Experiment No.7
Perform DCL and TCL commands
Date of Performance:
Date of Submission:

Aim :- Write a query to implement Data Control Language(DCL) and Transaction Control Language(TCL) commands

Objective :- To learn DCL commands like Grant and Revoke privileges to the user and TCL commands to commit the transactions and recover it using rollback and save points.

Theory:

Data Control Language:

DCL commands are used to grant and take back authority from any database user.

- Grant
- Revoke

a. Grant: It is used to give user access privileges to a database.

Example

```
1. GRANT SELECT, UPDATE ON MY_TABLE TO SOME_USER,
   ANOTHER_USER;
```

b. Revoke: It is used to take back permissions from the user.

Example

```
1. REVOKE SELECT, UPDATE ON MY_TABLE FROM USER1, USER2;
```

Transaction Control Language

TCL commands can only use with DML commands like INSERT, DELETE and UPDATE only.

These operations are automatically committed in the database that's why they cannot be used while creating tables or dropping them.

Here are some commands that come under TCL:

- COMMIT
- ROLLBACK
- SAVEPOINT

a. Commit: Commit command is used to save all the

transactions to the database. Syntax:

1. COMMIT;

Example:

1. DELETE FROM CUSTOMERS
2. WHERE AGE = 25;
3. COMMIT;

b. Rollback: Rollback command is used to undo transactions that have not already been saved to the database.

Syntax:

1. ROLLBACK;

Example:

1. DELETE FROM CUSTOMERS
2. WHERE AGE = 25;
3. ROLLBACK;

c. SAVEPOINT: It is used to roll the transaction back to a certain point without rolling back the entire transaction.

Syntax:

2. SAVEPOINT SAVEPOINT_NAME;

Implementation:

```
GRANT SELECT, UPDATE ON Employee TO Alice, Bob;

REVOKE SELECT, UPDATE ON Employee FROM Charlie, David;

DELETE FROM Employee

WHERE AGE = 25;

COMMIT;
DELETE FROM Employee

WHERE AGE = 35;

ROLLBACK;

SAVEPOINT delete_employee;
```

Conclusion:

1. Explain about issues faced during rollback in mysql and how it got resolved.

Ans In MySQL, rollback issues arise from transactional inconsistencies, such as deadlocks, lock timeouts, large transactions, or disk space exhaustion.

To resolve these issues, MySQL implements:

- Different transaction isolation levels to control visibility of changes.

- Proper transaction management practices to prevent long-running transactions.

- Monitoring and tuning of transaction activity and database performance.

- Continuous improvements in storage engine technology to enhance concurrency control and rollback performance.

2. Explain how to create a user in sql.

Ans To create a user in SQL:

1. Use the CREATE USER statement.
2. Specify the username and password.
3. Optionally, define privileges or roles for the user.

For example:
```sql
CREATE USER 'username'@'hostname' IDENTIFIED BY 'password';
```
This creates a user with the specified username and password.

You can also grant privileges to the user using the GRANT statement:
```sql
GRANT SELECT, INSERT ON database.* TO 'username'@'hostname';
```
This grants SELECT and INSERT privileges on a specific database to the user.

Finally, don't forget to flush privileges to apply the changes:
```sql
FLUSH PRIVILEGES;
```
This ensures that the changes take effect immediately.