**Module 4 – Introduction to DBMS**

**1. Introduction to SQL**

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

* SQL (Structured Query Language) is a standard language used to manage and manipulate databases. It's essential because it allows users to create, read, update, and delete data efficiently.

2. Explain the difference between DBMS and RDBMS.

* DBMS (Database Management System) is software for storing and managing data; RDBMS (Relational DBMS) is a type of DBMS that stores data in tables with relationships between them using keys.

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

* SQL is used in relational databases to define structures (CREATE), manipulate data (INSERT, UPDATE, DELETE), and query data (SELECT), enabling users to interact with the database.

4. What are the key features of SQL?

* Key features of SQL:
* Data definition and manipulation
* Data retrieval with queries
* Transaction control
* Built-in functions
* Support for data integrity and constraints

**2. SQL Syntax**

1. What are the basic components of SQL syntax?

* Keywords (e.g., SELECT, FROM), identifiers (table/column names), operators (=, >), clauses (WHERE, ORDER BY), and expressions (values, functions).

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

* SELECT column1, column2

FROM table\_name

WHERE condition

ORDER BY column1;

3. Explain the role of clauses in SQL statements.

* Clauses define specific parts of an SQL query (e.g., WHERE filters data, ORDER BY sorts data, GROUP BY groups rows) to control what data is retrieved and how it's processed.

**3. SQL Constraints**

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

* Constraints in SQL are rules applied to table columns to enforce data integrity and accuracy.

Types of constraints:

PRIMARY KEY: Uniquely identifies each record in a table.

FOREIGN KEY: Ensures referential integrity by linking to a primary key in another table.

NOT NULL: Prevents null (empty) values in a column.

UNIQUE: Ensures all values in a column are different.

CHECK: Validates that values meet a specific condition.

DEFAULT: Assigns a default value if none is provided.

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

* PRIMARY KEY uniquely identifies a row in the same table and cannot be NULL.

FOREIGN KEY links to a primary key in another table and can have duplicate or NULL values (depending on design).

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

* NOT NULL ensures that a column must always have a value (no NULLs allowed).  
  UNIQUE ensures that all values in a column (or combination of columns) are distinct.

**4. Data Definition Language (DDL)**

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

* DDL defines the structure of database objects like tables, schemas, etc.

2. Explain the CREATE command and its syntax.

* CREATE:- Is used to create database objects.  
  Syntax:- CREATE TABLE table\_name (column1 datatype, column2 datatype, ...);

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

* Data types and constraints ensure correct data format and maintain data integrity.

**5. ALTER Command**

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

* ALTER modifies the structure of an existing table.

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

* **Add** column: ALTER TABLE table\_name ADD column\_name datatype;

**Modify** column: ALTER TABLE table\_name MODIFY column\_name new\_datatype;

**Drop** column: ALTER TABLE table\_name DROP COLUMN column\_name;

**6. DROP Command**

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

* DROP permanently deletes a table or database.

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

* Implications: All data and structure are lost and cannot be recovered.

**7. Data Manipulation Language (DML)**

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

* INSERT adds new records, UPDATE modifies existing records, DELETE removes records.

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

* WHERE clause filters rows to prevent unintentional changes/deletions.

**8. Data Query Language (DQL)**

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

* SELECT retrieves data from a table.

SELECT column1, column2

FROM table\_name

WHERE condition

ORDER BY column1;

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

* WHERE filters rows based on conditions; ORDER BY sorts result by specified columns.

**9. Data Control Language (DCL)**

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

* GRANT gives privileges; REVOKE removes them.

1. How do you manage privileges using these commands?

* Used to manage access control on tables, views, and procedures.

**10. Transaction Control Language (TCL)**

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

* COMMIT saves changes, ROLLBACK undoes changes.

1. Explain how transactions are managed in SQL databases.

* Transactions ensure all operations complete successfully or none at all (atomicity).

**11. SQL Joins**

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

* JOIN combines rows from two or more tables:
  + **INNER JOIN**: matching rows only
  + **LEFT JOIN**: all rows from left, matched from right
  + **RIGHT JOIN**: all rows from right, matched from left
  + **FULL OUTER JOIN**: all rows from both tables

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

* Joins are used to combine related data from different tables.

**12. SQL GROUP BY**

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

* GROUP BY groups rows with the same values for aggregation (e.g., SUM, COUNT).

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

* GROUP BY groups data; ORDER BY sorts data.

**13. SQL Stored Procedure**

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

* A stored procedure is a precompiled set of SQL statements stored in the database, unlike a standard query which is written and run on the spot.

1. Explain the advantages of using stored procedures.

* Advantages: Improves performance, promotes code reuse, enhances security, and simplifies complex operations.

**14. SQL View**

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

* A view is a virtual table created using a SELECT query; it does not store data itself like a table.

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

* Advantages: Simplifies complex queries, provides data security, and allows logical data abstraction.

**15. SQL Triggers**

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

* A trigger is a block of SQL code automatically executed in response to certain events (INSERT, UPDATE, DELETE).  
  Types: BEFORE, AFTER, INSTEAD OF.

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

* Difference:
  + INSERT trigger: fires on new record addition
  + UPDATE trigger: fires on record modification
  + DELETE trigger: fires on record removal

**16. Introduction to PL/SQL**

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

* PL/SQL is Oracle’s procedural extension of SQL that allows use of variables, loops, and control logic.

2. List and explain the benefits of using PL/SQL.

* Benefits: Supports modular code, improves performance, enhances error handling, and allows procedural logic within SQL.

**17. PL/SQL Control Structures**

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

* Control structures guide the flow of execution.
  + IF-THEN: executes code conditionally
  + LOOP: repeats code blocks

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

* They enable complex decision-making and iterations, making queries dynamic and powerful.

**18. SQL Cursors**

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

* A cursor is a pointer used to process rows one at a time.
  + Implicit cursors are auto-managed by PL/SQL
  + Explicit cursors are manually declared and controlled.

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

* Explicit cursors are preferred when handling multiple rows with detailed control.

**19. Rollback, Commit, and Savepoint**

1. Explain the concept of SAVEPOINT in transaction management. How do ROLLBACK and

COMMIT interact with savepoints?

* A SAVEPOINT marks a point in a transaction to which you can roll back.
  + COMMIT finalizes changes
  + ROLLBACK undoes changes up to a savepoint or entirely.

1. When is it useful to use savepoints in a database transaction?

* Useful for partial rollbacks within large transactions, preserving earlier steps.