**Fundamental concepts of SQL**

**1. Introduction to SQL**

SQL (Structured Query Language) is a domain-specific language used for managing and manipulating relational databases. It is used to query, update, and manage databases. SQL commands are used to perform tasks such as data retrieval, data insertion, data modification, and database structure management.

**2. SQL Database Structure**

A database consists of several components:

\*Tables\*: Stores data in rows and columns.

\*Rows\* (Records): Individual data entries within a table.

\*Columns\*: Fields within a table that store specific data types.

\*Primary Key\*: A unique identifier for each record in a table.

\*Foreign Key\*: A field in one table that uniquely identifies a row of another table.

\*Indexes\*: Improve search performance.

**3. SQL Data Types**

Data types define the type of data that can be stored in a column. Common SQL data types include:

\*INT\*: Integer numbers.

\*VARCHAR\*: Variable-length string.

\*CHAR\*: Fixed-length string.

\*DATE\*: Date value (yyyy-mm-dd).

\*DATETIME\*: Date and time value.

\*BOOLEAN\*: True/False value.

\*FLOAT\*: Floating-point numbers.

**4. SQL Commands**

SQL commands are divided into various categories:

**a) DDL (Data Definition Language)**

DDL commands are used to define and manage database structures.

\*CREATE DATABASE\*: Creates a new database.

sql

CREATE DATABASE my\_database;

\*CREATE TABLE\*: Creates a new table.

sql

CREATE TABLE my\_table (

id INT PRIMARY KEY,

name VARCHAR(100),

age INT

);

\*ALTER TABLE\*: Modifies an existing table.

sql

ALTER TABLE my\_table ADD email VARCHAR(100);

\*DROP TABLE\*: Deletes an existing table.

sql

DROP TABLE my\_table;

\*DROP DATABASE\*: Deletes a database.

sql

DROP DATABASE my\_database;

**b) DML (Data Manipulation Language)**

DML commands are used to manage data within the tables.

\*SELECT\*: Retrieves data from one or more tables.

sql

SELECT name, age FROM my\_table WHERE age > 30;

\*INSERT INTO\*: Adds new data into a table.

sql

INSERT INTO my\_table (name, age) VALUES ('John Doe', 28);

\*UPDATE\*: Modifies existing data.

sql

UPDATE my\_table SET age = 29 WHERE name = 'John Doe';

\*DELETE\*: Removes data from a table.

sql

DELETE FROM my\_table WHERE age < 20;

**c) DCL (Data Control Language)**

DCL commands are used for defining user access and permissions.

\*GRANT\*: Gives a user access privileges.

sql

GRANT SELECT, INSERT ON my\_table TO 'user';

\*REVOKE\*: Removes a user's access privileges.

sql

REVOKE INSERT ON my\_table FROM 'user';

**d) TCL (Transaction Control Language)**

TCL commands are used for managing transactions in SQL.

\*COMMIT\*: Saves all changes made during the current transaction.

sql

COMMIT;

\*ROLLBACK\*: Undoes changes made during the current transaction.

sql

ROLLBACK;

\*SAVEPOINT\*: Creates a point in the transaction to which you can roll back.

sql

SAVEPOINT sp1;

\*SET TRANSACTION\*: Sets the properties of a transaction.

sql

SET TRANSACTION ISOLATION LEVEL SERIALIZABLE;

**5. SQL Query Clauses**

SQL queries often use clauses to filter and manipulate data:

\*WHERE\*: Filters records based on specified conditions.

sql

SELECT \* FROM my\_table WHERE age > 25;

\*ORDER BY\*: Sorts the result set by one or more columns.

sql

SELECT \* FROM my\_table ORDER BY age DESC;

\*GROUP BY\*: Groups rows that have the same values in specified columns.

sql

SELECT COUNT(\*), AVG(age) FROM my\_table GROUP BY age;

\*HAVING\*: Sets a condition for groups created by the GROUP BY clause.

sql

SELECT age, COUNT(\*) FROM my\_table GROUP BY age HAVING COUNT(\*) > 1;

\*LIMIT\*: Limits the number of rows returned by a query.

sql

SELECT \* FROM my\_table LIMIT 10;

**6. SQL Joins**

SQL joins are used to combine records from two or more tables:

\*INNER JOIN\*: Returns rows when there is a match in both tables.

sql

SELECT \* FROM table1 INNER JOIN table2 ON table1.id = table2.id;

\*LEFT JOIN\* (or LEFT OUTER JOIN): Returns all records from the left table and matched records from the right table.

sql

SELECT \* FROM table1 LEFT JOIN table2 ON table1.id = table2.id;

\*RIGHT JOIN\* (or RIGHT OUTER JOIN): Returns all records from the right table and matched records from the left table.

sql

SELECT \* FROM table1 RIGHT JOIN table2 ON table1.id = table2.id;

\*FULL JOIN\* (or FULL OUTER JOIN): Returns records when there is a match in one of the tables.

sql

SELECT \* FROM table1 FULL JOIN table2 ON table1.id = table2.id;

**7. SQL Functions**

SQL supports various built-in functions for manipulating data:

\***Aggregate Functions**\*: Perform calculations on multiple rows of data.

\*COUNT()\*: Counts the number of rows.

\*AVG()\*: Returns the average value.

\*MAX()\*: Returns the maximum value.

\*MIN()\*: Returns the minimum value.

\*SUM()\*: Returns the sum of values.

sql

SELECT COUNT(\*), MAX(age) FROM my\_table;

\***String Functions**\*:

\*CONCAT()\*: Concatenates two or more strings.

\*SUBSTRING()\*: Extracts part of a string.

\*UPPER()\*: Converts a string to uppercase.

sql

SELECT CONCAT(first\_name, ' ', last\_name) FROM my\_table;

**\*Date Functions\*:**

\*NOW()\*: Returns the current date and time.

\*DATEADD()\*: Adds a specified time interval to a date.

\*DATEDIFF()\*: Calculates the difference between two dates.

sql

SELECT NOW(), DATEDIFF('2024-12-09', '2024-01-01');

**8. Subqueries**

A subquery is a query within another query:

\*Subquery in SELECT\*: Used to retrieve a single value.

sql

SELECT name FROM employees WHERE salary > (SELECT AVG(salary) FROM employees);

\*Subquery in WHERE\*: Used to filter rows based on the result of another query.

sql

SELECT \* FROM orders WHERE customer\_id IN (SELECT customer\_id FROM customers WHERE region = 'North');

**9. Indexes**

Indexes are used to speed up search queries:

\*CREATE INDEX\*: Creates an index on a table.

sql

CREATE INDEX idx\_name ON my\_table(name);

\*DROP INDEX\*: Removes an index.

sql

DROP INDEX idx\_name;

**10. Views**

A view is a virtual table based on the result of a SELECT query.

\*CREATE VIEW\*: Creates a view.

sql

CREATE VIEW my\_view AS SELECT name, age FROM my\_table WHERE age > 30;

\*DROP VIEW\*: Removes a view.

sql

DROP VIEW my\_view;