1. What is a relational database management system (RDBMS)? What are the advantages of a database management system over a file system?

Ans:

* The software used to store, manage, query, and retrieve data stored in a relational database is called a relational database management system (RDBMS). It serves as an interface between the database and end users or application programs.

Advantages of a database management system over a file system:

* Data Redundancy: DBMS controls redundancy by maintaining a single repository of data that is defined once and is accessed by many users whereas the file system cannot control redundancy of data as each user defines and maintains the needed files for a specific application to run.
* Data Consistency: Date remains consistent with DBMS.
* Data sharing: File system does not allow sharing of data or sharing is too complex. Whereas in DBMS, data can be shared easily due to centralized system.
* Data Concurrency: File system does not provide any procedure to stop anomalies. Whereas DBMS provides a locking system to stop anomalies to occur.
* Data Searching: For every search operation performed on file system, a different application program has to be written. While DBMS provides inbuilt searching operations. User only have to write a small query to retrieve data from database.
* Data Security: A file system provides a password mechanism to protect the database but DBMS has specialized features that help provide shielding to its data.
* In file system, there wont be recovery of data when the system crashes but A DBMS will have the recovery manager which retrieves the data.

1. In a database management system, explain the ACID properties.

**Atomicity**:

the entire transaction takes place at once or doesn’t happen at all. transactions do not occur partially. Each transaction is considered as one unit and either runs to completion or is not executed at all. It involves the following two operations.   
 **Abort**: If a transaction aborts, changes made to database are not visible.   
 **Commit**: If a transaction commits, changes made are visible.

**Consistency**:  
This means that integrity constraints must be maintained so that the database is consistent before and after the transaction. It refers to the correctness of a database.

**Isolation**:  
This property ensures that multiple transactions can occur concurrently without leading to the inconsistency of database state. Transactions occur independently without interference. Changes occurring in a particular transaction will not be visible to any other transaction until that particular change in that transaction is written to memory or has been committed.

**Durability:**   
This property ensures that once the transaction has completed execution, the updates and modifications to the database are stored in and written to disk and they persist even if a system failure occurs. These updates now become permanent and are stored in non-volatile memory. The effects of the transaction, thus, are never lost.

1. Explain the concept of normalization.

Normalization: It is the process of minimizing redundancy from a relation or set of relations. Redundancy in relation may cause insertion, deletion, and update anomalies. So, it helps to minimize the redundancy in relations.

First Normal Form: In this Normal Form, we tackle the problem of atomicity. Here atomicity means values in the table should not be further divided. In simple terms, a single cell cannot hold multiple values. If a table contains a composite or multi-valued attribute, it violates the First Normal Form.

Second Normal Form: The first condition in the 2nd NF is that the table has to be in 1st NF. The table also should not contain partial dependency. Here partial dependency means the proper subset of candidate key determines a non-prime attribute.

**3rd Normal Form:** the table has to be in 2NF before proceeding to 3NF. The other condition is there should be no transitive dependency for non-prime attributes. That means non-prime attributes (which doesn’t form a candidate key) should not be dependent on other non-prime attributes in a given table.

1. Explain the many types of query languages used in relational databases. DQL, DML, DCL, and DDL are some examples.

There are 4 types of query languages in databases. They are,

* 1. Data query language: Ex: select

* 1. Data Definition Language: Ex: create, alter and drop
  2. Data Manipulation Language: Ex: Insert, update and delete
  3. Date Control Language: Grant and Revoke.

1. What is the difference between the main key and a composite key? Give instances of how primary key and composite are used.

Ans:

A **PRIMARY KEY** constraint uniquely identifies each record in a table. A Primary keys column must contain unique values and cannot have null values. A table can have only one primary key, which may consist of single or multiple columns.

A **COMPOSITE KEY** is a combination of two or more columns in a table that can be used to uniquely identify each row in the table when the columns are combined uniqueness is guaranteed, but when it taken individually it does not guarantee uniqueness. So basically, primary key becomes the composite key when more than column is used to uniquely identify each row in the table.