Department of Information Technology

Academic Year: 2022-23   Name of Student: Harsh Joshi
Semester: III   Student ID: 22204012
Class / Branch: SE (IT)   Subject: SQL Lab

Experiment No:09

Aim: To study and implement various functions and procedures in SQL.

Software used: MySQL

Theory :-

      Procedures and Functions are the subprograms which can be created and saved in the database as database objects. They can be called or referred inside the other blocks also.

Procedures & Functions

    "A **procedures** or **function** is a group or set of SQL and PL/SQL statements that perform a specific task." A function and procedure is a named PL/SQL Block which is similar.The major difference between a procedure and a function is, a function must always return a value, but a procedure may or may not return a value.

Procedures:

 A procedure is a named PL/SQL block which performs one or more specific task. This is similar to a procedure in other programming languages. A procedure has a header and a body. The header consists of the name of the procedure and the parameters or variables passed to the   procedure. The body consists or declaration section, execution section and exception section similar to a general PL/SQL Block. A procedure is similar to an anonymous PL/SQL Block but it is named for repeated usage.We can pass parameters to procedures in three ways :

Parameters	Description
IN type	These types of parameters are used to send values to stored procedures.
OUT type	These types of parameters are used to get values from stored procedures. This is similar to a return type in

	functions.
IN OUT type	These types of parameters are used to send values and get values from stored procedures

A procedure may or may not return any value.

Syntax:

```
CREATE [OR REPLACE] PROCEDURE procedure_name (<Argument> {IN, OUT, IN OUT}
<Datatype>,…)
IS
 Declaration section<variable, constant> ;
BEGIN
  Execution section
EXCEPTION
  Exception section
END
```

IS - marks the beginning of the body of the procedure and is similar to DECLARE in anonymous PL/SQL Blocks. The code between IS and BEGIN forms the Declaration section. The syntax within the brackets [ ] indicate they are optional. By using CREATE OR REPLACE together the procedure is created if no other procedure with the same name exists or the existing procedure is replaced with the current code.

How to execute a Procedure?

There are two ways to execute a procedure :

- From the SQL prompt : EXECUTE [or EXEC] procedure_name;
- Within another procedure – simply use the procedure name : procedure_name;

Example:

create table named emp have two column id and salary with number datatype.

```
CREATE OR REPLACE PROCEDURE p1(id IN NUMBER, sal IN NUMBER)
AS
BEGIN
 INSERT INTO emp VALUES(id, sal);
 DBMD_OUTPUT.PUT_LINE('VALUE INSERTED.');
END;
/
```

Output:

Run SQL Command Line

```
SQL>set serveroutput on
SQL>start D://pr.sql
Procedure created.

SQL>exec p1(5,4);
VALUE INSERTED.
PL/SQL procudere successfully completed.

SQL>select * from emp;
  ID    SALARY
-----  --------
  2     5000
```
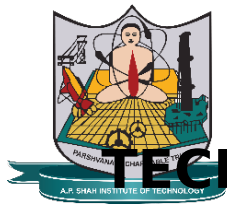
Functions:

A function is a named PL/SQL Block which is similar to a procedure. The major difference between a procedure and a function is, a function must always return a value, but a procedure may or may not return a value.

Syntax:

CREATE [OR REPLACE] FUNCTION function_name [parameters]

 RETURN return_datatype; {IS, AS}

 Declaration_section <variable,constant> ;

 BEGIN

```
   Execution_section
   Return return_variable;
 EXCEPTION
   exception section
   Return return_variable;
 END;
```

RETURN TYPE: The header section defines the return type of the function. The return datatype can be any of the oracle datatype like varchar, number etc.

The execution and exception section both should return a value which is of the datatype defined in the header section.

How to execute a Function?
 A function can be executed in the following ways.

- As a part of a SELECT statement : SELECT emp_details_func FROM dual;
- In a PL/SQL Statements like, : dbms_output.put_line(emp_details_func);

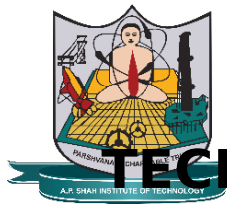This line displays the value returned by the function .

Example:

```
create or replace function getsal (no IN number) return number
is
 sal number(5);
begin
 select salary into sal from emp where id=no;
 return sal;
end;
/
```

Output:

Run SQL Command Line

```
SQL>select * from emp;
  ID    SALARY
```

```
-----  --------
  2      5000

SQL>start D://fun.sql
Function created.

SQL>select getsal(2) from dual;
GETSAL(2)
---------
   5000
```

In the example we are retrieving the 'salary' of employee with id 2 to variable 'sal'. The return type of the function is number.

Destroying procedure and function :

Syntax:

DROP PROCEDURE/**FUNCTION PROCEDURE**/**FUNCTION_NAME;**

Procedures VS Functions:

- A function MUST return a value
- A procedure cannot return a value
- Procedures and functions can both return data in OUT and IN OUT parameters
- The return statement in a function returns control to the calling program and returns the results of the function
- The return statement of a procedure returns control to the calling program and cannot return a value
- Functions can be called from SQL, procedure cannot
- Functions are considered expressions, procedure are not

Output:-

```
mysql> select *,calcProfit(prod_cost,prod_price) as profit from products;
    -> end$$
+---------+-----------+-----------+------------+--------+
| prod_id | prod_name | prod_cost | prod_price | profit |
+---------+-----------+-----------+------------+--------+
|       1 | Shampoo   |      5.95 |       8.35 |   2.40 |
|       2 | Biscuit   |      0.95 |       1.35 |   0.40 |
|       3 | Soap      |     99.95 |        140 |  40.05 |
+---------+-----------+-----------+------------+--------+
3 rows in set (0.01 sec)
```
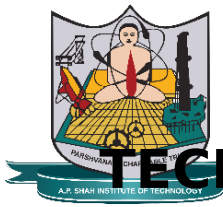
```
mysql> call procedureTest() \g
+------------+
| prod_name |
+------------+
| Shampoo    |
| Biscuit    |
| Soap       |
+------------+
3 rows in set (0.00 sec)

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

Conclusion :- In this experiment we studied about procedures & functions in SQL. A procedure is a named SQL block which performs one or more specific task whereas a function is a SQL block which is similar to a procedure. The main difference between a function and a procedure is that a function always returns a value & procedure may or may not return any value.