SQL-DBMS

THEORY:-

✓ Part 1: Introduction to SQL

Q1. What is SQL, and why is it essential in database management?

A: SQL (Structured Query Language) is a standard programming language used to manage and manipulate relational databases. It is essential because it enables users to efficiently retrieve, insert, update, and delete data from databases.

Q2. Explain the difference between DBMS and RDBMS.

A: DBMS is software for storing and managing data, whereas RDBMS is a DBMS that stores data in relational tables and enforces relationships using keys like primary and foreign keys.

Q3. Describe the role of SQL in managing relational databases.

A: SQL acts as a tool to query, manipulate, and manage data in relational databases using commands like SELECT, INSERT, UPDATE, DELETE, etc.

Q4. What are the key features of SQL?

A: Key features include:

- Data manipulation
- Data definition
- Transaction control
- Security and permissions
- Built-in functions and joins

Part 2: SQL Syntax

Q1. What are the basic components of SQL syntax?

A: Basic components include SQL keywords (SELECT, FROM, WHERE), clauses, identifiers (table and column names), operators, and values.

Q2. Write the general structure of an SQL SELECT statement.

A:

SELECT column1, column2 FROM table_name WHERE condition;

Q3. Explain the role of clauses in SQL statements.

A: Clauses like WHERE, ORDER BY, GROUP BY, etc., are used to refine, group, sort, or filter the data in a query.

Q1. What are constraints in SQL? List and explain the different types.

A: Constraints are rules applied to columns to enforce data integrity. Types include:

- NOT NULL
- UNIQUE
- PRIMARY KEY
- FOREIGN KEY
- CHECK
- DEFAULT

Q2. How do PRIMARY KEY and FOREIGN KEY constraints differ?

A: A PRIMARY KEY uniquely identifies each record in a table, while a FOREIGN KEY establishes a relationship between two tables by referencing the primary key of another table.

Q3. What is the role of NOT NULL and UNIQUE constraints?

A:

NOT NULL: Ensures a column cannot have NULL values.

UNIQUE: Ensures all values in a column are distinct.

✓ Part 4: DDL (Data Definition Language)

Q1. Define the SQL Data Definition Language (DDL).

A: DDL is used to define and modify the structure of database objects like tables, schemas, and indexes using commands like CREATE, ALTER, DROP.

Q2. Explain the CREATE command and its syntax.

A: The CREATE command is used to create database objects.

CREATE TABLE table_name (column1 datatype, column2 datatype, ...);

Q3. What is the purpose of specifying data types and constraints during table creation?

A: It ensures data validity, defines memory/storage allocation, and enforces business rules.

Part 5: ALTER Command

Q1. What is the use of the ALTER command in SQL?

A: ALTER modifies an existing table structure — adds, deletes, or modifies columns.

Q2. How can you add, modify, and drop columns from a table using ALTER?

A:

- Add: ALTER TABLE table_name ADD column_name datatype;
- Modify: ALTER TABLE table_name MODIFY column_name datatype;
- Drop: ALTER TABLE table_name DROP COLUMN column_name;

Part 6: DROP Command

Q1. What is the function of the DROP command in SQL?

A: It permanently deletes a table or database and all associated data.

Q2. What are the implications of dropping a table from a database?

A: All data, structure, and relationships linked to the table are lost permanently and cannot be recovered.

✓ Part 7: DML (Data Manipulation Language)

Q1. Define the INSERT, UPDATE, and DELETE commands in SQL.

A:

- INSERT: Adds new records
- UPDATE: Modifies existing records
- DELETE: Removes records

Q2. What is the importance of the WHERE clause in UPDATE and DELETE operations?

A: It ensures only the intended rows are affected; without WHERE, all rows will be updated or deleted.

Part 8: DQL (Data Query Language)

Q1. What is the SELECT statement, and how is it used to query data?

A: The SELECT statement is used to retrieve data from a table.

SELECT column1, column2 FROM table_name;

Q2. Explain the use of the ORDER BY and WHERE clauses in SQL queries.

A:

• WHERE: Filters rows

ORDER BY: Sorts the result set based on specified columns

✓ Part 9: DCL (Data Control Language)

Q1. What is the purpose of GRANT and REVOKE in SQL?

A: These commands control user access to database objects:

- GRANT: Gives permissions
- REVOKE: Removes permissions

Q2. How do you manage privileges using these commands?

A: By assigning or removing actions like SELECT, INSERT, DELETE on tables to/from users or roles.

V Part 10: TCL (Transaction Control Language)

Q1. What is the purpose of the COMMIT and ROLLBACK commands in SQL?

A:

- COMMIT saves all changes permanently.
- ROLLBACK undoes changes made during the transaction.

Q2. Explain how transactions are managed in SQL databases.

A: Transactions begin with a DML command and can be

controlled using COMMIT, ROLLBACK, and SAVEPOINT to ensure data consistency.

Part 11: SQL Joins

Q1. Explain the concept of JOIN in SQL. What is the difference between INNER, LEFT, RIGHT, and FULL OUTER JOIN?

A: JOIN combines rows from two or more tables.

- INNER JOIN: Only matched records
- LEFT JOIN: All from left + matched right
- RIGHT JOIN: All from right + matched left
- FULL OUTER JOIN: All records from both tables

Q2. How are joins used to combine data from multiple tables?

A: By linking tables using keys and JOIN conditions to create comprehensive result sets.

✓ Part 12: SQL GROUP BY

Q1. What is the GROUP BY clause in SQL? How is it used with aggregate functions?

A: GROUP BY groups rows with the same values, often used with COUNT, SUM, AVG, etc.

Q2. Explain the difference between GROUP BY and ORDER BY.

A:

- GROUP BY groups rows
- ORDER BY sorts rows
- **V** Part 13: Stored Procedure

Q1. What is a stored procedure in SQL, and how does it differ from a standard SQL query?

A: A stored procedure is a named set of SQL statements stored in the database. Unlike ad hoc queries, it can accept parameters and be reused.

Q2. Explain the advantages of using stored procedures. A:

- Reusability
- Performance
- Centralized logic
- Enhanced security

Q1. What is a view in SQL, and how is it different from a table?

A: A view is a virtual table based on a query. It does not store data but shows results dynamically.

Q2. Explain the advantages of using views in SQL databases.

A:

- Simplifies queries
- Adds security
- Hides complexity

✓ Part 15: SQL Triggers

Q1. What is a trigger in SQL? Describe its types and when they are used.

A: A trigger is a procedure that runs automatically in response to table events like INSERT, UPDATE, or DELETE. Types include BEFORE, AFTER, and INSTEAD OF.

Q2. Explain the difference between INSERT, UPDATE, and DELETE triggers.

A:

• INSERT: Fires after row insertion

- UPDATE: Fires when data is modified
- DELETE: Fires when row is deleted
- ✓ Part 16: Introduction to PL/SQL

Q1. What is PL/SQL, and how does it extend SQL's capabilities?

A: PL/SQL is Oracle's procedural extension to SQL, allowing loops, conditions, variables, and procedures for complex logic.

Q2. List and explain the benefits of using PL/SQL. A:

- Modular code
- Exception handling
- Better performance
- Enhanced reusability
- ✓ Part 17: PL/SQL Control Structures

Q1. What are control structures in PL/SQL? Explain the IF-THEN and LOOP control structures.

A:

- IF-THEN: Executes a block if condition is true
- LOOP: Repeats a block multiple times

Q2. How do control structures in PL/SQL help in writing complex queries?

A: They allow dynamic decision-making and iterative processing in programs.

Part 18: SQL Cursors

Q1. What is a cursor in PL/SQL? Explain the difference between implicit and explicit cursors.

A:

- Cursor: A pointer to a query result set
- Implicit: Automatic for single-row SELECTs
- Explicit: User-defined for multi-row SELECTs

Q2. When would you use an explicit cursor over an implicit one?

A: When processing multiple rows individually with custom logic.

Q1. Explain the concept of SAVEPOINT. How do ROLLBACK and COMMIT interact with it?

A: SAVEPOINT sets a temporary point in a transaction.

- ROLLBACK TO SAVEPOINT undoes changes after it
- COMMIT makes all changes permanent and clears savepoints

Q2. When is it useful to use SAVEPOINT in a transaction?

A: In long transactions, to rollback selectively without undoing the entire transaction.

Lab Exercise:-

- ✓ Part 1: Introduction to SQL
 - Lab 1: Create a new database named school_db and a table called students with the following columns: student_id, student_name, age, class, and address.
 - Lab 2: Insert five records into the students table and retrieve all records using the SELECT statement.

- Lab 1: Write SQL queries to retrieve specific columns (student_name and age) from the students table.
- Lab 2: Write SQL queries to retrieve all students whose age is greater than 10.

Part 3: SQL Constraints

- Lab 1: Create a table teachers with the following columns: teacher_id (Primary Key), teacher_name (NOT NULL), subject (NOT NULL), and email (UNIQUE).
- Lab 2: Implement a FOREIGN KEY constraint to relate the teacher_id from the teachers table with the students table.

Part 4: DDL (Data Definition Language)

- Lab 1: Create a table courses with columns: course_id, course_name, and course_credits. Set the course_id as the primary key.
- Lab 2: Use the CREATE command to create a database university_db.

Part 5: ALTER Command

- Lab 1: Modify the courses table by adding a column course_duration using the ALTER command.
- Lab 2: Drop the course_credits column from the courses table.

Part 6: DROP Command

- Lab 1: Drop the teachers table from the school_db database.
- Lab 2: Drop the students table from the school_db database and verify that the table has been removed.

Part 7: DML (Data Manipulation Language)

- Lab 1: Insert three records into the courses table using the INSERT command.
- Lab 2: Update the course duration of a specific course using the UPDATE command.
- Lab 3: Delete a course with a specific course_id from the courses table using the DELETE command.

✓ Part 8: DQL (Data Query Language)

- Lab 1: Retrieve all courses from the courses table using the SELECT statement.
- Lab 2: Sort the courses based on course_duration in descending order using ORDER BY.
- Lab 3: Limit the results of the SELECT query to show only the top two courses using LIMIT.

Part 9: DCL (Data Control Language)

- Lab 1: Create two new users user1 and user2, and grant user1 permission to SELECT from the courses table.
- Lab 2: Revoke the INSERT permission from user1 and give it to user2.

Part 10: TCL (Transaction Control Language)

- Lab 1: Insert a few rows into the courses table and use COMMIT to save the changes.
- Lab 2: Insert additional rows, then use ROLLBACK to undo the last insert operation.

 Lab 3: Create a SAVEPOINT before updating the courses table, and use it to roll back specific changes.

Part 11: SQL Joins

- Lab 1: Create two tables: departments and employees. Perform an INNER JOIN to display employees along with their respective departments.
- Lab 2: Use a LEFT JOIN to show all departments, even those without employees.

Part 12: SQL GROUP BY

- Lab 1: Group employees by department and count the number of employees in each department using GROUP BY.
- Lab 2: Use the AVG aggregate function to find the average salary of employees in each department.

Part 13: SQL Stored Procedure

• Lab 1: Write a stored procedure to retrieve all employees from the employees table based on department.

• Lab 2: Write a stored procedure that accepts course_id as input and returns the course details.

Part 14: SQL View

- Lab 1: Create a view to show all employees along with their department names.
- Lab 2: Modify the view to exclude employees whose salaries are below \$50,000.

Part 15: SQL Triggers

- Lab 1: Create a trigger to automatically log changes to the employees table when a new employee is added.
- Lab 2: Create a trigger to update the last_modified timestamp whenever an employee record is updated.

Part 16: Introduction to PL/SQL

- Lab 1: Write a PL/SQL block to print the total number of employees from the employees table.
- Lab 2: Create a PL/SQL block that calculates the total sales from an orders table.

V Part 17: PL/SQL Control Structures

- Lab 1: Write a PL/SQL block using an IF-THEN condition to check the department of an employee.
- Lab 2: Use a FOR LOOP to iterate through employee records and display their names.

Part 18: SQL Cursors

- Lab 1: Write a PL/SQL block using an explicit cursor to retrieve and display employee details.
- Lab 2: Create a cursor to retrieve all courses and display them one by one.

☑ Part 19: ROLLBACK, COMMIT, SAVEPOINT

- Lab 1: Perform a transaction where you create a SAVEPOINT, insert records, then rollback to the SAVEPOINT.
- Lab 2: Commit part of a transaction after using a SAVEPOINT, and then rollback the remaining changes.