EXPERIMENT NO. 08

AIM: To implement control flow function and user defined functions in SQL.

THEORY:

Procedures and functions are PL/SQL blocks that are named, and they are also known as subprograms. Procedures and Functions are the subprograms which can be created and saved in the database as database objects.

Control Flow Functions :-

Case when [test1] then [result1] …else [default] end	Returns resultN if testN evaluates as true, else return default.
Case [test] when [val1] then [result1] … Else [default] End	Returns resultN if test evaluates to valN, else return default.
If(test,t,f)	Returns t if test evaluates as true; else returns f
Ifnull (arg1, arg2)	Returns arg1 if arg1 is not null; else returns arg2
Nullif(arg1,arg2)	Returns null if arg1 equals arg2; else returns arg2

- Functions − These subprograms return a single value; mainly used to compute and return a value.

  Syntax is as follows :-

```
delimiter $$

CREATE FUNCTION function_name(
   param1,
   param2,…
)
RETURNS datatype
DETERMINISTIC
BEGIN
-- statements
END $$

delimiter ;
```

The DETERMINISTIC and NOT DETERMINISTIC characteristics indicate whether a function always produces the same result for given inputs.

```
mysql> select * from student;
+--------+--------+---------+-------+---------+
| fname  | lname  | english | maths | science |
+--------+--------+---------+-------+---------+
| john   | Doe    |      50 |    51 |      50 |
| sarah  | shaikh |      65 |    67 |      65 |
| joe    | wumba  |      45 |    47 |      43 |
| deniel | Clark  |      70 |    72 |      70 |
+--------+--------+---------+-------+---------+
4 rows in set (0.00 sec)

mysql> select fname,lname ,(english+maths+science) as 'total_marks', case when (english+maths+science) <140 then 'D'
    -> when (english+maths+science) between 145 and 175 then 'C'
    -> when (english+maths+science) between 176 and 200 then 'B'
    -> else 'A' end
    -> as grade from student;
+--------+--------+-------------+-------+
| fname  | lname  | total_marks | grade |
+--------+--------+-------------+-------+
| john   | Doe    |         151 | C     |
| sarah  | shaikh |         197 | B     |
| joe    | wumba  |         135 | D     |
| deniel | Clark  |         212 | A     |
+--------+--------+-------------+-------+
4 rows in set (0.00 sec)
```

```
mysql> select * from student;
+--------+--------+---------+-------+---------+
| fname  | lname  | english | maths | science |
+--------+--------+---------+-------+---------+
| john   | Doe    |      50 |    51 |      50 |
| sarah  | shaikh |      65 |    67 |      65 |
| joe    | wumba  |      45 |    47 |      43 |
| deniel | Clark  |      70 |    72 |      70 |
| Mery   | thumb  |      50 |    31 |      47 |
+--------+--------+---------+-------+---------+
5 rows in set (0.00 sec)

mysql> select concat(fname,lname) as name, if((maths)/100<0.32,'fail in maths','pass in math') as result_math
from student;
+-------------+---------------+
| name        | result_math   |
+-------------+---------------+
| johnDoe     | pass in math  |
| sarahshaikh | pass in math  |
| joewumba    | pass in math  |
| denielClark | pass in math  |
| Merythumb   | fail in maths |
+-------------+---------------+
5 rows in set (0.00 sec)
```

```
mysql> use ajk;
Database changed
mysql> delimiter $
mysql> create function calprofit(purches int,sell int) returns int
    -> deterministic
    -> begin
    -> declare profit int;
    -> set profit = purches-sell;
    -> return profit;
    -> end $
Query OK, 0 rows affected (0.03 sec)
```

```
mysql> select * from product;
+---------+-----------+-----------+------------+
| prod_id | prod_name | prod_cost | prod_price |
+---------+-----------+-----------+------------+
|      11 | TV        |     12000 |      15000 |
|      12 | AC        |     22000 |      25500 |
|      13 | Printer   |     16000 |      17000 |
+---------+-----------+-----------+------------+
3 rows in set (0.00 sec)
```

```
mysql> select *,calprofit(prod_price,prod_cost) as profit from product;
+---------+-----------+-----------+------------+--------+
| prod_id | prod_name | prod_cost | prod_price | profit |
+---------+-----------+-----------+------------+--------+
|      11 | TV        |     12000 |      15000 |   3000 |
|      12 | AC        |     22000 |      25500 |   3500 |
|      13 | Printer   |     16000 |      17000 |   1000 |
+---------+-----------+-----------+------------+--------+
3 rows in set (0.00 sec)
```

```
mysql> DELIMITER $$
mysql> CREATE FUNCTION tax(salary int) RETURNS int
    -> DETERMINISTIC
    -> BEGIN
    ->   DECLARE tax_cal int;
    ->   SET tax_cal = salary*0.08;
    ->   RETURN tax_cal;
    -> END$$
Query OK, 0 rows affected (0.01 sec)

mysql> DELIMITER ;
mysql> select * from employee;
+----------+--------+--------+
| fname    | lname  | salary |
+----------+--------+--------+
| mourgous | taylor |  58000 |
| david    | mantoo |  65000 |
| vargas   | rock   |  60000 |
+----------+--------+--------+
3 rows in set (0.00 sec)

mysql> select *,tax(salary) from employee;
+----------+--------+--------+-------------+
| fname    | lname  | salary | tax(salary) |
+----------+--------+--------+-------------+
| mourgous | taylor |  58000 |        4640 |
| david    | mantoo |  65000 |        5200 |
| vargas   | rock   |  60000 |        4800 |
+----------+--------+--------+-------------+
3 rows in set (0.01 sec)
```

CONCLUSION: - Hence studied the user defined functions in SQL.