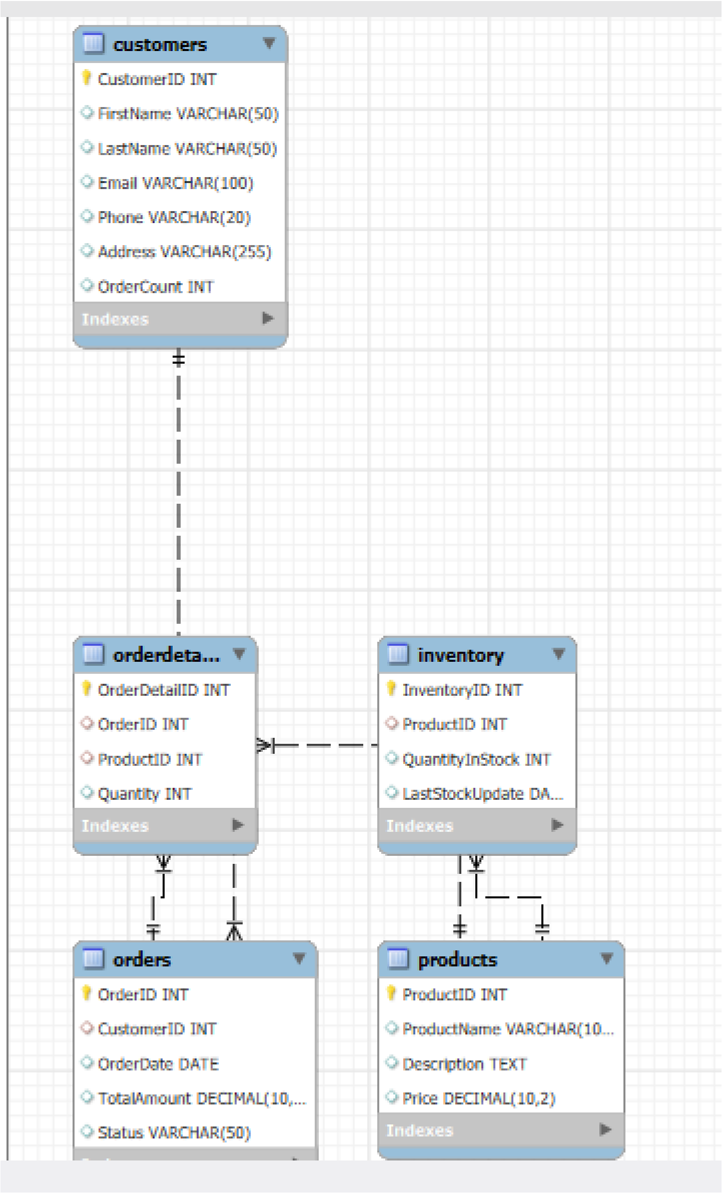
Assignment 1: Analyze a given business scenario and create an ER diagram that includes entities, relationships, attributes, and cardinality. Ensure that the diagram reflects proper normalization up to the third normal form.



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 library;

Query OK, 1 row affected (0.02 sec)


mysql> CREATE TABLE Member (

    ->     MemberID INT PRIMARY KEY,

    ->     Name VARCHAR(100) NOT NULL,

    ->     Email VARCHAR(100) UNIQUE NOT NULL,

    ->     Phone VARCHAR(15),

    ->     JoinDate DATE NOT NULL

    -> );

ERROR 1046 (3D000): No database selected

mysql> use library;

Database changed

mysql> CREATE TABLE Member (

    ->     MemberID INT PRIMARY KEY,

    ->     Name VARCHAR(100) NOT NULL,

    ->     Email VARCHAR(100) UNIQUE NOT NULL,

    ->     Phone VARCHAR(15),

    ->     JoinDate DATE NOT NULL

    -> );

Query OK, 0 rows affected (0.21 sec)


mysql> CREATE TABLE Category (

    ->     CategoryID INT PRIMARY KEY,

    ->     CategoryName VARCHAR(50) NOT NULL UNIQUE

    -> );

Query OK, 0 rows affected (0.12 sec)


mysql> CREATE TABLE Staff (
    ->     StaffID INT PRIMARY KEY,
    ->     Name VARCHAR(100) NOT NULL,
    ->     Role VARCHAR(50),
    ->     Email VARCHAR(100) UNIQUE,
    ->     HireDate DATE NOT NULL
    -> );
Query OK, 0 rows affected (0.19 sec)

mysql> CREATE TABLE Book (
    ->     BookID INT PRIMARY KEY,
    ->     Title VARCHAR(150) NOT NULL,
    ->     Author VARCHAR(100) NOT NULL,
    ->     ISBN VARCHAR(20) UNIQUE NOT NULL,
    ->     CategoryID INT,
    ->     PublishedYear INT CHECK (PublishedYear >= 1500 AND PublishedYear <= YEAR(CURDATE())),
    ->     CopiesAvailable INT NOT NULL CHECK (CopiesAvailable >= 0),
    ->     FOREIGN KEY (CategoryID) REFERENCES Category(CategoryID)
    -> );
ERROR 3814 (HY000): An expression of a check constraint 'book_chk_1' contains disallowed function: curdate.
mysql> INSERT INTO Member VALUES
    -> (1, 'Alice Johnson', 'alice@example.com', '9876543210', '2022-01-10'),
    -> (2, 'Bob Smith', 'bob@example.com', '9876543211', '2022-03-15'),
    -> (3, 'Charlie Brown', 'charlie@example.com', '9876543212', '2023-07-01'),
    -> (4, 'Diana Prince', 'diana@example.com', '9876543213', '2023-11-25'),
    -> (5, 'Ethan Hunt', 'ethan@example.com', '9876543214', '2024-02-08');
Query OK, 5 rows affected (0.02 sec)
Records: 5  Duplicates: 0  Warnings: 0

mysql> CREATE TABLE Book (
    ->     BookID INT PRIMARY KEY,
   ->     Title VARCHAR(150) NOT NULL,
   ->     Author VARCHAR(100) NOT NULL,
   ->     ISBN VARCHAR(20) UNIQUE NOT NULL,
   ->     CategoryID INT,
   ->     PublishedYear INT CHECK (PublishedYear >= 1500),
   ->     CopiesAvailable INT NOT NULL CHECK (CopiesAvailable >= 0),
   ->     FOREIGN KEY (CategoryID) REFERENCES Category(CategoryID)
   -> );
Query OK, 0 rows affected (0.09 sec)

mysql> INSERT INTO Staff VALUES
   -> (101, 'John Admin', 'Librarian', 'john@example.com', '2021-09-01'),
   -> (102, 'Maya Clerk', 'Clerk', 'maya@example.com', '2022-02-12'),
   -> (103, 'Zara Helper', 'Assistant', 'zara@example.com', '2022-11-18'),
   -> (104, 'Sam Monitor', 'Monitor', 'sam@example.com', '2023-01-05'),
   -> (105, 'Nina Audit', 'Auditor', 'nina@example.com', '2023-06-20');
Query OK, 5 rows affected (0.02 sec)
Records: 5  Duplicates: 0  Warnings: 0

mysql> INSERT INTO Category VALUES
   -> (1, 'Fiction'),
   -> (2, 'Non-fiction'),
   -> (3, 'Science'),
   -> (4, 'History'),
   -> (5, 'Biography');
Query OK, 5 rows affected (0.02 sec)
Records: 5  Duplicates: 0  Warnings: 0

mysql> INSERT INTO Book VALUES
 -> (201, 'The Silent Patient', 'Alex Michaelides', 'ISBN001', 1, 2019, 3),
 -> (202, 'Sapiens', 'Yuval Noah Harari', 'ISBN002', 2, 2014, 5),
   -> (203, 'Brief Answers to the Big Questions', 'Stephen Hawking', 'ISBN003', 3, 2018, 2),
   -> (204, 'The Wright Brothers', 'David McCullough', 'ISBN004', 4, 2015, 4),
   -> (205, 'Becoming', 'Michelle Obama', 'ISBN005', 5, 2018, 6);
Query OK, 5 rows affected (0.02 sec)
Records: 5  Duplicates: 0  Warnings: 0
```

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.

1.Atomicity
A transaction must be treated as a single unit, which either completes entirely or does nothing at all. If any part fails, the whole transaction is rolled back.
2.Consistency
A transaction must leave the database in a valid and consistent state. All rules, constraints, and relationships must remain intact.
3.Isolation
When multiple transactions occur at the same time, they should not interfere with each other. Intermediate results of one transaction must be invisible to others.
4.Durability
Once a transaction is committed, the changes it made are permanent, even if the system crashes right afterward.

```
mysql> CREATE TABLE BankAccount (
   ->    AccountID INT PRIMARY KEY,
   ->    AccountHolder VARCHAR(100),
   ->    Balance DECIMAL(10,2)
   -> );
Query OK, 0 rows affected (0.05 sec)


  mysql>
mysql> INSERT INTO BankAccount VALUES
    -> (1, 'Alice', 1000.00),
    -> (2, 'Bob', 1500.00);
Query OK, 2 rows affected (0.01 sec)
Records: 2  Duplicates: 0  Warnings: 0

mysql> SET autocommit = 0;
Query OK, 0 rows affected (0.00 sec)

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

mysql> SELECT * FROM BankAccount WHERE AccountID IN (1, 2) FOR UPDATE;
+-----------+---------------+---------+
| AccountID | AccountHolder | Balance |
+-----------+---------------+---------+
|         1 | Alice         | 1000.00 |
|         2 | Bob           | 1500.00 |
+-----------+---------------+---------+
2 rows in set (0.00 sec)

mysql> UPDATE BankAccount SET Balance = Balance - 200 WHERE AccountID = 1;
Query OK, 1 row affected (0.00 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> UPDATE BankAccount SET Balance = Balance + 200 WHERE AccountID = 2;
Query OK, 1 row affected (0.00 sec)
Rows matched: 1  Changed: 1  Warnings: 0

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

```
mysql> START TRANSACTION;

Query OK, 0 rows affected (0.01 sec)


mysql> SELECT * FROM BankAccount WHERE AccountID = 1;

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

| AccountID | AccountHolder | Balance |

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

|         1 | Alice         |  800.00 |

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

1 row in set (0.00 sec)
```

1. Read Uncommitted

```
mysql> SET SESSION TRANSACTION ISOLATION LEVEL READ UNCOMMITTED;

Query OK, 0 rows affected (0.00 sec)


mysql> START TRANSACTION;

Query OK, 0 rows affected (0.00 sec)


mysql> SELECT * FROM BankAccount WHERE AccountID = 1;

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

| AccountID | AccountHolder | Balance |

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

|         1 | Alice         |  800.00 |

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

1 row in set (0.00 sec)


mysql> COMMIT;

Query OK, 0 rows affected (0.00 sec)
```

2. Read Committed

```
mysql> SET SESSION TRANSACTION ISOLATION LEVEL READ COMMITTED;

Query OK, 0 rows affected (0.00 sec)
```

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

mysql> SELECT * FROM BankAccount WHERE AccountID = 1;
+-----------+---------------+---------+
| AccountID | AccountHolder | Balance |
+-----------+---------------+---------+
|         1 | Alice         |  800.00 |
+-----------+---------------+---------+
1 row in set (0.00 sec)

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

3. Repeatable Read

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

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

mysql> SELECT * FROM BankAccount WHERE AccountID = 1;
+-----------+---------------+---------+
| AccountID | AccountHolder | Balance |
+-----------+---------------+---------+
|         1 | Alice         |  800.00 |
+-----------+---------------+---------+
1 row in set (0.00 sec)

mysql> SELECT * FROM BankAccount WHERE AccountID = 1;
```

```
+-----------+---------------+---------+
| AccountID | AccountHolder | Balance |
+-----------+---------------+---------+
|         1 | Alice         |  800.00 |
+-----------+---------------+---------+
1 row in set (0.00 sec)

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

4. Serializable

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

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

mysql> SELECT * FROM BankAccount WHERE AccountID = 1;
+-----------+---------------+---------+
| AccountID | AccountHolder | Balance |
+-----------+---------------+---------+
|         1 | Alice         |  800.00 |
+-----------+---------------+---------+
1 row in set (0.00 sec)

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.

ALTER Table Statements

```
mysql> ALTER TABLE Member
    -> ADD Address VARCHAR(200);
Query OK, 0 rows affected (0.03 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> select * from member;
+----------+---------------+---------------------+------------+------------+---------+
| MemberID | Name          | Email               | Phone      | JoinDate   | Address |
+----------+---------------+---------------------+------------+------------+---------+
|        1 | Alice Johnson | alice@example.com   | 9876543210 | 2022-01-10 | NULL    |
|        2 | Bob Smith     | bob@example.com     | 9876543211 | 2022-03-15 | NULL    |
|        3 | Charlie Brown | charlie@example.com | 9876543212 | 2023-07-01 | NULL    |
|        4 | Diana Prince  | diana@example.com   | 9876543213 | 2023-11-25 | NULL    |
|        5 | Ethan Hunt    | ethan@example.com   | 9876543214 | 2024-02-08 | NULL    |
+----------+---------------+---------------------+------------+------------+---------+
5 rows in set (0.00 sec)

mysql> ALTER TABLE Member
    -> MODIFY Phone VARCHAR(20);
Query OK, 0 rows affected (0.04 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql>
mysql> select * from member;
+----------+---------------+---------------------+------------+------------+---------+
| MemberID | Name          | Email               | Phone      | JoinDate   | Address |
+----------+---------------+---------------------+------------+------------+---------+
|        1 | Alice Johnson | alice@example.com   | 9876543210 | 2022-01-10 | NULL    |
|        2 | Bob Smith     | bob@example.com     | 9876543211 | 2022-03-15 | NULL    |
|        3 | Charlie Brown | charlie@example.com | 9876543212 | 2023-07-01 | NULL    |
|        4 | Diana Prince  | diana@example.com   | 9876543213 | 2023-11-25 | NULL    |
|        5 | Ethan Hunt    | ethan@example.com   | 9876543214 | 2024-02-08 | NULL    |
+----------+---------------+---------------------+------------+------------+---------+
5 rows in set (0.00 sec)

mysql> ALTER TABLE Book
    -> ADD Publisher VARCHAR(100);
Query OK, 0 rows affected (0.01 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> ALTER TABLE Book
    -> ;
Query OK, 0 rows affected (0.01 sec)

mysql> select * from member;
+----------+---------------+---------------------+------------+------------+---------+
| MemberID | Name          | Email               | Phone      | JoinDate   | Address |
+----------+---------------+---------------------+------------+------------+---------+
|        1 | Alice Johnson | alice@example.com   | 9876543210 | 2022-01-10 | NULL    |
|        2 | Bob Smith     | bob@example.com     | 9876543211 | 2022-03-15 | NULL    |
|        3 | Charlie Brown | charlie@example.com | 9876543212 | 2023-07-01 | NULL    |
|        4 | Diana Prince  | diana@example.com   | 9876543213 | 2023-11-25 | NULL    |
|        5 | Ethan Hunt    | ethan@example.com   | 9876543214 | 2024-02-08 | NULL    |
+----------+---------------+---------------------+------------+------------+---------+
5 rows in set (0.00 sec)
```

DROP a Redundant Table

```
mysql> CREATE TABLE OldCategory (
    ->     ID INT,
    ->     Name VARCHAR(100)
    -> );
Query OK, 0 rows affected (0.03 sec)

mysql> INSERT INTO OldCategory (ID, Name) VALUES
    -> (1, 'Science'),
    -> (2, 'Literature'),
    -> (3, 'Technology'),
    -> (4, 'History'),
    -> (5, 'Children');
Query OK, 5 rows affected (0.00 sec)
Records: 5  Duplicates: 0  Warnings: 0

mysql> drop table oldcategory;
Query OK, 0 rows affected (0.02 sec)
```

Assignment 6: 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.

```
mysql> -- CREATING INDEX ON TABLE COLUMN--
mysql> CREATE INDEX idx_author ON Book(Author);
Query OK, 0 rows affected (0.08 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

- The index idx_author helps the database engine **quickly locate records** based on the Author column.
- Without the index, a query like this:
  SELECT * FROM Book WHERE Author = 'Dan Brown'; With the index, the query uses the index to **quickly jump to matching rows**, improving performance especially for **large tables.**

Drop the Index :

```
DROP INDEX idx_author ON Book;
```

- After dropping the index, the same SELECT query: **SELECT * FROM Book WHERE Author = 'Dan Brown';**
- will perform **slower**, especially as the number of rows grows.
- The database engine will revert to a **full scan** of the Book table to find matches.

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> -- CREATING USER AND DROPING USER ON THE DATABASE--

mysql> CREATE USER 'library_user'@'localhost' IDENTIFIED BY 'password123';

Query OK, 0 rows affected (0.05 sec)


mysql> GRANT SELECT, INSERT ON LibraryDB.* TO 'library_user'@'localhost';

Query OK, 0 rows affected (0.02 sec)


mysql> REVOKE INSERT ON LibraryDB.* FROM 'library_user'@'localhost';

Query OK, 0 rows affected (0.01 sec)


mysql> DROP USER 'library_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.

1. Insert New Records

```
mysql> INSERT INTO Member (MemberID, Name, Email, Phone, JoinDate)

    -> VALUES

    -> (6, 'John Doe', 'john.doe@example.com', '123-456-7890', '2025-05-01'),

    -> (7, 'Jane Smith', 'jane.smith@example.com', '987-654-3210', '2025-04-15');

Query OK, 2 rows affected (0.00 sec)
```

```
Records: 2  Duplicates: 0  Warnings: 0

mysql> INSERT INTO Book (BookID, Title, Author, ISBN, CategoryID, PublishedYear, CopiesAvailable)
    -> VALUES
    -> (6, 'The Great Gatsby', 'F. Scott Fitzgerald', '9780743273566', 1, 1925, 5),
    -> (7, '1984', 'George Orwell', '9780451524936', 2, 1949, 3);
Query OK, 2 rows affected (0.00 sec)
Records: 2  Duplicates: 0  Warnings: 0
```

2. Update Existing Records

```
mysql> UPDATE Member
    -> SET Phone = '111-222-3333'
    -> WHERE MemberID = 1;
Query OK, 1 row affected (0.00 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> UPDATE Book
    -> SET CopiesAvailable = 8
    -> WHERE BookID = 1;
Query OK, 0 rows affected (0.00 sec)
Rows matched: 0  Changed: 0  Warnings: 0
```

3. Delete Records Based on Specific Criteria

```
mysql> DELETE FROM Member
    -> WHERE MemberID = 2;
Query OK, 1 row affected (0.00 sec)

mysql> DELETE FROM Book
    -> WHERE CategoryID = 1;
Query OK, 2 rows affected (0.00 sec)
```

4. Bulk Insert Data from an External Source

```
mysql> INSERT INTO Member (MemberID, Name, Email, Phone, JoinDate) VALUES
    -> (8, 'John Doe', 'john.doe8@example.com', '111-222-3333', '2025-01-10'),
    -> (9, 'Jane Smith', 'jane.smith9@example.com', '222-333-4444', '2025-01-12'),
    -> (10, 'Alice Brown', 'alice.brown10@example.com', '333-444-5555', '2025-02-01'),
    -> (11, 'Bob Lee', 'bob.lee11@example.com', '444-555-6666', '2025-03-05'),
    -> (12, 'Emma Davis', 'emma.davis12@example.com', '555-666-7777', '2025-04-20');
Query OK, 5 rows affected (0.00 sec)
Records: 5  Duplicates: 0  Warnings: 0

mysql> INSERT INTO Book (BookID, Title, Author, ISBN, CategoryID, PublishedYear, CopiesAvailable)
VALUES
    -> (8, 'The Great Gatsby', 'F. Scott Fitzgerald', '9780743273588', 1, 1925, 5),
    -> (9, '1984', 'George Orwell', '9780451524959', 2, 1949, 3),
    -> (10, 'To Kill a Mockingbird', 'Harper Lee', '9780061120099', 1, 1960, 4),
    -> (11, 'Pride and Prejudice', 'Jane Austen', '9780141439529', 2, 1813, 6),
    -> (12, 'The Alchemist', 'Paulo Coelho', '9780062315019', 3, 1988, 7);
Query OK, 5 rows affected (0.00 sec)
Records: 5  Duplicates: 0  Warnings: 0
```