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

SQL, or Structured Query Language, is a standardized programming language used to manage and manipulate relational databases. It allows users to perform various operations such as querying data, updating records, and managing database structures. SQL is essential in database management because it provides a consistent and efficient way to interact with relational databases, ensuring data integrity and facilitating complex data operations.

**2. Explain the difference between DBMS and RDBMS.**

A Database Management System (DBMS) is software that provides an interface to perform various operations like creating, storing, retrieving, and managing data in databases. In contrast, a Relational Database Management System (RDBMS) is a type of DBMS that stores data in a tabular form and establishes relationships between the data points. While DBMS stores data as files without enforcing relationships, RDBMS stores data in tables with enforced relationships, supporting constraints like primary and foreign keys to maintain data integrity.

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

SQL plays a pivotal role in managing relational databases by providing the means to define, manipulate, and control data. It allows users to create and modify database structures (Data Definition Language), insert, update, delete, and retrieve data (Data Manipulation Language), and control access to the data (Data Control Language). Through SQL, users can perform complex queries to extract meaningful insights from large datasets, ensuring efficient data management and retrieval.

**4. What are the key features of SQL?**

SQL offers several key features that make it a powerful tool for database management:

* **Data Querying**: Retrieve specific data from one or more tables using the SELECT statement.
* **Data Manipulation**: Insert, update, and delete records in the database.
* **Data Definition**: Define and modify the structure of database objects such as tables and indexes.
* **Data Control**: Control access to data through permissions and roles.
* **Transaction Management**: Ensure data integrity by grouping a set of operations into a single transaction.
* **Built-in Functions**: Perform calculations and data transformations using aggregate and scalar functions.