Chapter 15:

Database Transactions

Informatics Practices

Class XII

By- Rajesh Kumar Mishra

PGT (Comp.Sc.)

KV No.1, AFS, Suratgarh (Raj.)

e-mail : rkmalld@gmail.com

What is Database Transaction?

- ☐ In general, an event of access or modify the record from a database is called Transaction.

- ☐ **A Transaction is a Logical Unit of Work (LUW) on the database that must succeed or fail entirely.**

- ☐ A Transaction may contains several statement but works as an atomic operation on a database.

Properties of a Transaction (ACID Properties)

A Transaction possesses the following ACID properties.

- ❑ **Atomicity: (All-or-None)**

  A transaction is executed entirely or none. No any transaction is allowed to run partially.

- ❑ **Consistency:**

A transaction must leave the database in Consistent state after completion. No any transaction is allowed to leave the database in Inconsistent state, because before execution it was in consistent state.

- ❑ **Integrity:**

Transaction is an atomic unit, it must be executed independently, no any other transaction can interfere during the transaction.

- ❑ **Durability:**

The changes made by the transaction are permanent in nature.

Representation of a Transaction

A transaction may contains several operations like Read() and Write() etc. All the operations are carried out to perform an specific action (work) on the database.

Suppose an amount of 100/- is transferred from X account to Y account, then it can be represented as-

```
T1:  Begin
       Read(X);
       X:=X-100;
       Write(X)
       Read(Y)
       Y:=Y+100;
       Write(Y);
     End;
```

Transaction Control Statements in MySQL

MySQL offers the following Transaction handling statements-

❑ **BEGIN | START TRANSACTION**

Represents the start of a transaction.

❑ **COMMIT [Work]**

Represents the end of a transaction. It saves all the changes on a database permanently.

❑ **SAVEPOINT**

It creates a flag or mark during the execution which can be used to cancel the transaction partially.

❑ **ROLLBACK [To SAVEPOINT <savepoint_Name>]**

It restores the last status of the database i.e. cancels the effect of a transaction (works like UNDO). A partial rollback can be done using Save Points created in the transaction.

❑ **Set Autocommit**

With autocommit feature of MySQL, changes are immediately saved after completion of the command, without using Commit command explicitly. It can disabled by :– mysql>Set Autocommit=0

Execution Mode of Transactions

MySQL offers the following Transaction handling statements-

❑ **BEGIN | START TRANSACTION**

 Represents the start of a transaction.

❑ **COMMIT [Work]**

Represents the end of a transaction. It saves all the changes on a database permanently.

❑ **SAVEPOINT**

It creates a flag or mark during the execution which can be used to cancel the transaction partially.

❑ **ROLLBACK [To SAVEPOINT <savepoint_Name>]**

It restores the last status of the database i.e. cancels the effect of a transaction (works like UNDO). A partial rollback can be done using Save Points created in the transaction.

❑ **Set Autocommit**

With autocommit feature of MySQL, changes are immediately saved after completion of the command, without using Commit command explicitly. It can disabled by :– mysql>Set Autocommit=0