**SQL Assignment 1**

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

**The software used to store, manage, query, and retrieve data stored in a relational database** is called a relational database management system (RDBMS). The RDBMS provides an interface between users and applications and the database, as well as administrative functions for managing data storage, access, and performance.

**There are several advantages of Database management system over file system. Few of them are as follows:**

* **No redundant data**:- Redundancy removed by data [normalization](https://beginnersbook.com/2015/05/normalization-in-dbms/). No data duplication saves storage and improves access time.
* **Data Consistency and Integrity**:-As we discussed earlier the root cause of data inconsistency is data redundancy, since data normalization takes care of the data redundancy, data inconsistency also been taken care of as part of it.
* **Data Security**:-It is easier to apply access constraints in database systems so that only authorized user is able to access the data. Each user has a different set of access thus data is secured from the issues such as identity theft, data leaks and misuse of data.
* **Privacy**: Limited access means privacy of data.
* **Easy access to data**:-Database systems manages data in such a way so that the data is easily accessible with fast response times.
* **Easy recovery**:-Since database systems keeps the backup of data, it is easier to do a full recovery of data in case of a failure.
* **Flexible**:-Database systems are more flexible than file processing systems.

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

**ACID (Atomicity, Consistency, Isolation, Durability)** is a set of properties of database transactions intended to guarantee validity even in the event of errors, power failures, etc.

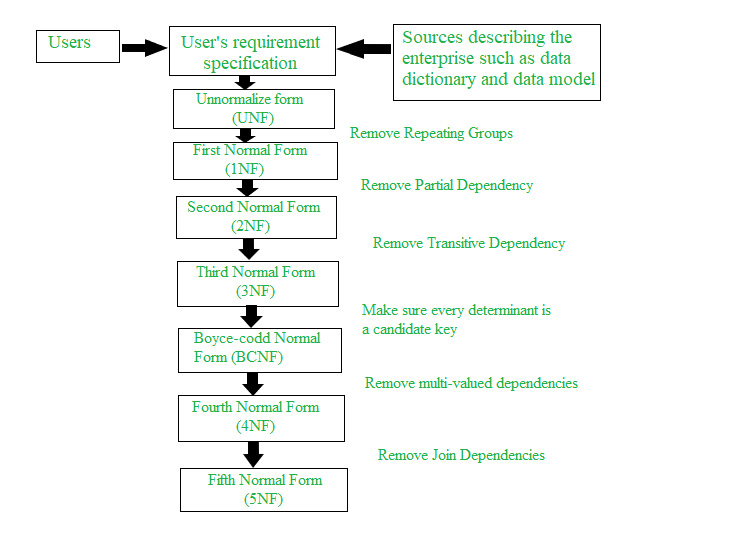
In the context of databases, a sequence of database operations that satisfies the ACID properties, and thus can be perceived as a single logical operation on the data, is called a transaction.

For example, a transfer of funds from one bank account to another involves debiting from one account and crediting to another, and this whole process is a single transaction.

* **Atomicity**  
  All statements of a transaction must succeed completely, or fail completely in each and every situation, including power failures, errors and crashes. Example - Debiting and crediting in a money transfer transaction, both must happen either together or not at all.
* **Consistency**  
  The database must remain in a consistent state after any transaction. Data in the database should not have any changes other than intended after the transaction completion.
* **Isolation**  
  Isolation ensures that concurrent execution of transactions leaves the database in the same state that would have been obtained if the transactions were executed sequentially.
* **Durability**  
  Durability guarantees that once a transaction has been committed, it will remain committed even in the case of a system failure which actually means recording the completed transactions (or their effects) in non-volatile memory.

1. Explain the concept of normalization.

Normalization 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. Normal forms are used to eliminate or reduce redundancy in database tables.



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

Here are list of database languages:-

### SQL

SQL, which stands for Structured Query Language, is one of the most well-known and longest-running database languages. It features both data definition and data manipulation languages and allows you to write queries in a database.

### XQuery

XQuery is a database language that allows you to extract and manipulate data in XML formats, which is a way to share data on the internet.

### OQL

OQL, which stands for Object Query Language, is the standard language for object-oriented databases, which represent data as variables, functions or data structures. These databases are popular with companies that want to store large amounts of complex data.

### SQL/XML

The SQL/XML language is a combination of SQL and XQuery that supports the manipulation and storage of XML data in a database that works with SQL. It enables applications to perform SQL statements on XML data and vice versa.

### GraphQL

GraphQL is an open-source language that works with APIs, which are interfaces that allow users to interact with data.

### LINQ

LINQ, or Language Integrated Query, is a language that extracts and processes data from XML documents, relational databases and other third-party sources.

**Examples:-**

DQL- **SQL SELECT statement** is strictly speaking an example of DQL.

DML- **INSERT :** It is used to insert data into a table.

**UPDATE**: It is used to update existing data within a table.

**DELETE** : It is used to delete records from a database table.

DCL- **GRANT:** Allow specified users to perform specified tasks.

**REVOKE**:Remove the user accessibility to database object.

DDL- DDL is Data Definition Language which is used to define data structures. For example: **create table, alter table are instructions in SQL**.

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

While a primary key and a composite key might do the same things, **the primary key will consist of one column, where the composite key will consist of two or more columns**. The relationship between a primary key and a foreign key is quite different.

An example would be a list of homes on a real estate market. In a well-ordered database, there should be a primary key that uniquely identifies each record.

How this works may have to do with the sophistication of the database.

In some cases, the homes may only be uniquely identified by a mortgage number — all other data (towns, streets, house numbers) is not unique to each record. The mortgage number would be the primary key. Suppose, however, that an MLS realtor’s listing technology assigns its own unique numbers to the records in the table.

Then, there will be two keys that developers might identify as “candidate keys”:

* The mortgage number.
* The MLS number.

One of them will qualify as the “primary key” in what some would consider an arbitrary way.

A composite key, then, would be the combination of two keys.

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

CREATE TABLE Persons (

ID int NOT NULL,  
     LastName varchar(255) NOT NULL,  
     FirstName varchar(255),  
     Age int,  
     CONSTRAINT PK\_Person PRIMARY KEY (ID,LastName));