Practical - 4

**/*
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
***/**

```
mysql> create database DBMSL_Exp_4;
Query OK, 1 row affected (0.02 sec)

mysql> use DBMSL_Exp_4;
Database changed

mysql> create table Borrower (RollNo int primary key, Name varchar(45) not null, DateOfIssue
date, NameOfBook varchar(45), Status char);
Query OK, 0 rows affected (0.05 sec)

mysql> desc Borrower;
+-------------+-------------+------+-----+---------+-------+
| Field       | Type        | Null | Key | Default | Extra |
+-------------+-------------+------+-----+---------+-------+
| RollNo      | int         | NO   | PRI | NULL    |       |
| Name        | varchar(45) | NO   |     | NULL    |       |
| DateOfIssue | date        | YES  |     | NULL    |       |
| NameOfBook  | varchar(45) | YES  |     | NULL    |       |
| Status      | char(1)     | YES  |     | NULL    |       |
+-------------+-------------+------+-----+---------+-------+
5 rows in set (0.01 sec)

mysql> create table Fine (RollNo int primary key, DateOfIssue date, Amount int);
Query OK, 0 rows affected (0.05 sec)

mysql> desc Fine;
+-------------+------+------+-----+---------+-------+
| Field       | Type | Null | Key | Default | Extra |
+-------------+------+------+-----+---------+-------+
| RollNo      | int  | NO   | PRI | NULL    |       |
| DateOfIssue | date | YES  |     | NULL    |       |
| Amount      | int  | YES  |     | NULL    |       |
```

```
+-------------+------+------+-----+---------+-------+
3 rows in set (0.00 sec)

mysql> delimiter #

create procedure Calculate_Fine (in roll int, in name varchar(45))
begin
declare d date;
declare difference int;
declare fine_amount int;
select DateOfIssue into d from Borrower where RollNo=roll and NameOfBook=name;
select datediff (curdate(),d) into difference;
if (difference>15) and (difference<=30) then
set fine_amount= 5 * difference;
insert into Fine values (roll,d,fine_amount);
elseif (difference>30) then
set fine_amount= 50 * difference;
insert into Fine values (roll,d,fine_amount);
end if;
update Borrower set status='R' where RollNo=roll;
end #

mysql> delimiter ;

mysql> insert into Borrower values (1, 'Adil Khan', '2023-10-31', 'Murder On The Orient Express',
'I'), (2, 'Suhani Badhe, '2023-11-11', 'Ikigai', 'I'), (3, 'Atharva Gholap', '2023-10-08', 'Harry Potter &
The Goblet Of Fire', 'I');
Query OK, 3 rows affected (0.02 sec)
Records: 3  Duplicates: 0  Warnings: 0

mysql> call Calculate_Fine (1, 'Murder On The Orient Express');
Query OK, 1 row affected (0.03 sec)

mysql> call Calculate_Fine (2, 'Ikigai');
Query OK, 1 row affected (0.01 sec)

mysql> call Calculate_Fine (3, 'Harry Potter & The Goblet Of Fire');
Query OK, 1 row affected (0.03 sec)

mysql> select * from Fine;
+--------+-------------+--------+
| RollNo | DateOfIssue | Amount |
+--------+-------------+--------+
|      1 | 2023-10-31  |     80 |
|      3 | 2023-10-08  |   1950 |
+--------+-------------+--------+
2 rows in set (0.00 sec)
```