P2

Problem Statement:

Unnamed PL/SQL: Borrower(Rollin, Name, DateofIssue, NameofBook, Status), Fine(Roll_no, Date, Amt). Accept roll_no & book name; calculate days since issue; fines: 15-30 => Rs5/day; >30 => Rs50/day; after submitting set status I to R; if fine applies, insert into Fine.

Code:

```
-- P2: MySQL-style stored procedure / block
CREATE TABLE Borrower (
  Roll_no INT,
  Name VARCHAR(100),
  DateofIssue DATE,
  NameofBook VARCHAR(200),
  Status CHAR(1),
  PRIMARY KEY (Roll_no, NameofBook)
);

CREATE TABLE Fine (
  id INT AUTO_INCREMENT PRIMARY KEY,
  Roll_no INT,
  Date_paid DATE,
  Amt DOUBLE
);

-- Procedure to process return and fines
DELIMITER $$
CREATE PROCEDURE process_return(IN in_roll INT, IN in_book VARCHAR(200))
BEGIN
  DECLARE v_date DATE;
  DECLARE v_days INT;
  DECLARE v_amt DOUBLE DEFAULT 0;
  SELECT DateofIssue INTO v_date FROM Borrower WHERE Roll_no = in_roll AND NameofBook = in_book;
  SET v_days = DATEDIFF(CURDATE(), v_date);
  IF v_days BETWEEN 15 AND 30 THEN
    SET v_amt = (v_days - 15) * 5;
  ELSEIF v_days > 30 THEN
    SET v_amt = (v_days - 30) * 50 + (15 * 5);
  ELSE
    SET v_amt = 0;
  END IF;
  UPDATE Borrower SET Status='R' WHERE Roll_no=in_roll AND NameofBook=in_book;
  IF v_amt > 0 THEN
    INSERT INTO Fine(Roll_no, Date_paid, Amt) VALUES (in_roll, CURDATE(), v_amt);
  END IF;
END$$
DELIMITER ;
```