ASSIGNMENT No 4 : PLSQL(Procedures,Cursors and Functions)

Shreya Pawaskar
Cno - C22018881961
Roll no - 3947
Division - C

Problem statement:
Execute the following stored procedures and functions:
1. Write a procedure which accepts the E_id as parameter and displays the Ename, Post, Dept and DOJ of the employee formatted in upper case

```
mysql> CREATE PROCEDURE procedure1 (in id1 integer)
    -> begin
    -> select upper(ename) as ename,upper(post) as post,upper(dname) as
dname, date_of_joining as date_of_joining
    -> from department,employee
    -> WHERE employee.e_id=id1
    -> AND
    -> employee.did = department.did;
    -> END //
Query OK, 0 rows affected (1.43 sec)

mysql> delimiter ;

mysql> call procedure1(3);
+-------+---------+----------+-----------------+
| ename | post    | dname    | date_of_joining |
+-------+---------+----------+-----------------+
| RAJAS | MANAGER | TREASURY | 2005-02-12      |
+-------+---------+----------+-----------------+
1 row in set (0.68 sec)
Query OK, 0 rows affected (0.70 sec)

mysql> call procedure1(13);
+-------+---------+--------------------+-----------------+
| ename | post    | dname              | date_of_joining |
+-------+---------+--------------------+-----------------+
| SOHA  | MANAGER | PRODUCTION SUPPORT | 2003-07-11      |
+-------+---------+--------------------+-----------------+
1 row in set (0.10 sec)

Query OK, 0 rows affected (0.11 sec)
```

2. Consider the Employee relation and Write a PL/SQL stored procedure for calculating the income tax of all the employees (using cursors). Consider the following rules:
If annual income of employee is less than 2.5 lakh then no tax, If annual income of employee is from 2.5 lakh to 5 lakh (slab1) then tax = 5% of annual income and If annual income of employee is from 5 lakh to 10 lakh (slab 2)then tax =10% of annual income.

```
mysql> delimiter //
mysql> CREATE PROCEDURE empl_relations()
    -> BEGIN
    -> DECLARE done Integer DEFAULT 0;
    -> DECLARE n Integer;
    -> DECLARE id Integer;
    -> DECLARE eid Integer;
    -> DECLARE income Integer;
    -> DECLARE c CURSOR FOR SELECT E_id,Annual_Income FROM Employee;
    -> DECLARE CONTINUE handler FOR NOT FOUND SET done=1;
    -> SELECT COUNT(*) INTO n FROM income_tax;
    -> IF n=0 THEN
    -> SET id=0;
    -> END IF;
    -> IF n>0 THEN
    -> SELECT tax_id INTO id FROM Income_tax ORDER BY tax_id DESC LIMIT
1;
    -> END IF;
    -> OPEN c;
    -> l1:LOOP
    -> FETCH c INTO eid,income;
    -> SET id=id+1;
    -> IF done=1 THEN
    -> LEAVE l1;
    -> END IF;
    -> IF income<250000 THEN
    -> INSERT INTO income_tax VALUES(id,'less than 2.5
lac',year(curdate()),0,eid);
    -> END IF;
    -> IF income>=250000 AND income<500000 THEN
    -> INSERT INTO income_tax VALUES(id,'2.5 lac to 5
lac',year(curdate()),income*0.05,eid);
    -> END IF;
    -> IF income>=500000 AND income<1000000 THEN
    -> INSERT INTO income_tax VALUES(id,'5 lac to 10
lac',year(curdate()),income*0.1,eid);
    -> END IF;
    -> IF income>=1000000 THEN
    -> INSERT INTO income_tax VALUES(id,'above 10
lac',year(curdate()),income*0.15,eid);
    -> END IF;
    -> END LOOP l1;
```

```
    -> CLOSE c;
    -> END
    -> //
Query OK, 0 rows affected (0.20 sec)
mysql> SELECT * FROM Income_Tax;
+--------+------+-------+------+------------+
| Tax_id | E_id | slab  | year | Tax_amount |
+--------+------+-------+------+------------+
|      1 |    1 | first | 2020 |       5000 |
|      2 |    2 | first | 2020 |       5000 |
|      3 |    3 | first | 2020 |       8000 |
|      4 |    4 | first | 2021 |      10000 |
|      5 |    5 | first | 2021 |       1000 |
|      6 |    6 | first | 2021 |       1000 |
|      7 |    7 | first | 2020 |       6000 |
|      8 |    8 | first | 2020 |       9000 |
|      9 |    9 | first | 2021 |       6000 |
|     10 |   10 | first | 2021 |       6000 |
|     11 |   11 | first | 2021 |      10000 |
|     12 |   12 | first | 2019 |      17000 |
|     13 |   13 | first | 2020 |       7000 |
|     14 |   14 | first | 2021 |       7000 |
|     15 |   15 | first | 2021 |       7000 |
+--------+------+-------+------+------------+
15 rows in set (0.00 sec)
mysql> call empl_relations();
Query OK, 0 rows affected (5.36 sec)

mysql> select * from income_tax;
+--------+------+-------------------+------+------------+
| Tax_id | e_id | slab              | year | tax_amount |
+--------+------+-------------------+------+------------+
|      1 |    1 | first             | 2020 |       5000 |
|      2 |    2 | first             | 2020 |       5000 |
|      3 |    3 | first             | 2020 |       8000 |
|      4 |    4 | first             | 2021 |      10000 |
|      5 |    5 | first             | 2021 |       1000 |
|      6 |    6 | first             | 2021 |       1000 |
|      7 |    7 | first             | 2020 |       6000 |
|      8 |    8 | first             | 2020 |       9000 |
|      9 |    9 | first             | 2021 |       6000 |
|     10 |   10 | first             | 2021 |       6000 |
|     11 |   11 | first             | 2021 |      10000 |
|     12 |   12 | first             | 2019 |      17000 |
|     13 |   13 | first             | 2020 |       7000 |
|     14 |   14 | first             | 2021 |       7000 |
|     15 |   15 | first             | 2021 |       7000 |
|     18 |   18 | first             | 2020 |       5000 |
```

```
|     19 |    1 | 5 lac to 10 lac   | 2021 |      50000 |
|     20 |    2 | 5 lac to 10 lac   | 2021 |      70000 |
|     21 |    3 | 5 lac to 10 lac   | 2021 |      90000 |
|     22 |    4 | above 10 lac      | 2021 |     150000 |
|     23 |    5 | less than 2.5 lac | 2021 |          0 |
|     24 |    6 | less than 2.5 lac | 2021 |          0 |
|     25 |    7 | 5 lac to 10 lac   | 2021 |      60000 |
|     26 |    8 | 5 lac to 10 lac   | 2021 |      90000 |
|     27 |    9 | 5 lac to 10 lac   | 2021 |      60000 |
|     28 |   10 | 5 lac to 10 lac   | 2021 |      70000 |
|     29 |   11 | 5 lac to 10 lac   | 2021 |      90000 |
|     30 |   12 | 5 lac to 10 lac   | 2021 |      90000 |
|     31 |   13 | above 10 lac      | 2021 |     150000 |
|     32 |   14 | 5 lac to 10 lac   | 2021 |      90000 |
|     33 |   15 | 5 lac to 10 lac   | 2021 |      90000 |
|     34 |   18 | above 10 lac      | 2021 |     300000 |
+--------+------+-------------------+------+------------+
```

---

3. Write a procedure that accepts the DName as parameter and returns the number of employees of that department.

```
mysql> delimiter //
mysql> CREATE PROCEDURE procedure3 (in dept_name varchar(255), out
no_of_emp integer)
    -> begin
    -> SELECT COUNT(*) INTO no_of_emp
    -> from department,employee
    -> where employee.did = department.did
    -> AND Department.Dname= dept_name;
    -> END
    -> //
Query OK, 0 rows affected (0.31 sec)

mysql> Delimiter ;
mysql> SET @temp1=0;
Query OK, 0 rows affected (0.08 sec)

mysql> CALL procedure3('Management',@temp1);
Query OK, 1 row affected (0.25 sec)

mysql> select @temp1;
+--------+
| @temp1 |
+--------+
|      1 |
```

```
+--------+
1 row in set (0.00 sec)
```

---

4. Write a function which accepts the ClientName and returns the number of projects of that client

```
mysql> Delimiter //
mysql> CREATE FUNCTION function_new(name1 varchar(255)) RETURNS Integer
deterministic
    -> BEGIN
    -> DECLARE total_no Integer;
    -> SELECT COUNT(*) INTO total_no
    -> FROM Project
    -> WHERE Project.client_name=name1;
    -> return total_no;
    -> END
    -> //
Query OK, 0 rows affected (0.20 sec)

mysql> Delimiter ;
mysql> SELECT function_new('XYZ') AS COUNT;
+-------+
| COUNT |
+-------+
|     5 |
+-------+
1 row in set (0.02 sec)
```