1. Key Terms Related to Databases

1. **Database:** A structured collection of data organized for efficient retrieval, storage, and manipulation.

2. **Table:** A data structure within a database that organizes data into rows and columns.

3. **Record:** A single instance or row of data within a table, containing information about a specific entity or object.

4. **Field:** A single piece of data within a record, corresponding to a specific attribute or characteristic.

5. **Primary Key:** A unique identifier for each record in a table, used to ensure data integrity and enable efficient retrieval of specific records.

6. **SQL (Structured Query Language):** A programming language used to manage and manipulate data in relational databases.

7. **Query:** A request for information from a database, typically written in SQL, used to retrieve, update, or manipulate data.

8. **Index:** A data structure that improves the speed of data retrieval operations on a database table by providing quick access to specific rows based on the values of certain columns.

9. **Normalization:** The process of organizing data in a database to minimize redundancy and dependency, ensuring data integrity and optimizing database performance.

10. **Database Management System (DBMS):** Software that enables users to interact with and manage databases, providing tools for data storage, retrieval, manipulation, and security.

2. Discussions

1. **Purpose of a Primary Key:** A primary key uniquely identifies each record in a table, ensuring data integrity by preventing duplicate or null values in the key field. For example, in a "Students" table, the "Student ID" field can serve as the primary key, ensuring that each student has a unique identifier.

2. **Difference between DBMS and Database:** A database is a structured collection of data, while a DBMS is software that manages and facilitates access to the database. In other

words, a database is the repository of data, while a DBMS provides the tools and functionality to store, retrieve, and manipulate that data.

3. **Importance of Normalization:** Normalization reduces data redundancy and dependency, improving data integrity and database efficiency. For example, by breaking down a "Customer" table into separate "Customers" and "Orders" tables and linking them with a foreign key, normalization ensures that each piece of data is stored only once, reducing the risk of inconsistencies and anomalies.