SQL Basics

1. **What is SQL?**
   SQL stands for Structured Query Language, and it's a programming language used for interaction with relational database management systems (RDBMS). This includes fetching, updating, inserting, and removing data from tables.

2. **What are the different types of SQL databases (SQL vs NoSQL)?**

   - **SQL (Relational Databases)**: Use structured data with predefined schemas, support ACID properties, and employ SQL for querying (e.g., MySQL, PostgreSQL, Oracle).
   - **NoSQL (Non-relational Databases)**: Handle unstructured or semi-structured data, offer flexible schemas, and focus on scalability and performance (e.g., MongoDB, Cassandra, Redis).

3. **What is a relational database?**
   A relational database is a type of database that stores and organizes data in a structured manner using tables. Each table consists of:

   - **Rows** (also called records or tuples): Represent individual entries, such as a customer or an order.
   - **Columns** (also called attributes or fields): Represent the properties of those entries, such as customer name or order date.

4. **What are the main applications of SQL?**
   Using SQL, we can:

   - Create, delete, and update tables in a database.
   - Access, manipulate, and modify data in a table.
   - Retrieve and summarize the necessary information from a table or several tables.
   - Add or remove certain rows or columns from a table.

5. **What is a database?**
   A structured storage space where the data is kept in many tables and organized so that the necessary information can be easily fetched, manipulated, and summarized.

6. **What is DBMS, and what types of DBMS do you know?**
   DBMS stands for Database Management System, a software package used to perform various operations on the data stored in a database, such as accessing, updating, wrangling, inserting, and removing data. Types of DBMS include:

   - Relational
   - Hierarchical
   - Network
   - Graph
   - Object-oriented

SQL Queries

7. **What is an SQL query, and what types of queries do you know?**
   A query is a piece of code written in SQL to access or modify data from a database. There are two types of SQL queries:

   - **Select queries**: Used to retrieve the necessary data (including limiting, grouping, ordering, and extracting data from multiple tables).
   - **Action queries**: Used to create, add, delete, update, and rename data.

8. **What is the basic structure of an SQL statement?**
   The basic structure typically includes the following clauses:

   ```
   SELECT column1, column2
   FROM table_name
   WHERE condition
   ORDER BY column1;
   ```

   - **SELECT**: Specifies the columns to retrieve.
   - **FROM**: Indicates the table to query.
   - **WHERE**: Filters records based on specified conditions.
   - **ORDER BY**: Sorts the results based on one or more columns.

9. **What is an SQL statement? Give some examples.**
   An SQL statement is a command used to perform specific operations on a database. Here are a few common examples:

   - **SELECT Statement**:

     ```
     SELECT first_name, last_name FROM employees WHERE department = 'Sales';
     ```

   - **INSERT Statement**:

     ```
     INSERT INTO employees (id, first_name, last_name, department, salary)
     VALUES (1, 'John', 'Doe', 'Sales', 60000.00);
     ```

   - **UPDATE Statement**:

     ```
     UPDATE employees SET salary = 65000.00 WHERE id = 1;
     ```

   - **DELETE Statement**:

     ```
     DELETE FROM employees WHERE id = 1;
     ```

- **CREATE TABLE Statement**:

  ```
  CREATE TABLE employees (
      id INT PRIMARY KEY,
      first_name VARCHAR(50),
      last_name VARCHAR(50),
      department VARCHAR(50),
      salary DECIMAL(10, 2)
  );
  ```

10. **What are primary keys and foreign keys?**

    - **Primary Key**: A unique identifier for each record in a table, ensuring no duplicate entries. It cannot be NULL.

    - **Foreign Key**: A field in one table that refers to the primary key of another table, establishing a relationship between the two tables.

11. **What is a unique key?**
    A unique key is a column (or multiple columns) of a table to which the UNIQUE constraint was imposed to ensure unique values, including a possible NULL value (the only one).

12. **What are indexes and how do they work?**
    Indexes are data structures that improve the speed of data retrieval operations on a database table by providing quick access to rows based on the values in one or more columns.

SQL Command Types

13. **What types of SQL commands (or SQL subsets) do you know?**

    - **Data Definition Language (DDL)**: Defines and modifies the structure of a database.
    - **Data Manipulation Language (DML)**: Accesses, manipulates, and modifies data in a database.
    - **Data Control Language (DCL)**: Controls user access to data and gives or revokes privileges to specific users or groups.
    - **Transaction Control Language (TCL)**: Controls transactions in a database.
    - **Data Query Language (DQL)**: Performs queries on the data to retrieve necessary information.

14. **Examples of common SQL commands of each type:**

    - **DDL**: CREATE, ALTER TABLE, DROP, TRUNCATE, ADD COLUMN
    - **DML**: UPDATE, DELETE, INSERT
    - **DCL**: GRANT, REVOKE
    - **TCL**: COMMIT, SET TRANSACTION, ROLLBACK, SAVEPOINT
    - **DQL**: SELECT

Constraints

15. **What is a constraint, and why use constraints?**
    A constraint is a set of conditions defining the type of data that can be input into each column of a

table. Constraints ensure data integrity in a table and block undesired actions.

16. **What SQL constraints do you know?**

    - **DEFAULT**: Provides a default value for a column.
    - **UNIQUE**: Allows only unique values.
    - **NOT NULL**: Allows only non-null values.
    - **PRIMARY KEY**: Allows only unique and strictly non-null values (NOT NULL and UNIQUE).
    - **FOREIGN KEY**: Provides shared keys between two or more tables.

Joins

17. **What is a join?**
    A clause used to combine and retrieve records from two or multiple tables based on the relationship between the columns of those tables.

18. **What types of joins do you know?**

    - **(INNER) JOIN**: Returns only those records that satisfy a defined join condition in both (or all) tables. It's a default SQL join.
    - **LEFT (OUTER) JOIN**: Returns all records from the left table and those records from the right table that satisfy a defined join condition.
    - **RIGHT (OUTER) JOIN**: Returns all records from the right table and those records from the left table that satisfy a defined join condition.
    - **FULL (OUTER) JOIN**: Returns all records from both (or all) tables, combining left and right joins.

19. **What is NULL value? How is it different from zero or a blank space?**
    A NULL value indicates the absence of data for a certain cell in a table. In contrast, zero is a valid numeric value, and an empty string is a legal string of zero length.

Aggregate Functions

20. **What are aggregate functions? Give examples.**
    Aggregate functions perform calculations on a set of values and return a single value. Examples include:

    - **COUNT()**: Counts rows
    - **SUM()**: Adds values
    - **AVG()**: Calculates average
    - **MAX()**: Finds the maximum value
    - **MIN()**: Finds the minimum value

21. **How do you use string functions such as CONCAT, SUBSTRING, and LENGTH?**

    - **CONCAT**: Combines two or more strings. Example:

      ```
      SELECT CONCAT(first_name, ' ', last_name) FROM employees;
      ```

    - **SUBSTRING**: Extracts a portion of a string. Example:

```sql
SELECT SUBSTRING(name, 1, 3) FROM employees;
```

- **LENGTH**: Returns the length of a string. Example:

```sql
SELECT LENGTH(name) FROM employees;
```

22. **What date functions are available?**
    Common date functions include:

    - **NOW()**: Returns the current date and time.
    - **CURDATE()**: Returns the current date.
    - **DATEDIFF()**: Returns the difference between two dates.

Subqueries

23. **What is a subquery?**
    A subquery is a SQL query nested inside another SQL query. It is used to retrieve data that will be used in the main query and can be employed in various clauses, such as SELECT, FROM, or WHERE, returning single or multiple values.

24. **Examples of Subqueries:**

    - **Using a Subquery in the WHERE Clause:**

```sql
SELECT first_name, last_name FROM employees
WHERE department_id = (SELECT id FROM departments WHERE name =
'Sales');
```

- **Using a Subquery in the FROM Clause:**

```sql
SELECT AVG(salary) FROM (SELECT salary FROM employees WHERE department
= 'Sales') AS sales_salaries;
```

Performance

25. **How to improve SQL query performance?**
    - Use indexes on frequently queried columns.
    - Limit the number of rows returned with a WHERE clause.
    - Optimize JOINs by minimizing the number of joins and filtering data as early as possible.
    - Avoid using SELECT * and specify only the required columns.
    - Regularly analyze and optimize the database.

Transactions

26. **What is a transaction?**
    A transaction is a sequence of one or more SQL operations treated as a single logical unit. A transaction follows the ACID properties:

    - **Atomicity**: Ensures all operations succeed or none at all.
    - **Consistency**: Maintains database integrity.
    - **Isolation**: Ensures transactions are isolated from each other.
    - **Durability**: Ensures the results of a transaction persist after completion.

27. **What are the transaction control commands?**
    Transaction control commands include:

    - **COMMIT**: Saves all changes made during the current transaction.
    - **ROLLBACK**: Reverts changes made during the current transaction.
    - **SAVEPOINT**: Sets a point in a transaction to which you can later roll back.