**📘 TCL Statements in MySQL**

**🔹 What is TCL in MySQL?**

**TCL (Transaction Control Language)** deals with transactions in a database.  
A **transaction** is a logical unit of work that consists of one or more SQL statements executed together.

👉 Example: In a **bank transfer** (₹1000 from Account A → Account B):

* Deduct 1000 from A
* Add 1000 to B  
  Both must happen **together**, or neither should.

TCL ensures **data consistency, reliability, and atomicity (all-or-nothing execution).**

**🔹 Why TCL is Used**

* To **maintain data integrity** in case of errors or failures.
* To **control multiple SQL statements** as a single transaction.
* To allow **rollback (undo)** or **commit (save permanently)** changes.
* To handle **concurrent access** safely.

**🔹 TCL Statements in MySQL**

| **TCL Statement** | **Description** |
| --- | --- |
| START TRANSACTION / BEGIN | Begins a new transaction. |
| COMMIT | Saves all changes made in the current transaction permanently. |
| ROLLBACK | Undo all changes made in the current transaction. |
| SAVEPOINT | Marks a point in a transaction that you can roll back to. |
| ROLLBACK TO SAVEPOINT | Rollback only to a specific savepoint. |
| RELEASE SAVEPOINT | Deletes a savepoint. |
| SET AUTOCOMMIT | Enables/disables auto-commit mode. |

**🔹 Syntax & Examples**

**1. START TRANSACTION**

START TRANSACTION;

-- OR

BEGIN;

👉 Begins a new transaction. Changes won’t be permanent until COMMIT.

**2. COMMIT**

COMMIT;

👉 Saves all operations permanently.

**Example:**

START TRANSACTION;

UPDATE accounts SET balance = balance - 1000 WHERE id = 1; -- deduct

UPDATE accounts SET balance = balance + 1000 WHERE id = 2; -- add

COMMIT;

✅ Both updates are saved.

**3. ROLLBACK**

ROLLBACK;

👉 Undoes all operations in the transaction since START TRANSACTION.

**Example:**

START TRANSACTION;

UPDATE accounts SET balance = balance - 1000 WHERE id = 1;

UPDATE accounts SET balance = balance + 1000 WHERE id = 2;

ROLLBACK;

❌ Both updates are undone (no money lost).

**4. SAVEPOINT**

SAVEPOINT sp\_name;

👉 Creates a bookmark within a transaction.

**Example:**

START TRANSACTION;

UPDATE accounts SET balance = balance - 500 WHERE id = 1;

SAVEPOINT sp1;

UPDATE accounts SET balance = balance - 300 WHERE id = 1;

Now you can roll back to sp1.

**5. ROLLBACK TO SAVEPOINT**

ROLLBACK TO sp\_name;

👉 Undo only part of a transaction.

**Example:**

ROLLBACK TO sp1; -- only undo the -300 update

COMMIT; -- save the -500 update

**6. RELEASE SAVEPOINT**

RELEASE SAVEPOINT sp\_name;

👉 Deletes a savepoint.

**Example:**

RELEASE SAVEPOINT sp1;

**7. SET AUTOCOMMIT**

By default, MySQL runs in **autocommit mode (every statement is saved immediately).**

SET AUTOCOMMIT = 0; -- Disable autocommit

START TRANSACTION;

INSERT INTO orders VALUES (101, 'Laptop');

ROLLBACK; -- undo insert

SET AUTOCOMMIT = 1; -- Enable autocommit (default)

**✅ Summary**

* **Transactions = group of SQL statements executed as one unit.**
* **Important TCL Commands:**
  + START TRANSACTION / BEGIN → Start a transaction
  + COMMIT → Save changes
  + ROLLBACK → Undo changes
  + SAVEPOINT → Mark a point in transaction
  + ROLLBACK TO SAVEPOINT → Undo up to a savepoint
  + RELEASE SAVEPOINT → Remove savepoint
  + SET AUTOCOMMIT → Control automatic saving