Experiment No.10
Implementation and demonstration of Transaction and Concurrency control techniques using locks
Date of Performance:
Date of Submission:

Experiment No.10

Aim :- Write a query to lock and unlock a table for transaction and concurrency control.

Objective :- To learn locking of tables for transaction processing and concurrency

control. **Theory:**

A lock is a mechanism associated with a table used to restrict the unauthorized access of the data in a table. MySQL allows a client session to acquire a table lock explicitly to cooperate with other sessions to access the table's data. MySQL also allows table locking to prevent unauthorized modification into the same table during a specific period.

Table Locking in MySQL is mainly used to solve concurrency problems. It will be used while running a transaction, i.e., first read a value from a table (database) and then write it into the table (database).

MySQL provides two types of locks onto the table, which are:

READ LOCK: This lock allows a user to only read the data from a table. WRITE

LOCK: This lock allows a user to do both reading and writing into a table. The

following is the syntax that allows us to acquire a table lock explicitly: LOCK

TABLES table_name [READ | WRITE];

The following is the syntax that allows us to release a lock for a table in MySQL:

UNLOCK TABLES;

Conclusion: Locking and unlocking of tables is achieved and verified using insert command in the same table of a database system.

1. Explain Transaction and Concurrency control techniques using locks.
2. Transactions ensure that a series of database operations are executed as a single unit of
3. work, either all succeed or none do. Concurrency control techniques using locks manage
4. access to shared resources in a multi-user environment. Locks can be applied at various
5. levels, such as database, table, or row, to prevent conflicts between transactions.
6. In short, transactions ensure atomicity, consistency, isolation, and durability of database
7. operations, while concurrency control techniques using locks prevent data inconsistency and
8. ensure data integrity by managing concurrent access to shared resources