**7.Database Trigger (All Types: Row level and Statement level triggers, Before and After**

**Triggers).**

**Write a database trigger on Library table. The System should keep track of the records that are**

**being updated or deleted. The old value of updated or deleted records should be added in**

**Library\_Audit table.**

create table library(rollnum int,name varchar(100),issuedate date,bookname varchar(100));

create table library\_audit(rollnum int,name varchar(100),issuedate date,bookname varchar(100),operationTime timestamp);

insert Library values (1,'Aditi','2021-08-15','JAVA')’, (2,'Aditya','2021-08-01','DBMS'),(3,'Anurag','2021-08-31','TOC'),(4,'Asim','2021-09-10','CNS'),(5,'Aryan','2021-09-04','DS'),(6,'Aneesh','2021-08-20','HCI'),(7,'Abhishek','2021-08-13','PYTHON');

mysql> select\* from library;

+---------+----------+------------+----------+

| rollnum | name | issuedate | bookname |

+---------+----------+------------+----------+

| 1 | Aditi | 2021-08-15 | JAVA |

| 2 | Aditya | 2021-08-01 | DBMS |

| 3 | Anurag | 2021-08-31 | TOC |

| 4 | Asim | 2021-09-10 | CNS |

| 5 | Aryan | 2021-09-04 | DS |

| 6 | Aneesh | 2021-08-20 | HCI |

| 7 | Abhishek | 2021-08-13 | PYTHON |

+---------+----------+------------+----------+

--Update Trigger

delimiter //

create trigger after\_update

after update

on library for each row

begin

insert into library\_audit values(old.rollnum,old.name,old.issuedate,old.bookname,now());

end //

delimiter //

create trigger before\_update

before update

on library for each row

begin

insert into library\_audit values(old.rollnum,old.name,old.issuedate,old.bookname,now());

end //

update library set issuedate=current\_date(),bookname='C++' where rollnum=1;

mysql> select \* from library;

+---------+----------+------------+----------+

| rollnum | name | issuedate | bookname |

+---------+----------+------------+----------+

| 1 | Aditi | 2021-09-30 | C++ |

| 2 | Aditya | 2021-08-01 | DBMS |

| 3 | Anurag | 2021-08-31 | TOC |

| 4 | Asim | 2021-09-10 | CNS |

| 5 | Aryan | 2021-09-04 | DS |

| 6 | Aneesh | 2021-08-20 | HCI |

| 7 | Abhishek | 2021-08-13 | PYTHON |

+---------+----------+------------+----------+

mysql> select \* from library\_audit;

+---------+-------+------------+----------+---------------------+

| rollnum | name | issuedate | bookname | operationTime |

+---------+-------+------------+----------+---------------------+

| 1 | Aditi | 2021-08-15 | JAVA | 2021-09-30 13:14:36 |

+---------+-------+------------+----------+---------------------+

--Delete Trigger

delimiter //

create trigger after\_delete

after delete

on library for each row

begin

insert into library\_audit values(old.rollnum,old.name,old.issuedate,old.bookname,now());

end //

delimiter //

create trigger before\_delete

before delete

on library for each row

begin

insert into library\_audit values(old.rollnum,old.name,old.issuedate,old.bookname,now());

end //

delete from library where rollnum=7;

mysql> select \* from library;

+---------+--------+------------+----------+

| rollnum | name | issuedate | bookname |

+---------+--------+------------+----------+

| 1 | Aditi | 2021-09-30 | C++ |

| 2 | Aditya | 2021-08-01 | DBMS |

| 3 | Anurag | 2021-08-31 | TOC |

| 4 | Asim | 2021-09-10 | CNS |

| 5 | Aryan | 2021-09-04 | DS |

| 6 | Aneesh | 2021-08-20 | HCI |

+---------+--------+------------+----------+

mysql> select \* from library\_audit;

**theory**

**Table**: In a database, a table is a collection of related data organized in rows and columns. Each table represents a specific entity, like the library table, which holds information on book issuances, and library\_audit, which logs historical data about changes to the library table.

**Trigger:** A trigger is a set of SQL statements that automatically execute in response to certain events on a table, such as an INSERT, UPDATE, or DELETE. Triggers can be defined to run before or after these events. Here, triggers are used to automatically log changes in the library\_audit table whenever records in the library table are updated or deleted.

**Before and After Triggers:**

Before Trigger: Executes before a change is made to a table. For example, before\_update and before\_delete triggers in this code capture the state of a library record before it’s modified or deleted, logging it in library\_audit.

After Trigger: Executes after a change is made. after\_update and after\_delete triggers log the state of the library record after modifications or deletions are performed.

Timestamp: A timestamp records the exact date and time of an operation. The operationTime field in the library\_audit table logs the moment a change occurs, allowing for precise tracking of when records were modified or removed.

Audit Log: An audit log is a record of actions taken within a system. Here, library\_audit serves as an audit log by storing historical data for every update or deletion in library, providing a complete history of actions for accountability and tracking purposes.

This system of tables and triggers helps ensure data integrity and transparency by keeping a complete record of changes to important library data.