

# **PROJECT-2**

## **Part-3**

### **Library Management System**

**CSE3330-004**

**BY:**

Sarvesh Ingale, 1002071081, [smi1081@mavs.uta.edu](mailto:smi1081@mavs.uta.edu)  
Ajay Ahluwalia, 1001933276, [aaa3276@mavs.uta.edu](mailto:aaa3276@mavs.uta.edu)  
Austin Mitchell, 1001876256, [axm6256@mavs.uta.edu](mailto:axm6256@mavs.uta.edu)

## **HONOR CODE**

I pledge, on my honor, to uphold UT Arlington's tradition of academic integrity, a tradition that values hard work and honest effort in the pursuit of academic excellence.

I promise that I will submit only work that I personally create or that I contribute to group collaborations, and I will appropriately reference any work from other sources. I will follow the highest standards of integrity and uphold the spirit of the Honor Code.

Ajay Ahluwalia, 1001933276, [aaa3276@mavs.uta.edu](mailto:aaa3276@mavs.uta.edu)

Sarvesh Ingale, 1002071081, [smi1081@mavs.uta.edu](mailto:smi1081@mavs.uta.edu)

Austin Mitchell, 1001876256, [axm6256@mavs.uta.edu](mailto:axm6256@mavs.uta.edu)

## **Task-1**

### **Query 1:**

```
ALTER TABLE Book_Loans
```

```
ADD Late INT;
```

```
UPDATE Book_Loans
```

```
SET Late =
```

```
    CASE
```

```
        WHEN (Returned_date IS NULL OR Returned_date = 'NULL') AND  
(julianday(Date('now')) > julianday(due_date)) THEN 1
```

```
        WHEN julianday(Returned_date) > julianday(due_date) THEN 1
```

```
        ELSE 0
```

```
END;
```

```

sqlite> ALTER TABLE Book_Loans
ADD Late INT;

UPDATE Book_Loans
SET Late =
CASE
WHEN (Returned_date IS NULL OR Returned_date = 'NULL') AND (julianday(Date('now')) > julianday(due_date)) THEN 1
WHEN julianday(Returned_date) > julianday(due_date) THEN 1
ELSE 0
END;
sqlite> SELECT * FROM BOOK_LOANS;

```

Book_id	Branch_id	Card_no	Date_out	Due_date	Returned_date	Late
1	1	123456	2022-01-01	2022-02-01	2022-02-01	0
2	1	789012	2022-01-02	2022-02-02	NULL	1
3	2	345678	2022-01-03	2022-02-03	NULL	1
4	3	901234	2022-01-04	2022-02-04	2022-02-04	0
5	1	567890	2022-01-05	2022-02-05	2022-02-09	1
6	2	234567	2022-01-06	2022-02-06	2022-02-10	1
7	2	890123	2022-01-07	2022-02-07	2022-03-08	1
8	3	456789	2022-01-08	2022-02-08	2022-03-10	1
9	1	111111	2022-01-09	2022-02-09	2022-02-06	0
10	2	222222	2022-01-10	2022-02-10	2022-02-07	0
11	1	333333	2022-03-01	2022-03-08	2022-03-08	0
12	3	444444	2022-03-03	2022-03-10	2022-03-10	0
13	3	555555	2022-02-03	2022-03-03	2022-02-18	0
14	1	565656	2022-01-14	2022-02-14	2022-03-31	1
15	3	676767	2022-01-15	2022-02-15	2022-02-21	1
16	2	787878	2022-03-05	2022-03-12	2022-03-24	1
17	3	989898	2022-03-23	2022-03-30	2022-03-30	0
18	3	121212	2022-01-18	2022-02-18	2022-02-18	0
19	1	232323	2022-03-24	2022-03-31	2022-03-31	0
20	3	343434	2022-01-21	2022-02-21	2022-02-21	0
21	3	454545	2022-01-24	2022-02-24	2022-02-24	0

Rows Affected = 21

## Query 2:

```
ALTER Table LIBRARY_BRANCH
```

```
ADD Late_fee REAL;
```

```
UPDATE LIBRARY_BRANCH
```

```
SET Late_fee =
```

```
CASE
```

```
WHEN Branch_id = 1 THEN 2
```

```
WHEN Branch_id = 2 THEN 5
```

```
WHEN Branch_id = 3 THEN 1
```

```
ELSE 2 -- Default Late_fee for a branch
```

```
END;
```

```

sqlite> ALTER Table LIBRARY_BRANCH
ADD Late_fee REAL;

UPDATE LIBRARY_BRANCH
SET Late_fee =
CASE
    WHEN Branch_id = 1 THEN 2
    WHEN Branch_id = 2 THEN 5
    WHEN Branch_id = 3 THEN 1
    ELSE 2 -- Default Late_fee for a branch
END;
sqlite> SELECT * FROM LIBRARY_BRANCH;

```

Branch_id	Branch_name	Branch_address	Late_fee
1	Main Branch	123 Main St, New York, NY 10003	2.0
2	West Branch	456 West St, Arizona, AR 70622	5.0
3	East Branch	789 East St, New Jersey, NY 32032	1.0

Rows Affected = 3

### Query 3:

```

CREATE VIEW BookLoanInfo AS
SELECT BO.Card_No, BO.Name AS Borrower_name, BL.Date_out, BL.Due_Date,
BL.Returned_Date, B.Title AS Book_title,
(CASE
    WHEN (BL.Returned_date = 'NULL' OR BL.Returned_date IS NULL) THEN
julianday(Date('now')) - julianday(Date_out)
    WHEN Returned_date > Date_out
    THEN julianday(BL.Returned_date) - julianday(BL.Date_out)
    ELSE
0
END
) AS TotalDays,
(CASE
    WHEN (BL.Returned_date = 'NULL' OR BL.Returned_date IS NULL) AND
(julianday(Date('now')) > julianday(BL.Due_date)) THEN
(julianday(Date('now')) - julianday(BL.Due_date))
    WHEN
julianday(BL.returned_date) > julianday(BL.due_date) THEN
julianday(BL.returned_date) - julianday(BL.Due_date)
    ELSE
0
END) AS Late_days,
BL.Branch_id,
( CASE

```

```

        WHEN (BL.Returned_date = 'NULL' OR BL.Returned_date IS NULL) AND
(julianday(Date('now')) > julianday(BL.Due_date)) THEN
(julianday(Date('now')) - julianday(BL.Due_date)) * LB.Late_fee
    WHEN
        julianday(BL.Returned_date) > julianday(BL.Due_date) THEN
        (julianday(BL.Returned_date) - julianday(BL.Due_date)) *
LB.Late_fee
    ELSE
    0
END) AS LateFeeBalance
FROM Library_Branch LB JOIN Book_Loans BL ON LB.Branch_id =
BL.Branch_id
JOIN Borrower BO ON BL.Card_no = BO.Card_no JOIN Book B ON BL.Book_id =
B.Book_id;

```

```
sqlite> select * from BookLoanInfo;
```

Card_No	Borrower_name	Date_out	Due_date	Returned_date	Book_title	TotalDays	Late_days	Branch_id	LateFeeBalance
123456	John Smith	2022-01-01	2022-02-01	2022-02-01	To Kill a Mockingbird	31.0	0	1	0
789012	Jane Doe	2022-01-02	2022-02-02	NULL	1984	1066.0	1035.0	1	2070.0
345678	Bob Johnson	2022-01-03	2022-02-03	NULL	Pride and Prejudice	1065.0	1034.0	2	5170.0
901234	Sarah Kim	2022-01-04	2022-02-04	2022-02-04	The Great Gatsby	31.0	0	3	0
567890	Tom Lee	2022-01-05	2022-02-05	2022-02-09	One Hundred Years of Solitude	35.0	4.0	1	8.0
234567	Emily Lee	2022-01-06	2022-02-06	2022-02-10	Animal Farm	35.0	4.0	2	20.0
890123	Michael Park	2022-01-07	2022-02-07	2022-03-08	The Catcher in the Rye	60.0	29.0	2	145.0
456789	Laura Chen	2022-01-08	2022-02-08	2022-03-10	Lord of the Flies	61.0	30.0	3	30.0
111111	Alex Kim	2022-01-09	2022-02-09	2022-02-06	Brave New World	28.0	0	1	0
222222	Rachel Lee	2022-01-10	2022-02-10	2022-02-07	The Picture of Dorian Gray	28.0	0	2	0
333333	William Johnson	2022-03-01	2022-03-08	2022-03-08	The Alchemist	7.0	0	1	0
444444	Ethan Martinez	2022-03-03	2022-03-10	2022-03-10	The God of Small Things	7.0	0	3	0
555555	Grace Hernandez	2022-02-03	2022-03-03	2022-02-18	Wuthering Heights	15.0	0	3	0
565656	Sophia Park	2022-01-14	2022-02-14	2022-03-31	The Hobbit	76.0	45.0	1	90.0
676767	Olivia Lee	2022-01-15	2022-02-15	2022-02-21	The Lord of the Rings	37.0	6.0	3	6.0
787878	Noah Thompson	2022-03-05	2022-03-12	2022-03-24	The Hitchhiker's Guide to the Galaxy	19.0	12.0	2	60.0
989898	Olivia Smith	2022-03-23	2022-03-30	2022-03-30	The Diary of a Young Girl	7.0	0	3	0
121212	Chloe Park	2022-01-18	2022-02-18	2022-02-18	The Da Vinci Code	31.0	0	3	0
232323	William Chen	2022-03-24	2022-03-31	2022-03-31	The Adventures of Huckleberry Finn	7.0	0	1	0
343434	Olivia Johnson	2022-01-21	2022-02-21	2022-02-21	The Adventures of Tom Sawyer	31.0	0	3	0
454545	Dylan Kim	2022-01-24	2022-02-24	2022-02-24	A Tale of Two Cities	31.0	0	3	0

Rows Returned = 21

## Task-2

### Query 1:

Checkout Book

Provide Book ID, BranchID, Card\_no, Date\_out, and Due\_date to checkout

Book ID:

9

Branch ID:

1

Card\_no

787878

Date\_out

2024-12-03

Due\_date

2024-12-08

Add New Loan

Branch ID	Branch Name	No Of Copies
9	1	3

Loan Added!

```
CREATE TRIGGER IF NOT EXISTS UpdateBookCopiesOnLoan
AFTER INSERT ON BOOK_LOANS
FOR EACH ROW
BEGIN
    UPDATE BOOK_COPIES
    SET No_Of_Copies = No_Of_Copies - 1
```

```
        WHERE Book_id = NEW.Book_id AND Branch_id =  
NEW.Branch_id;  
    END;
```

```
INSERT INTO BOOK_LOANS VALUES (?, ?, ?, ?, ?, ?, ?)
```

```
SELECT * FROM BOOK_COPIES WHERE Book_id = ? AND Branch_id = ?
```

## Query 2: Adding new borrower

Add New Borrower

Provide Name, Address, Phone(Not necessary) to register

Borrower Name:

Stardust

Borrower Address:

1220 Nemo Street

Borrower Phone

123-678-2346

Add New Borrower

Your Library Card

Borrower ID	Name	Address	Phone
989903	Stardust	1220 Nemo Street	123-678-2346

Borrower Added Successfully!

sqlite> select * from borrower;			
Card_No	Name	Address	Phone
111111	Alex Kim	983 Sine St, Arizona, AR 70451	678-784-5563
121212	Chloe Park	345 Shark St, Arizona, AR 72213	755-905-5572
123456	John Smith	456 Oak St, Arizona, AR 70010	205-555-5555
222222	Rachel Lee	999 Apple Ave, Arizona, AR 70671	231-875-5564
232323	William Chen	890 Sting St, New York, NY 10459	406-755-5580
234567	Emily Lee	389 Oaklay St, Arizona, AR 70986	231-678-5560
333333	William Johnson	705 Paster St, New Jersey 32002	235-525-5567
343434	Olivia Johnson	345 Pine St, New Jersey, NJ 32095	662-554-5575
345678	Bob Johnson	12 Elm St, Arizona, AR 70345	545-234-5557
444444	Ethan Martinez	466 Deeplm St, New York, NY 10321	555-555-5569
454545	Dylan Kim	567 Cowboy way St, New Jersey, NJ 32984	435-254-5578
456789	Laura Chen	345 Mapman Ave, Arizona, AR 70776	565-985-9962
555555	Grace Hernandez	315 Babes St, Arizona, AR 70862	455-567-5587
565656	Sophia Park	678 Dolphin St, New York, NY 10062	675-455-5568
567890	Tom Lee	678 S Oak St, New York, NY 10045	209-525-5559
676767	Olivia Lee	345 Spine St, New York, NY 10092	435-878-5569
787878	Noah Thompson	189 GreenOak Ave, New Jersey, NJ 32453	245-555-5571
789012	Jane Doe	789 Maple Ave, New Jersey, NJ 32542	555-235-5556
890123	Michael Park	123 Pinewood St, New Jersey, NJ 32954	655-890-2161
901234	Sarah Kim	345 Pine St, New York, NY 10065	515-325-2158
989898	Olivia Smith	178 Elm St, New Jersey, NJ 32124	325-500-5579
989899	Ajay	8785 copper	3339998888
989900	Austin	5879 mitchell street	682-857-8567

INSERT INTO BORROWER VALUES (?, ?, ?, ?)



### Query 3: Add a new book

Add New Book

Provide Publisher (already exists) and Author to add new book

Publisher Name:

Scribner

Author Name:

George Orwell

Book Title:

Percy Jackson

Add New Book

Book_id	Title	Publisher_name
1	To Kill a Mockingbird	HarperCollins
2	1984	Penguin Books
3	Pride and Prejudice	Penguin Classics
4	The Great Gatsby	Scribner
5	One Hundred Years of Solitude	Harper & Row
6	Animal Farm	Penguin Books
7	The Catcher in the Rye	Little, Brown and Company
8	Lord of the Flies	Faber and Faber
9	Brave New World	Chatto & Windus
10	The Picture of Dorian Gray	Ward, Lock and Co.
11	The Alchemist	HarperCollins
12	The God of Small Things	Random House India
13	Wuthering Heights	Thomas Cautley Newby
14	The Hobbit	Allen & Unwin
15	The Lord of the Rings	Allen & Unwin
16	The Hitchhiker's Guide to the Galaxy	Pan Books
17	The Diary of a Young Girl	Bantam Books
18	The Da Vinci Code	Doubleday
19	The Adventures of Huckleberry Finn	Penguin Classics
20	The Adventures of Tom Sawyer	American Publishing Company
21	A Tale of Two Cities	Chapman and Hall
22	Harry Potter and the Sorcerers Stone	Oxford Publishing
23	Where the wild things are	HarperCollins
24	Percy Jackson	Scribner

```
INSERT INTO BOOK(Title, Publisher_name) VALUES (?, ?)
```

```
INSERT INTO BOOK_AUTHORS(Book_id, Author_name) VALUES (?, ?)
```

```
SELECT Branch_id FROM LIBRARY_BRANCH
```

```
INSERT INTO BOOK_COPIES VALUES (?, ?, ?)
```

#### Query 4: Find loaned out copies

List Loaned Copies Per Branch

Provide Book Title to get listed all loaned copies per branch

Book Title:

A Tale of Two Cities

View List

Branch ID	Branch Name	Copies Loaned
3	East Branch	1

List Retrieved Successfully!

.

```
SELECT LB.Branch_id, LB.Branch_name, COUNT(*) AS Copies_loaned
      FROM BOOK_LOANS BL JOIN
LIBRARY_BRANCH LB ON BL.Branch_id = LB.Branch_id
      JOIN BOOK B ON BL.Book_id =
B.Book_id
```

```
WHERE B.Title = ?
GROUP BY LB.Branch_id,
LB.Branch_name
```

**Query 5: Give date range to find due books in this range**

```
SELECT Book_id, Branch_id, Card_no, Date_out, Due_date, Returned_date,
       (julianday(Returned_date) - julianday(Due_date)) AS
Days_Late
FROM BOOK_LOANS
WHERE Due_date BETWEEN ? AND ? AND Returned_date > Due_date
ORDER BY Days_Late DESC
"""
```



**Query 6a: Shows late fee of borrowers depending on inserted info or input no information and it will display all**

List Late Fee for Borrower(s)

Provide borrower id, name, or part-name to get late fee for borrower or leave it blank to see latefee for all borrowers

Borrower ID [Card\_on] (Blank Allowed)

Borrower Name or Part-name (Blank Allowed)

View List

Borrower ID	Borrower Name	LateFeeBalance
345678	Bob Johnson	\$5,175.00
789012	Jane Doe	\$2,072.00
890123	Michael Park	\$145.00
565656	Sophia Park	\$90.00
787878	Noah Thompson	\$60.00
456789	Laura Chen	\$30.00
234567	Emily Lee	\$20.00
567890	Tom Lee	\$8.00
676767	Olivia Lee	\$6.00
111111	Alex Kim	\$0.00
121212	Chloe Park	\$0.00
123456	John Smith	\$0.00
222222	Rachel Lee	\$0.00
232323	William Chen	\$0.00
333333	William Johnson	\$0.00
343434	Olivia Johnson	\$0.00
444444	Ethan Martinez	\$0.00
454545	Dylan Kim	\$0.00
555555	Grace Hernandez	\$0.00
901234	Sarah Kim	\$0.00
989898	Olivia Smith	\$0.00

List Retrieved Successfully!

```

SELECT Card_no as Borrower_ID, Borrower_name, SUM(LateFeeBalance) as
LateFeeBalance

FROM BookLoanInfo
GROUP BY Card_no,
Borrower_name

ORDER BY LateFeeBalance DESC'''

```

## Query 6b: Shows book information when card number of Borrower inserted

List Book Info for Borrower(s)

Provide Borrower ID (necessary) and any of book id, book title or part-title

Borrower ID [Card\_on] (Necessary)

456789

Book ID (Blank Allowed)

Book Title (Blank Allowed)

View List

Book ID	Book Title	LateFeeBalance
8	Lord of the Flies	\$30.00

List Retrieved Successfully!

```
SELECT B.Book_id, BLI.Book_title, BLI.LateFeeBalance
      FROM BookLoanInfo BLI
      JOIN Book_Loans BL ON
BLI.Card_No = BL.Card_no
      JOIN Book B ON BL.Book_id =
B.Book_id
      WHERE BLI.Card_no = ?
      ORDER BY BLI.LateFeeBalance
DESC
```

## Contribution List

Task 1 Query: Sarvesh Ingale, Ajay Ahluwalia, Austin Mitchell

Task 2 Gui creation: Sarvesh Ingale, Ajay Ahluwalia, Austin Mitchell

Document Creation: Austin Mitchell, Sarvesh Ingale, Ajay Ahluwalia