create table Borrower(

Rollin int primary key,

Name Varchar(30),

Date\_of\_issue Date,

Name\_of\_book varchar(30),

Status varchar(30));

CREATE TABLE Fine (

roll\_no INT PRIMARY KEY,

Date DATE,

Amount INT

);

show tables;

INSERT INTO Borrower (Rollin, Name, Date\_of\_issue, Name\_of\_book, Status)

VALUES(1, 'Nishant', '2024-07-10', 'Toc', 'issued'),

(2, 'Vibhav', '2024-07-13', 'Dbms', 'issued'),

(3, 'Aaditya', '2024-07-09', 'Spm', 'issued'),

(4, 'Prathamesh', '2024-07-21', 'Spos', 'issued');

select \* from Borrower;

DELIMITER //

CREATE PROCEDURE finecal(IN roll\_no INT, IN book\_name CHAR(20))

BEGIN

DECLARE v\_doi DATE;

DECLARE noofdays INT;

DECLARE fine DOUBLE;

DECLARE f1 INT DEFAULT 0;

-- Handler for the case when no record is found

DECLARE CONTINUE HANDLER FOR NOT FOUND SET f1 = 1;

-- Fetch the date of issue for the given roll number and book name

SELECT Date\_of\_issue INTO v\_doi

FROM Borrower

WHERE Rollin = roll\_no AND Name\_of\_book = book\_name;

-- Check if the record was found

IF f1 = 1 THEN

SELECT 'Record not found' AS Message;

ELSE

-- Calculate the number of overdue days

SET noofdays = DATEDIFF(CURDATE(), v\_doi);

-- Calculate the fine based on the number of overdue days

IF noofdays >= 15 AND noofdays <= 30 THEN

SET fine = ((noofdays - 15) \* 50) + 150;

ELSE

SET fine = 0;

END IF;

-- Output the fine amount

SELECT fine AS FineAmount;

-- Update the status of the book in the Borrower table

UPDATE Borrower

SET Status = 'Return'

WHERE Rollin = roll\_no AND Name\_of\_book = book\_name;

-- Insert the fine record into the Fine table

INSERT INTO Fine (roll\_no, Date, Amount)

VALUES (roll\_no, CURDATE(), fine);

END IF;

END //

DELIMITER ;

CALL finecal(1, 'Toc');

CALL finecal(2, 'Dbms');

CALL finecal(3, 'Spm');

CALL finecal(4, 'Spos');

select \* from Fine;

select \* from Borrower;