What is Transaction in DBMS?

- Transactions are a set of operations that are used to perform some logical set of work.
- A transactionis made to change data in a database which can be done by inserting new data, updating the existingdata, or by deleting the data that is no longer required.
- There are certain types of transaction states which tell the user about the current condition of that database transaction and what further steps to be followed for the processing.

Introduction

- You might have encountered a situation when your system got crashed due to some hardware or software issues and got rebooted to ensure that all the data is restored in a consistent state.
- This protection of user's data even in case of a system failure marks as one of the major advantages of database management system.
- Various transactions are done as a part of manipulating the data in a database, these transactions can be seen as a set of operations that are executed by the user program in DBMS.
- Execution of a similar transaction multiple times will lead to the generation of multiple transactions.
- For example, Withdrawing some amount of money from the ATM can be seenas a transaction that can be done multiple times also.

Operations in Transaction

- A certain set of operations takes place when a transaction is done that is used to perform some logical set of operations.
- For example: When we go to withdraw money from ATM, we encounter the following set of operations:

1. Transaction Initiated
2. You have to insert an ATM card
3. Select your choice of language
4. Select whether savings or current account
5. Enter the amount to withdraw
6. Entering your ATM pin
7. Transaction processes
8. You collect the cash
9. You press finish to end transaction

- The above mentioned are the set of operations done by you.
- But in the case of a transaction inDBMS there are three major operations that are used for a transaction to get executed in an efficient manner. These are:

1. **Read/ Access Data**

2. **Write/ Change Data**

3. **Commit**

Properties of Transaction in DBMS

- There are four major properties that are vital for a transaction to be successful.

- These are used to maintain state consistency in the database, both before and after the transaction.

- These are called ACID properties.

Atomicity:

- This property means that either the transaction takes place completely at once or doesn't happen at all.
- There is no middle option, i.e., transactions do not occur partially.
- Each transaction is considered as one single step which either runs completely or is not executed at all.

Consistency:

- This property means that the integrity constraints of a database are maintained so thatthe database is consistent before and after the transaction.
- It refers to the correctness of a database.

Isolation:

- This property means that multiple transactions can occur concurrently without causing anyinconsistency to the database state.
- These transactions occur independently without any external interference.
- Changes that occur in a particular transaction are not visible/ accessible to any othertransaction until that particular change in that transaction 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 remain intact even if a system failure occurs.
- These updates become permanent and are stored in the non-volatile memory.

Conclusion

- A Transaction can be seen as a set of operations that are used to perform some logical set of work.
- Atransaction is used to make changes in data in a database which can be done by inserting new data, altering the existing data, or by deleting the already data.
- ACID properties of a transaction provide a method of ensuring consistency of a database in a waysuch that each transaction is a set of operations that acts a one single step, produces consistent results, acts in isolation from other transactions, and provides durability that makes a database resistant to system failures.