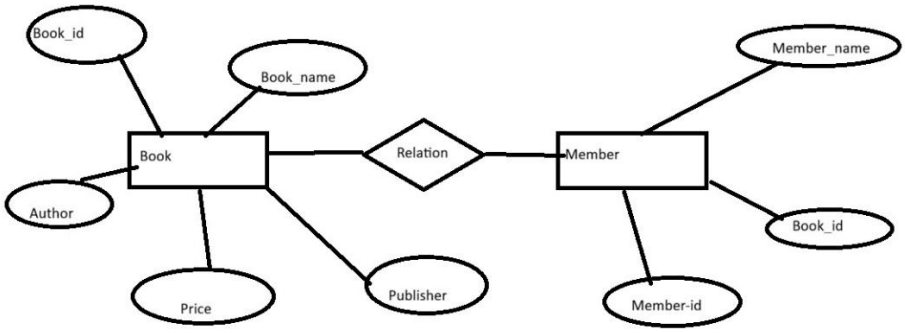
DBMS & SQL ASSIGNMENT – 1

- Harshinie M

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.

Scenario Overview

A library maintains a collection of books and manages its members. Each member can borrow books from the library, and the library needs to track which member has borrowed which book. The system should also store essential details about books and members.



Relationship:

- Member BORROWS Book
- Relationship type: One-to-Many (1:N)
- Explanation:
    - One Book can be borrowed by multiple Members
    - One Member borrows one Book at a time

Cardinality:

- 1 Book → many Members
- 1 Member → borrows 1 Book

Normalization (Up to 3NF):

1. 1NF (First Normal Form)
    - All attributes are atomic
2. 2NF (Second Normal Form)
    - All non-key attributes fully depend on the primary key
3. 3NF (Third Normal Form)
    - No transitive dependency

Tables are normalized up to 3NF to eliminate redundancy and maintain data integrity**.**

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.

```
mysql> CREATE database Harshinie_sdb;
Query OK, 1 row affected (0.01 sec)

mysql> SHOW databases;
+--------------------+
| Database           |
+--------------------+
| harshinie_sdb      |
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
5 rows in set (0.00 sec)

mysql> CREATE TABLE Book (
    ->     Book_ID INT PRIMARY KEY,
    ->     Book_Name VARCHAR(100) NOT NULL,
    ->     Author VARCHAR(100) NOT NULL,
    ->     Price INT CHECK (Price > 0)
    -> );
```

```
ERROR 1046 (3D000): No database selected
mysql> use Harshinie_sdb;
Database changed
mysql> CREATE TABLE Book (
    ->     Book_ID INT PRIMARY KEY,
    ->     Book_Name VARCHAR(100) NOT NULL,
    ->     Author VARCHAR(100) NOT NULL,
    ->     Price INT CHECK (Price > 0)
    -> );
Query OK, 0 rows affected (0.02 sec)

mysql> SELECT * FROM Book;
Empty set (0.00 sec)

mysql> desc book;
+-----------+--------------+------+-----+---------+-------+
| Field     | Type         | Null | Key | Default | Extra |
+-----------+--------------+------+-----+---------+-------+
| Book_ID   | int          | NO   | PRI | NULL    |       |
| Book_Name | varchar(100) | NO   |     | NULL    |       |
| Author    | varchar(100) | NO   |     | NULL    |       |
| Price     | int          | YES  |     | NULL    |       |
+-----------+--------------+------+-----+---------+-------+
4 rows in set (0.00 sec)

mysql> SELECT * from Book;
Empty set (0.00 sec)

mysql> CREATE TABLE Member (
    ->     Member_ID INT PRIMARY KEY,
    ->     Member_Name VARCHAR(100) NOT NULL,
    ->     Book_ID INT,
    ->     FOREIGN KEY (Book_ID) REFERENCES Book(Book_ID)
    -> );
Query OK, 0 rows affected (0.02 sec)

mysql> SHOW tables;
+------------------------+
| Tables_in_harshinie_sdb |
+------------------------+
| book                   |
| member                 |
+------------------------+
2 rows in set (0.00 sec)

mysql>
```

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 guarantee the reliable transaction where logical unit of work that performs one or more database operations. For example, bank transfer, where debit from account A and credit to account B must be succeed together. The acid properties are atomicity, consistency, isolation, and durability.

Atomicity is either all succeed or none are applied. It ensures rollback and undo locks.

Consistency is it moves database from one valid state to another valid state. It takes the constraints and triggers. For example, if the primary key is null, then the error will come because the primary key should not be null.

Isolation Each transaction runs as if it is the only transaction in the database, even when many transactions are running at the same time.

Durability Once committed, the data will not be lost even if the system fails or crashes. It ensures committed data survive failures.

```
mysql> use harshinie_sdb;
Database changed
mysql> show tables;
+------------------------+
| Tables_in_harshinie_sdb |
+------------------------+
| book                   |
| member                 |
+------------------------+
2 rows in set (0.00 sec)

mysql> select * from book;
```

```
+---------+-----------+---------+-------+
| Book_ID | Book_Name | Author  | Price |
+---------+-----------+---------+-------+
|       1 | DBMS      | Navathe |   500 |
+---------+-----------+---------+-------+
1 row in set (0.00 sec)

mysql> desc book;
+-----------+--------------+------+-----+---------+-------+
| Field     | Type         | Null | Key | Default | Extra |
+-----------+--------------+------+-----+---------+-------+
| Book_ID   | int          | NO   | PRI | NULL    |       |
| Book_Name | varchar(100) | NO   |     | NULL    |       |
| Author    | varchar(100) | NO   |     | NULL    |       |
| Price     | int          | YES  |     | NULL    |       |
+-----------+--------------+------+-----+---------+-------+
4 rows in set (0.00 sec)

mysql> desc member;
+-------------+--------------+------+-----+---------+-------+
| Field       | Type         | Null | Key | Default | Extra |
+-------------+--------------+------+-----+---------+-------+
| Member_ID   | int          | NO   | PRI | NULL    |       |
| Member_Name | varchar(100) | NO   |     | NULL    |       |
| Book_ID     | int          | YES  | MUL | NULL    |       |
+-------------+--------------+------+-----+---------+-------+
3 rows in set (0.00 sec)

mysql> START TRANSACTION;
Query OK, 0 rows affected (0.00 sec)

mysql> SELECT * FROM book
    -> WHERE Book_ID = 1
    -> FOR UPDATE;
+---------+-----------+---------+-------+
| Book_ID | Book_Name | Author  | Price |
+---------+-----------+---------+-------+
|       1 | DBMS      | Navathe |   500 |
+---------+-----------+---------+-------+
1 row in set (0.00 sec)

mysql> UPDATE book
    -> SET Price = 600
    -> WHERE Book_ID = 1;
Query OK, 1 row affected (0.00 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

```
mysql> COMMIT;
Query OK, 0 rows affected (0.00 sec)

mysql> SELECT * FROM book;
+---------+-----------+---------+-------+
| Book_ID | Book_Name | Author  | Price |
+---------+-----------+---------+-------+
|       1 | DBMS      | Navathe |   600 |
+---------+-----------+---------+-------+
1 row in set (0.00 sec)

mysql> update book set price = 700 where book_id = 1;
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> SELECT * FROM book;
+---------+-----------+---------+-------+
| Book_ID | Book_Name | Author  | Price |
+---------+-----------+---------+-------+
|       1 | DBMS      | Navathe |   700 |
+---------+-----------+---------+-------+
1 row in set (0.00 sec)

mysql> SET TRANSACTION ISOLATION LEVEL READ COMMITTED;
Query OK, 0 rows affected (0.00 sec)

mysql> SET TRANSACTION ISOLATION LEVEL SERIALIZABLE;
Query OK, 0 rows affected (0.00 sec)

mysql>
```

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.

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

mysql> USE library_db;
Database changed
mysql> CREATE TABLE book (
    ->     Book_ID INT PRIMARY KEY,
    ->     Book_Name VARCHAR(100),
    ->     Author VARCHAR(100),
```

```
  ->     Price INT
  -> );
Query OK, 0 rows affected (0.03 sec)

mysql> CREATE TABLE member (
  ->     Member_ID INT PRIMARY KEY,
  ->     Member_Name VARCHAR(100),
  ->     Book_ID INT,
  ->     FOREIGN KEY (Book_ID) REFERENCES book(Book_ID)
  -> );
Query OK, 0 rows affected (0.03 sec)

mysql> ALTER TABLE book
  -> ADD Publisher VARCHAR(100);
Query OK, 0 rows affected (0.06 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> ALTER TABLE member
  -> MODIFY Member_Name VARCHAR(150);
Query OK, 0 rows affected (0.01 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> CREATE TABLE temp_books (
  ->     Temp_ID INT PRIMARY KEY,
  ->     Book_Name VARCHAR(100)
  -> );
Query OK, 0 rows affected (0.03 sec)

mysql> select * from book;
Empty set (0.00 sec)

mysql> desc book;
+-----------+--------------+------+-----+---------+-------+
| Field     | Type         | Null | Key | Default | Extra |
+-----------+--------------+------+-----+---------+-------+
| Book_ID   | int          | NO   | PRI | NULL    |       |
| Book_Name | varchar(100) | YES  |     | NULL    |       |
| Author    | varchar(100) | YES  |     | NULL    |       |
| Price     | int          | YES  |     | NULL    |       |
| Publisher | varchar(100) | YES  |     | NULL    |       |
+-----------+--------------+------+-----+---------+-------+
5 rows in set (0.00 sec)

mysql> desc member;
+-------------+--------------+------+-----+---------+-------+
| Field       | Type         | Null | Key | Default | Extra |
+-------------+--------------+------+-----+---------+-------+
```

```
| Member_ID   | int         | NO   | PRI | NULL    |      |
| Member_Name | varchar(150) | YES  |     | NULL    |      |
| Book_ID     | int         | YES  | MUL | NULL    |      |
+-------------+--------------+------+-----+---------+-------+
3 rows in set (0.00 sec)

mysql> desc temp_books;
+-----------+--------------+------+-----+---------+-------+
| Field     | Type         | Null | Key | Default | Extra |
+-----------+--------------+------+-----+---------+-------+
| Temp_ID   | int          | NO   | PRI | NULL    |       |
| Book_Name | varchar(100) | YES  |     | NULL    |       |
+-----------+--------------+------+-----+---------+-------+
2 rows in set (0.00 sec)

mysql> DROP TABLE temp_books;
Query OK, 0 rows affected (0.02 sec)

mysql> show tables;
+----------------------+
| Tables_in_library_db |
+----------------------+
| book                 |
| member               |
+----------------------+
2 rows in set (0.00 sec)

mysql>
```

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 analyze the impact on query execution.

An index is a database object that improves the speed of data retrieval operations on a table.

```
mysql> select * from book;
+---------+------------------+--------------+-------+-----------+
| Book_ID | Book_Name        | Author       | Price | Publisher |
+---------+------------------+--------------+-------+-----------+
|       1 | DBMS             | Navathe      |   700 | NULL      |
|       2 | Operating System | Silberschatz |   650 | NULL      |
|       3 | Computer Networks | Tanenbaum   |   600 | NULL      |
|       4 | SQL Fundamentals | Navathe      |   550 | NULL      |
```

```
+---------+------------------+-------------+-------+-----------+
4 rows in set (0.00 sec)

mysql> CREATE INDEX idx_author
   -> ON book(Author);
ERROR 1061 (42000): Duplicate key name 'idx_author'
mysql> SELECT * FROM book WHERE Author = 'Navathe';
+---------+-----------------+---------+-------+-----------+
| Book_ID | Book_Name       | Author  | Price | Publisher |
+---------+-----------------+---------+-------+-----------+
|       1 | DBMS            | Navathe |   700 | NULL      |
|       4 | SQL Fundamentals | Navathe |   550 | NULL      |
+---------+-----------------+---------+-------+-----------+
2 rows in set (0.00 sec)

mysql> SHOW INDEX FROM book;
+-------+------------+------------+--------------+-------------+-----------+-------------+----------+--------
--+------+------------+---------+---------------+---------+------------+
| Table | Non_unique | Key_name   | Seq_in_index | Column_name | Collation |
Cardinality | Sub_part | Packed | Null | Index_type | Comment | Index_comment | Visible |
Expression |
+-------+------------+------------+--------------+-------------+-----------+-------------+----------+--------
--+------+------------+---------+---------------+---------+------------+
| book  |          0 | PRIMARY    |            1 | Book_ID     | A         |           4 |     NULL |   NULL |      |
BTREE     |       |            | YES     | NULL       |
| book  |          1 | idx_author |            1 | Author      | A         |           3 |     NULL |   NULL | YES  |
BTREE     |       |            | YES     | NULL       |
+-------+------------+------------+--------------+-------------+-----------+-------------+----------+--------
--+------+------------+---------+---------------+---------+------------+
2 rows in set (0.01 sec)

mysql> DROP INDEX idx_author ON book;
Query OK, 0 rows affected (0.02 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> SHOW INDEX FROM book;
+-------+------------+----------+--------------+-------------+-----------+-------------+----------+------
--+------+------------+---------+---------------+---------+------------+
| Table | Non_unique | Key_name | Seq_in_index | Column_name | Collation | Cardinality
| Sub_part | Packed | Null | Index_type | Comment | Index_comment | Visible | Expression
|
+-------+------------+----------+--------------+-------------+-----------+-------------+----------+------
--+------+------------+---------+---------------+---------+------------+
| book  |          0 | PRIMARY  |            1 | Book_ID     | A         |           4 |     NULL |   NULL |      |
BTREE     |       |            | YES     | NULL       |
+-------+------------+----------+--------------+-------------+-----------+-------------+----------+------
--+------+------------+---------+---------------+---------+------------+
```

```
1 row in set (0.00 sec)

mysql> SELECT * FROM book WHERE Author = 'Navathe';
+---------+-----------------+---------+-------+-----------+
| Book_ID | Book_Name       | Author  | Price | Publisher |
+---------+-----------------+---------+-------+-----------+
|       1 | DBMS            | Navathe |   700 | NULL      |
|       4 | SQL Fundamentals | Navathe |   550 | NULL      |
+---------+-----------------+---------+-------+-----------+
2 rows in set (0.00 sec)

mysql>
```

An index idx_author was created on the Author column to improve query performance. The index allows faster data retrieval by reducing full table scans. After dropping the index, queries on the Author column require scanning the entire table, which impacts performance.

After DROP INDEX

- SELECT queries become **slower**
- Database does **full table scan**
- Performance degrades for large data

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.

```
mysql> CREATE USER 'lib_user'@'localhost' IDENTIFIED BY 'lib123';
Query OK, 0 rows affected (0.01 sec)

mysql> GRANT SELECT, INSERT ON harshinie_sdb.* TO 'lib_user'@'localhost';
Query OK, 0 rows affected (0.00 sec)

mysql> SELECT User, Host FROM mysql.user;
+------------------+-----------+
| User             | Host      |
+------------------+-----------+
| lib_user         | localhost |
| mysql.infoschema | localhost |
| mysql.session    | localhost |
| mysql.sys        | localhost |
| root             | localhost |
```

```
+------------------+-----------+
5 rows in set (0.00 sec)

mysql> SHOW GRANTS FOR 'lib_user'@'localhost';
+-----------------------------------------------------------------------+
| Grants for lib_user@localhost                                         |
+-----------------------------------------------------------------------+
| GRANT USAGE ON *.* TO `lib_user`@`localhost`                          |
| GRANT SELECT, INSERT ON `harshinie_sdb`.* TO `lib_user`@`localhost` |
+-----------------------------------------------------------------------+
2 rows in set (0.00 sec)

mysql> REVOKE INSERT ON harshinie_sdb.* FROM 'lib_user'@'localhost';
Query OK, 0 rows affected (0.00 sec)

mysql> SHOW GRANTS FOR 'lib_user'@'localhost';
+-------------------------------------------------------------+
| Grants for lib_user@localhost                               |
+-------------------------------------------------------------+
| GRANT USAGE ON *.* TO `lib_user`@`localhost`                |
| GRANT SELECT ON `harshinie_sdb`.* TO `lib_user`@`localhost` |
+-------------------------------------------------------------+
2 rows in set (0.00 sec)

mysql> DROP USER 'lib_user'@'localhost';
Query OK, 0 rows affected (0.00 sec)

mysql> SELECT User, Host FROM mysql.user;
+------------------+-----------+
| User             | Host      |
+------------------+-----------+
| mysql.infoschema | localhost |
| mysql.session    | localhost |
| mysql.sys        | localhost |
| root             | localhost |
+------------------+-----------+
4 rows in set (0.00 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.

```
mysql> SELECT * FROM member;
+-----------+-------------+---------+
| Member_ID | Member_Name | Book_ID |
+-----------+-------------+---------+
|         1 | Alice       |       1 |
|         2 | Bob         |       2 |
|         3 | Charlie     |       3 |
|         4 | David       |       4 |
+-----------+-------------+---------+
4 rows in set (0.00 sec)

mysql> UPDATE member
    -> SET book_id 5
    -> where member_name=charlie;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that
corresponds to your MySQL server version for the right syntax to use near '5
where member_name=charlie' at line 2
mysql> where member_name = charlie;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that
corresponds to your MySQL server version for the right syntax to use near 'where
member_name = charlie' at line 1
mysql> UPDATE book
    -> SET Price = 750
    -> WHERE Book_ID = 1;
Query OK, 1 row affected (0.00 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> UPDATE member
    -> SET Member_Name = 'Alice Smith'
    -> WHERE Member_ID = 1;
Query OK, 1 row affected (0.00 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select * from member;
+-----------+-------------+---------+
| Member_ID | Member_Name | Book_ID |
+-----------+-------------+---------+
|         1 | Alice Smith |       1 |
|         2 | Bob         |       2 |
|         3 | Charlie     |       3 |
|         4 | David       |       4 |
+-----------+-------------+---------+
4 rows in set (0.00 sec)

mysql> delete from member where book_id = 4;
Query OK, 1 row affected (0.00 sec)
```

```
mysql> select * from member;
+-----------+-------------+---------+
| Member_ID | Member_Name | Book_ID |
+-----------+-------------+---------+
|         1 | Alice Smith |       1 |
|         2 | Bob         |       2 |
|         3 | Charlie     |       3 |
+-----------+-------------+---------+
3 rows in set (0.00 sec)
```