**Unnamed PL/SQL code block: Use of Control structure and Exception handling is mandatory.**

Suggested Problem statement:

Consider Tables:

1. Borrower(Roll\_no, Name, DateofIssue, NameofBook, Status)

2. Fine(Roll\_no,Date,Amt)

* Accept Roll\_no and NameofBook from user.
* Check the number of days (from date of issue).
* If days are between 15 to 30 then fine amount will be Rs 5per day.
* If no. of days>30, per day fine will be Rs 50 per day and for days less than 30, Rs. 5 per day.
* After submitting the book, status will change from I to R.
* If condition of fine is true, then details will be stored into fine table.
* Also handles the exception by named exception handler or user define exception handler.

**Delimiter //**

**create procedure BookFine(roll\_new integer,book\_name varchar(30))**

**begin**

**declare X integer;**

**DECLARE CONTINUE HANDLER FOR NOT FOUND**

**begin**

**select CONCAT('ROLLNO', roll\_new, ' NOT FOUND') as NoRecord;**

**end;**

**select datediff(curdate(),DateofIssue) into X from Borrower where Rollno=roll\_new;**

**if (X>15 && X<30) then**

**insert into Fine values(roll\_new,curdate(),(X\*5));**

**end if;**

**if (X>30) then**

**insert into Fine values(roll\_new,curdate(),(X\*50));**

**end if;**

**update Borrower set status='R' where Rollno=roll\_new;**

**end;**

**//**

**Delimiter ;**

DELIMITER //

CREATE PROCEDURE BookFine(roll\_new INT, book\_name VARCHAR(30))

BEGIN

DECLARE X INT;

DECLARE No\_Record INT DEFAULT 0;

-- Declare a CONTINUE HANDLER to handle the case where no rows are found

DECLARE CONTINUE HANDLER FOR NOT FOUND SET No\_Record = 1;

-- Try to get the difference between current date and the DateofIssue

SELECT DATEDIFF(CURDATE(), DateofIssue) INTO X

FROM Borrower

WHERE Rollno = roll\_new AND NameofBook = book\_name;

-- If no record is found

IF No\_Record = 1 THEN

SELECT CONCAT('ROLLNO ', roll\_new, ' NOT FOUND') AS NoRecord;

LEAVE BookFine;

END IF;

-- If the fine is applicable between 16 to 30 days overdue

IF (X > 15 AND X <= 30) THEN

INSERT INTO Fine VALUES (roll\_new, CURDATE(), (X \* 5));

END IF;

-- If the fine is applicable for more than 30 days overdue

IF (X > 30) THEN

INSERT INTO Fine VALUES (roll\_new, CURDATE(), (X \* 50));

END IF;

-- Update the status of the borrower to 'R' (Returned)

UPDATE Borrower

SET Status = 'R'

WHERE Rollno = roll\_new;

END //

DELIMITER ;

Save above program with extension .sql(as4.sql)

How to run:

1. Source as4.sql;
2. Call **BookFine(1,”DBMS”);**
3. **Select \* from Fine**