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

* A database is information that is set up for easy access, management and updating. Computer databases typically store aggregations of data records or files that contain information, such as sales transactions, customer data, financials and product information.
* Databases are used for storing, maintaining and accessing any sort of data. They collect information on people, places or things. That information is gathered in one place so that it can be observed and analyzed. Databases can be thought of as an organized collection of information.
* Here's an example of a customer database for an online retail store:
* The customer database consists of a table with columns like CustomerID, FirstName, LastName, Email, and Address.
* Each row in the table represents a unique customer entry.
* CustomerID serves as the primary key, ensuring each customer has a unique identifier.
* The FirstName and LastName columns store the customer's name.
* The Email column stores the customer's email address for communication purposes.
* The Address column stores the customer's shipping address.
* The database allows the store to efficiently retrieve customer information based on their ID or search for specific customers by name or email.
* It also enables the store to track customer orders and link them to their respective profiles.
* The database can be used to analyze customer behavior, such as purchase history and preferences, for targeted marketing campaigns.
* Regular backups and data security measures ensure the safety and integrity of customer data.

1. **Write a short note on File base storage system. Explain the major challenges of a File-based storage system.**

* A file-based storage system is a traditional method of storing data in individual files. It is simple and familiar, but it suffers from challenges such as data redundancy, inconsistency, and limited scalability. Managing data integrity, security, and sharing can be difficult in file-based systems compared to databases. They lack built-in mechanisms for enforcing constraints and may not be optimized for handling large volumes of data efficiently.
* A file-based storage system faces major challenges such as data redundancy, inconsistency, and limited data integrity. Sharing data between applications becomes complex, scalability becomes an issue with increasing data volumes, and data security measures are often lacking. Data recovery from system failures or corruption can be difficult, and integrating data from multiple sources becomes complex due to the lack of standardized formats. Inefficient data access and slower retrieval times are common, and maintaining and backing up files require manual effort. Overall, file-based systems struggle with maintaining data consistency, ensuring data integrity, and providing efficient data management and retrieval capabilities compared to modern database systems.

1. **What is DBMS? What was the need for DBMS?**

* Database management system is a software which is used to manage the database. For example: MySQL, Oracle, etc are a very popular commercial database which is used in different applications.DBMS provides an interface to perform various operations like database creation, storing data in it, updating data, creating a table in the database and a lot more.It provides protection and security to the database. In the case of multiple users, it also maintains data consistency.
* DBMS centralized data storage and management.
* It ensures data integrity and provides security features.
* DBMS facilitates data sharing and integration.
* It offers data independence for easier modifications.
* Concurrent access control ensures data consistency.
* Powerful query languages enable efficient data retrieval and analysis.
* DBMS supports scalability and optimized performance.
* It includes data recovery and backup mechanisms.

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

Here are five challenges of the file-based storage system

* Data Redundancy and Inconsistency: In a file-based system, data duplication and inconsistencies were common due to the lack of centralized control. DBMS eliminated redundancy by providing a centralized database where data is stored in a structured manner, reducing redundancy and ensuring data consistency.
* Data Integrity and Security: File-based systems lacked built-in mechanisms for enforcing data integrity constraints and implementing robust security measures. DBMS introduced features like referential integrity, data validation rules, and access control mechanisms to ensure data integrity and enhance data security.
* Data Sharing and Integration: File-based systems made data sharing and integration complex, as each application had its own separate files and formats. DBMS enabled efficient data sharing and integration by providing a standardized format, a common data repository, and query languages to retrieve and manipulate data from multiple applications.
* Concurrent Access and Data Consistency: In a file-based system, concurrent access to data by multiple users or applications often led to data inconsistencies or conflicts. DBMS implemented concurrency control mechanisms to handle simultaneous data access, ensuring data consistency and preventing data conflicts.
* Data Scalability and Performance: As data volumes grew in file-based systems, managing and organizing numerous files became complex, leading to performance issues. DBMS introduced optimized data storage structures, indexing techniques, and query optimization to handle large volumes of data efficiently and deliver improved performance.

1. **List out the different types of classification in DBMS and explain them in depth.**

* **Classification of Database Management System**
* Based on Data Model
* Based on Number of Users
* Based on Database Distribution
* Based on Cost of Database
* Based on Usage
* Based on Flow Control
* Based on Data Model:
* Relational Databases: They organize data into tables with predefined relationships, making them suitable for structured data and complex querying.
* Hierarchical Databases: They arrange data in a parent-child tree-like structure, allowing for one-to-many relationships but limiting flexibility.
* Network Databases: They extend the hierarchical model by enabling multiple parent-child relationships, enhancing data flexibility.
* Object-Oriented Databases: They store data as objects with associated methods, supporting complex data structures and object-oriented programming paradigms.
* NoSQL Databases: They are designed for unstructured or semi-structured data, providing flexibility, horizontal scalability, and high performance.
* Based on Data Distribution:
* Centralized Databases: Data is stored and managed in a single location or server, simplifying administration but risking a single point of failure.
* Distributed Databases: Data is distributed across multiple servers or locations, offering scalability, fault tolerance, and reduced network traffic.
* Federated Databases: They integrate multiple autonomous databases into a unified logical database, enabling transparent access to data from various sources.
* Based on Application Domain:
* Operational Databases: These databases handle day-to-day transactional operations in real-time, supporting data modifications and concurrent access.
* Analytical Databases: They focus on data analysis and complex queries, enabling business intelligence, reporting, and decision-making.
* Data Warehouses: These large-scale databases consolidate data from multiple sources for analysis, reporting, and decision support, often used in business intelligence applications.
* Each classification has its own characteristics and benefits, allowing organizations to choose the most suitable database type based on their data structure, distribution requirements, and intended use cases.

1. **What is the significance of Data Modelling and explain the types of data modeling.**

* The significance of data modeling lies in its ability to structure and organize data, provide a clear representation of data relationships, facilitate requirement analysis, and ensure alignment with business needs. It serves as a blueprint for database design, enabling efficient data management and communication among stakeholders.
* There are three main types of data modeling:
* Conceptual Data Modeling: Focuses on high-level understanding and representation of the business concepts, entities, and relationships. It helps in capturing the essence of the data requirements without concerns about technical implementation details.
* Logical Data Modeling: Translates the conceptual model into a more detailed representation that is independent of any specific database management system. It defines entities, attributes, relationships, and constraints in a structured manner to serve as a foundation for database design.
* Physical Data Modeling: Focuses on the implementation aspects of data modeling. It defines how the logical data model is transformed into a specific database system, including data types, indexes, partitions, and storage considerations.

1. **Explain 3 schema architecture along with its advantages.**

* **External Schema:**
* Represents the database view from the perspective of individual users or applications.Customizes the view by defining specific data elements and structures for each user.
* Advantages: Provides customized views, ensures data independence, and supports data security.
* **Conceptual Schema:**
* Represents the overall logical structure of the entire database system.
* Defines entities, relationships, and constraints without specifying physical implementation details.
* Advantages: Global view of the database, data integration, effective communication, data independence.
* **Internal Schema:**
* Represents the physical storage and implementation details of the database system.Defines how data is stored, indexed, and accessed in the underlying storage medium.
* Advantages: Optimizes data storage and retrieval, ensures efficient performance, and provides data security and integrity measures.