## **Assignment Answers**

### 

### **1. Components of a DBMS (Database Management System)**

A **DBMS** consists of several key components that work together to efficiently manage, store, and retrieve data. These components are:

* **Database Engine**: This is the core component that is responsible for storing, modifying, and retrieving data from the database. It handles queries and updates to the database and ensures that data integrity and security are maintained.
* **Database Schema**: The schema defines the structure of the database, including tables, columns, data types, and relationships between tables. It is essentially the blueprint of the database.
* **Query Processor**: This component interprets and executes SQL queries. It translates the user's requests into actions that the database engine can understand and execute.
* **Data Dictionary**: The data dictionary stores metadata about the database, such as information about tables, columns, constraints, and relationships. It is used by the DBMS to ensure the structure and integrity of the database.
* **Transaction Manager**: This component manages database transactions, ensuring that they are processed reliably and consistently. It handles tasks like rolling back incomplete transactions to maintain data integrity.
* **Security Manager**: This component manages user access to the database and ensures that data is kept secure. It controls permissions, authentication, and authorization.
* **Backup and Recovery Manager**: This component ensures that the database can be backed up and restored if needed. It protects data from loss due to system failures.

### **2. What is a Relational Database? Give 4 Examples.**

A **relational database** is a type of database that stores data in tables (also called relations). These tables consist of rows (records) and columns (fields), where each row represents a unique entity and each column represents an attribute of that entity. Data in different tables are related using **foreign keys**.

#### **Four examples of relational databases:**

* **MySQL**: An open-source relational database management system widely used for web applications.
* **PostgreSQL**: A powerful open-source relational database known for its advanced features and standards compliance.
* **Oracle Database**: A commercial relational database used by enterprises for large-scale data management.
* **Microsoft SQL Server**: A relational database management system developed by Microsoft, commonly used in business applications.

### **3. Three Classifications of SQL**

SQL (Structured Query Language) can be classified into three main categories based on the type of operations they perform:

* **DQL (Data Query Language)**: SQL commands used to query or retrieve data from the database. The most common DQL command is:
  + **SELECT**: Retrieves data from one or more tables.
* **DML (Data Manipulation Language)**: SQL commands used to manipulate the data within the database. These commands are used to insert, update, and delete data:
  + **INSERT**: Adds new data into a table.
  + **UPDATE**: Modifies existing data.
  + **DELETE**: Removes data from a table.
* **DDL (Data Definition Language)**: SQL commands used to define or modify the structure of the database, such as creating or altering tables:
  + **CREATE**: Creates a new table or database.
  + **ALTER**: Modifies an existing database object.
  + **DROP**: Deletes an existing database object like a table.

### **4. Difference Between a Primary Key and a Foreign Key**

* **Primary Key**:
  + A **primary key** is a field (or a combination of fields) in a table that uniquely identifies each row in that table. It ensures that no two rows have the same values for the primary key field.
  + **Characteristics**: Must be unique for every row and cannot be null.
  + **Example**: In a "Customers" table, the customer\_id might be the primary key.
* **Foreign Key**:
  + A **foreign key** is a field (or a combination of fields) in a table that links to the primary key in another table. It creates a relationship between two tables.
  + **Characteristics**: A foreign key can have duplicate values, and it can be null if the relationship allows it.
  + **Example**: In an "Orders" table, the customer\_id could be a foreign key that references the customer\_id in the "Customers" table.

### **5. What is an Entity-Relationship Diagram (ERD)?**

An **Entity-Relationship Diagram (ERD)** is a visual representation of the entities (tables) in a database and the relationships between them. It shows how entities interact with one another through relationships, typically represented by lines connecting entities. ERDs use symbols like rectangles for entities, diamonds for relationships, and ovals for attributes. They help in designing and understanding the structure of a database.

### **6. Advantages of Relational Databases**

* **Data Integrity**: Relational databases enforce rules (such as primary and foreign keys) that ensure data consistency and accuracy.
* **Flexibility**: Data can be easily modified, added, or deleted using SQL queries.
* **Normalization**: Relational databases support normalization, which helps reduce data redundancy and improve storage efficiency.
* **Data Security**: DBMSs offer features such as user access control and encryption to protect sensitive data.
* **Scalability**: Many relational databases can handle large amounts of data and scale as needed.

### **7. Four Types of Data Types Used to Store Data in Tables**

* **INT**: Used to store integers (whole numbers).
* **VARCHAR**: Used to store variable-length strings (text).
* **DATE**: Used to store date values (e.g., YYYY-MM-DD).
* **DECIMAL**: Used to store exact numeric values with decimal points (e.g., for financial data).

### **8. Purpose of a Database Management System (DBMS)**

The purpose of a **Database Management System (DBMS)** is to efficiently manage and organize large amounts of data. It provides an interface for users to interact with the database, perform queries, and manage the data. A DBMS ensures data integrity, security, and consistency, and it supports the efficient retrieval, updating, and deletion of data. Additionally, it handles transaction management, concurrency control, and backup/recovery processes. Essentially, the DBMS acts as an intermediary between users and the database, making it easier to store, manipulate, and retrieve data in a structured way.