

Name: Paramprakash Makwana

Module 4 – Introduction to DBMS Introduction to SQL(Theory)

4.1) What is SQL, and why is it essential in database management?

Ans:- SQL (Structured Query Language) is used to store, retrieve, and manage data in databases. It is essential because it lets users interact with databases efficiently and safely.

4.2. Explain the difference between DBMS and RDBMS.

Ans:- DBMS stores data as files or tables without relationships. RDBMS stores data in related tables using keys and follows rules like data integrity.

4.3. Describe the role of SQL in managing relational databases.

Ans:- SQL is used to create tables, insert and update data, retrieve records, and control access in relational databases.

4.4. What are the key features of SQL?

Ans:- SQL is easy to learn, supports data security, and handles large amounts of data. It works with multiple database systems like MySQL and Oracle.

4.5. What are the basic components of SQL syntax?

Ans:- SQL syntax includes keywords (SELECT, INSERT), clauses (WHERE, GROUP BY), expressions, tables, and operators.

4.6. Write the general structure of an SQL SELECT statement.

Ans:- `SELECT column_name FROM table_name WHERE condition;`

4.7. Explain the role of clauses in SQL statements.

Ans:- Clauses define different parts of a query and control how data is selected, filtered, grouped, and sorted.

4.8) What are constraints in SQL? List and explain the different types of constraints.

Ans:- Constraints are rules applied to table columns to ensure valid and accurate data.

NOT NULL: Prevents NULL values in a column.

UNIQUE: Ensures all values in a column are different.

PRIMARY KEY: Uniquely identifies each record in a table.

FOREIGN KEY: Links one table to another using a referenced key.

CHECK: Allows only values that satisfy a condition.

DEFAULT: Assigns a default value if no value is provided.

4.9) How do PRIMARY KEY and FOREIGN KEY constraints differ?

Ans:- A PRIMARY KEY uniquely identifies records in a table and cannot be NULL. A FOREIGN KEY links a column to the PRIMARY KEY of another table and allows relationships.

4.10) What is the role of NOT NULL and UNIQUE constraints?

Ans:- NOT NULL ensures a column always has a value.

UNIQUE ensures no duplicate values are stored in a column.

4.11) Define the SQL Data Definition Language (DDL).

Ans:- DDL is a set of SQL commands used to create, modify, and delete database objects like tables, views, and indexes.

4.12) Explain the CREATE command and its syntax.

Ans:- `CREATE TABLE table_name (column_name datatype constraints);`

4.13) What is the purpose of specifying data types and constraints during table creation?

Ans:- Data types define what kind of data a column can store.

Constraints ensure data accuracy, consistency, and integrity in the table.

4.14) What is the use of the ALTER command in SQL?

Ans:- The ALTER command is used to change the structure of an existing table, such as adding, modifying, or deleting columns.

4.15) How can you add, modify, and drop columns from a table using ALTER?

Ans:- Add a column:-

```
ALTER TABLE table_name
```

```
ADD column_name datatype;
```

Modify a column:-

```
ALTER TABLE table_name
```

```
MODIFY column_name new_datatype;
```

Drop a column:-

```
ALTER TABLE table_name
```

```
DROP column_name;
```

4.15) What is the function of the DROP command in SQL?

Ans:- The DROP command is used to permanently delete database objects like tables, views, or databases.

4.16) What are the implications of dropping a table from a database?

Ans:- Dropping a table removes the table structure and all its data permanently.

It cannot be undone and any dependent objects like constraints or views may also be affected.

4.17) Define the INSERT, UPDATE, and DELETE commands in SQL.

Ans:- INSERT adds new records into a table.

UPDATE modifies existing records in a table.

DELETE removes records from a table.

4.18) What is the importance of the WHERE clause in UPDATE and DELETE operations?

Ans:- The WHERE clause specifies which rows should be updated or deleted.

Without it, all rows in the table will be affected.

4.19) What is the SELECT statement, and how is it used to query data?

Ans:- The SELECT statement is used to retrieve data from one or more tables. It allows you to choose specific columns and rows.

4.20) Explain the use of the ORDER BY and WHERE clauses in SQL queries.

Ans:- WHERE filters records based on a condition.

ORDER BY sorts the result set in ascending or descending order.

4.21) What is the purpose of GRANT and REVOKE in SQL?

Ans:- GRANT is used to give permissions to users on database objects.

REVOKE is used to remove previously given permissions.

4.22) How do you manage privileges using these commands?

Ans:- You manage privileges by granting specific rights like SELECT or INSERT using GRANT, and removing them when no longer needed using REVOKE.

4.23) What is the purpose of the COMMIT and ROLLBACK commands in SQL?

Ans:- COMMIT permanently saves all changes made in a transaction.

ROLLBACK cancels changes and restores the database to its previous state.

4.24) Explain how transactions are managed in SQL databases.

Ans:- A transaction groups multiple SQL operations into a single unit.

The database ensures all operations succeed together or none are applied, maintaining data consistency.

4.25) Explain the concept of JOIN in SQL. What is the difference between INNER JOIN, LEFT JOIN, RIGHT JOIN, and FULL OUTER JOIN?

Ans:- A JOIN is used to combine rows from two or more tables based on a related column.

INNER JOIN: Returns only matching records from both tables.

LEFT JOIN: Returns all records from the left table and matching records from the right table.

RIGHT JOIN: Returns all records from the right table and matching records from the left table.

FULL OUTER JOIN: Returns all records when there is a match in either table.

4.26) How are joins used to combine data from multiple tables?

Ans:- Joins link tables using common columns, usually primary and foreign keys, to display related data in a single result set.

4.27) What is the GROUP BY clause in SQL? How is it used with aggregate functions?

Ans:- GROUP BY is used to group rows that have the same values in specified columns. It works with aggregate functions like COUNT, SUM, AVG, MIN, and MAX to return one result per group.

4.28) Explain the difference between GROUP BY and ORDER BY.

Ans:- GROUP BY groups rows to perform calculations on each group. ORDER BY sorts the result set in ascending or descending order.

4.29) What is a stored procedure in SQL, and how does it differ from a standard SQL query?

Ans:- A stored procedure is a precompiled set of SQL statements stored in the database.

A standard SQL query runs a single command, while a procedure can contain logic like loops and conditions.

4.30) Explain the advantages of using stored procedures.

Ans:- Stored procedures improve performance, reduce repeated code, and enhance security by controlling direct access to tables.

4.31) What is a view in SQL, and how is it different from a table?

Ans:- A view is a virtual table created using a SELECT query.

A table stores data physically, while a view stores only the query, not the data itself.

4.32) Explain the advantages of using views in SQL databases.

Ans:- Views simplify complex queries, improve security by limiting data access, and provide consistent data presentation.

4.33) What is a trigger in SQL? Describe its types and when they are used.

Ans:- A trigger is a special stored program that runs automatically when a specific event occurs on a table.

Types of triggers include BEFORE and AFTER triggers, used to validate data or log changes.

4.34) Explain the difference between INSERT, UPDATE, and DELETE triggers.

Ans:- INSERT trigger runs when a new record is added.

UPDATE trigger runs when an existing record is modified.

DELETE trigger runs when a record is removed.

4.35) What is PL/SQL, and how does it extend SQL's capabilities?

Ans:- PL/SQL is Oracle's procedural extension of SQL that allows programming constructs like loops, conditions, and variables.

It extends SQL by enabling complex logic and control structures inside the database.

4.36) List and explain the benefits of using PL/SQL.

Ans:- Allows procedural programming with SQL.

Reduces network traffic by executing multiple statements at once.

Improves code reusability with procedures, functions, and packages.

Enhances security by controlling data access.

4.37) What are control structures in PL/SQL? Explain the IF-THEN and LOOP control structures.

Ans:- Control structures guide the flow of execution in PL/SQL programs.

IF-THEN executes a block of code only if a specified condition is true.

LOOP repeatedly executes a block of code until a condition is met (simple LOOP, WHILE LOOP, and FOR LOOP are common types).

4.38) How do control structures in PL/SQL help in writing complex queries?

Ans:- They allow conditional processing, repeated operations, and procedural logic, making it possible to handle complex data manipulations inside the database efficiently.

4.39) What is a cursor in PL/SQL? Explain the difference between implicit and explicit cursors.

Ans:- A cursor is a pointer that allows you to fetch and process rows returned by a query one at a time.

Implicit cursor: Automatically created by PL/SQL for single-row queries.

Explicit cursor: Defined and controlled by the programmer, used for multi-row queries.

4.40) When would you use an explicit cursor over an implicit one?

Ans:- Use an explicit cursor when you need to process multiple rows one by one or control the fetching of records, such as in loops or conditional processing.

4.41) Explain the concept of SAVEPOINT in transaction management. How do ROLLBACK and COMMIT interact with savepoints?

Ans:- A SAVEPOINT is a marker within a transaction that allows partial rollback.

ROLLBACK TO savepoint undoes changes only up to that point without affecting earlier operations.

COMMIT saves all changes permanently, including those before the savepoint.

4.42) When is it useful to use savepoints in a database transaction?

Ans:- Savepoints are useful in complex transactions to undo specific parts of work without discarding the entire transaction, helping maintain data integrity.