1. What is a database? Explain with an example on why should we need a database.

A **database** is an organized collection of data that is stored and accessed electronically. Databases allow for the efficient retrieval, insertion, and deletion of data. They are managed by Database Management Systems (DBMS).

Example: Imagine an online bookstore. The store needs to keep track of its inventory, customer orders, and payment details. All this data is stored in a database. If a customer buys a book, the database updates the inventory and records the sale. Without a database, managing and retrieving this information would be inefficient and error-prone.

Why do we need a database?

- **Efficiency:** Databases allow for efficient data management, making it easy to retrieve, update, and manage large amounts of data.
- **Data Integrity:** Databases help ensure data is accurate and consistent.
- **Security:** Sensitive data can be protected through authentication and authorization.
- **Scalability:** Databases can handle large volumes of data as the system grows.

---

2. Write a short note on file-based storage system. Explain the major challenges of a file-based storage system.

A **file-based storage system** is a system where data is stored in files within a directory structure. Each file can contain various types of data, and access to the files is managed by the operating system.

Major Challenges:

1. **Data Redundancy:** The same data might be stored in multiple files, leading to unnecessary duplication and wasted storage space.
2. **Data Inconsistency:** If the same data is stored in multiple places, updating it in one file might not update it in others, leading to inconsistencies.
3. **Lack of Data Security:** File-based systems generally lack robust security features. Anyone with access to the files can read, modify, or delete them.
4. **Limited Data Sharing:** File-based systems are not designed for easy sharing of data between different users or systems.
5. **Difficulty in Accessing Data:** Retrieving specific information from a large number of files can be time-consuming and error-prone.

---

3. What is DBMS? What was the need for DBMS?

A **Database Management System (DBMS)** is software that interacts with the user, applications, and the database itself to capture and analyze data. A DBMS provides tools for data storage, retrieval, and management.

Need for DBMS:

- To address the limitations of file-based systems, such as data redundancy, inconsistency, and lack of security.
- To manage large volumes of data efficiently.
- To ensure data integrity and consistency across multiple users and applications.
- To support complex queries and data manipulation operations that would be difficult to implement with file-based systems.

---

4. Explain 5 challenges of the file-based storage system which was tackled by DBMS?

1. **Data Redundancy:** DBMS reduces redundancy by normalizing the data, ensuring that each piece of data is stored only once.
2. **Data Inconsistency:** DBMS ensures that updates to data are consistent across the system, preventing discrepancies between different data files.
3. **Data Security:** DBMS offers advanced security features such as access control, authentication, and encryption to protect data.
4. **Data Integrity:** DBMS enforces rules and constraints to maintain the accuracy and consistency of data.
5. **Data Sharing:** DBMS allows multiple users to access and manipulate data simultaneously without conflicts, thanks to transaction management.

---

5. List out the different types of classification in DBMS and explain?

DBMS classifications can be done based on several criteria:

1. **Based on Data Model:**
   - **Hierarchical DBMS:** Data is organized in a tree-like structure.
   - **Network DBMS:** More flexible than hierarchical, allows multiple parent-child relationships.
   - **Relational DBMS (RDBMS):** Data is organized in tables (relations). This is the most common type.
   - **Object-Oriented DBMS:** Stores data as objects, similar to object-oriented programming.
2. **Based on Number of Users:**
   - **Single-user DBMS:** Supports one user at a time.
   - **Multi-user DBMS:** Supports multiple users simultaneously.
3. **Based on Distribution:**
   - **Centralized DBMS:** All data is stored and managed on a single location.
   - **Distributed DBMS:** Data is distributed across multiple locations but appears as a single database to the user.
4. **Based on Usage:**
   - **OLTP (Online Transaction Processing) DBMS:** Optimized for transactional data.
   - **OLAP (Online Analytical Processing) DBMS:** Optimized for querying and reporting.

---

6. What is the significance of Data Modelling and explain the types of Data Modelling?

Data Modelling is the process of creating a visual representation of either a whole information system or parts of it to communicate connections between data points and structures.

Significance:

- **Clarifies Data Structure:** Helps in understanding the relationships and flow of data within the system.
- **Improves Database Design:** Ensures that the database structure is logical and efficient.
- **Facilitates Communication:** Provides a blueprint for developers and stakeholders to discuss and refine.
- **Ensures Data Integrity:** Helps in identifying and implementing constraints and rules for data.

Types of Data Modelling:

1. **Conceptual Data Modelling:** High-level model, defines what data should be included in the system. It focuses on the user's view of the data.
2. **Logical Data Modelling:** Details the structure of the data elements and relationships between them. It includes entities, attributes, and relationships.
3. **Physical Data Modelling:** Describes how the data will be physically stored in the database, including tables, indexes, and partitions.

---

7. Explain 3 schema architecture along with its advantages?

Three-Schema Architecture is a framework that separates a database system into three levels:

1. **Internal Level:** This is the physical storage level where the actual data is stored in the database. It deals with the physical storage of data on devices.
2. **Conceptual Level:** This is the logical level where the structure of the entire database is defined. It is independent of any physical considerations and focuses on data models and relationships.
3. **External Level:** This is the view level where users interact with the data. It involves user-specific views of the data, customized to their needs.

Advantages:

- **Data Abstraction:** Separates the physical storage from the user’s perspective, allowing for flexibility in managing changes.
- **Improved Security:** Each user only accesses the external schema relevant to their role, protecting the underlying data.
- **Easier Maintenance:** Changes at one level (like the physical storage) do not require changes at the other levels.