Q1: What is the difference between delete and drop?

The DELETE command is a Data Manipulation Language operation used to remove specific rows from a table based on a condition, but it keeps the table structure, indexes, and constraints intact. In contrast, the DROP command is a Data Definition Language operation that completely removes the entire table, its data, and its metadata from the database schema, meaning the table no longer exists after the command is executed.

Q2: What is the difference with drop and truncate?

The DROP command is used to delete the entire table object from the database, which means both the data and the structure are permanently removed. The TRUNCATE command is a Data Definition Language operation that deletes all the records within a table but preserves the structure, columns, and constraints for future data insertion. TRUNCATE is generally faster than DELETE because it does not log individual row removals.

Q3: Explain acid properties?

The ACID properties consist of Atomicity, Consistency, Isolation, and Durability, which serve as the foundation for reliable database transactions. Atomicity ensures a transaction is an all or nothing proposition, Consistency ensures that a transaction takes the database from one valid state to another, Isolation guarantees that concurrent transactions do not interfere with each other, and Durability ensures that once a transaction is committed, it remains saved even in the event of a system failure.

Q4: How many types of normalization's are there?

There are several levels of normalization designed to reduce data redundancy and improve integrity, including First Normal Form, Second Normal Form, Third Normal Form, and Boyce-Codd Normal Form. Beyond these common stages, there are also higher levels known as Fourth Normal Form which deals with multi-valued dependencies, and Fifth Normal Form which manages join dependencies, making a total of six primary types used in database design.

Q5: How to fetch the second highest salary from the employee and the department?

To fetch the second highest salary from an employee table, one can use a correlated subquery that finds the maximum salary which is strictly less than the overall maximum salary. An example of the logic is selecting the maximum salary where the value is not in the set of the top maximum salary, or using a limit and offset clause to sort the salaries in descending order and skipping the first record to retrieve the second one.

INTERNAL

Q6: Explain crud in the dbms?

CRUD is an acronym representing the four basic functions of persistent storage which are Create, Read, Update, and Delete. Create refers to adding new records to the database using the INSERT statement, Read involves retrieving data using the SELECT statement, Update is used to modify existing records, and Delete is used to remove records that are no longer required.

Q7: What is entity diagram?

An entity diagram, or Entity Relationship Diagram, is a graphical representation of the logical structure of a database that shows how different entities such as people or objects relate to one another. It uses specific symbols like rectangles for entities, ovals for attributes, and diamonds for relationships to provide a visual map of how data is organized and connected within a system.

Q8: How do you describe the relationship?

Relationships in a database are described through cardinality, which defines the numerical relationship between occurrences of one entity and occurrences of another. These descriptions are typically categorized as one-to-one, one-to-many, or many-to-many, and they are established in the database through the use of primary keys and foreign keys to link related rows across different tables.

Q9: Difference between unique key and primary key?

The primary difference is that a Primary Key uniquely identifies each record in a table and strictly prohibits NULL values, while a Unique Key also ensures all values in a column are distinct but can allow for a single NULL value. Furthermore, a table can have only one Primary Key defined, whereas it is possible to have multiple Unique Keys on different columns within the same table.

Q10: Why do we use NOT NULL?

We use the NOT NULL constraint to ensure that a specific column in a database table must always contain a value and cannot be left empty. This is crucial for maintaining data integrity in fields that are essential for business logic or identification, such as a primary key or a username, as it prevents errors and inconsistencies that occur when mandatory data is missing.