import sqlite3

# Create database and tables

def create\_database():

conn = sqlite3.connect('library.db')

c = conn.cursor()

# Create Books table

c.execute('''CREATE TABLE IF NOT EXISTS Books (

BookID INTEGER PRIMARY KEY,

Title TEXT NOT NULL,

Author TEXT NOT NULL,

ISBN TEXT NOT NULL,

Status TEXT NOT NULL)''')

# Create Users table

c.execute('''CREATE TABLE IF NOT EXISTS Users (

UserID INTEGER PRIMARY KEY,

Name TEXT NOT NULL,

Email TEXT NOT NULL)''')

# Create Reservations table

c.execute('''CREATE TABLE IF NOT EXISTS Reservations (

ReservationID INTEGER PRIMARY KEY,

BookID INTEGER,

UserID INTEGER,

ReservationDate TEXT,

FOREIGN KEY (BookID) REFERENCES Books (BookID),

FOREIGN KEY (UserID) REFERENCES Users (UserID))''')

conn.commit()

conn.close()

# Add a new book to the database

def add\_book():

title = input("Enter the title of the book: ")

author = input("Enter the author of the book: ")

isbn = input("Enter the ISBN of the book: ")

status = "Available"

conn = sqlite3.connect('library.db')

c = conn.cursor()

c.execute("INSERT INTO Books (Title, Author, ISBN, Status) VALUES (?, ?, ?, ?)",

(title, author, isbn, status))

conn.commit()

conn.close()

print("Book added successfully!")

# Find a book's detail based on BookID

def find\_book\_details(book\_id):

conn = sqlite3.connect('library.db')

c = conn.cursor()

c.execute('''SELECT Books.BookID, Books.Title, Books.Author, Books.ISBN, Books.Status,

Users.UserID, Users.Name, Users.Email, Reservations.ReservationDate

FROM Books

LEFT JOIN Reservations ON Books.BookID = Reservations.BookID

LEFT JOIN Users ON Reservations.UserID = Users.UserID

WHERE Books.BookID = ?''', (book\_id,))

result = c.fetchone()

conn.close()

if result:

book\_details = {

'Book ID': result[0],

'Title': result[1],

'Author': result[2],

'ISBN': result[3],

'Status': result[4]

}

if result[5]:

user\_details = {

'User ID': result[5],

'Name': result[6],

'Email': result[7]

}

book\_details['Reserved By'] = user\_details

return book\_details

else:

return None

# Find a book's reservation status based on BookID, Title, UserID, or ReservationID

def find\_reservation\_status(text):

conn = sqlite3.connect('library.db')

c = conn.cursor()

if text.startswith('LB'): # BookID

c.execute('''SELECT Books.Status, Users.Name, Users.Email

FROM Books

LEFT JOIN Reservations ON Books.BookID = Reservations.BookID

LEFT JOIN Users ON Reservations.UserID = Users.UserID

WHERE Books.BookID = ?''', (text,))

elif text.startswith('LU'): # UserID

c.execute('''SELECT Books.Status, Books.Title, Books.Author, Books.ISBN

FROM Books

LEFT JOIN Res