CREATE TABLE Books (

book\_id NUMBER PRIMARY KEY,

title VARCHAR2(255),

author VARCHAR2(255),

genre VARCHAR2(100),

availability\_status VARCHAR2(20) DEFAULT 'Available'

);

CREATE TABLE Librarians (

librarian\_id NUMBER PRIMARY KEY,

name VARCHAR2(255),

email VARCHAR2(255) UNIQUE,

phone VARCHAR2(15)

);

CREATE TABLE Transactions (

transaction\_id NUMBER PRIMARY KEY,

book\_id NUMBER,

student\_id NUMBER,

issue\_date DATE DEFAULT SYSDATE,

return\_date DATE,

status VARCHAR2(20) DEFAULT 'Issued',

FOREIGN KEY (book\_id) REFERENCES Books(book\_id),

FOREIGN KEY (student\_id) REFERENCES Students(student\_id)

);

INSERT INTO Books (book\_id, title, author, genre) VALUES (1, 'The Alchemist', 'Paulo Coelho', 'Fiction');

INSERT INTO Books (book\_id, title, author, genre) VALUES (2, 'Atomic Habits', 'James Clear', 'Self-Help');

COMMIT;

INSERT INTO Students (student\_id, name, email, phone) VALUES (1, 'Alice', 'alice@gmail.com', '9876543210');

INSERT INTO Students (student\_id, name, email, phone) VALUES (2, 'Bob', 'bob@gmail.com', '8765432109');

COMMIT;

INSERT INTO Librarians (librarian\_id, name, email, phone) VALUES (1, 'John Doe', 'john.doe@gmail.com', '7896541230');

COMMIT;

INSERT INTO Transactions (transaction\_id, book\_id, student\_id, return\_date, status)

VALUES (1, 1, 1, NULL, 'Issued');

UPDATE Books SET availability\_status = 'Not Available' WHERE book\_id = 1;

COMMIT;

UPDATE Transactions

SET return\_date = SYSDATE, status = 'Returned'

WHERE transaction\_id = 1;

UPDATE Books SET availability\_status = 'Available' WHERE book\_id = 1;

COMMIT;

SELECT t.transaction\_id, b.title, s.name, t.issue\_date

FROM Transactions t

JOIN Books b ON t.book\_id = b.book\_id

JOIN Students s ON t.student\_id = s.student\_id

WHERE t.status = 'Issued';

CREATE OR REPLACE PROCEDURE IssueBook(p\_book\_id NUMBER, p\_student\_id NUMBER) AS

BEGIN

INSERT INTO Transactions (transaction\_id, book\_id, student\_id, status)

VALUES ((SELECT NVL(MAX(transaction\_id),0) + 1 FROM Transactions), p\_book\_id, p\_student\_id, 'Issued');

UPDATE Books SET availability\_status = 'Not Available' WHERE book\_id = p\_book\_id;

COMMIT;

END;

EXEC IssueBook(2, 2);

CREATE VIEW AvailableBooks AS

SELECT \* FROM Books WHERE availability\_status = 'Available';

SELECT \* FROM AvailableBooks;