



SQL Mini Project

Library Management System

Problem Statement

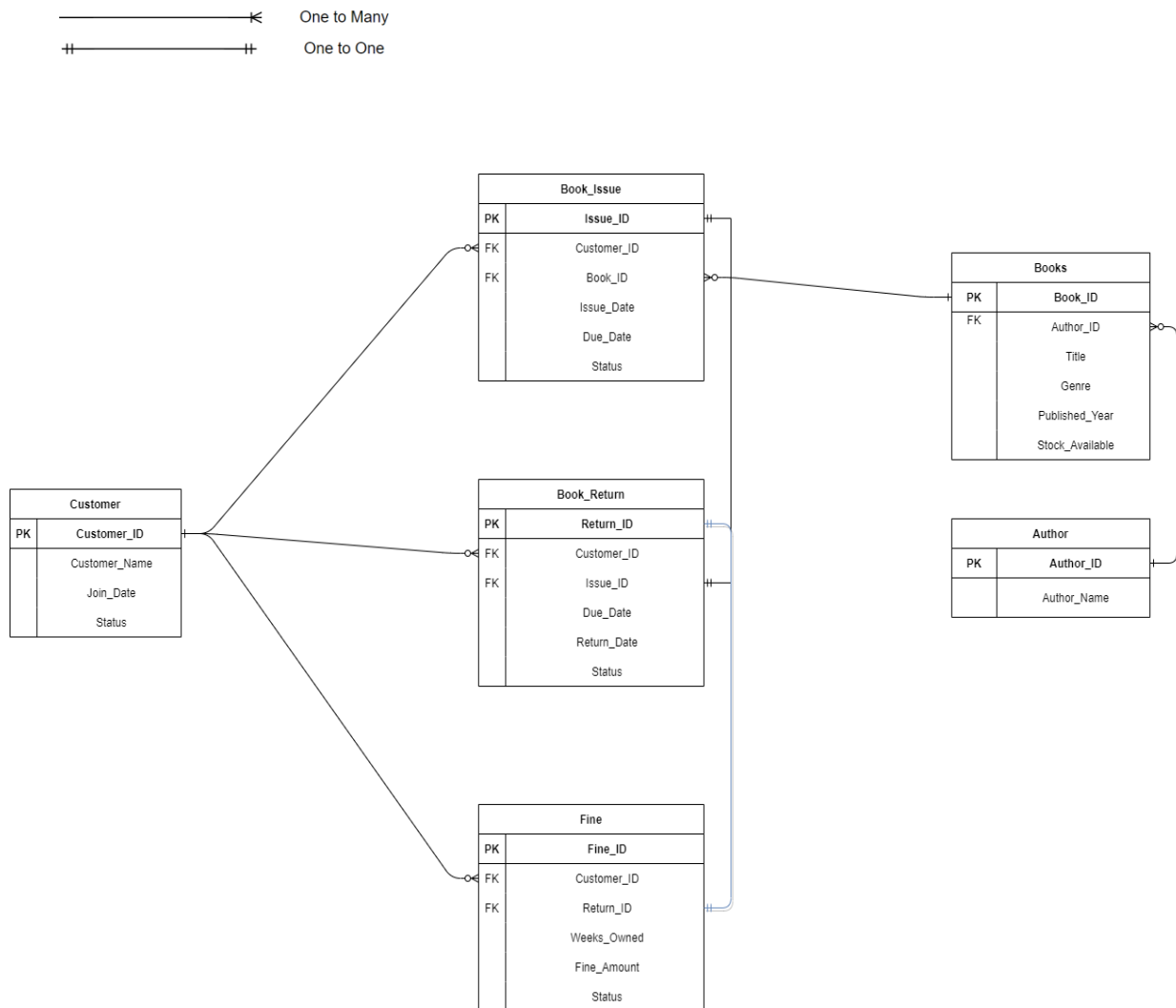
**Query Design to Solve 5 Major Applications Used
in Library Management**

Developed & programmed by:

Rithik Tripathi

Entity Relationship (ER) Diagram

Library_Database



Tables Creation DDL Commands

```
CREATE TABLE Book
(
    Book_ID INT PRIMARY KEY IDENTITY(100,1),
    Author_ID INT NOT NULL,
    Title NVARCHAR(40) NOT NULL UNIQUE,
    Genre NVARCHAR(20),
    Published_Year INT,
    Stock INT NOT NULL CHECK(Stock >= 0),
    FOREIGN KEY(Author_ID) REFERENCES Author(Author_ID)
)
```

```
CREATE TABLE Author
(
    Author_ID INT PRIMARY KEY IDENTITY(1,1),
    Author_Name NVARCHAR(40) NOT NULL UNIQUE
)
```

```
CREATE TABLE Customer
(
    Customer_ID INT PRIMARY KEY IDENTITY(10,1),
    Customer_Name NVARCHAR(40) NOT NULL,
    Join_Date DATE,
    Customer_Status NVARCHAR(15) NOT NULL
)
```

```
CREATE TABLE Issue
(
    Issue_ID INT PRIMARY KEY IDENTITY(10,1),
    Customer_ID INT NOT NULL,
    Book_ID INT NOT NULL,
    Issue_Date DATE NOT NULL,
    Due_Date DATE,
    Status NVARCHAR(10),
    FOREIGN KEY(Customer_ID) REFERENCES Customer(Customer_ID)
)
```

```
CREATE TABLE Submit
(
    Return_ID INT PRIMARY KEY IDENTITY(10,1),
    Customer_ID INT NOT NULL,
    Issue_ID INT NOT NULL,
    Due_Date DATE,
    Return_Date DATE NOT NULL,
    Status NVARCHAR(10),
    FOREIGN KEY(Customer_ID) REFERENCES Customer(Customer_ID)
)
```

```
CREATE TABLE fine
(
    Fine_ID INT PRIMARY KEY IDENTITY(10,1),
    Return_ID INT NOT NULL,
    Customer_ID INT NOT NULL,
    Weeks_Owned INT NOT NULL,
    Fine_Amount INT NOT NULL,
    Status NVARCHAR(10),
    FOREIGN KEY(Customer_ID) REFERENCES Customer(Customer_ID)
)
```

Tables & Description

Customer_ID	Customer_Name	Join_Date	customer_Status
10	Rithik Tripathi	2020-10-15	Active
11	Krati Tripathi	2020-10-15	Active
12	Aditya Singh	2012-05-20	Active
13	Nikita Meena	2020-08-31	Active
14	Abhinav Anand	2022-09-06	Active
15	Manish Pandey	2022-02-16	Active
16	Surya Singh	2019-11-18	In Active
17	Aman Sharma	2012-12-12	Active
18	Nidhi Kumawat	2021-06-17	Suspended
19	Mahesh Saini	2019-01-01	Suspended

Customer_ID : unique ID given to each customer.

Customer_Name : Full name of customers

Join_Date : Joining date of customers to endorse loyalty in future.

Customer_Status : status of customers if they are Active, In-Active or Suspended due reason.

Book_ID	Author_ID	Title	Genre	Published_Year	Stock
100	1	Half Girlfriend	Romance	2014	20
101	1	2 States	Romance	2009	28
102	1	400 Days	Mystery	2021	12
103	1	One Indian Girl	Romance	2016	26
104	2	Romeo and Juliet	Romance	1597	2
105	3	The Alchemist	Adventure	1988	9
106	3	Eleven Minutes	Fiction	2003	5
107	4	Harry Potter and the Philosopher's Stone	Fiction	1997	50
108	4	Harry Potter and the Chamber of Secrets	Fiction	1998	50
109	4	Harry Potter and the Prisoner of Azkaban	Fiction	1999	50
110	4	Harry Potter and the Goblet of Fire	Fiction	2000	50
111	4	Harry Potter and the Order of the Phoenix	Fiction	2003	50
112	4	Harry Potter and the Half-Blood Prince	Fiction	2005	50
113	4	Harry Potter and the Deathly Hallows	Fiction	2007	50
114	5	Azadi	Fiction	2020	5
115	5	War Talk	Fiction	2003	3
116	5	The Cost of Living	Thriller	1999	8

Book_ID : unique ID for each Book.

Author_ID : ID to uniquely identify author of book. There can be multiple authors for 1 book.

Title : Name of the book.

Genre : Genre of the book.

Published_Year : Publishing year of book

Stock : Number of copies available for each book. Updated Dynamically with Issue & Return.

Author_Name	Author_ID
Chetan Bhagat	1
William Shakespeare	2
Paulo Coelho	3
J K Rowling	4
Arundhati Roy	5

Author_Name : Name of author

Author_ID : ID to uniquely identify author of book. There can be multiple authors for 1 book.

Issue_ID	Customer_ID	Book_ID	Issue_Date	Due_Date	Status
10	10	101	2022-11-15	2022-11-22	Issued
11	10	105	2022-11-15	2022-11-22	Issued
12	11	104	2022-11-15	2022-11-22	Issued
13	12	106	2022-11-17	2022-11-24	Issued
14	13	107	2022-11-20	2022-11-27	Issued
15	13	108	2022-11-20	2022-11-27	Issued
16	10	116	2022-11-20	2022-11-27	Issued
17	14	100	2022-11-21	2022-11-28	Issued
18	15	106	2022-11-24	2022-11-30	Issued
19	11	100	2022-11-24	2022-11-30	Issued
20	17	114	2022-11-26	2022-12-02	Issued
21	17	115	2022-11-26	2022-12-02	Issued
22	18	111	2022-09-01	2022-09-08	Issued
23	18	109	2022-09-01	2022-09-08	Issued
24	19	103	2022-09-01	2022-09-08	Issued
25	19	106	2022-09-01	2022-09-08	Issued
31	12	101	2022-11-17	2022-11-24	Issued

Issue_ID : Unique ID to identify Book Issue to Customers

Customer_ID : ID to identify customer details who issued the book.

Book_ID : ID to identify book and Author details

Issue_Date : Date on which book was Issued

Due_Date : Expected maximum return date of book

Status : If Returned or Issued

Return_ID	Issue_ID	Customer_ID	Due_Date	Return_Date	Status
10	10	10	2022-11-22	2022-11-28	Returned
11	11	10	2022-11-22	2022-11-28	Returned
12	12	11	2022-11-22	2022-11-30	Returned
13	13	12	2022-11-24	2022-11-26	Returned
14	14	13	2022-11-27	2022-12-02	Returned
15	15	13	2022-11-27	2022-12-02	Returned
16	16	10	2022-11-27	2022-12-02	Returned
17	17	14	2022-11-28	2022-11-30	Returned
18	18	15	2022-11-25	2022-11-30	Returned
19	19	11	2022-11-25	2022-12-02	Returned
20	20	17	2022-12-02	2022-12-22	Returned
21	21	17	2022-12-02	2022-12-22	Returned
22	22	18	2022-09-08	2022-10-08	Returned
23	23	18	2022-09-08	2022-10-08	Returned
24	24	19	2022-09-08	2022-10-11	Returned
25	25	19	2022-09-08	2022-10-11	Returned

Return_ID : Unique ID to identify Book Returned by Customers

Customer_ID : ID to identify customer details who issued the book.

Issue_ID : ID to identify all Issue Details

Due_Date : Expected maximum return date of book

Return_Date : Date on which book was Returned by Customer.

Status : If Returned or Issued

Fine_ID	Return_ID	customer_ID	Weeks_Owned	Fine_Amount	Status
10	10	10	1	0	Returned
11	11	10	1	0	Returned
12	12	11	2	100	Returned
13	13	12	1	0	Returned
14	14	13	1	0	Returned
15	15	13	1	0	Returned
16	16	10	1	0	Returned
17	17	14	1	0	Returned
18	18	15	1	0	Returned
19	19	11	1	0	Returned
20	20	17	3	200	Returned
21	21	17	3	200	Returned
22	22	18	5	400	Returned
23	23	18	5	400	Returned
24	24	19	5	400	Returned
25	25	19	5	400	Returned

Fine_ID : Unique ID to identify Fine for Each Return

Return_ID : ID to identify Return Details

Customer_ID : ID to identify customer

Weeks_Owned : How many weeks Customer Kept the Book

Fine_Amount : Amount as 100 Rs fine for Each week Extra

Status : If Returned or Still UnReturned

```
CREATE DATABASE Library_Management_System -- Creating Database for Library
USE Library_Management_System -- Activating Database
```

Problem 1 : Query to Issue Book to Customer

```
-- Required :
--     Customer_ID: 10 (used below)  => to which customer book is being issued
--     book_id: 101 (used below)    => which book is being issued
-- with above 2 values, we can issue book using query below

-- checking if this customer already have same book Issued
IF EXISTS (SELECT * FROM issue WHERE customer_id = 15 and book_id = 101)
    PRINT('Same Book Already Issued To Customer.')
ELSE
    IF EXISTS (SELECT * FROM book WHERE book_id = 101 AND stock > 0) -- if Book Exists & In Stock
    AND EXISTS(SELECT * FROM customer WHERE customer_id = 15 AND customer_Status = 'Active') -- if Customer is Registered Member
    BEGIN
        PRINT('Book Available & Customer Active');

        INSERT INTO issue --Creating Record
        VALUES(15,101, CAST( GETDATE() AS Date) , DATEADD(day, 7, CAST( GETDATE() AS Date)), 'Issued');
        -- issue date : Today Date -- due date : 7 Days post Issue date
        PRINT('Book Issued Successfully.');
```

```
        UPDATE book -- Updating Stock
        SET stock = (stock-1)
        WHERE Book_ID = 101
        PRINT('Stock Updated');
```

```
    END
    ELSE -- Troubleshooting : Root Cause Analysis for Management
    BEGIN
        IF EXISTS (SELECT * FROM book WHERE book_id = 101 AND stock > 0) PRINT('book available')
        ELSE PRINT('book not available') -- checking Book Availability

        IF EXISTS(SELECT * FROM customer WHERE customer_id = 15 AND customer_Status = 'Active') PRINT('customer active')
        ELSE PRINT('customer not registered/Active') -- Checking Customer Status
    END
```

Result & Updates

Book Available & Customer Active

(1 row affected)

Book Issued Successfully.

(1 row affected)

Stock Updated

Issue_ID	Customer_ID	Book_ID	Issue_Date	Due_Date	Status
10	10	101	2022-11-15	2022-11-22	Issued
11	10	105	2022-11-15	2022-11-22	Issued
12	11	104	2022-11-15	2022-11-22	Issued
13	12	106	2022-11-17	2022-11-24	Issued
14	13	107	2022-11-20	2022-11-27	Issued
15	13	108	2022-11-20	2022-11-27	Issued
16	10	116	2022-11-20	2022-11-27	Issued
17	14	100	2022-11-21	2022-11-28	Issued
18	15	106	2022-11-24	2022-11-30	Issued
19	11	100	2022-11-24	2022-11-30	Issued
20	17	114	2022-11-26	2022-12-02	Issued
21	17	115	2022-11-26	2022-12-02	Issued
22	18	111	2022-09-01	2022-09-08	Issued
23	18	109	2022-09-01	2022-09-08	Issued
24	19	103	2022-09-01	2022-09-08	Issued
25	19	106	2022-09-01	2022-09-08	Issued
31	12	101	2022-11-17	2022-11-24	Issued
32	15	101	2022-11-19	2022-11-26	Issued

Book
Issued in
Database



Stock Updation

Author_ID	Title	Genre	Published_Year	Stock	Book_ID
1	Half Girlfriend	Romance	2014	20	100
1	2 States	Romance	2009	28	101
1	400 Days	Mystery	2021	12	102
1	One Indian Girl	Romance	2016	26	103
2	Romeo and Juliet	Romance	1597	2	104
3	The Alchemist	Adventure	1988	9	105
3	Eleven Minutes	Fiction	2003	5	106
4	Harry Potter and the Philosopher's Stone	Fiction	1997	50	107
4	Harry Potter and the Chamber of Secrets	Fiction	1998	50	108
4	Harry Potter and the Prisoner of Azkaban	Fiction	1999	50	109
4	Harry Potter and the Goblet of Fire	Fiction	2000	50	110
4	Harry Potter and the Order of the Phoenix	Fiction	2003	50	111
4	Harry Potter and the Half-Blood Prince	Fiction	2005	50	112
4	Harry Potter and the Deathly Hallows	Fiction	2007	50	113



Author_ID	Title	Genre	Published_Year	Stock	Book_ID
1	Half Girlfriend	Romance	2014	20	100
1	2 States	Romance	2009	27	101
1	400 Days	Mystery	2021	12	102
1	One Indian Girl	Romance	2016	26	103
2	Romeo and Juliet	Romance	1597	2	104
3	The Alchemist	Adventure	1988	9	105
3	Eleven Minutes	Fiction	2003	5	106
4	Harry Potter and the Philosopher's Stone	Fiction	1997	50	107
4	Harry Potter and the Chamber of Secrets	Fiction	1998	50	108
4	Harry Potter and the Prisoner of Azkaban	Fiction	1999	50	109
4	Harry Potter and the Goblet of Fire	Fiction	2000	50	110
4	Harry Potter and the Order of the Phoenix	Fiction	2003	50	111
4	Harry Potter and the Half-Blood Prince	Fiction	2005	50	112
4	Harry Potter and the Deathly Hallows	Fiction	2007	50	113



Book Stock
Updated
After Issue

Problem 2 : Query to Return Book to Library

```
-- Required :
-- Issue ID : 32 (used below) => only Issue ID is required to Return book and Update Fine (if applicable)
-- Book ID : 101 (used below) => used to match if Same book is being returned as issued
-- with above 2 values, we can return book & update fine dynamically using query below

IF NOT EXISTS (SELECT * FROM issue WHERE issue_id = 32 ) -- checking if Customer Issued Book or Not
    print('Book Not Identified')
ELSE
    BEGIN
        IF((SELECT book_id FROM issue WHERE issue_id = 32) = (SELECT 101)) -- checking if returning same book which was issued
        -- fetching book id from Issue table -- book id being returned
        BEGIN
            PRINT('Book Identified');

            INSERT INTO submit -- Returing Book with Today date
            VALUES(32, (SELECT Customer_ID FROM issue WHERE Issue_ID = 32)
            , (SELECT due_date FROM issue WHERE Issue_ID = 32)
            , CAST( GETDATE() AS Date), 'Returned');

            PRINT('Book Returned');

            UPDATE book -- Updating Stock Availability
            SET stock = (stock+1)
            WHERE Book_ID = (SELECT Book_ID FROM issue WHERE Issue_ID=32);
            PRINT('Stock Updated');

            -- creating null record in fine
            INSERT INTO fine
            VALUES((SELECT Return_ID FROM submit WHERE Issue_ID=32), -- fetching respective Return ID
            (SELECT Customer_ID FROM issue WHERE Issue_ID = 32) , -- fetching respective Customer ID
            -- calculating number of weeks book owned
            DATEDIFF(WEEK, (SELECT issue_date FROM issue WHERE Issue_ID=32), -- fetching book issue date
            (SELECT return_date FROM submit WHERE Issue_ID=32)) -- fetching book return date
            , NULL, -- null record for fine (updated next)
            'Returned' );
            PRINT('Record Created in Fine');

            -- updating fine = 0 if book was returned before/ on due date
            IF ((SELECT weeks_owed FROM fine WHERE Return_ID = (SELECT Return_ID FROM Submit WHERE Issue_ID=32 )) = 0 )
            BEGIN
                UPDATE fine
                SET Fine_Amount = 0
                WHERE Return_ID = (SELECT Return_ID FROM Submit WHERE Issue_ID=32 )
            END
            -- updating fine = 100* extra weeks owned
            ELSE
            BEGIN
                UPDATE fine
                SET Fine_Amount = (Weeks_Owned-1)*100
                WHERE Return_ID = (SELECT Return_ID FROM Submit WHERE Issue_ID=32 )
            END
            PRINT('Fine Updated');

        END
    END
ELSE
    PRINT('Please Return Same Book As Issued')
END
```

END

Result & Updates

Book Identified

(1 row affected)

Book Returned

(1 row affected)

Stock Updated

(1 row affected)

Record Created in Fine

(1 row affected)

Fine Updated

Return_ID	Issue_ID	Customer_ID	Due_Date	Return_Date	Status
10	10	10	2022-11-22	2022-11-28	Returned
11	11	10	2022-11-22	2022-11-28	Returned
12	12	11	2022-11-22	2022-11-30	Returned
13	13	12	2022-11-24	2022-11-26	Returned
14	14	13	2022-11-27	2022-12-02	Returned
15	15	13	2022-11-27	2022-12-02	Returned
16	16	10	2022-11-27	2022-12-02	Returned
17	17	14	2022-11-28	2022-11-30	Returned
18	18	15	2022-11-25	2022-11-30	Returned
19	19	11	2022-11-25	2022-12-02	Returned
20	20	17	2022-12-02	2022-12-22	Returned
21	21	17	2022-12-02	2022-12-22	Returned
22	22	18	2022-09-08	2022-10-08	Returned
23	23	18	2022-09-08	2022-10-08	Returned
24	24	19	2022-09-08	2022-10-11	Returned
25	25	19	2022-09-08	2022-10-11	Returned
26	32	15	2022-11-26	2022-11-20	Returned



Book Returned
in Database

Fine_ID	Return_ID	customer_ID	Weeks_Owned	Fine_Amount	Status
10	10	10	1	0	Returned
11	11	10	1	0	Returned
12	12	11	2	100	Returned
13	13	12	1	0	Returned
14	14	13	1	0	Returned
15	15	13	1	0	Returned
16	16	10	1	0	Returned
17	17	14	1	0	Returned
18	18	15	1	0	Returned
19	19	11	1	0	Returned
20	20	17	3	200	Returned
21	21	17	3	200	Returned
22	22	18	5	400	Returned
23	23	18	5	400	Returned
24	24	19	5	400	Returned
25	25	19	5	400	Returned
26	26	15	1	0	Returned



Book Return
Fine Updated

Stock Before
Returning
Book 101

Author_ID	Title	Genre	Published_Year	Stock	Book_ID
1	Half Girlfriend	Romance	2014	20	100
1	2 States	Romance	2009	27	101
1	400 Days	Mystery	2021	12	102
1	One Indian Girl	Romance	2016	26	103
2	Romeo and Juliet	Romance	1597	2	104
3	The Alchemist	Adventure	1988	9	105
3	Eleven Minutes	Fiction	2003	5	106
4	Harry Potter and the Philosopher's Stone	Fiction	1997	50	107
4	Harry Potter and the Chamber of Secrets	Fiction	1998	50	108
4	Harry Potter and the Prisoner of Azkaban	Fiction	1999	50	109
4	Harry Potter and the Goblet of Fire	Fiction	2000	50	110
4	Harry Potter and the Order of the Phoenix	Fiction	2003	50	111
4	Harry Potter and the Half-Blood Prince	Fiction	2005	50	112
4	Harry Potter and the Deathly Hallows	Fiction	2007	50	113



Stock After
Returning
Book 101

Author_ID	Title	Genre	Published_Year	Stock	Book_ID
1	Half Girlfriend	Romance	2014	20	100
1	2 States	Romance	2009	28	101
1	400 Days	Mystery	2021	12	102
1	One Indian Girl	Romance	2016	26	103
2	Romeo and Juliet	Romance	1597	2	104
3	The Alchemist	Adventure	1988	9	105
3	Eleven Minutes	Fiction	2003	5	106
4	Harry Potter and the Philosopher's Stone	Fiction	1997	50	107
4	Harry Potter and the Chamber of Secrets	Fiction	1998	50	108
4	Harry Potter and the Prisoner of Azkaban	Fiction	1999	50	109
4	Harry Potter and the Goblet of Fire	Fiction	2000	50	110
4	Harry Potter and the Order of the Phoenix	Fiction	2003	50	111
4	Harry Potter and the Half-Blood Prince	Fiction	2005	50	112
4	Harry Potter and the Deathly Hallows	Fiction	2007	50	113



Problem 3 : Query to Display Fine Due for all Borrowers

Result & Updates

```
-- for all borrowers
SELECT f.Customer_ID, c.Customer_Name, SUM(f.Fine_Amount) AS Fine_Due
FROM fine f
INNER JOIN Customer c
ON f.customer_ID = c.Customer_ID
GROUP BY f.customer_ID, c.Customer_Name
ORDER BY Fine_Due DESC
```

Customer_ID	Customer_Name	Fine_Due
18	Nidhi Kumawat	800
19	Mahesh Saini	800
17	Aman Sharma	400
11	Krati Tripathi	100
12	Aditya Singh	0
13	Nikita Meena	0
14	Abhinav Anand	0
15	Manish Pandey	0
10	Rithik Tripathi	0

Result & Updates

```
-- for those only who have some fine due
SELECT f.Customer_ID, c.Customer_Name, SUM(f.Fine_Amount) AS Fine_Due
FROM fine f
INNER JOIN Customer c
ON f.customer_ID = c.Customer_ID
GROUP BY f.customer_ID, c.Customer_Name
HAVING SUM(f.fine_amount) > 0
ORDER BY Fine_Due DESC
```

Customer_ID	Customer_Name	Fine_Due
18	Nidhi Kumawat	800
19	Mahesh Saini	800
17	Aman Sharma	400
11	Krati Tripathi	100

Problem 4 : Query to Display Fine Due for a Specific Customer

```
-- fine due
-- Required :
--      Customer Name
SELECT f.customer_ID, c.Customer_Name, SUM(f.Fine_Amount) AS Fine_Due
FROM fine f
INNER JOIN customer c
ON f.customer_ID = c.Customer_ID
GROUP BY f.customer_ID, c.Customer_Name
HAVING c.Customer_Name = 'Krati Tripathi'
```

Result & Updates

customer_ID	Customer_Name	Fine_Due
11	Krati Tripathi	100

```
-- fine due
-- Required :
--      Customer ID
SELECT f.customer_ID, c.Customer_Name, SUM(f.Fine_Amount) AS Fine_Due
FROM fine f
INNER JOIN customer c
ON f.customer_ID = c.Customer_ID
GROUP BY f.customer_ID, c.Customer_Name
HAVING f.Customer_ID = 11
```

Result & Updates

customer_ID	Customer_Name	Fine_Due
11	Krati Tripathi	100

Problem 5 : Query to Display Stock of Books Availability

```
-- stock of available books
SELECT b.Book_ID , a.author_name, b.Title
FROM Book b
INNER JOIN Author a
ON a.Author_ID = b.Author_ID
```

Result & Updates

Book_ID	author_name	Title	Stock
100	Chetan Bhagat	Half Girlfriend	20
101	Chetan Bhagat	2 States	28
102	Chetan Bhagat	400 Days	12
103	Chetan Bhagat	One Indian Girl	26
104	William Shakespeare	Romeo and Juliet	2
105	Paulo Coelho	The Alchemist	9
106	Paulo Coelho	Eleven Minutes	5
107	J K Rowling	Harry Potter and the Philosopher's Stone	50
108	J K Rowling	Harry Potter and the Chamber of Secrets	50
109	J K Rowling	Harry Potter and the Prisoner of Azkaban	50
110	J K Rowling	Harry Potter and the Goblet of Fire	50
111	J K Rowling	Harry Potter and the Order of the Phoenix	50
112	J K Rowling	Harry Potter and the Half-Blood Prince	50
113	J K Rowling	Harry Potter and the Deathly Hallows	50
114	Arundhati Roy	Azadi	5
115	Arundhati Roy	War Talk	3
116	Arundhati Roy	The Cost of Living	8

```
----- VIEWS TO EXTRACT INFORMATION -----
SELECT * FROM Book -- entire employee table
SELECT * FROM Author -- entire employee_contact table
SELECT * FROM customer -- entire organisation details
SELECT * FROM Issue -- employees having Multiple Contact
SELECT * FROM Submit -- employees having Multiple Email
SELECT * FROM Fine -- employees having Multiple Contact with Count
SELECT * FROM vw_allFineDue -- Fine Due for All Customers
SELECT * FROM vw_FineDue -- Only customers with some fine due
SELECT * FROM vw_stock -- Stock Availability
```

Thank You