K DEEPAK ASSIGNMENT (DAY -9)

ASSIGNMENT 1: Analyze a given business scenario and create an ER diagram that includes entities, relationships, attributes, and cardinality. Ensure that the diagram reflects proper normalization up to the third normal form.

Creating a business scenario of the IT department along with includes entities,relationships,attributes and cardinality

Entities and Attributes:

- First step to create is employees have an attributes of EmployeeID, name, email, and job title.
- Projects have with attributes of ProjectID, project name, start date, and end date.
- Departments with attributes of DepartmentID, department name, and manager.
- Tasks with attributes of TaskID, task description, due date, and are assigned to an employee within a project.

Relationships and Cardinality :

- **Employees and Departments:** one to many department has one manager, but an employee can manage one or no departments.

- **Employees and projects:**Many to Many employee can work on multiple projects, and a project can have multiple employees working on it

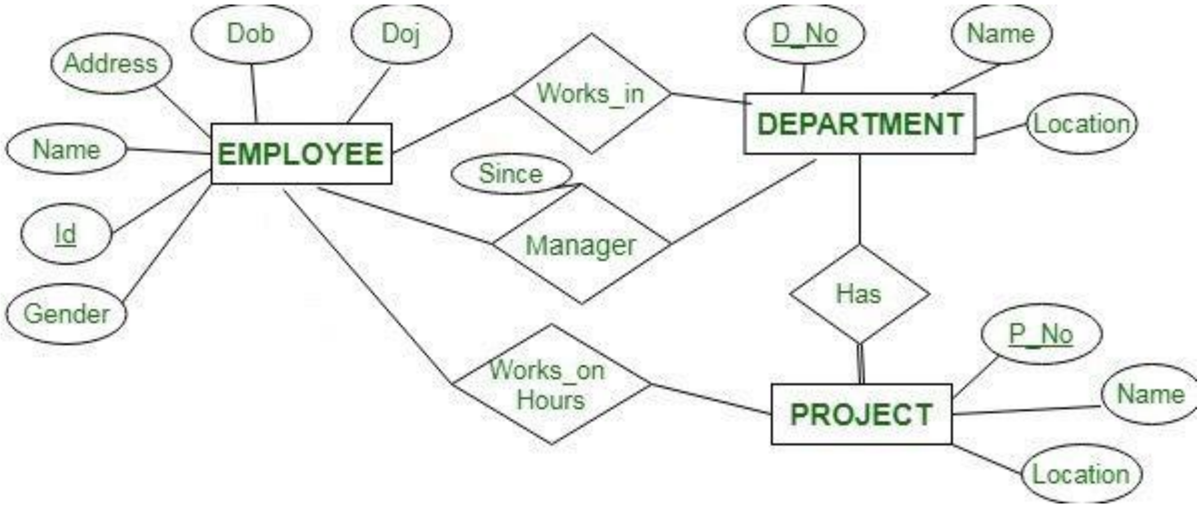
- **Employee and Task:**one to many employee can have multiple tasks, but each task is assigned to one employee.

- **Project and tasks:** one to many A project can have multiple tasks, but each task belongs to one project.

Normalization:

- **First Normal form(1NF):** All attributes contain only atomic values.
- **Second Normal form(2NF):** All non-key attributes are fully functional dependent on the primary key.
- **Third Normal form(3NF):** No transitive dependencies. All attributes are dependent only on the primary key.

ER Diagram:



ASSIGNMENT 2: Design a database schema for a library system, including tables, fields, and constraints like NOT NULL, UNIQUE, and CHECK. Include primary and foreign keys to establish relationships between tables

Library System Database Schema:

Library database schema of library system,tables,fields.

Tables for Library:

Creating a Books (Table):

- BookID(Primary Key, NOT NULL ,Title(VARCHAR(30),NOTNULL),AuthorID(INT,FOREIGN KEY),PublisherID(INT ,FOREIGN KEY),YearPublished, ISBN(UNIQUE)

Members:

- MemberID (PK), Name, Address, Phone, Email (UNIQUE).

Authors:

- AuthorID (PK), Name, Bio .

BookAuthors:

- BookID (FK), AuthorID (FK).

Loans:

- LoanID (PK), BookID (FK), MemberID (FK), LoanDate, ReturnDate.

Constraints:

- NOT NULL: Guarantee that essential fields are always populated.
- UNIQUE: Make sure certain values are distinct where required.
- CHECK: Impose conditions (e.g., CHECK (YearPublished > 1900)).

ASSIGNMENT 3: Explain the ACID properties of a transaction in your own words. Write SQL statements to simulate a transaction that includes locking and demonstrate different isolation levels to show concurrency control.

ACID PROPERTIES AND TRANSACTION:

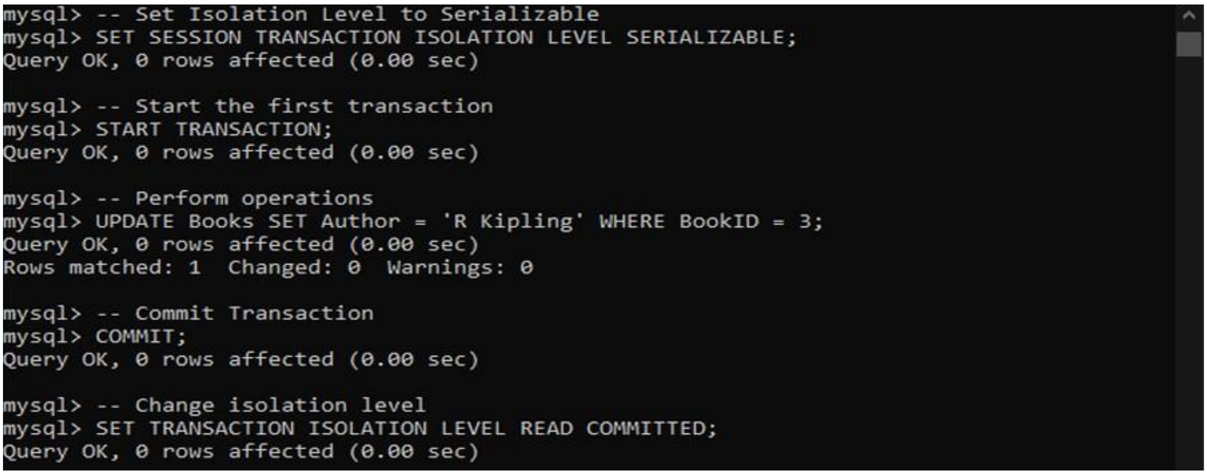
ACID PROPERTIES:

Atomicity: Ensures that all operations in a transaction are completed; if not, the transaction is aborted.

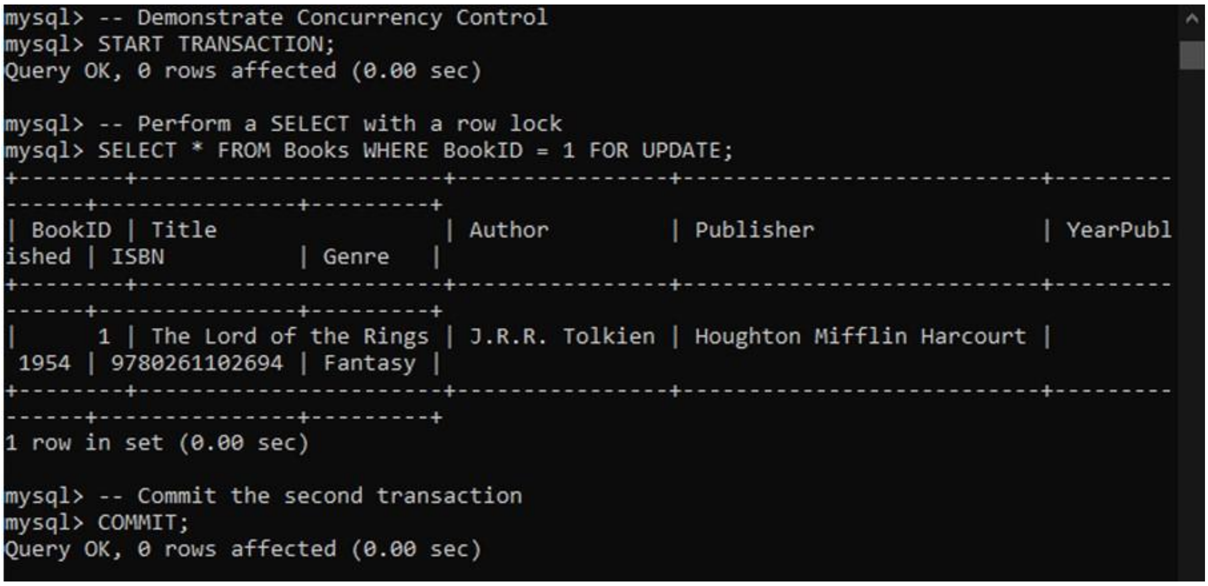
Consistency: Ensures the database remains in a consistent state before and after the transaction.

Isolation: Ensures that transactions are executed independently.

Durability: Ensures that once a transaction is committed, it remains in the system even in case of a failure.



```
mysql> -- Set Isolation Level to Serializable
mysql> SET SESSION TRANSACTION ISOLATION LEVEL SERIALIZABLE;
Query OK, 0 rows affected (0.00 sec)

mysql> -- Start the first transaction
mysql> START TRANSACTION;
Query OK, 0 rows affected (0.00 sec)

mysql> -- Perform operations
mysql> UPDATE Books SET Author = 'R Kipling' WHERE BookID = 3;
Query OK, 0 rows affected (0.00 sec)
Rows matched: 1  Changed: 0  Warnings: 0

mysql> -- Commit Transaction
mysql> COMMIT;
Query OK, 0 rows affected (0.00 sec)

mysql> -- Change isolation level
mysql> SET TRANSACTION ISOLATION LEVEL READ COMMITTED;
Query OK, 0 rows affected (0.00 sec)
```



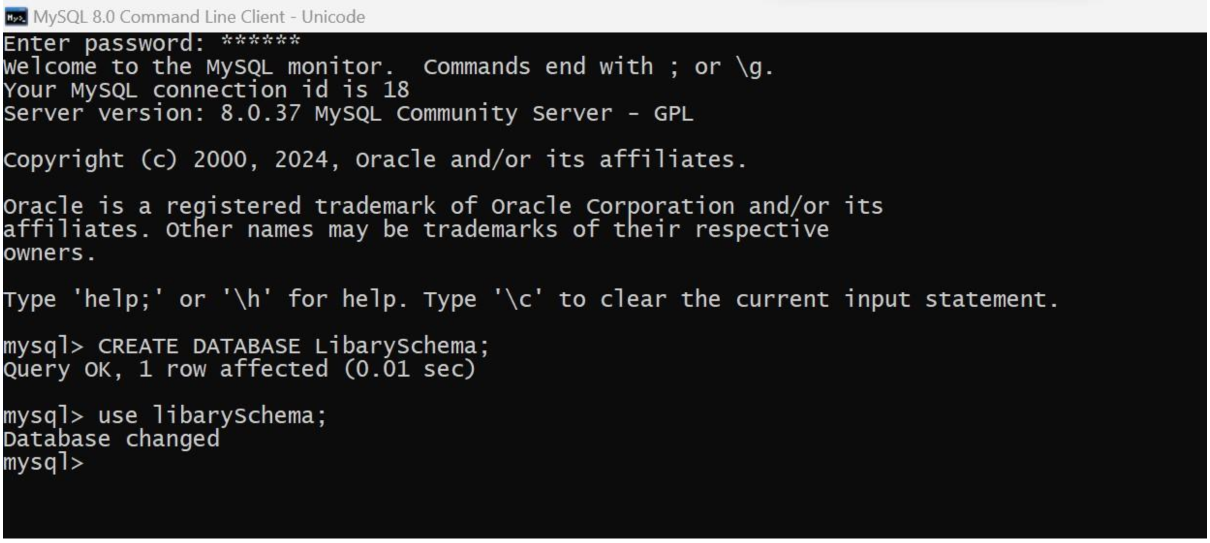
```
mysql> -- Demonstrate Concurrency Control
mysql> START TRANSACTION;
Query OK, 0 rows affected (0.00 sec)

mysql> -- Perform a SELECT with a row lock
mysql> SELECT * FROM Books WHERE BookID = 1 FOR UPDATE;
+--------+-----------------------+----------------+---------------------------+---------
------+---------------+---------+
| BookID | Title                 | Author         | Publisher                 | YearPubl
ished | ISBN          | Genre   |
+--------+-----------------------+----------------+---------------------------+---------
------+---------------+---------+
|      1 | The Lord of the Rings | J.R.R. Tolkien | Houghton Mifflin Harcourt |
 1954 | 9780261102694 | Fantasy |
+--------+-----------------------+----------------+---------------------------+---------
------+---------------+---------+
1 row in set (0.00 sec)

mysql> -- Commit the second transaction
mysql> COMMIT;
Query OK, 0 rows affected (0.00 sec)
```

ASSIGNMENT 4: Write SQL statements to CREATE a new database and tables that reflect the library schema you designed earlier. Use ALTER statements to modify the table structures and DROP statements to remove a redundant table.

SQL Statements for Database and Table Creation for Library Schema:

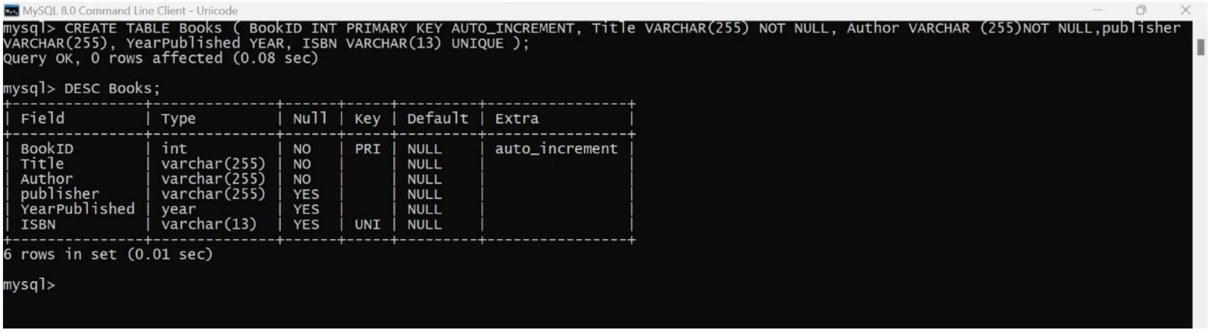
Creating a Database :

- Creating a database named libarySchema and using LibarySchema database



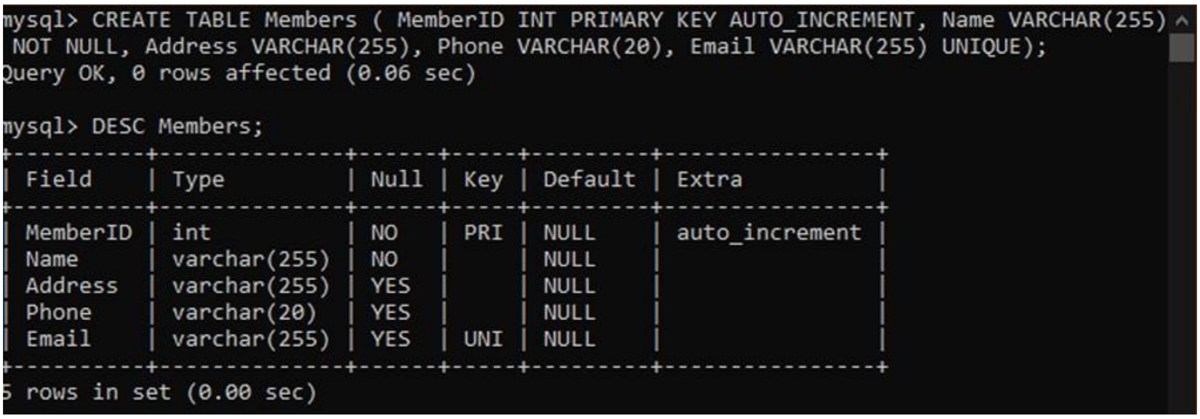
```
Enter password: ******
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 18
Server version: 8.0.37 MySQL Community Server - GPL

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> CREATE DATABASE LibarySchema;
Query OK, 1 row affected (0.01 sec)

mysql> use libarySchema;
Database changed
mysql>
```

CREATING A TABLE:



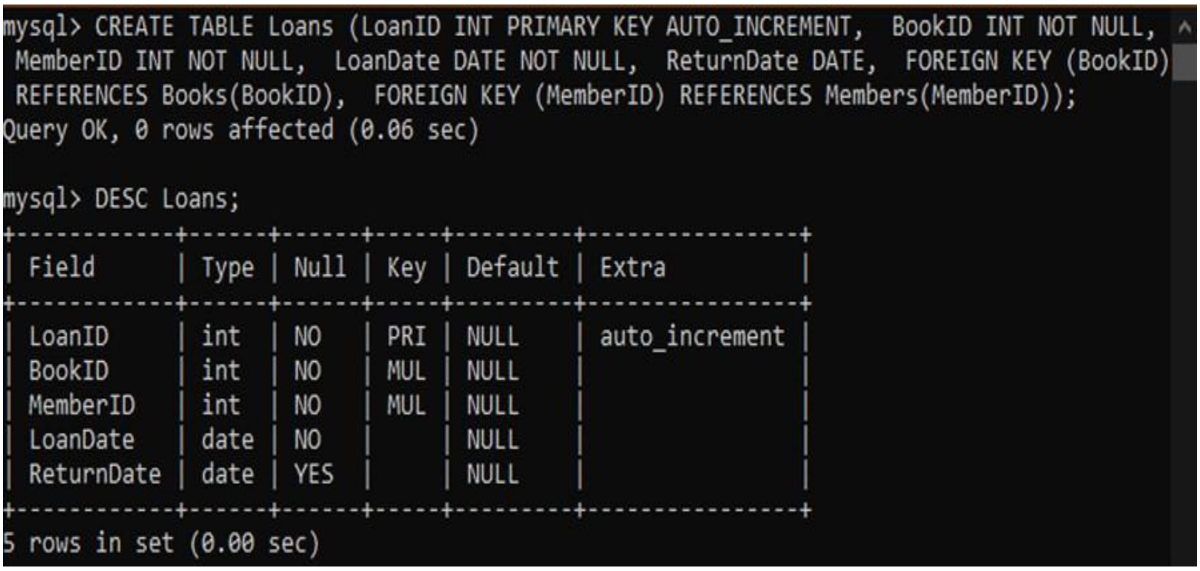
```
mysql> CREATE TABLE Books ( BookID INT PRIMARY KEY AUTO_INCREMENT, Title VARCHAR(255) NOT NULL, Author VARCHAR (255)NOT NULL,publisher
VARCHAR(255), YearPublished YEAR, ISBN VARCHAR(13) UNIQUE );
Query OK, 0 rows affected (0.08 sec)

mysql> DESC Books;
+---------------+--------------+------+-----+---------+----------------+
| Field         | Type         | Null | Key | Default | Extra          |
+---------------+--------------+------+-----+---------+----------------+
| BookID        | int          | NO   | PRI | NULL    | auto_increment |
| Title         | varchar(255) | NO   |     | NULL    |                |
| Author        | varchar(255) | NO   |     | NULL    |                |
| publisher     | varchar(255) | YES  |     | NULL    |                |
| YearPublished | year         | YES  |     | NULL    |                |
| ISBN          | varchar(13)  | YES  | UNI | NULL    |                |
+---------------+--------------+------+-----+---------+----------------+
6 rows in set (0.01 sec)

mysql>
```

CREATING TABLE MEMBERS :



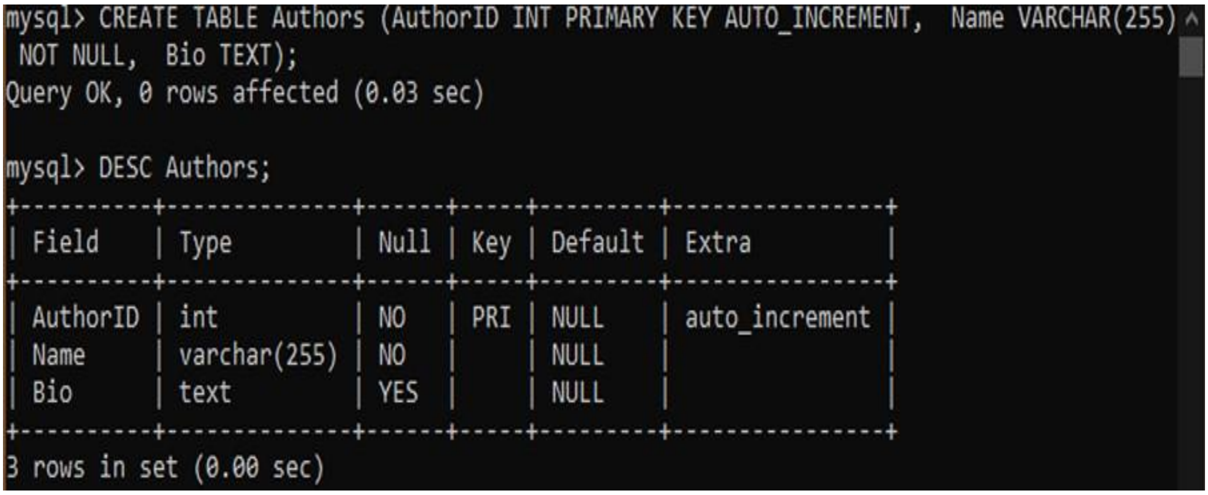
```
mysql> CREATE TABLE Members ( MemberID INT PRIMARY KEY AUTO_INCREMENT, Name VARCHAR(255)
 NOT NULL, Address VARCHAR(255), Phone VARCHAR(20), Email VARCHAR(255) UNIQUE);
Query OK, 0 rows affected (0.06 sec)

mysql> DESC Members;
+----------+--------------+------+-----+---------+----------------+
| Field    | Type         | Null | Key | Default | Extra          |
+----------+--------------+------+-----+---------+----------------+
| MemberID | int          | NO   | PRI | NULL    | auto_increment |
| Name     | varchar(255) | NO   |     | NULL    |                |
| Address  | varchar(255) | YES  |     | NULL    |                |
| Phone    | varchar(20)  | YES  |     | NULL    |                |
| Email    | varchar(255) | YES  | UNI | NULL    |                |
+----------+--------------+------+-----+---------+----------------+
5 rows in set (0.00 sec)
```

CREATING TABLE LOANS:



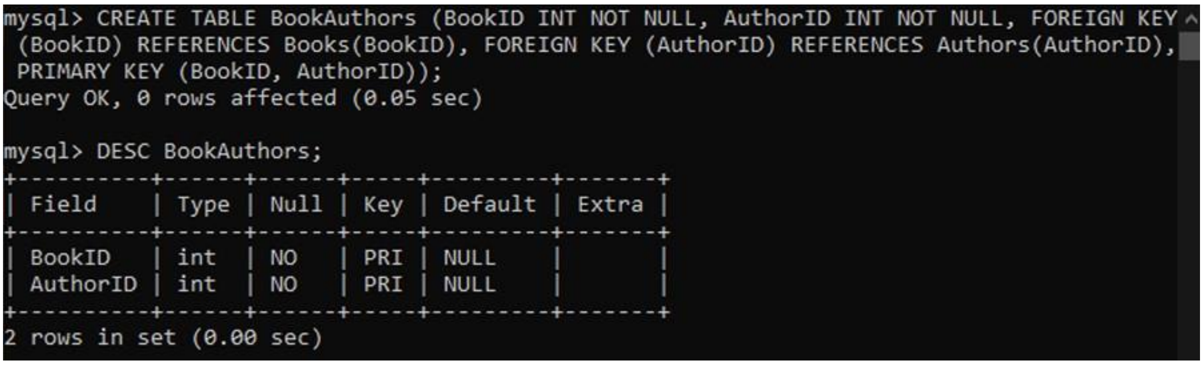
```
mysql> CREATE TABLE Loans (LoanID INT PRIMARY KEY AUTO_INCREMENT,  BookID INT NOT NULL,
 MemberID INT NOT NULL,  LoanDate DATE NOT NULL,  ReturnDate DATE,  FOREIGN KEY (BookID)
 REFERENCES Books(BookID),  FOREIGN KEY (MemberID) REFERENCES Members(MemberID));
Query OK, 0 rows affected (0.06 sec)

mysql> DESC Loans;
+------------+------+------+-----+---------+----------------+
| Field      | Type | Null | Key | Default | Extra          |
+------------+------+------+-----+---------+----------------+
| LoanID     | int  | NO   | PRI | NULL    | auto_increment |
| BookID     | int  | NO   | MUL | NULL    |                |
| MemberID   | int  | NO   | MUL | NULL    |                |
| LoanDate   | date | NO   |     | NULL    |                |
| ReturnDate | date | YES  |     | NULL    |                |
+------------+------+------+-----+---------+----------------+
5 rows in set (0.00 sec)
```

CREATING TABLE AUTHORS:



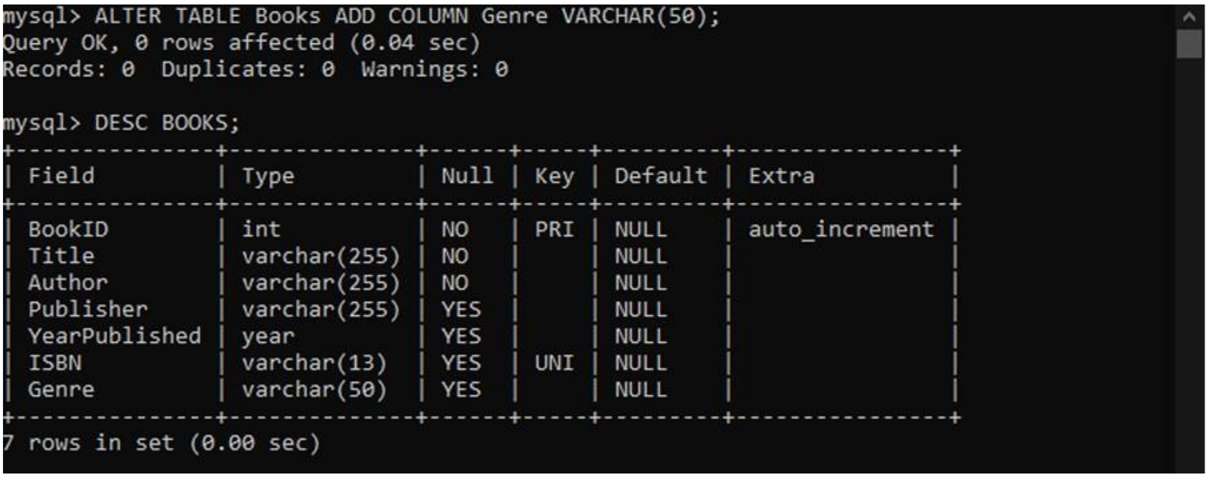
```
mysql> CREATE TABLE Authors (AuthorID INT PRIMARY KEY AUTO_INCREMENT,  Name VARCHAR(255)
 NOT NULL,  Bio TEXT);
Query OK, 0 rows affected (0.03 sec)

mysql> DESC Authors;
+----------+--------------+------+-----+---------+----------------+
| Field    | Type         | Null | Key | Default | Extra          |
+----------+--------------+------+-----+---------+----------------+
| AuthorID | int          | NO   | PRI | NULL    | auto_increment |
| Name     | varchar(255) | NO   |     | NULL    |                |
| Bio      | text         | YES  |     | NULL    |                |
+----------+--------------+------+-----+---------+----------------+
3 rows in set (0.00 sec)
```

CREATING TABLE AUTHORS:



```
mysql> CREATE TABLE BookAuthors (BookID INT NOT NULL, AuthorID INT NOT NULL, FOREIGN KEY
 (BookID) REFERENCES Books(BookID), FOREIGN KEY (AuthorID) REFERENCES Authors(AuthorID),
 PRIMARY KEY (BookID, AuthorID));
Query OK, 0 rows affected (0.05 sec)

mysql> DESC BookAuthors;
+----------+------+------+-----+---------+-------+
| Field    | Type | Null | Key | Default | Extra |
+----------+------+------+-----+---------+-------+
| BookID   | int  | NO   | PRI | NULL    |       |
| AuthorID | int  | NO   | PRI | NULL    |       |
+----------+------+------+-----+---------+-------+
2 rows in set (0.00 sec)
```

CREATING A TABLE:



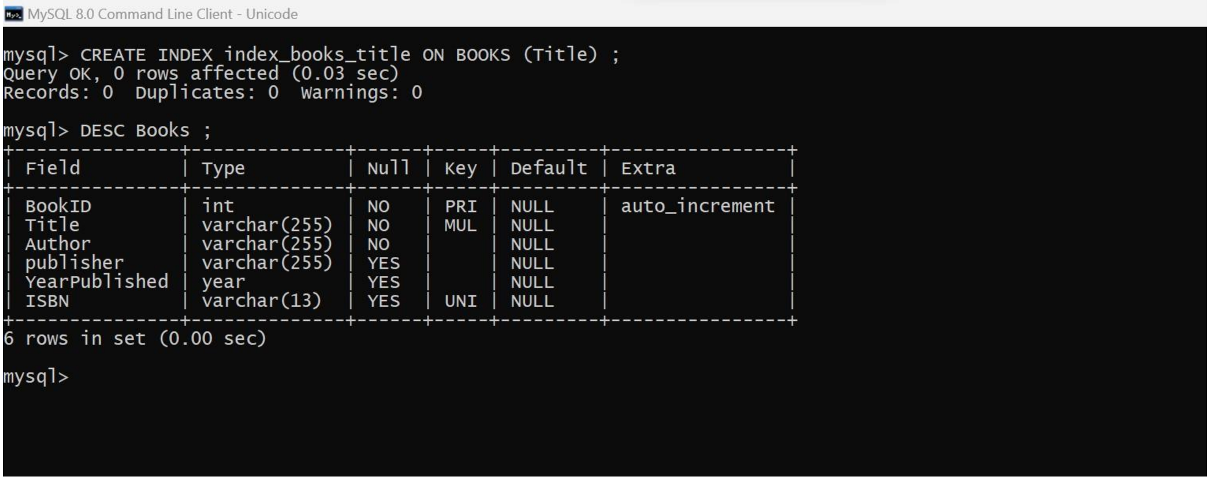
```
mysql> ALTER TABLE Books ADD COLUMN Genre VARCHAR(50);
Query OK, 0 rows affected (0.04 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> DESC BOOKS;
+---------------+--------------+------+-----+---------+----------------+
| Field         | Type         | Null | Key | Default | Extra          |
+---------------+--------------+------+-----+---------+----------------+
| BookID        | int          | NO   | PRI | NULL    | auto_increment |
| Title         | varchar(255) | NO   |     | NULL    |                |
| Author        | varchar(255) | NO   |     | NULL    |                |
| Publisher     | varchar(255) | YES  |     | NULL    |                |
| YearPublished | year         | YES  |     | NULL    |                |
| ISBN          | varchar(13)  | YES  | UNI | NULL    |                |
| Genre         | varchar(50)  | YES  |     | NULL    |                |
+---------------+--------------+------+-----+---------+----------------+
7 rows in set (0.00 sec)
```

ASSIGNMENT 5: Demonstrate the creation of an index on a table and discuss how it improves query performance. Use a DROP INDEX statement to remove the index and analyse the impact on query execution.

CREATING AND DROPPING AN INDEX:

Creating index:

- Here we Create an index using the SQL commands



```
mysql> CREATE INDEX index_books_title ON BOOKS (Title) ;
Query OK, 0 rows affected (0.03 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> DESC Books ;
+---------------+--------------+------+-----+---------+----------------+
| Field         | Type         | Null | Key | Default | Extra          |
+---------------+--------------+------+-----+---------+----------------+
| BookID        | int          | NO   | PRI | NULL    | auto_increment |
| Title         | varchar(255) | NO   | MUL | NULL    |                |
| Author        | varchar(255) | NO   |     | NULL    |                |
| publisher     | varchar(255) | YES  |     | NULL    |                |
| YearPublished | year         | YES  |     | NULL    |                |
| ISBN          | varchar(13)  | YES  | UNI | NULL    |                |
+---------------+--------------+------+-----+---------+----------------+
6 rows in set (0.00 sec)

mysql>
```

Query to search for a book by title

SELECT * FROM Books WHERE Title = 'Sample Book';

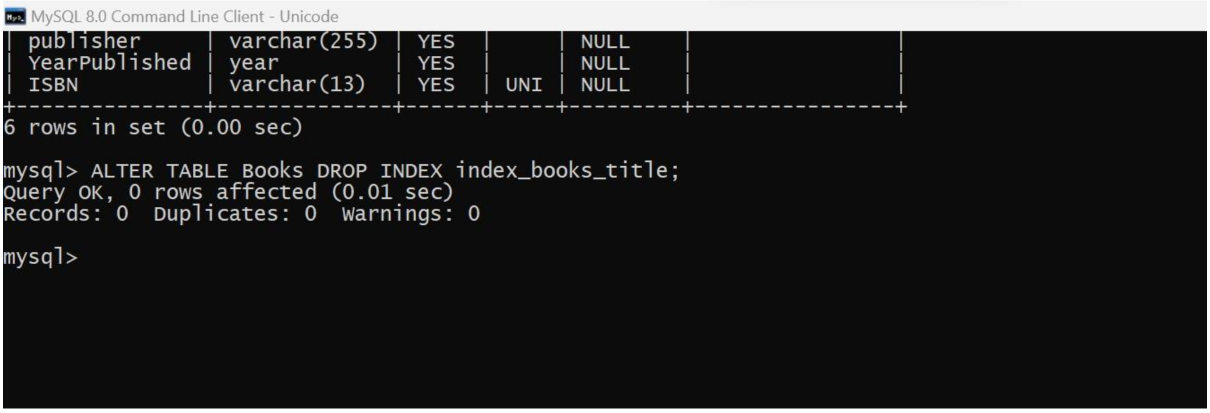
Query without Index:

- Without an index, the database must scan the entire table to locate rows where the Title is 'Sample Book'. This approach is inefficient for large tables, as every row needs to be examined.

Query with Index:

- With the index “ index_books_Title” established, the database leverages this index to quickly find rows where the Title matches 'Sample Book'. This significantly reduces the number of rows to be checked, thereby accelerating the query execution.

DROPPING AN INDEX:



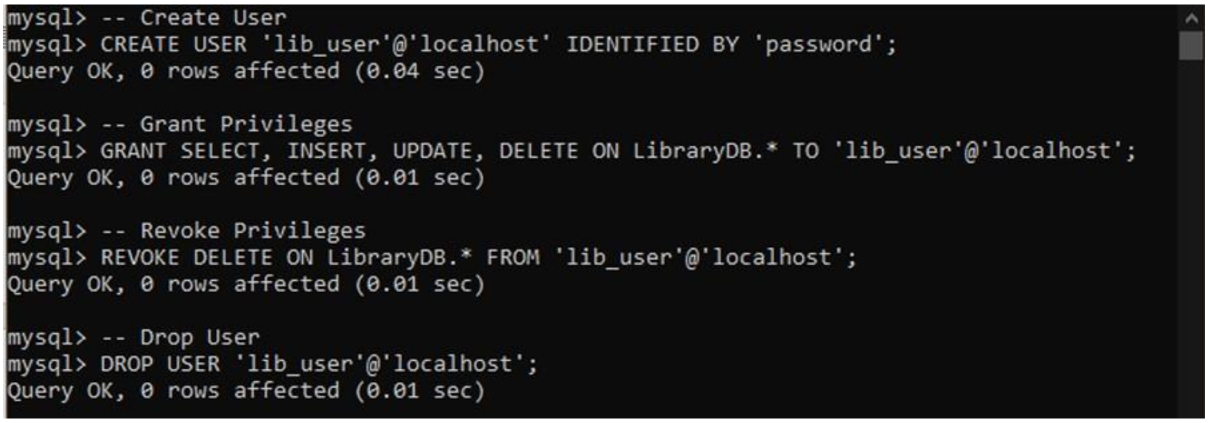
MySQL 8.0 Command Line Client - Unicode

```
| publisher     | varchar(255) | YES  |     | NULL    |                |
| YearPublished | year         | YES  |     | NULL    |                |
| ISBN          | varchar(13)  | YES  | UNI | NULL    |                |
+---------------+--------------+------+-----+---------+----------------+
6 rows in set (0.00 sec)

mysql> ALTER TABLE Books DROP INDEX index_books_title;
Query OK, 0 rows affected (0.01 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql>
```

ASSIGNMENT 6: Create a new database user with specific privileges using the CREATE USER and GRANT commands. Then, write a script to REVOKE certain privileges and DROP the user.

CREATE AND MANAGE THE DATABASE USERS:

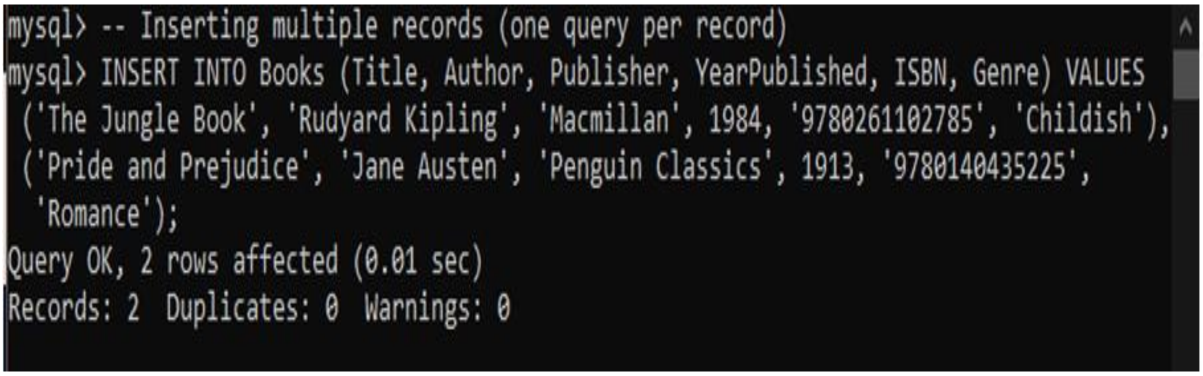
- Creating user command new user named “lib_user” that can connect to the MySQL server from “localhost” (the local machine). The user is assigned a password 'password'.
- Grant commands gives “Lib_user” permission to “Select,insert,update and Delete “ operations on the table (*) with the “librarydb”
- Revoke command gives the “DELETE” permission to “lib_user” and for the all the tables in libarydb
- Drop command deletes the “Lib_user” from MySql server and executed once “Lib_user” will no longer be able to log in or perform any operations on the database.



```
mysql> -- Create User
mysql> CREATE USER 'lib_user'@'localhost' IDENTIFIED BY 'password';
Query OK, 0 rows affected (0.04 sec)

mysql> -- Grant Privileges
mysql> GRANT SELECT, INSERT, UPDATE, DELETE ON LibraryDB.* TO 'lib_user'@'localhost';
Query OK, 0 rows affected (0.01 sec)

mysql> -- Revoke Privileges
mysql> REVOKE DELETE ON LibraryDB.* FROM 'lib_user'@'localhost';
Query OK, 0 rows affected (0.01 sec)

mysql> -- Drop User
mysql> DROP USER 'lib_user'@'localhost';
Query OK, 0 rows affected (0.01 sec)
```

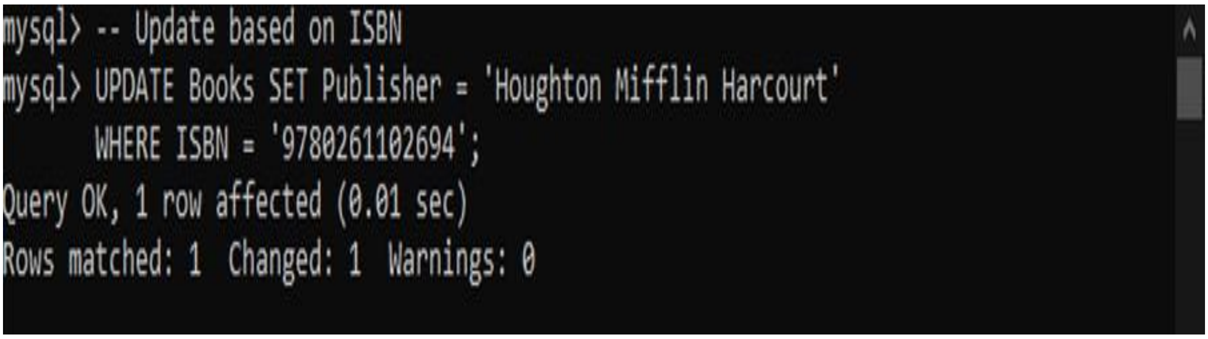
ASSIGNMENT 7: Prepare a series of SQL statements to INSERT new records into the library tables, UPDATE existing records with new information, and DELETE records based on specific criteria. Include BULK INSERT operations to load data from an external source.

SQL STATEMENTS :



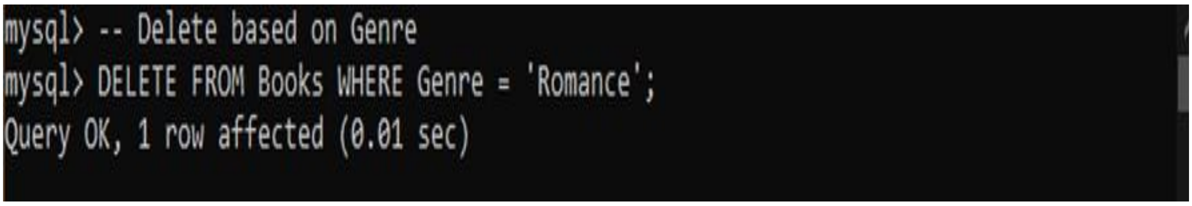
```
mysql> -- Inserting multiple records (one query per record)
mysql> INSERT INTO Books (Title, Author, Publisher, YearPublished, ISBN, Genre) VALUES
 ('The Jungle Book', 'Rudyard Kipling', 'Macmillan', 1984, '9780261102785', 'Childish'),
 ('Pride and Prejudice', 'Jane Austen', 'Penguin Classics', 1913, '9780140435225',
  'Romance');
Query OK, 2 rows affected (0.01 sec)
Records: 2  Duplicates: 0  Warnings: 0
```

UPDATING EXISITING RECORDS:



```
mysql> -- Update based on ISBN
mysql> UPDATE Books SET Publisher = 'Houghton Mifflin Harcourt'
       WHERE ISBN = '9780261102694';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

DELETING RECORDS:



```
mysql> -- Delete based on Genre
mysql> DELETE FROM Books WHERE Genre = 'Romance';
Query OK, 1 row affected (0.01 sec)
```

Bulk Insert:

```
mysql> LOAD DATA LOCAL INFILE
'C:\Users\Deepak\Documents\Desktop\Books.csv'
INTO TABLE Books
FIELDS TERMINATED BY ','
LINES TERMINATED BY '\n'
IGNORE 1 LINES;
```