SQL Server

1. What is Microsoft SQL Server?

Microsoft SQL Server is a **Relational Database Management System (RDBMS)** developed by Microsoft. Its primary function is to store and retrieve data as requested by other software applications. It can run on a single machine or across a network.

The system uses **Transact-SQL** (**T-SQL**) as its primary query language, which is Microsoft's proprietary extension of the standard SQL (Structured Query Language).

2. Core Components

SQL Server is not just a database engine; it's a comprehensive data platform that includes several key services:

- Database Engine: The core service responsible for storing, processing, and securing data. It handles
 transactions, locking, and concurrency.
- **SQL Server Integration Services (SSIS):** A platform for building high-performance data integration and Extract, Transform, Load (ETL) solutions.
- SQL Server Analysis Services (SSAS): Provides data analysis and business intelligence capabilities,
 allowing users to create and manage multidimensional data cubes (OLAP) and tabular models.
- SQL Server Reporting Services (SSRS): A server-based reporting platform used to create, manage, and deliver a wide variety of interactive and printed reports.

3. Common SQL Server Editions

SQL Server comes in several editions tailored for different needs:

• **Enterprise:** The premium, full-featured edition designed for large-scale, mission-critical applications. It includes advanced features for high availability, security, and performance.

- Standard: Provides core database capabilities, reporting, and analytics for mid-tier applications and data marts. It has fewer features and lower capacity limits than Enterprise.
- Express: A free, entry-level edition ideal for learning, development, and small-scale applications. It
 has significant limitations on database size, memory, and CPU usage.
- **Developer:** A free, full-featured edition (identical to Enterprise) licensed only for non-production development and testing environments.

4. Key Concepts in SQL Server

- Database: A structured collection of data. It contains objects like tables, views, stored procedures, and functions.
- **Table:** The primary object for storing data, organized into rows (records) and columns (attributes).
- Schema: A container for database objects. It helps in organizing objects and managing permissions.

 The default schema is dbo (database owner).
- T-SQL (Transact-SQL): The language used to interact with the database. It includes:
 - DDL (Data Definition Language): Commands like CREATE, ALTER, DROP (to define database structure).
 - DML (Data Manipulation Language): Commands like SELECT, INSERT, UPDATE,
 DELETE (to query and modify data).
 - DCL (Data Control Language): Commands like GRANT, REVOKE (to manage permissions).
- Stored Procedure: A pre-compiled set of one or more T-SQL statements saved in the database. It
 improves performance, security, and reusability.
- **Index:** A data structure that improves the speed of data retrieval operations on a database table at the cost of slower writes and increased storage space.

5. Common T-SQL Command Examples

Here are practical examples of the most frequently used Transact-SQL commands for daily database operations.

A. Data Manipulation Language (DML) - Working with Data

Assume we have a table named Employees with columns: EmployeeID, FirstName, LastName, and Department.

1. SELECT (Querying Data) To retrieve all employees from the 'Sales' department:

```
SQL

SELECT EmployeeID, FirstName, LastName
FROM Employees
WHERE Department = 'Sales';
```

2. INSERT (Adding New Data) *To add a new employee record to the table:*

```
SQL

INSERT INTO Employees (EmployeeID, FirstName, LastName, Department)

VALUES (101, 'John', 'Doe', 'Marketing');
```

3. UPDATE (Modifying Existing Data) *To change an existing employee's department:*

```
UPDATE Employees
SET Department = 'Human Resources'
WHERE EmployeeID = 101;
```

4. DELETE (Removing Data) *To remove an employee's record from the table:*

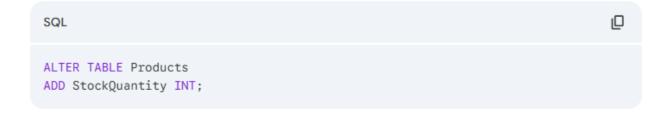
```
SQL

DELETE FROM Employees
WHERE EmployeeID = 101;
```

- B. Data Definition Language (DDL) Defining Database Structure
- **1. CREATE TABLE** *To create a new table to store product information:*

```
CREATE TABLE Products (
    ProductID INT PRIMARY KEY,
    ProductName VARCHAR(100) NOT NULL,
    Price DECIMAL(10, 2),
    DateAdded DATETIME DEFAULT GETDATE()
);
```

2. ALTER TABLE *To add a new column for 'StockQuantity' to the Products table:*



3. DROP TABLE *To completely delete the Products table and all its data (use with caution):*

```
SQL

DROP TABLE Products;
```

C. Stored Procedures and Joins

1. CREATE STORED PROCEDURE To create a reusable command that gets employees by department:

```
CREATE PROCEDURE sp_GetEmployeesByDept
    @DeptName VARCHAR(50)

AS

BEGIN
    SELECT EmployeeID, FirstName, LastName
    FROM Employees
    WHERE Department = @DeptName
    ORDER BY LastName;

END;
```

To use (execute) this procedure:

```
SQL

EXEC sp_GetEmployeesByDept @DeptName = 'Sales';
```

2. JOIN (Combining Data from Multiple Tables) Assume we have a second table Departments (DeptName,

Manager). To get a list of employees and their managers:

```
SELECT
e.FirstName,
e.LastName,
d.Manager

FROM
Employees AS e
INNER JOIN
Departments AS d ON e.Department = d.DeptName;
```