Assignment 4

Roll no: 31413

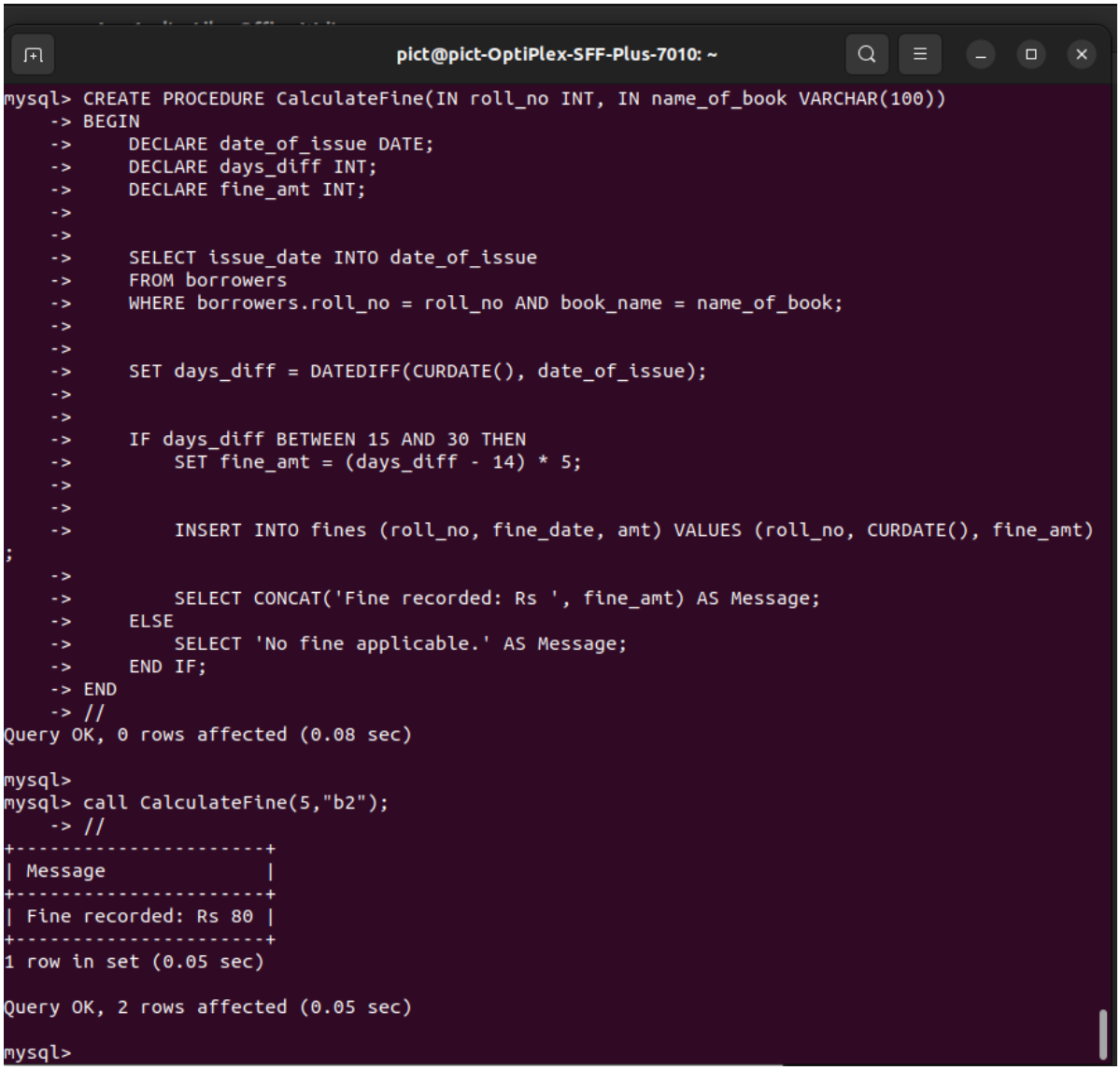
1) Unnamed PL/SQL code block: Use of Control structure and Exception handling is mandatory. Write a PL/SQL block of code for the following requirements:-
Schema:
1. Borrower(Roll,Name,DateofIssue, NameofBook, Status)
2. Fine (Roll, Date, Amt)
Accept Roll & N ame of book from user. Check the number of 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 & 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.

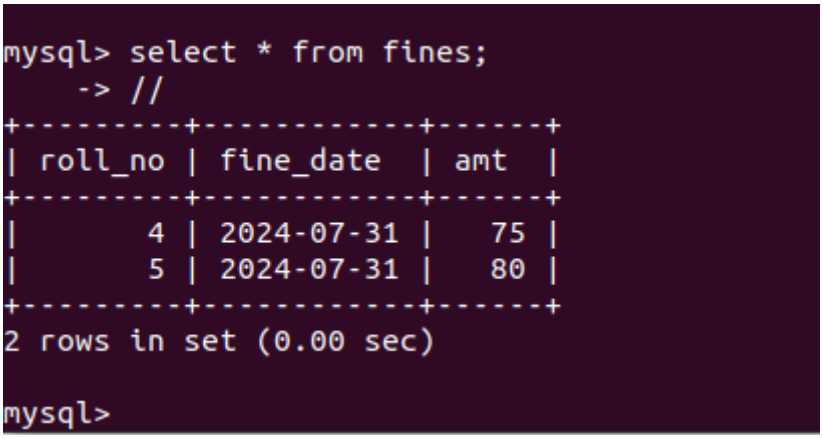
```
create table borrowers(
        roll_no INT,
        name VARCHAR(20),
        issue_date DATE,
        book_name VARCHAR(20),
        status CHAR(10),
        primary key( roll_no )
);

create table fines(
        roll_no INT,
        fine_date DATE,
        amt INT,
        foreign key( roll_no ) references borrowers( roll_no )
);

insert into borrowers values
( 1 , 'p1' , '2024-7-31' , 'b1' , 'I' ) ,
( 2 , 'p2' , '2024-7-17' , 'b2' , 'I' ) ,
( 3 , 'p2' , '2024-6-18' , 'b3' , 'I' ) ,
( 4 , 'p2' , '2024-7-16' , 'b1' , 'I' ) ,
( 5 , 'p2' , '2024-7-01' , 'b2' , 'I' ) ,
( 6 , 'p2' , '2024-6-10' , 'b3' , 'I' ) ;
```

```
mysql> select * from borrowers;
+---------+------+------------+-----------+--------+
| roll_no | name | issue_date | book_name | status |
+---------+------+------------+-----------+--------+
|       1 | p1   | 2024-07-31 | b1        | I      |
|       2 | p2   | 2024-07-17 | b2        | I      |
|       3 | p2   | 2024-06-18 | b3        | I      |
|       4 | p2   | 2024-07-16 | b1        | I      |
|       5 | p2   | 2024-07-01 | b2        | I      |
|       6 | p2   | 2024-06-10 | b3        | I      |
+---------+------+------------+-----------+--------+
6 rows in set (0.00 sec)
```

Procedure :

```
DELIMITER //

CREATE PROCEDURE CalculateFine(IN roll_no INT, IN name_of_book VARCHAR(100))
BEGIN
   DECLARE date_of_issue DATE;
   DECLARE days_diff INT;
   DECLARE fine_amt INT;


   SELECT issue_date INTO date_of_issue
   FROM borrowers
   WHERE borrowers.roll_no = roll_no AND book_name = name_of_book;


   SET days_diff = DATEDIFF(CURDATE(), date_of_issue);


   IF days_diff BETWEEN 15 AND 30 THEN
      SET fine_amt = (days_diff - 14) * 5;


         INSERT INTO fines (roll_no, fine_date, amt) VALUES (roll_no, CURDATE(), fine_amt);

      SELECT CONCAT('Fine recorded: Rs ', fine_amt) AS Message;
   ELSE
      SELECT 'No fine applicable.' AS Message;
   END IF;
END
//
```



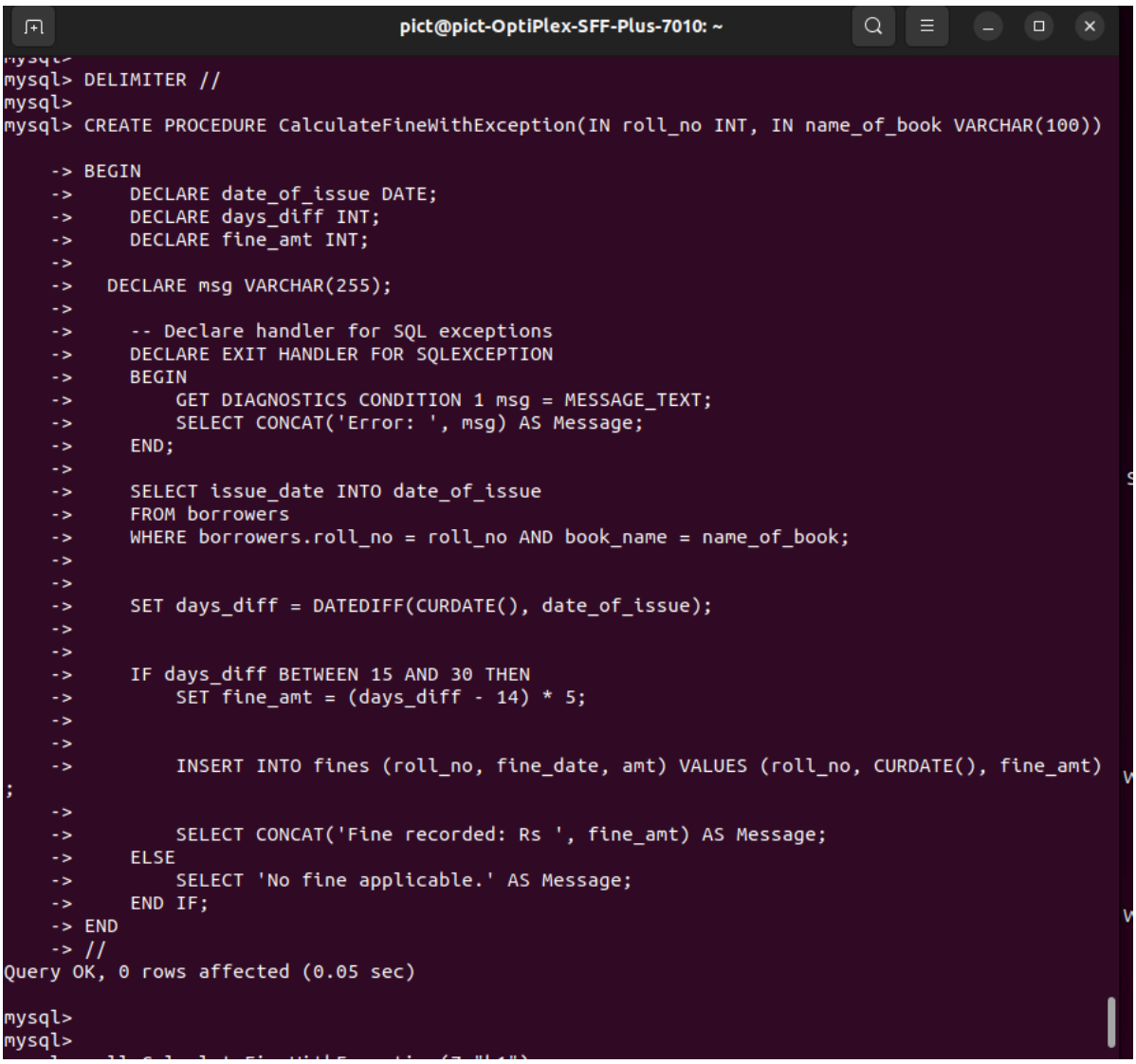


Using Exception Handling :

DELIMITER //

CREATE PROCEDURE CalculateFineWithException(IN roll_no INT, IN name_of_book VARCHAR(100))
BEGIN

```
    DECLARE date_of_issue DATE;
    DECLARE days_diff INT;
    DECLARE fine_amt INT;

          DECLARE msg VARCHAR(255);

    -- Declare handler for SQL exceptions
    DECLARE EXIT HANDLER FOR SQLEXCEPTION
    BEGIN
       GET DIAGNOSTICS CONDITION 1 msg = MESSAGE_TEXT;
       SELECT CONCAT('Error: ', msg) AS Message;
    END;

    SELECT issue_date INTO date_of_issue
    FROM borrowers
    WHERE borrowers.roll_no = roll_no AND book_name = name_of_book;


    SET days_diff = DATEDIFF(CURDATE(), date_of_issue);


    IF days_diff BETWEEN 15 AND 30 THEN
       SET fine_amt = (days_diff - 14) * 5;


           INSERT INTO fines (roll_no, fine_date, amt) VALUES (roll_no, CURDATE(),
fine_amt);

       SELECT CONCAT('Fine recorded: Rs ', fine_amt) AS Message;
    ELSE
       SELECT 'No fine applicable.' AS Message;
    END IF;
END
//
```



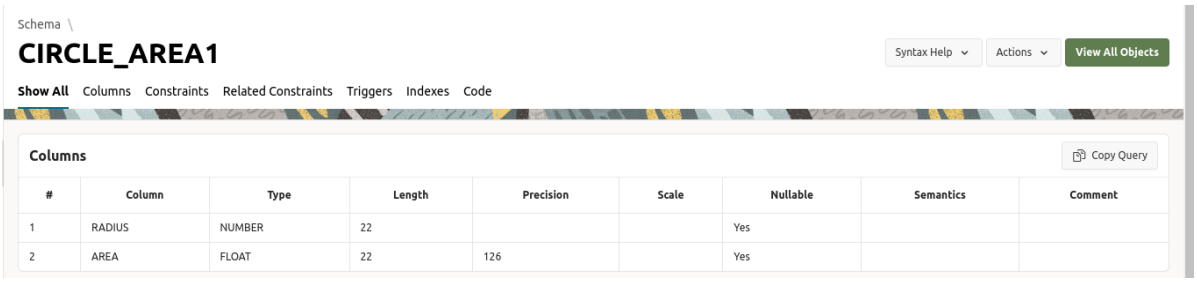

```
mysql> DELIMITER //
mysql>
mysql> CREATE PROCEDURE CalculateFineWithException(IN roll_no INT, IN name_of_book VARCHAR(100))

    -> BEGIN
    ->     DECLARE date_of_issue DATE;
    ->     DECLARE days_diff INT;
    ->     DECLARE fine_amt INT;
    ->
    ->   DECLARE msg VARCHAR(255);
    ->
    ->     -- Declare handler for SQL exceptions
    ->     DECLARE EXIT HANDLER FOR SQLEXCEPTION
    ->     BEGIN
    ->         GET DIAGNOSTICS CONDITION 1 msg = MESSAGE_TEXT;
    ->         SELECT CONCAT('Error: ', msg) AS Message;
    ->     END;
    ->
    ->     SELECT issue_date INTO date_of_issue
    ->     FROM borrowers
    ->     WHERE borrowers.roll_no = roll_no AND book_name = name_of_book;
    ->
    ->
    ->     SET days_diff = DATEDIFF(CURDATE(), date_of_issue);
    ->
    ->
    ->     IF days_diff BETWEEN 15 AND 30 THEN
    ->         SET fine_amt = (days_diff - 14) * 5;
    ->
    ->
    ->         INSERT INTO fines (roll_no, fine_date, amt) VALUES (roll_no, CURDATE(), fine_amt)
;
    ->
    ->         SELECT CONCAT('Fine recorded: Rs ', fine_amt) AS Message;
    ->     ELSE
    ->         SELECT 'No fine applicable.' AS Message;
    ->     END IF;
    -> END
    -> //
Query OK, 0 rows affected (0.05 sec)

mysql>
mysql>
```

```
mysql> call CalculateFineWithException(7,"b1");
    -> //
+--------------------+
| Message            |
+--------------------+
| No fine applicable. |
+--------------------+
1 row in set (0.00 sec)

Query OK, 0 rows affected, 1 warning (0.00 sec)

mysql> call CalculateFineWithException(7);
    -> //
ERROR 1318 (42000): Incorrect number of arguments for PROCEDURE te31413_db.CalculateFineWithExce
ption; expected 2, got 1
mysql>
```

2) Write a PLSQL code block to calculate the area of a circle for a value of radius varying from S to 9. Store the radius and the corresponding values of calculated area in an empty table named areas, consisting of two columns, radius and area (Oracle live)

```
create table circle_area1(radius Number, area float )
```

```
Table created.
```



#	Column	Type	Length	Precision	Scale	Nullable	Semantics	Comment
1	RADIUS	NUMBER	22			Yes		
2	AREA	FLOAT	22	126		Yes		

SQL Worksheet

```
1  declare
2  a number;
3  area1 number;
4  pi constant number:=3.1413;
5
6  begin
7      for a in 5 .. 9 loop
8      area1:=a*a;
9
10     insert into circle_area1(radius,area) values(a,area1);
11
12     dbms_output.put_line('Radius is : '||a||' Area is : '||area1);
13
14     end loop;
15 end;
```

```
Statement processed.
Radius is : 5 Area is : 25
Radius is : 6 Area is : 36
Radius is : 7 Area is : 49
Radius is : 8 Area is : 64
Radius is : 9 Area is : 81
```

SQL Worksheet

```
1  select * from circle_area1;
```

RADIUS	AREA
5	25
6	36
7	49
8	64
9	81

Download CSV

5 rows selected.