

```

-- =====
-- Q1: LIBRARY MANAGEMENT SYSTEM
-- =====

CREATE TABLE Authors (
    AuthorID INT PRIMARY KEY,
    Name VARCHAR(100) NOT NULL,
    Country VARCHAR(50)
);

CREATE TABLE Books (
    BookID INT PRIMARY KEY,
    Title VARCHAR(100) NOT NULL,
    AuthorID INT,
    PublishedYear INT,
    Genre VARCHAR(50),
    FOREIGN KEY (AuthorID) REFERENCES Authors(AuthorID)
);

CREATE TABLE Borrowers (
    BorrowerID INT PRIMARY KEY,
    Name VARCHAR(100) NOT NULL,
    Contact VARCHAR(15)
);

CREATE TABLE BorrowedBooks (
    BorrowID INT PRIMARY KEY,
    BorrowerID INT,
    BookID INT,
    BorrowDate DATE,
    ReturnDate DATE,
    FOREIGN KEY (BorrowerID) REFERENCES Borrowers(BorrowerID),
    FOREIGN KEY (BookID) REFERENCES Books(BookID)
);

INSERT INTO Authors VALUES (1, 'George Orwell', 'UK');
INSERT INTO Authors VALUES (2, 'J.K. Rowling', 'UK');

```

```
INSERT INTO Authors VALUES (3, 'Dan Brown', 'USA');
INSERT INTO Authors VALUES (4, 'Chetan Bhagat', 'India');
INSERT INTO Authors VALUES (5, 'Haruki Murakami', 'Japan');

INSERT INTO Books VALUES (101, '1984', 1, 1949, 'Dystopian');
INSERT INTO Books VALUES (102, 'Animal Farm', 1, 1945,
'Satire');
INSERT INTO Books VALUES (103, 'Harry Potter', 2, 2001,
'Fantasy');
INSERT INTO Books VALUES (104, 'The Da Vinci Code', 3, 2003,
'Thriller');
INSERT INTO Books VALUES (105, 'Digital Fortress', 3, 1998,
'Thriller');
INSERT INTO Books VALUES (106, '2 States', 4, 2009,
'Romance');
INSERT INTO Books VALUES (107, 'The Alchemist', 5, 1988,
'Fiction');
INSERT INTO Books VALUES (108, 'Kafka on the Shore', 5, 2002,
'Fiction');
INSERT INTO Books VALUES (109, 'Norwegian Wood', 5, 1987,
'Fiction');
INSERT INTO Books VALUES (110, 'Half Girlfriend', 4, 2014,
'Romance');

INSERT INTO Borrowers VALUES (201, 'Alice', '1234567890');
INSERT INTO Borrowers VALUES (202, 'Bob', '9876543210');
INSERT INTO Borrowers VALUES (203, 'Charlie', '5555555555');
INSERT INTO Borrowers VALUES (204, 'David', '1111111111');
INSERT INTO Borrowers VALUES (205, 'Eva', '2222222222');

INSERT INTO BorrowedBooks VALUES (1, 201, 103, '2025-06-01',
NULL);
INSERT INTO BorrowedBooks VALUES (2, 202, 104, '2025-06-05',
'2025-06-15');
INSERT INTO BorrowedBooks VALUES (3, 203, 108, '2025-07-01',
NULL);
```

```
INSERT INTO BorrowedBooks VALUES (4, 201, 107, '2025-06-10',  
'2025-06-20');
```

```
-- Q1 Queries
```

```
SELECT * FROM Books WHERE PublishedYear > 2015;
```

```
SELECT DISTINCT b.Name  
FROM Borrowers b  
JOIN BorrowedBooks bb ON b.BorrowerID = bb.BorrowerID  
JOIN Books bk ON bb.BookID = bk.BookID  
WHERE bk.Genre = 'Fiction';
```

```
SELECT bk.Title, COUNT(*) AS BorrowCount  
FROM BorrowedBooks bb  
JOIN Books bk ON bb.BookID = bk.BookID  
GROUP BY bb.BookID  
ORDER BY BorrowCount DESC  
LIMIT 1;
```

```
SELECT DISTINCT a.Name  
FROM Authors a  
JOIN Books bk ON a.AuthorID = bk.AuthorID  
JOIN BorrowedBooks bb ON bk.BookID = bb.BookID  
WHERE bb.ReturnDate IS NULL;
```

```
SELECT Genre, COUNT(*) AS TotalBooks  
FROM Books  
GROUP BY Genre;
```

```
-- =====
```

```
-- Q2 - Q4
```

```
-- =====
```

```
CREATE TABLE Salesman (  
    salesman_id INT PRIMARY KEY,  
    name TEXT,
```

```
    city TEXT,
    commission REAL
);

CREATE TABLE Customer (
    customer_id INT PRIMARY KEY,
    cust_name TEXT,
    city TEXT,
    grade INT,
    salesman_id INT,
    FOREIGN KEY (salesman_id) REFERENCES Salesman(salesman_id)
);

CREATE TABLE Orders (
    ord_no INT PRIMARY KEY,
    purch_amt REAL,
    ord_date DATE,
    customer_id INT,
    salesman_id INT,
    FOREIGN KEY (customer_id) REFERENCES Customer(customer_id),
    FOREIGN KEY (salesman_id) REFERENCES Salesman(salesman_id)
);

INSERT INTO Salesman VALUES (5001, 'James Hoog', 'New York',
0.15);
INSERT INTO Salesman VALUES (5002, 'Nail Knite', 'Paris',
0.13);
INSERT INTO Salesman VALUES (5005, 'Pit Alex', 'London',
0.11);
INSERT INTO Salesman VALUES (5006, 'Mc Lyon', 'Paris', 0.14);
INSERT INTO Salesman VALUES (5003, 'Lauson Hen', 'San Jose',
0.12);
INSERT INTO Salesman VALUES (5007, 'Paul Adam', 'Rome', 0.13);

INSERT INTO Customer VALUES (3002, 'Nick Rimando', 'New York',
100, 5001);
```

```

INSERT INTO Customer VALUES (3005, 'Graham Zusi',
'California', 200, 5002);
INSERT INTO Customer VALUES (3001, 'Brad Guzan', 'London',
100, 5005);
INSERT INTO Customer VALUES (3004, 'Fabian Johnson', 'Paris',
300, 5006);
INSERT INTO Customer VALUES (3007, 'Brad Davis', 'New York',
200, 5001);
INSERT INTO Customer VALUES (3009, 'Geoff Cameron', 'Berlin',
100, 5003);
INSERT INTO Customer VALUES (3008, 'Julian Green', 'London',
300, 5002);
INSERT INTO Customer VALUES (3003, 'Jozy Altidor', 'Moscow',
200, 5007);

INSERT INTO Orders VALUES (70001, 150.50, '2012-10-05', 3005,
5002);
INSERT INTO Orders VALUES (70009, 270.65, '2012-09-10', 3001,
5005);
INSERT INTO Orders VALUES (70002, 65.26, '2012-10-05', 3002,
5001);
INSERT INTO Orders VALUES (70004, 110.50, '2012-08-17', 3009,
5003);
INSERT INTO Orders VALUES (70007, 948.50, '2012-09-10', 3005,
5002);
INSERT INTO Orders VALUES (70005, 2400.60, '2012-07-27', 3007,
5001);
INSERT INTO Orders VALUES (70008, 5760.00, '2012-09-10', 3002,
5001);
INSERT INTO Orders VALUES (70010, 1983.43, '2012-10-10', 3004,
5006);

-- Q2
SELECT *
FROM Customer
WHERE customer_id = 2001 AND salesman_id < (
    SELECT salesman_id FROM Salesman WHERE name = 'Mc Lyon'

```

```

);

-- Q3
SELECT
    c.cust_name AS CustomerName,
    c.city AS CustomerCity,
    s.name AS SalesmanName,
    s.city AS SalesmanCity,
    s.commission
FROM Customer c
JOIN Salesman s ON c.salesman_id = s.salesman_id
WHERE c.city != s.city AND s.commission > 0.12;

-- Q4
SELECT
    c.cust_name,
    c.city,
    o.ord_no,
    o.ord_date,
    o.purch_amt
FROM Orders o
LEFT JOIN Customer c ON o.customer_id = c.customer_id
WHERE o.customer_id IS NOT NULL;

```

Final\_output.txt

```

Charlie
Alice
Kafka on the Shore|1
J.K. Rowling
Haruki Murakami
Dystopian|1
Fantasy|1
Fiction|3
Romance|2
Satire|1
Thriller|2
Graham Zusij|California|Nail Knite|Paris|0.13
Julian Green|London|Nail Knite|Paris|0.13

```

Jozy Altidor|Moscow|Paul Adam|Rome|0.13  
Graham Zusi|California|70001|2012-10-05|150.5  
Brad Guzan|London|70009|2012-09-10|270.65  
Nick Rimando|New York|70002|2012-10-05|65.26  
Geoff Cameron|Berlin|70004|2012-08-17|110.5  
Graham Zusi|California|70007|2012-09-10|948.5  
Brad Davis|New York|70005|2012-07-27|2400.6  
Nick Rimando|New York|70008|2012-09-10|5760.0  
Fabian Johnson|Paris|70010|2012-10-10|1983.43