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

A relational database management system (RDBMS) is, as the name suggests, software that manages a relational database. It is a set of smaller programs designed to work together, allowing the developer to store, access, and modify data in tables, transparently and without having to know where the data is physically stored on disk. You can interact with any relational database management system by using the SQL programming language. SQL stands for “Structured Query Language”, which allows you to interrogate structured data stored in tables. Advantages of DBMS over File system : **Data redundancy and inconsistency** – Redundancy is the concept of repetition of data i.e. each data may have more than a single copy. The file system cannot control redundancy of data as each user defines and maintains the needed files for a specific application to run. There may be a possibility that two users are maintaining same files data for different applications. Hence changes made by one user does not reflect in files used by second users, which leads to inconsistency of data. \*\*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 – \*\*

Concurrent access to data means more than one user is accessing the same data at the same time. Anomalies occur when changes made by one user gets lost because of changes made by other user. 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 integrity – \*\*

There may be cases when some constraints need to be applied on the data before inserting it in database. The file system does not provide any procedure to check these constraints automatically. Whereas DBMS maintains data integrity by enforcing user defined constraints on data by itself

.

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

The ACID principles are a foundation for all RDBMSs. They are composed of the following rules.

“A” stands for “atomicity.” It guarantees that either all the changes made by a user are applied to the table or no changes are applied at all. “C” stands for “consistency.” It guarantees that the execution of a transaction keeps the database and its data in a consistent state. “I” stands for “isolation.” It allows for multiple transactions to execute at the same time with transparency to other users, provided they do not change the same data. “D” stands for “durability.” It ensures that once a transaction is committed and applied, it will remain committed even if the database shuts down.

1. **Explain the concept of normalization.**

Apply the so-called normalization rules to check whether your database is structurally correct and optimal. First Normal Form (1NF): A table is 1NF if every cell contains a single value, not a list of values. This property is known as atomic. 1NF also prohibits a repeating group of columns such as item1, item2, itemN. Instead, you should create another table using a one-to-many relationship.

Second Normal Form (2NF) − A table is 2NF if it is 1NF and every non-key column is fully dependent on the primary key. Furthermore, if the primary key is made up of several columns, every non-key column shall depend on the entire set and not part of it.

For example, the primary key of the OrderDetails table comprising orderID and productID. If unitPrice is dependent only on productID, it shall not be kept in the OrderDetails table (but in the Products table). On the other hand, if the unit price is dependent on the product as well as the particular order, then it shall be kept in the OrderDetails table.

Third Normal Form (3NF) − A table is 3NF if it is 2NF and the non-key columns are independent of each other. In other words, the non-key columns are dependent on primary key, only on the primary key and nothing else. For example, suppose that we have a Products table with columns productID (primary key), name and unitPrice. The column discountRate shall not belong to the Products table if it is also dependent on the unitPrice, which is not part of the primary key.

Higher Normal Form: 3NF has its inadequacies, which leads to a higher Normal form, such as Boyce/Codd Normal form, Fourth Normal Form (4NF) and Fifth Normal Form (5NF)

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

An RDBMS actually has a more extended API that allows the developer to interact with it and perform multiple types of operations. These operations for interacting with the data are split into four major categories: DDL, DCL, DML, and DQL.

**DDL Operations** DDL operations are the first type of operations you will execute on your database through the RDBMS once it is installed.

DDL stands for “data definition language.” It is a list of SQL commands you can use not only to create your first objects in a database (such as tables) but also to make changes (such as adding new columns). They are not limited to creating tables; they can create other objects in a database as well.

The DDL statements begin with the same common keywords:

CREATE is used to create any type of database object, for example, tables. DROP is used to delete any database object including tables, users, functions, etc. ALTER is used to modify any object in a database. TRUNCATE is used to remove all the data from a table. **DQL Operations** DQL operations are executed to query the data from the tables. DQL stands for “data query language.” There is only one keyword used for DQL operations: SELECT.

Most of the queries you will write when interacting with a database will be SELECT queries. They take data from the tables and return it to the application from which the database was queried, be it your IDE or a website.

When an RDBMS executes an SQL query, it tries to return the data in the most efficient way possible by minimizing the resources used by the database to find the data on disk. It then retrieves it into memory and finally sends it back to your application.

But DQL operations aren’t limited to just returning data. You can also ask to do computations, like calculating the average hours worked by an employee per month or even multiplying the values between two or more columns and return the result.

**DML Operations** DML stands for “data manipulation language.” DML operations in an RDBMS are used to modify the data already stored in tables and use the following types of commands:

INSERT is a command used to insert data into tables. UPDATE is a command executed on a table to allow the user to modify values in columns. DELETE is a command that allows the user to delete entire rows from a table. **DCL Operations** DCL stands for “data control language.” DCL operations in an RDBMS are specifically targeted for the administrator of the database. They provide user management features, allowing the administrator to set different combinations of permissions for each user who accesses the database. The DCL commands are the following.

GRANT is a command that allows the administrator to grant privileges to a user for accessing or even modifying database objects. REVOKE is a command that allows the administrator to remove privileges from a user, thus restricting the types of operations the user can perform.

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

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. SYNTAX

create table table\_name ( column1 datatype1, column1 datatype1, column1 datatype1,.. primary key (column\_name) Lets look at an example of primary key in SQL Server.

Following statement create a new table named as Customer with primary key on column custId

CREATE Table dbo.Customer ( CustId INT NOT NULL, CustName VARCHAR(150) , CustCode VARCHAR(20), CustMailId VARCHAR(30) PRIMARY KEY (custId) ) A primary key is specifiedon column CustId, it means CustId column can not have null values also duplicate values can not be allowed. An error message will be returned when duplicate value or null are inserted into custId column.

1. **Create a table with a primary key, a column default value, and a column unique constraint in SQL.**

create table customer( id int primary key, not null, name varchar(30), age int, address varchar(120));