TOPSTechnologies

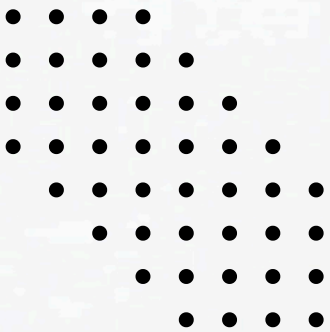
Introduction to DBMS

Presented for :

TOPs Technologies

Presented by :

Sumit B Yadav



Introduction to DBMS

Que 1

SQL (Structured Query Language) is a standardized programming language designed for managing and manipulating relational databases. It provides a systematic way to interact with data, enabling users to create, read, update, and delete data within a database.

Key Features of SQL:

1. Data Manipulation: SQL allows users to perform CRUD operations (Create, Read, Update, Delete) on data in a database.
   - SELECT: Retrieve data from one or more tables.
   - INSERT: Add new data to tables.
   - UPDATE: Modify existing data.
   - DELETE: Remove data from tables.

- **Data Definition: SQL provides commands to define and modify the structure of a database.**
- **CREATE: Create new tables, schemas, or databases.**
- **ALTER: Modify the structure of existing tables or schemas.**
- **DROP: Delete tables or databases.**
- **Data Querying: SQL supports powerful querying capabilities using filters, aggregations, and joins to retrieve specific and complex datasets.**
- **Data Integrity and Constraints: SQL allows the enforcement of rules (e.g., primary keys, foreign keys, unique constraints) to ensure data accuracy and integrity.**
- **Access Control: SQL provides mechanisms to grant or revoke permissions, enhancing database security.**
- **Transaction Management: SQL supports transactions to ensure data consistency through atomic operations (ACID properties).**

Why SQL is Essential:

- Universal Standard: SQL is the industry standard for relational database management systems (RDBMS) like MySQL, PostgreSQL, Oracle, and SQL Server.

- Efficient Data Handling: SQL queries are optimized for handling large datasets, making data processing faster and more efficient.

- Interoperability: SQL allows applications to access and manipulate data stored in relational databases, enabling seamless integration between systems.

- Data Analysis: SQL's querying capabilities are widely used for extracting and analyzing data, making it crucial in fields like data science, analytics, and business intelligence.

- Scalability: SQL databases can handle large volumes of data and are suitable for applications of varying sizes, from small projects to enterprise-level systems.
- Reliability: SQL ensures robust data management with features like backups, recovery options, and support for concurrency.

Que. 2

The terms DBMS (Database Management System) and RDBMS (Relational Database Management System) are both related to systems for managing data, but they have key differences in functionality, structure, and application.

DBMS (Database Management System)

Definition:

A DBMS is software that provides a way to store, retrieve, and manage data in a database. It does not necessarily enforce relationships between the data.

Key Features:

1. Data Structure:
   - DBMS stores data in a file-based system or hierarchical, network, or object-oriented formats.
   - Relationships between data are not explicitly defined.

- Normalization:
- Typically does not enforce data normalization (organizing data to reduce redundancy).
- ACID Compliance:
- May not fully support ACID (Atomicity, Consistency, Isolation, Durability) properties, making it less reliable for critical systems.
- Scalability:
- Generally suitable for smaller or single-user systems.

RDBMS (Relational Database Management System)

Definition:

An RDBMS is a specific type of DBMS that uses a relational model for data storage, where data is organized into tables (rows and columns), and relationships between tables are defined.

Key Features:

1. Data Structure:
   - Data is stored in tables with rows (records) and columns (fields).
   - Enforces relationships using keys (e.g., primary and foreign keys).
2. Normalization:
   - Encourages normalization to minimize redundancy and improve data integrity.
3. ACID Compliance:
   - Fully supports ACID properties, ensuring data reliability and consistency.
4. Scalability:
   - Designed for multi-user environments and can handle large-scale databases with complex relationships.

Que. 3

SQL (Structured Query Language) plays a central role in managing relational databases by providing a standardized way to define, manipulate, query, and control data stored in relational tables. It serves as the primary interface for interacting with relational database management systems (RDBMS).

1. Data Definition (DDL - Data Definition Language)
SQL allows users to define and modify the structure of databases and their objects, such as tables, schemas, indexes, and constraints.

Key Commands:

- CREATE: Creates new database objects like tables, indexes, and schemas.
  - Example: CREATE TABLE employees (id INT, name VARCHAR(100));
- ALTER: Modifies the structure of existing objects.
  - Example: ALTER TABLE employees ADD COLUMN salary DECIMAL(10, 2);
- DROP: Deletes database objects.
  - Example: DROP TABLE employees;

2. Data Manipulation (DML - Data Manipulation Language)

SQL enables users to interact with and modify the data stored in relational tables.

Key Commands:

- INSERT: Adds new rows of data.
  - Example: INSERT INTO employees (id, name, salary) VALUES (1, 'Alice', 50000);
- UPDATE: Modifies existing data.
  - Example: UPDATE employees SET salary = 55000 WHERE id = 1;
- DELETE: Removes rows of data.
  - Example: DELETE FROM employees WHERE id = 1;

3. Data Querying (DQL - Data Query Language)

SQL provides powerful querying capabilities to retrieve specific information from one or more tables.

Key Command:

- SELECT: Extracts data based on specified criteria.
  - Example: SELECT name, salary FROM employees WHERE salary > 50000;

Advanced querying features include:

- Filtering: Using WHERE clauses for conditions.
- Joins: Combining data from multiple tables.
  - Example: SELECT e.name, d.name FROM employees e JOIN departments d ON e.dept_id = d.id;

4. Data Control (DCL - Data Control Language)

SQL manages access and permissions for database users to ensure data security.

Key Commands:

- GRANT: Gives specific permissions to users.
  - Example: GRANT SELECT, INSERT ON employees TO user1;
- REVOKE: Removes permissions.
  - Example: REVOKE INSERT ON employees FROM user1;

5. Transaction Control (TCL - Transaction Control Language)

SQL ensures data consistency and reliability by supporting transaction management.

Key Commands:

- BEGIN TRANSACTION: Starts a transaction.
- COMMIT: Saves changes made during a transaction.
  - Example: COMMIT;
- ROLLBACK: Reverts changes made during a transaction in case of errors.
  - Example: ROLLBACK;

Que.4

Here are the key features of SQL:

1. Data Manipulation

SQL allows users to perform operations on data stored in relational databases:

- Insert: Add new data using INSERT.
- Update: Modify existing data using UPDATE.
- Delete: Remove data using DELETE.
- Retrieve: Fetch data with SELECT.

2. Data Definition

SQL provides tools to define and modify the structure of databases:

- Create: Define new tables, schemas, or entire databases using CREATE.
- Alter: Modify the structure of existing tables with ALTER.
- Drop: Remove database objects like tables or databases using DROP.

3. Data Querying

SQL's powerful querying capabilities enable users to extract specific data from large datasets:

- Filters: Use WHERE clauses to filter data.
- Sorting: Use ORDER BY to sort data in ascending or descending order.
- Aggregations: Apply functions like SUM, AVG, MAX, MIN, and COUNT.
- Joins: Combine data from multiple tables using INNER JOIN, LEFT JOIN, RIGHT JOIN, etc.
- Grouping: Group data using GROUP BY.

4. Data Control

SQL offers commands for managing user access and permissions:

- Grant: Assign permissions to users using GRANT.
- Revoke: Remove permissions using REVOKE.

5. Transaction Management

SQL supports transaction control to ensure data integrity and consistency:

- BEGIN TRANSACTION: Starts a transaction.
- COMMIT: Saves all changes made in the transaction.
- ROLLBACK: Reverts changes if an error occurs during a transaction.

6. Constraints and Data Integrity

SQL enforces data integrity using constraints:

- Primary Key: Ensures each row in a table is unique.
- Foreign Key: Maintains referential integrity between tables.
- Unique: Ensures all values in a column are distinct.
- Not Null: Prevents null values in a column.
- Check: Validates data based on a condition.

7. Standardized Language

SQL is a widely adopted standard across major relational database management systems (RDBMS) like MySQL, PostgreSQL, Oracle Database, and Microsoft SQL Server.

8. Flexibility

SQL supports various operations on data:

- Dynamic Queries: Create reusable and parameterized queries.
- Complex Queries: Handle nested queries with subqueries.