Practical - 5

/*
Named PL/SQL Block: PL/SQL Stored Procedure and Stored Function.
Write a Stored Procedure namely proc_Grade for the categorization of student.
If marks scored by students in examination is <=1500 and marks>=990 then student will
be placed in distinction category if marks scored are between 989 and 900 category is
first class, if marks 899 and 825 category is Higher Second Class.

Write a PL/SQL block to use procedure created with above requirements:
Stud_Marks(name, total_marks)   Result(Roll,Name, Class)
*/

mysql> create database DBMSL_Exp_5;
Query OK, 1 row affected (0.02 sec)

mysql> use DBMSL_Exp_5;
Database changed

mysql> create table StudentMarks (Roll int primary key, Name varchar(30) not null, TotalMarks int not null);
Query OK, 0 rows affected (0.05 sec)

mysql> desc StudentMarks;
```
+------------+-------------+------+-----+---------+-------+
| Field      | Type        | Null | Key | Default | Extra |
+------------+-------------+------+-----+---------+-------+
| Roll       | int         | NO   | PRI | NULL    |       |
| Name       | varchar(30) | NO   |     | NULL    |       |
| TotalMarks | int         | NO   |     | NULL    |       |
+------------+-------------+------+-----+---------+-------+
```
3 rows in set (0.00 sec)

mysql> create table Result (Roll int primary key, Name varchar(30) not null, Class varchar(5));
Query OK, 0 rows affected (0.06 sec)

mysql> desc Result;
```
+-------+-------------+------+-----+---------+-------+
| Field | Type        | Null | Key | Default | Extra |
+-------+-------------+------+-----+---------+-------+
| Roll  | int         | NO   | PRI | NULL    |       |
| Name  | varchar(30) | NO   |     | NULL    |       |
| Class | varchar(5)  | YES  |     | NULL    |       |
+-------+-------------+------+-----+---------+-------+
```
3 rows in set (0.00 sec)

mysql> delimiter #

```
create procedure proc_Grade(in marks int, out class varchar(10))
begin
if marks<=1500 and marks>=990 then set class='DIST';
```

```
end if;
if marks<=989 and marks>=900 then set class='FC';
end if;
if marks<=899 and marks>=825 then set class='HSC';
end if;
if marks<=824 and marks>=750 then set class='SC';
end if;
if marks<=749 and marks>=650 then set class='PC';
end if;
if marks < 650 then set class='FAIL';
end if;
end #

create function Find_Result(Roll_In int) returns varchar(20) deterministic
begin
declare marks int;
declare Grade varchar(5);
declare StuName varchar(30);
select StudentMarks.TotalMarks,StudentMarks.Name into marks, StuName from StudentMarks
where StudentMarks.Roll = Roll_In;
call proc_Grade(marks, @Grade);
insert into Result values (Roll_In, StuName, @Grade);
return @Grade;
end #
```

mysql> delimiter ;

mysql> insert into StudentMarks(Roll,Name,TotalMarks) values (23,'Adil',788),(55,'Suhani',1400),
(64,'Atharva',989),(69,'Samruddhi',600);
Query OK, 4 rows affected (0.01 sec)
Records: 4 Duplicates: 0 Warnings: 0

mysql> select * from StudentMarks;
+------+--------+------------+
| Roll | Name | TotalMarks |
+------+--------+------------+
23	Adil	888
55	Suhani	1400
64	Atharva	989
69	Samruddhi	600
+------+--------+------------+
4 rows in set (0.00 sec)

mysql> select Find_Result (55);
+------------------+
| Find_Result (55) |
+------------------+
| 55 |
+------------------+
1 row in set (0.02 sec)

mysql> select Find_Result (23);

```
+------------------+
| Find_Result (23) |
+------------------+
|               23 |
+------------------+
1 row in set (0.01 sec)

mysql> select Find_Result (64);
+------------------+
| Find_Result (64) |
+------------------+
|               64 |
+------------------+
1 row in set (0.01 sec)

mysql> select Find_Result (69);
+------------------+
| Find_Result (69) |
+------------------+
|               69 |
+------------------+
1 row in set (0.02 sec)

mysql> select * from Result;
+------+----------+-------+
| Roll | Name     | Class |
+------+----------+-------+
|   23 | Adil     | HSC   |
|   55 | Suhani   | DIST  |
|   64 | Atharva  | FC    |
|   69 | Samruddhi| FAIL  |
+------+----------+-------+
4 rows in set (0.00 sec)
```