**CSE 3330 – 004 (Project 2 – Part 3)**

**GROUP 14 (Araohat Kokate, Inshaad Merchant , Athrva Arora)**

**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

**TASK1:** We are using view.sql file for displaying the given tables.

view.sql –

.headers on

.mode column

SELECT \* FROM BOOK\_LOANS ;

Query 1:

ALTER TABLE BOOK\_LOANS

ADD COLUMN Late INT;

UPDATE BOOK\_LOANS

SET Late = CASE

    WHEN Returned\_Date IS NULL OR Due\_Date IS NULL THEN NULL

    WHEN Returned\_Date > Due\_Date THEN 1

    WHEN Returned\_Date > Due\_Date THEN 1

    ELSE 0

END;

Output :

A screenshot of a computer screen

Description automatically generated

view.sql –

.headers on

.mode column

SELECT \* FROM LIBRARY\_BRANCH ;

Query2:

ALTER TABLE LIBRARY\_BRANCH

ADD COLUMN LateFee INT;

UPDATE LIBRARY\_BRANCH

SET LateFee = CASE

    WHEN Branch\_id = 1 THEN 1

    WHEN Branch\_id = 2 THEN 2

    WHEN Branch\_id = 3 THEN 3

    WHEN Branch\_id = 4 THEN 4

    WHEN Branch\_id = 5 THEN 5

    ELSE 0

END;

Output:

A computer screen shot of address

Description automatically generated

view.sql –

.headers on

.mode column

SELECT \* FROM vBookLoanInfo ;

Query3:

DROP VIEW IF EXISTS vBookLoanInfo;

CREATE VIEW vBookLoanInfo AS

SELECT

    bl.Card\_no,

    b.Name AS "Borrower Name",

    bl.Date\_out,

    bl.Due\_Date,

    bl.Returned\_date,

    CAST((julianday(bl.Returned\_date) - julianday(bl.Date\_out)) AS INTEGER) AS TotalDays,

    bo.Title AS "Book Title",

    ba.Book\_id,

    CASE

        WHEN bl.Returned\_date <= bl.Due\_Date THEN 0

        ELSE CAST((julianday(bl.Returned\_date) - julianday(bl.Due\_Date)) AS INTEGER)

    END AS "Number of days returned late",

    bl.Branch\_id,

    CASE

        WHEN bl.Returned\_date <= bl.Due\_Date THEN 0

        ELSE CAST((julianday(bl.Returned\_date) - julianday(bl.Due\_Date)) AS INTEGER) \* lb.LateFee

    END AS LateFeeBalance

FROM

    BOOK\_LOANS bl

JOIN BORROWER b ON bl.Card\_no = b.Card\_no

JOIN BOOK bo ON bl.Book\_id = bo.Book\_id

JOIN BOOK\_AUTHORS ba ON bo.Book\_id = ba.Book\_id

JOIN LIBRARY\_BRANCH lb ON bl.Branch\_id = lb.Branch\_id;

Output:

A black screen with white text

Description automatically generated

**TASK 2:**

1. **Query 1**

Editable Queries Used :

SELECT B.Book\_id, B.Title, COUNT(BC.no\_of\_copies) AS per\_Book\_copies, BC.Branch\_Id

        FROM BOOK\_COPIES as BC

        INNER JOIN BOOK as B ON B.Book\_id = BC.Book\_id

        GROUP BY B.Book\_id, BC.Branch\_Id

Output - A screenshot of a computer

Description automatically generated

Explanation –

There are total of 3 copies of Book with Book\_id – 1 at the Branch\_id – 1 and when we checkout one of the copies and it gives us the remaining number of copies :2 as the output.

1. **Query 2**

Editable Queries Used:

"INSERT INTO BORROWER(Name, address, phone) VALUES (:Name, :address, :phone)",

                   {

                       'Name': name.get(),

                       'address': addr.get(),

                       'phone': phone.get()

                   }

Output:

**A screenshot of a computer

Description automatically generated**

Explanation:

Since Card\_no is Autoincremented when we add a borrower with no Card\_no it will automatically assign it a new Card\_no which will be one greater than the Highest Card\_no value.

1. **Query 3**

Editable Queries Used:

'INSERT INTO BOOK (Title, Publisher\_Name) VALUES (?, ?)', (book\_title\_t, publisher\_name\_t)

'INSERT INTO BOOK\_AUTHORS (Book\_Id, Author\_Name) VALUES (?, ?)', (book\_id, author\_name\_t)

'INSERT INTO BOOK\_COPIES (Book\_Id, Branch\_Id, no\_of\_copies) VALUES (?, ?, ?)', (book\_id, i, 5)

Output:

A screenshot of a computer

Description automatically generated

A computer screen shot of a black screen

Description automatically generated

Explanation:

It can be confirmed from the table BOOK in lms.db that the entry was added.

1. **Query 4**

Editable SQL Queries –

SELECT Branch\_Name, COUNT(\*) FROM BOOK\_LOANS as BL,LIBRARY\_BRANCH as LB, BOOK as B WHERE BL.Branch\_id = LB.Branch\_id AND BL.Book\_id = B.Book\_id AND Title like ? GROUP BY Branch\_name",(t,)

Output:

A screenshot of a computer

Description automatically generated

Explanation:

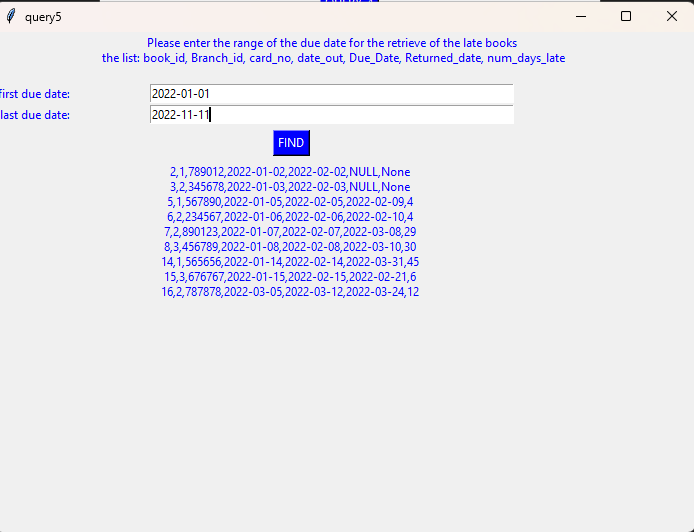
There is only 1 book at branch “Main Branch” of the book named 1984 it can also be seen in BOOK\_LOANS Table as well.

1. **Query 5**

Editable SQL Query-

SELECT Book\_id, Branch\_id, Card\_No, Date\_Out, Due\_Date, Returned\_date, CAST(JULIANDAY(Returned\_date) AS INTEGER) - CAST(JULIANDAY(Due\_Date)AS INTEGER) AS num\_days\_late FROM BOOK\_LOANS WHERE Late = '1' AND (Due\_Date < ? AND   Due\_Date > ?) "

Output-



Explanation-

There are a total of 9 Books that have Due Dates between “2022-01-01” and “2022-11-11” and the attributes Book\_id, Branch\_id, Card\_No, Date\_Out, Due\_Date, Returned\_date and Number of Days Late have been listed in the above output.

1. Query 6a –

Editable Sql Query-

When Both Borrower\_id and Name is Entered:

"SELECT DISTINCT vBookLoanInfo.Card\_no, BORROWER.Name, IFNULL(LateFeeBalance, 0) "

"FROM vBookLoanInfo "

"LEFT JOIN BORROWER ON vBookLoanInfo.Card\_no = BORROWER.Card\_no "

"WHERE vBookLoanInfo.Card\_no = ? AND BORROWER.Name LIKE '%'||?||'%'"

When Just Borrower\_id is entered:

"SELECT DISTINCT vBookLoanInfo.Card\_no, BORROWER.Name, IFNULL(LateFeeBalance, 0) "

"FROM vBookLoanInfo "

"LEFT JOIN BORROWER ON vBookLoanInfo.Card\_no = BORROWER.Card\_no "

"WHERE vBookLoanInfo.Card\_no = ?"

When Just Name is entered:

"SELECT DISTINCT vBookLoanInfo.Card\_no, BORROWER.Name, IFNULL(LateFeeBalance, 0) "

"FROM BORROWER "

"LEFT JOIN vBookLoanInfo ON BORROWER.Card\_no = vBookLoanInfo.Card\_no "

"WHERE BORROWER.Name LIKE '%'||?||'%'"

When Nothing is Entered:

"SELECT DISTINCT vBookLoanInfo.Card\_no, BORROWER.Name, IFNULL(LateFeeBalance, 0) "

"FROM vBookLoanInfo "

"LEFT JOIN BORROWER ON vBookLoanInfo.Card\_no = BORROWER.Card\_no "

"ORDER BY LateFeeBalance DESC"

Output –

When Nothing is Entered:

A screenshot of a search box

Description automatically generated

When Just Name is entered:

A screenshot of a computer

Description automatically generated

When Just Borrower\_id is entered:

A screenshot of a computer

Description automatically generated

When Both Borrower\_id and Name is Entered:

A screenshot of a computer

Description automatically generated

1. Query 6b-

Editable Query –

"SELECT v.Card\_no, BORROWER.name, b.Title, "

"CASE WHEN LateFeeBalance IS NULL THEN 'Non-Applicable' "

"ELSE '$' || printf('%.2f', LateFeeBalance) END AS LateFeeBalance "

"FROM vBookLoanInfo v "

"JOIN BOOK b ON v.Book\_id = b.Book\_id "

"JOIN BORROWER ON v.Card\_no = BORROWER.Card\_no "

"WHERE v.Card\_no = ? OR v.Book\_id = ? OR b.Title LIKE '%' || ? || '%' "

"ORDER BY LateFeeBalance DESC",

(borrower\_id, book\_info, book\_info)

Output –

When Nothing is Entered -

A screenshot of a computer

Description automatically generated

When Partial Name is Entered with Borrower Id–

A screenshot of a computer

Description automatically generated

When Book\_id is Entered with Borrower Id-

A screenshot of a computer

Description automatically generated

**TEAM CONTRIBUTIONS**

1. GUI Frontend- Inshaad Merchant
2. GUI Backend- Athrva Arora and Araohat Kokate
3. Task 1 –
4. Query1- Inshaad Merchant
5. Query2- Araohat Kokate
6. Query3- Athrva Arora
7. Task 2 –
8. Query 1- Inshaad Merchant
9. Query 2- Araohat Kokate
10. Query 3- Athrva Arora
11. Query 4- Inshaad Merchant
12. Query 5- Araohat Kokate
13. Query 6- Athrva Arora and Inshaad Merchant and Araohat Kokate
14. README – Athrva Arora
15. Report - Athrva Arora and Araohat Kokate