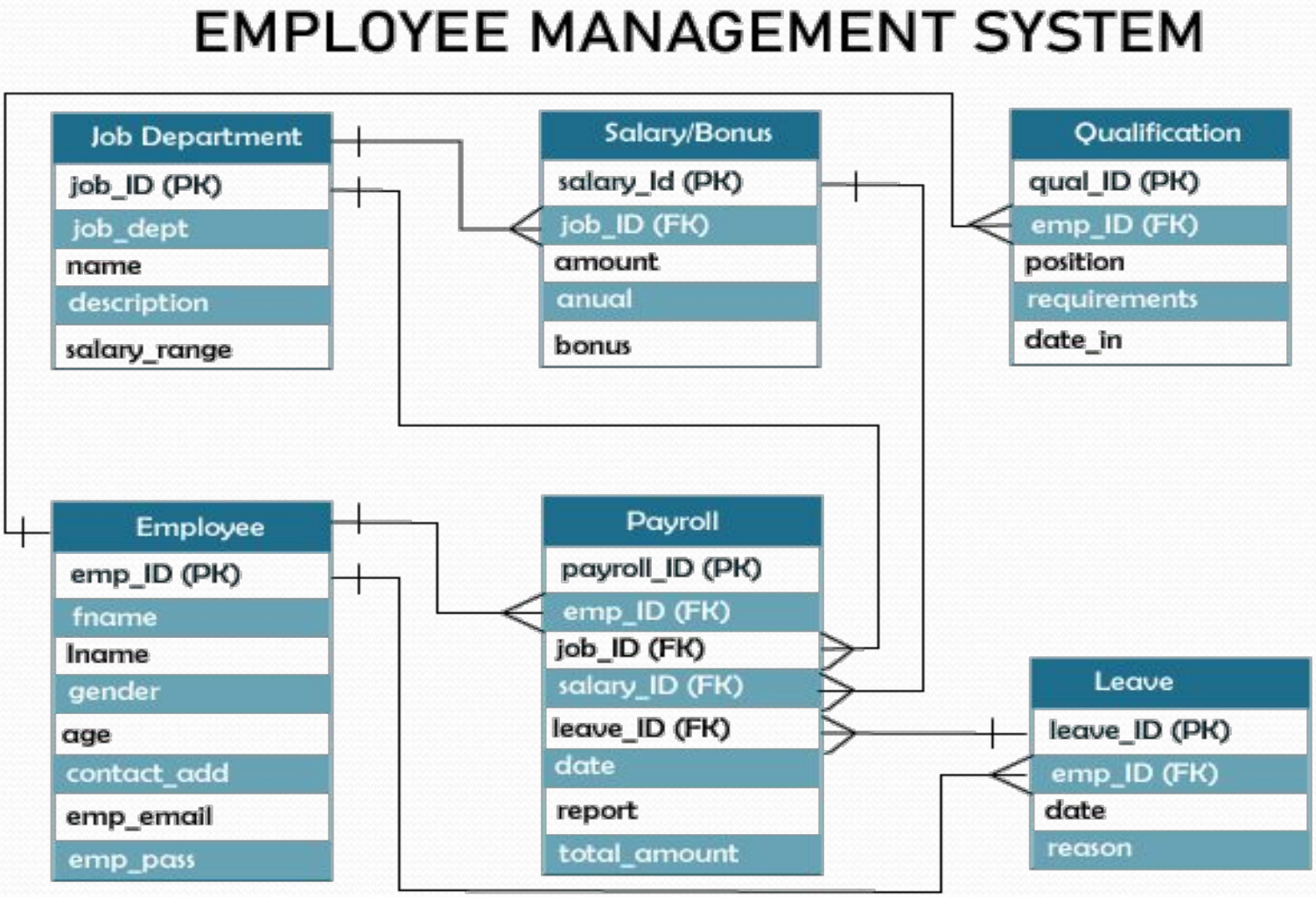
Why ER Diagram?

- ER diagrams are used to represent the E-R model in a database, which makes them easy to convert into relations (tables).
- ER diagrams provide the purpose of real-world modeling of objects which makes them intently useful.
- ER diagrams require no technical knowledge and no hardware support.
- These diagrams are very easy to understand and easy to create even for a naive user. It gives a standard solution for visualizing the data logically.

ER Diagram Symbols

Figures	Symbols	Represents
Rectangle		Entities in ER Model
Ellipse		Attributes in ER Model
Diamond		Relationships among Entities
Line		Attributes to Entities and Entity Sets with Other Relationship Types
Double Ellipse		Multi-Valued Attributes
Double Rectangle		Weak Entity

Sample ER Diagram

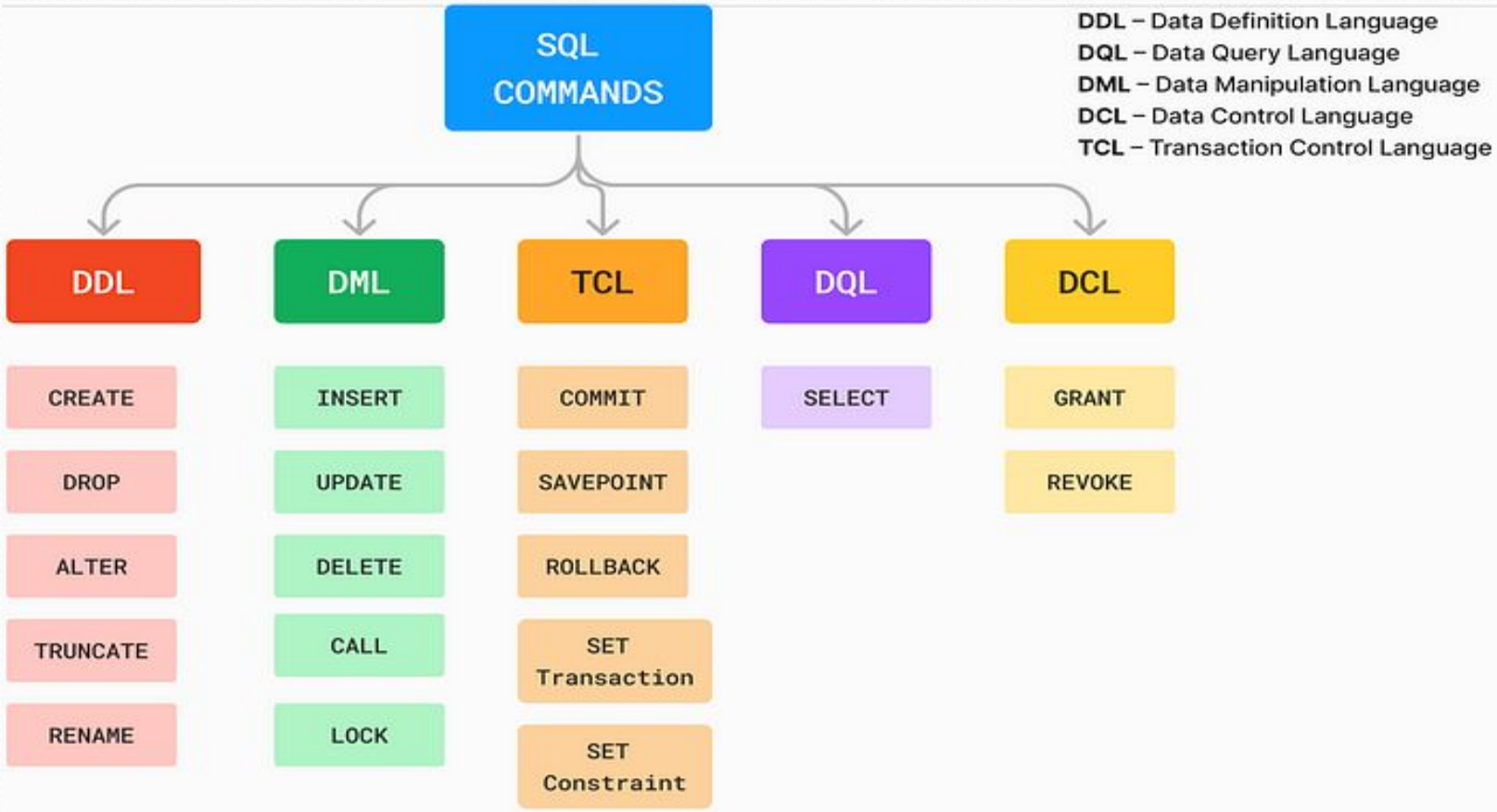


SQL

What is SQL?

- Structured query language (SQL) is a programming language for storing and processing information in a relational database.
- A relational database stores information in tabular form, with rows and columns representing different data attributes and the various relationships between the data values. You can use SQL statements to store, update, remove, search, and retrieve information from the database.
- You can also use SQL to maintain and optimize database performance.

Types of SQL commands.



Thank You

Valan Arasu M