Exp. No: 11

Date: 10-11-2020

MySQL Stored Procedure using Cursor

AIM:

Implement MySQL Stored Procedure using Cursor.

Theoretical Background:
Defining a Cursor

Define a cursor with the DECLARE statement, which has the following syntax:
DECLARE cursor_name CURSOR FOR SELECT_statement;

OPEN

Initializes the result set for the cursor. We must open a cursor before fetching any rows from that cursor. The syntax for the OPEN statement :
OPEN cursor_name;

FETCH

Retrieves the next row from the cursor and moves the cursor "pointer" to the following row in the result set. It has the following syntax:
FETCH cursor_name INTO variable list;

CLOSE

Deacctivates the cursor and releases the memory associated with that cursor. The syntax for this statement is:
CLOSE cursor_name;

Queries and Results:

1. Write a stored procedure using cursor to calculate the total and the percentage of marks of the students in four subjects from the table - Student with the schema given below.

STUDENT (RNO , S1 , S2, S3, S4, total, percentage)

 [Initially the table is partially filled except the last two columns. Those columns should be updated from your procedure]

delimiter $$

```sql
drop Procedure if exists create_student$$

CREATE PROCEDURE create_student()

BEGIN

    drop table if exists STUDENT;

    create table STUDENT ( RNO int , S1 int , S2 int , S3 int , S4 int , total int,
percentage float);

    insert into STUDENT (RNO , S1 , S2, S3, S4) values(10,98,70,50,34);

    insert into STUDENT (RNO , S1 , S2, S3, S4) values(11,89,50,40,49);

    insert into STUDENT (RNO , S1 , S2, S3, S4) values(12,78,73,90,94);

    insert into STUDENT (RNO , S1 , S2, S3, S4) values(13,98,90,90,91);

    insert into STUDENT (RNO , S1 , S2, S3, S4) values(14,38,70,57,67);

END$$


drop Procedure if exists total_percentage_mark$$

CREATE PROCEDURE total_percentage_mark()

BEGIN

    DECLARE last_row_fetched INT DEFAULT FALSE;

    declare r int;

    declare b1 int;

    declare b2 int;

    declare b3 int;

    declare b4 int;

    declare t int;

    declare p float;

    DECLARE csr cursor FOR
```

```
    SELECT RNO , S1 , S2, S3, S4 FROM STUDENT;

DECLARE CONTINUE HANDLER FOR

NOT FOUND SET last_row_fetched=1;

SET last_row_fetched=0;

OPEN csr;

cursor_loop:LOOP

    FETCH csr INTO r,b1,b2,b3,b4;

    IF last_row_fetched=1 THEN

            LEAVE cursor_loop;

    END IF;

    set t=b1+b2+b3+b4;

    set p=(t/400)*100;

    update STUDENT set total=t , percentage =p where RNO=r;

    END LOOP cursor_loop;

    CLOSE csr;

END$$

delimiter ;
```

```
mysql> source /home/anjana-anjali/Documents/asd_lab/exp11/q1.sql
Query OK, 0 rows affected (0.78 sec)

Query OK, 0 rows affected (0.51 sec)

Query OK, 0 rows affected (0.44 sec)

Query OK, 0 rows affected (0.29 sec)

mysql> ^C

^C
mysql> call create_student();
Query OK, 1 row affected (3.40 sec)

mysql> call total_percentage_mark();
Query OK, 0 rows affected (1.72 sec)

mysql> select * from STUDENT;
+-------+-------+-------+-------+-------+-------+------------+
| RNO   | S1    | S2    | S3    | S4    | total | percentage |
+-------+-------+-------+-------+-------+-------+------------+
|    10 |    98 |    70 |    50 |    34 |   252 |         63 |
|    11 |    89 |    50 |    40 |    49 |   228 |         57 |
|    12 |    78 |    73 |    90 |    94 |   335 |      83.75 |
|    13 |    98 |    90 |    90 |    91 |   369 |      92.25 |
|    14 |    38 |    70 |    57 |    67 |   232 |         58 |
+-------+-------+-------+-------+-------+-------+------------+
5 rows in set (0.00 sec)
```

 2. Write a stored procedure using cursor to display employee SSN, name and basic_pay of 5 highest paid employees. For this create the following tables and populate it.

delimiter $$

drop Procedure if exists create_table_emp_details$$

CREATE PROCEDURE create_table_emp_details()

BEGIN

    drop table if exists emp_details_13;

```sql
    create table emp_details_13(ssn int PRIMARY KEY, name
varchar(30),address varchar(50),year_of_exp int);

    insert into emp_details_13 values(30001,'Akhil
Ragu','Ernakulam,683581',9);

    insert into emp_details_13 values(30002,'Achu M','Thrissur,680001',4);

    insert into emp_details_13 values(30003,'Siva
Kumar','Thiruvananthapuram,695001',2);

    insert into emp_details_13 values(30004,'Krish Raj','Kannur,670001',6);

    insert into emp_details_13 values(30005,'Anu
Ramesh','Palakkad,678001',5);

    insert into emp_details_13 values(30006,'paul K','Palakkad,678001',3);
END$$


drop Procedure if exists create_table_emp_sal$$

CREATE PROCEDURE create_table_emp_sal()

BEGIN

    drop table if exists emp_sal_13;

    create table emp_sal_13(ssn int PRIMARY KEY,basic_pay int,da int,hra
int,gross_sal int);

    insert into emp_sal_13 values(30001,80000,8000,5000,1120000);

    insert into emp_sal_13 values(30002,70000,7000,4000,974000);

    insert into emp_sal_13 values(30003,30000,3000,1000,410000);

    insert into emp_sal_13 values(30004,39000,3900,1500,535000);

    insert into emp_sal_13 values(30005,120000,12000,8000,1690000);

    insert into emp_sal_13 values(30006,10000,1000,500,140000);

END$$
```

```sql
drop Procedure if exists display$$

CREATE PROCEDURE display()

BEGIN

        DECLARE flag INT DEFAULT 0;

        declare i int;

        declare s int;

        declare n varchar(30);

        declare bp int;

        declare cur_emp cursor for select t1.ssn,t1.name,t2.basic_pay

        from emp_details_13 t1, emp_sal_13 t2 where t1.ssn=t2.ssn order by
t2.gross_sal desc;

        declare continue handler for not found set flag=1;

        set i=0;

        open cur_emp;

        getemp: LOOP

                FETCH cur_emp into s,n,bp;

                set i=i+1;

                IF (flag=1 OR i>5)THEN

                        LEAVE getemp;

                END IF;

                select s as ssn,n as name,bp as basic_pay;


        END LOOP getemp;

        close cur_emp;

END$$
```

delimiter ;

```
mysql> source /home/anjana-anjali/Documents/asd_lab/exp11/2.sql
Query OK, 0 rows affected (0.16 sec)

Query OK, 0 rows affected (0.15 sec)

Query OK, 0 rows affected (0.13 sec)

Query OK, 0 rows affected (0.15 sec)

Query OK, 0 rows affected (0.12 sec)

Query OK, 0 rows affected (0.20 sec)

mysql> CALL display();
Query OK, 0 rows affected (0.00 sec)
```

```
mysql> call display;
+-------+--------------+-----------+
| ssn   | name         | basic_pay |
+-------+--------------+-----------+
| 50005 | Saju Ramesh  |    120000 |
+-------+--------------+-----------+
1 row in set (0.00 sec)


+-------+--------------+-----------+
| ssn   | name         | basic_pay |
+-------+--------------+-----------+
| 50001 | Raju Raghav  |     80000 |
+-------+--------------+-----------+
1 row in set (0.00 sec)


+-------+--------------+-----------+
| ssn   | name         | basic_pay |
+-------+--------------+-----------+
| 50002 | Nancy Paul   |     70000 |
+-------+--------------+-----------+
1 row in set (0.00 sec)


+-------+--------------+-----------+
| ssn   | name         | basic_pay |
+-------+--------------+-----------+
| 50004 | Krish K K    |     39000 |
+-------+--------------+-----------+
1 row in set (0.00 sec)


+-------+--------------+-----------+
| ssn   | name         | basic_pay |
+-------+--------------+-----------+
| 50003 | Midhun Raj   |     30000 |
+-------+--------------+-----------+
1 row in set (0.00 sec)

Query OK, 0 rows affected (0.00 sec)
```

3. Write a stored procedure using cursor to calculate the total salary of first 'n' records of EMP_SAL table. The value of n is passed to procedure as parameter.

delimiter $$

drop Procedure if exists display$$

CREATE PROCEDURE display(n int)

BEGIN

      DECLARE flag1 INT DEFAULT 0;

      declare i int;

```
declare s int;

declare gs int;

declare cur cursor for select gross_sal from emp_sal_13;

declare continue handler for not found set flag1=1;

set i=0;

set s=0;

open cur;

get_emp: LOOP

        FETCH cur into gs;

        set i=i+1;

        IF (flag1=1 OR i>n)THEN

                LEAVE get_emp;

        END IF;

        set s=s+gs;


END LOOP get_emp;

close cur;

select s as sum;

END$$

delimiter ;
```

```
mysql> source /home/anjana-anjali/Documents/asd_lab/exp11/3.sql
Query OK, 0 rows affected, 1 warning (0.04 sec)

Query OK, 0 rows affected (0.30 sec)

mysql> CALL display(2);
+---------+
| sum     |
+---------+
| 2094000 |
+---------+
1 row in set (0.00 sec)

Query OK, 0 rows affected (0.00 sec)

mysql>
```

4. Write a stored procedure using cursor to update the salary of all employees who earn less than the average salary. Use EMP_SAL table. All the updations must be recorded in a separate table with details: employee SSN, date of updation and updated salary.

delimiter $$

drop Procedure if exists display$$

CREATE PROCEDURE display()

BEGIN

DECLARE flag INT DEFAULT 0;

declare avgs int;

declare gs int;

declare s int;

drop table if exists update_sal_12;

create table update_sal_12(ssn int PRIMARY KEY, date_of_updation varchar(15), updated_sal int);

select avg(gross_sal) into avgs from emp_sal_12;

```
select avgs;

declare cur_emp cursor for select ssn,gross_sal from emp_sal_12;

declare continue handler for not found set flag=1;

open cur_emp;

getemp: LOOP

        FETCH cur_emp into s,gs;

        IF (flag=1)THEN

                LEAVE getemp;

        END IF;

        IF (gs<avgs) THEN

                set gs=gs+gs*10;

                insert into update_sal_12 values(s,'10-11-2020',gs);

        END IF;


END LOOP getemp;

close cur_emp;

select * from update_sal_12;

END$$

delimiter ;
```

 5. Write a stored procedure using cursor to delete employees whose experience is less then 2 years. USE tables EMP_DETAILS and EMP_SAL.

```
delimiter $$

drop Procedure if exists display$$

CREATE PROCEDURE display()
```

```
BEGIN

    DECLARE flag INT DEFAULT 0;

    declare i int;

    declare s int;

    declare y int;

    declare cur_emp cursor for select ssn,year_of_exp from emp_details_13;

    declare continue handler for not found set flag=1;

    set i=0;

    set s=0;

    open cur_emp;

    get_emp: LOOP

        FETCH cur_emp into s,y;

        IF flag=1 THEN

            LEAVE get_emp;

        END IF;

        IF y<2 THEN

            delete from emp_details_13 where ssn=s;

            delete from emp_sal_13 where ssn=s;

        END IF;


    END LOOP get_emp;

    close cur_emp;

END$$

delimiter ;
```

```
mysql> source /home/anjana-anjali/Desktop/anjali/asdlab/exp11/q5.sql
Query OK, 0 rows affected (0.21 sec)

Query OK, 0 rows affected (0.24 sec)

mysql> select * from emp_details_13;
+-------+-------------+----------------------------+-------------+
| ssn   | name        | address                    | year_of_exp |
+-------+-------------+----------------------------+-------------+
| 30001 | Akhil Ragu  | Ernakulam,683581           |           9 |
| 30002 | Achu M      | Thrissur,680001            |           4 |
| 30003 | Siva Kumar  | Thiruvananthapuram,695001  |           2 |
| 30004 | Krish Raj   | Kannur,670001              |           6 |
| 30005 | Anu  Ramesh | Palakkad,678001            |           5 |
| 30006 | paul K      | Palakkad,678001            |           3 |
+-------+-------------+----------------------------+-------------+
6 rows in set (0.00 sec)

mysql> select * from emp_details_13;
+-------+-------------+----------------------------+-------------+
| ssn   | name        | address                    | year_of_exp |
+-------+-------------+----------------------------+-------------+
| 30001 | Akhil Ragu  | Ernakulam,683581           |           9 |
| 30002 | Achu M      | Thrissur,680001            |           4 |
| 30003 | Siva Kumar  | Thiruvananthapuram,695001  |           2 |
| 30004 | Krish Raj   | Kannur,670001              |           6 |
| 30005 | Anu  Ramesh | Palakkad,678001            |           5 |
| 30006 | paul K      | Palakkad,678001            |           3 |
+-------+-------------+----------------------------+-------------+
6 rows in set (0.00 sec)
```