CODE:

```sql
CREATE DATABASE `LibraryManagementSystem`;

USE LibraryManagementSystem;


CREATE TABLE Person (

        Person_ID INT PRIMARY KEY AUTO_INCREMENT,

        Name VARCHAR(100) NOT NULL,

        Address VARCHAR(255),

        Phone_Number VARCHAR(15),

        Card_Number INT UNIQUE,

        Username VARCHAR(50) UNIQUE NOT NULL,

        Password VARCHAR(100) NOT NULL

);


CREATE TABLE Customer (

        Person_ID INT PRIMARY KEY,

        Date_Sign_Up DATE,

        FOREIGN KEY (Person_ID) REFERENCES Person(Person_ID)

);


CREATE TABLE Employee (

        Person_ID INT PRIMARY KEY,
```

```sql
        Paycheck DECIMAL(10,2),

        FOREIGN KEY (Person_ID) REFERENCES Person(Person_ID)

);


CREATE TABLE Card (

        Card_Number INT PRIMARY KEY,

        Status ENUM('Active', 'Inactive') DEFAULT 'Active',

        Fines DECIMAL(8,2) DEFAULT 0.00,

        FOREIGN KEY (Card_Number) REFERENCES Person(Card_Number)

);


CREATE TABLE Branch (

        Branch_ID INT PRIMARY KEY AUTO_INCREMENT,

        Name VARCHAR(100) NOT NULL,

        Phone_Number VARCHAR(15)

);


CREATE TABLE Location (

        Location_ID INT PRIMARY KEY AUTO_INCREMENT,

        Address VARCHAR(255),

        Branch_ID INT,

        FOREIGN KEY (Branch_ID) REFERENCES Branch(Branch_ID)

);
```

```sql
CREATE TABLE Item (

        Item_ID INT PRIMARY KEY AUTO_INCREMENT,

        Availability BOOLEAN DEFAULT TRUE,

        State VARCHAR(50),

        Delay_Cost DECIMAL(6,2),

        Lost_Cost DECIMAL(6,2),

        Location_ID INT,

        FOREIGN KEY (Location_ID) REFERENCES Location(Location_ID)

);


CREATE TABLE Book (

        Item_ID INT PRIMARY KEY,

        ISBN VARCHAR(20) UNIQUE,

        FOREIGN KEY (Item_ID) REFERENCES Item(Item_ID)

);


CREATE TABLE Video (

        Item_ID INT PRIMARY KEY,

        Title VARCHAR(255),

        Year YEAR,

        FOREIGN KEY (Item_ID) REFERENCES Item(Item_ID)

);
```

```sql
CREATE TABLE Rent (

        Card_Number INT,

        Item_ID INT,

        Date DATE,

        Return_Date DATE,

        PRIMARY KEY (Card_Number, Item_ID, Date),

        FOREIGN KEY (Card_Number) REFERENCES Card(Card_Number),

        FOREIGN KEY (Item_ID) REFERENCES Item(Item_ID)

);


CREATE TABLE Work (

        Person_ID INT,

        Branch_ID INT,

        PRIMARY KEY (Person_ID, Branch_ID),

        FOREIGN KEY (Person_ID) REFERENCES Employee(Person_ID),

        FOREIGN KEY (Branch_ID) REFERENCES Branch(Branch_ID)

);


INSERT INTO Person (Name, Address, Phone_Number, Card_Number, Username, Password) VALUES

('Aarav Sharma', 'MG Road, Bengaluru, Karnataka', '9876543210', 1001, 'aaravsharma', 'pass123'),

('Sanya Mehta', 'Anna Salai, Chennai, Tamil Nadu', '9811122233', 1002, 'sanyamehta', 'pass234'),
```

('Rohan Das', 'Park Street, Kolkata, West Bengal', '9834567890', 1003, 'rohandas', 'pass345'),

('Meera Iyer', 'FC Road, Pune, Maharashtra', '9898765432', 1004, 'meera.iyer', 'pass456'),

('Aditya Verma', 'Hazratganj, Lucknow, Uttar Pradesh', '9911223344', 1005, 'adityav', 'pass567');

INSERT INTO Customer (Person_ID, Date_Sign_Up) VALUES

(1, '2023-05-20'),

(2, '2023-07-15'),

(3, '2024-01-10');

INSERT INTO Employee (Person_ID, Paycheck) VALUES

(4, 35000.00),

(5, 40000.00);

INSERT INTO Card (Card_Number, Status, Fines) VALUES

(1001, 'Active', 0.00),

(1002, 'Active', 20.00),

(1003, 'Inactive', 0.00),

(1004, 'Active', 15.00),

(1005, 'Active', 0.00);

INSERT INTO Branch (Name, Phone_Number) VALUES

('Bengaluru Central Library', '08022334455'),

('Chennai City Library', '04433445566');

INSERT INTO Location (Address, Branch_ID) VALUES

('MG Road, Bengaluru', 1), ('Jayanagar, Bengaluru', 1),

('T. Nagar, Chennai', 2), ('Velachery, Chennai', 2);

INSERT INTO Item (Availability, State, Delay_Cost, Lost_Cost, Location_ID) VALUES

(TRUE, 'New', 5.00, 300.00, 1),

(TRUE, 'Good', 4.00, 200.00, 2),

(FALSE, 'Damaged', 6.00, 250.00, 3),

(TRUE, 'Fair', 3.00, 150.00, 4);

INSERT INTO Book (Item_ID, ISBN) VALUES

(1, '9789354600854'),

(3, '9789385999380');

INSERT INTO Video (Item_ID, Title, Year) VALUES

(2, 'History of India', 2020),

(4, 'Yoga for Beginners', 2021);

INSERT INTO Rent (Card_Number, Item_ID, Date, Return_Date) VALUES

(1001, 1, '2024-03-10', '2024-03-20'),

(1002, 2, '2024-03-12', NULL),

(1003, 3, '2024-01-01', '2024-01-15');

INSERT INTO Work (Person_ID, Branch_ID) VALUES

(4, 1), (5, 2);

```sql
SELECT * FROM Card;
SELECT * FROM Customer;
SELECT * FROM Employee;
SELECT * FROM Branch;
SELECT * FROM Location;
SELECT * FROM Book;
SELECT * FROM Video;
SELECT * FROM Rent;

DELIMITER //
CREATE PROCEDURE UserLogin(
    IN p_username VARCHAR(50),
    IN p_password VARCHAR(100)
)
BEGIN
    DECLARE userCount INT;
    DECLARE userRole VARCHAR(20);

    SELECT COUNT(*), Role INTO userCount, userRole
    FROM USERS
    WHERE Username = p_username AND Password = p_password;

    IF userCount = 1 THEN
        SELECT CONCAT('Login successful. Role: ', userRole) AS Message;
    ELSE
        SELECT 'Invalid username or password' AS Message;
    END IF;
END;
//
DELIMITER ;

CALL UserLogin('cust1', 'custpass');
```

```sql
CALL UserLogin('admin1', 'wrongpass');


DELIMITER //

CREATE PROCEDURE RentItem (

        IN p_Card_Number INT,

        IN p_Item_ID INT,

        IN p_Date DATE

)

BEGIN

        DECLARE item_available BOOLEAN;


        SELECT Availability INTO item_available

        FROM Item WHERE Item_ID = p_Item_ID;


        IF item_available THEN

        INSERT INTO Rent (Card_Number, Item_ID, Date, Return_Date)

        VALUES (p_Card_Number, p_Item_ID, p_Date, NULL);


        UPDATE Item SET Availability = FALSE

        WHERE Item_ID = p_Item_ID;

        ELSE

        SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'Item is not available for
rent.';

        END IF;
```

```sql
END;

//

DELIMITER ;


DELIMITER //

CREATE PROCEDURE ReturnItem (

        IN p_Card_Number INT,

        IN p_Item_ID INT,

        IN p_Return_Date DATE

)

BEGIN

        DECLARE rent_date DATE;

        DECLARE delay_cost DECIMAL(6,2);

        DECLARE days_late INT DEFAULT 0;


        SELECT Date INTO rent_date

        FROM Rent WHERE Card_Number = p_Card_Number AND Item_ID =
p_Item_ID AND Return_Date IS NULL;


        IF rent_date IS NULL THEN

        SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'No active rent record
found.';

        ELSE

        SET days_late = DATEDIFF(p_Return_Date, rent_date) - 10; -- 10 days allowed
```

```
        IF days_late < 0 THEN SET days_late = 0; END IF;


        SELECT Delay_Cost INTO delay_cost

        FROM Item WHERE Item_ID = p_Item_ID;


        UPDATE Rent

        SET Return_Date = p_Return_Date

        WHERE Card_Number = p_Card_Number AND Item_ID = p_Item_ID AND
Return_Date IS NULL;


        UPDATE Item SET Availability = TRUE

        WHERE Item_ID = p_Item_ID;


        UPDATE Card

        SET Fines = Fines + (days_late * delay_cost)

        WHERE Card_Number = p_Card_Number;

        END IF;
END;
//
DELIMITER ;
```