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

A **database** is a structured collection of data that is organized in such a way that it can be easily accessed, managed, and updated. Databases are used to store vast amounts of data and allow users to perform operations such as creating, reading, updating, and deleting data (CRUD operations).

Example: Consider an online e-commerce website. Each product, customer, order, and transaction needs to be stored systematically. If there is no database, storing this information becomes chaotic, and retrieving customer orders or searching products becomes inefficient. A database like MySQL or Oracle allows the website to store and retrieve data efficiently and reliably, ensuring that customers have a smooth experience while shopping.

Why we need a database:

- **Data Management:** It provides a central place to manage and organize data systematically.
- **Data Security:** Offers a secure way to store sensitive information such as passwords and personal details.
- **Data Integrity:** Ensures consistency and accuracy of data.
- **Scalability:** Easily allows handling increasing volumes of data.

---

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 way of organizing and storing data in individual files that are managed directly by the operating system. These files could be in formats like text, binary, CSV, or Excel files, and they are often used when only small-scale storage is required, and no complex relationships exist between the data elements.

Major Challenges of a File-Based Storage System:

1. **Data Redundancy:** Data is often duplicated across files leading to wastage of storage and inconsistency.
2. **Data Inconsistency:** Due to duplication, changes in one file may not reflect in others, leading to inconsistencies.
3. **Difficulty in Access:** Searching and retrieving data from multiple files can be time-consuming and inefficient.
4. **Lack of Security:** File systems typically don't offer robust mechanisms to ensure that unauthorized users cannot access or modify the data.
5. **Limited Query Capabilities:** File systems lack sophisticated querying options that allow users to search data efficiently.

---

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

A **Database Management System (DBMS)** is software designed to store, retrieve, manage, and manipulate data in databases. It provides a systematic and organized environment to handle large amounts of data, ensuring that data is consistent, accurate, secure, and easily accessible.

Need for DBMS:

- **Data Redundancy Control:** DBMS helps in eliminating duplication of data by centralizing the storage and updating of data.
- **Data Integrity:** It ensures that the data stored is accurate and consistent across the database.
- **Efficient Data Access:** It offers efficient query handling and data retrieval options.
- **Data Security:** Allows controlled access to data, ensuring that unauthorized users cannot access sensitive information.
- **Backup and Recovery:** DBMS systems provide robust backup and recovery mechanisms, protecting against data loss.

---

4. Explain 5 challenges of file-based storage systems that were tackled by DBMS.

1. **Data Redundancy and Inconsistency:** In file-based systems, the same data may be duplicated in several files. DBMS centralizes the data storage, reducing redundancy and inconsistency.
2. **Difficulty in Data Access:** Searching for data in multiple files can be tedious. DBMS uses structured querying (SQL) to provide efficient data access.
3. **Lack of Data Security:** File systems have limited options for securing data, whereas DBMS allows for fine-grained access control, ensuring that only authorized users can access specific data.
4. **Data Isolation:** In file-based systems, data is scattered across various files, making integration difficult. DBMS integrates all data in one place, ensuring smooth relationships between datasets.
5. **Data Integrity Problems:** File systems struggle to enforce data integrity. DBMS ensures that data constraints, such as primary keys and foreign keys, maintain data integrity across the system.

---

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

DBMS can be classified based on various criteria:

1. **Based on Data Model:**
    - **Hierarchical DBMS:** Organizes data in a tree-like structure with a single parent-child relationship. (e.g., IBM IMS)
    - **Network DBMS:** Uses a graph structure allowing more complex relationships. (e.g., Integrated Data Store (IDS))

- **Relational DBMS (RDBMS):** Stores data in tables and uses relationships between them (e.g., MySQL, Oracle).
- **Object-Oriented DBMS:** Stores data in the form of objects, similar to object-oriented programming (e.g., db4o, ObjectDB).

2. **Based on Number of Users:**
   - **Single-user DBMS:** Supports one user at a time (e.g., Microsoft Access).
   - **Multi-user DBMS:** Allows multiple users to access the system simultaneously (e.g., Oracle, SQL Server).
3. **Based on the Location of Data:**
   - **Centralized DBMS:** Stores data at a single location.
   - **Distributed DBMS:** Data is stored across different physical locations but managed as a single database.
4. **Based on Data Access:**
   - **Online Transaction Processing (OLTP):** Handles transaction-oriented applications where many users perform short and frequent transactions.
   - **Online Analytical Processing (OLAP):** Used for complex queries, often in data warehouses, to support decision-making.

---

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 a complex system's data. It defines how data is structured, the relationships between different data elements, and the rules governing data consistency. Data modeling is significant because it helps to ensure that the database system will meet both current and future requirements efficiently and with minimal redundancy.

Types of Data Modelling:

1. **Conceptual Data Model:** Focuses on the high-level design of the data, capturing key entities and relationships between them. It is a blueprint for the overall structure of the database.
2. **Logical Data Model:** Provides detailed descriptions of data without considering how it will be physically stored. It includes attributes, relationships, and constraints.
3. **Physical Data Model:** Focuses on how the data will be stored in a database. It deals with tables, indexes, and storage allocation mechanisms.

---

7. Explain 3-schema architecture along with its advantages.

The **3-schema architecture** in DBMS is a framework that divides the database system into three levels:

1. **Internal Schema (Physical Level):** Defines how the data is physically stored in the database, including storage space, indexing, and data structures.

2. **Conceptual Schema (Logical Level):** Defines what data is stored and the relationships between the data without worrying about the physical storage.
3. **External Schema (View Level):** Defines how the data is viewed by end users or specific applications. Multiple views can be created for different users, hiding unnecessary details.

Advantages of 3-Schema Architecture:

- **Data Abstraction:** It separates the physical and logical aspects of data, allowing changes in one level without affecting the others.
- **Data Independence:** Changes in the internal schema don't affect the conceptual schema or external schemas, providing logical and physical data independence.
- **Multiple Views:** Allows the creation of multiple views for different users, providing customized access to data depending on user roles or application needs.