**ASSIGNMENT NO- 05**

TCOA35 -Shrutee Pimpare

**TITLE:**

To Study and Implement PL/SQL program along with Procedure and Functions.

**SOLUTION**

mysql> CREATE TABLE STUD\_MARKS( -> NAME VARCHAR(50),

-> TOTAL\_MARKS INTEGER);

Query OK, 0 rows affected (0.04 sec)

mysql> CREATE TABLE RESULT(

-> ROLL INTEGER PRIMARY KEY,

-> NAME VARCHAR(50),

-> CLASS VARCHAR(50));

Query OK, 0 rows affected (0.02 sec)

mysql> INSERT INTO STUD\_MARKS VALUES('AA', 1400);

Query OK, 1 row affected (0.01 sec)

mysql> INSERT INTO STUD\_MARKS VALUES('BB', 950);

Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO STUD\_MARKS VALUES('CC', 1099);

Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO STUD\_MARKS VALUES('DD', 750);

Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO STUD\_MARKS VALUES('EE', 850);

Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO RESULT VALUES(1,'AA', NULL);

Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO RESULT VALUES(2,'BB', NULL);

Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO RESULT VALUES(3,'CC', NULL);

Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO RESULT VALUES(4,'DD', NULL);

Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO RESULT VALUES(5,'EE', NULL);

Query OK, 1 row affected (0.00 sec)

mysql> DELIMITER $ mysql> CREATE PROCEDURE PROC\_GRADE(IN RNO INT, OUT GRADE VARCHAR(30))

-> BEGIN

-> DECLARE M INT;

-> SELECT TOTAL\_MARKS INTO M FROM STUD\_MARKS WHERE NAME = (SELECT NAME FROM

RESULT WHERE ROLL = RNO);

-> IF M>=990 AND M<=1500 THEN

-> SELECT 'DISTINCTION' INTO GRADE;

-> UPDATE RESULT SET CLASS='DISTINCTION' WHERE ROLL=RNO;

-> ELSEIF M>=900 AND M<=989 THEN

-> SELECT 'FIRST CLASS' INTO GRADE;

-> UPDATE RESULT SET CLASS='FIRST CLASS' WHERE ROLL=RNO;

-> ELSEIF M>=825 AND M<=899 THEN

-> SELECT 'HIGHER SECOND CLASS' INTO GRADE;

-> UPDATE RESULT SET CLASS='HIGHER SECOND CLASS' WHERE ROLL=RNO;

-> ELSE SELECT 'NA' INTO GRADE;

-> UPDATE RESULT SET CLASS='NA' WHERE ROLL=RNO;

-> END IF;

-> END;

-> $

Query OK, 0 rows affected (0.00 sec)

mysql> DELIMITER $ mysql> CREATE FUNCTION FUNC\_GRADE(RNO INT)

-> RETURNS VARCHAR(25)

-> DETERMINISTIC

-> BEGIN

-> DECLARE GRADE VARCHAR(25);

-> CALL PROC\_GRADE(RNO, GRADE);

-> RETURN GRADE;

-> END;

-> $

Query OK, 0 rows affected (0.00 sec)

mysql> SELECT FUNC\_GRADE(1);

+---------------+

| FUNC\_GRADE(1) |

+---------------+

| DISTINCTION |

+---------------+

1 row in set (0.00 sec)

mysql> SELECT FUNC\_GRADE(2);

+---------------+

| FUNC\_GRADE(2) |

+---------------+

| FIRST CLASS |

+---------------+

1 row in set (0.00 sec)

mysql> SELECT FUNC\_GRADE(3);

+---------------+

| FUNC\_GRADE(3) |

+---------------+

| DISTINCTION |

+---------------+

1 row in set (0.00 sec)

mysql> SELECT FUNC\_GRADE(4);

+---------------+

| FUNC\_GRADE(4) |

+---------------+

| NA |

+---------------+

1 row in set (0.00 sec)

mysql> SELECT FUNC\_GRADE(5); +---------------------+

| FUNC\_GRADE(5) |

+---------------------+

| HIGHER SECOND CLASS |

+---------------------+

1 row in set (0.00 sec)

mysql> SELECT \*FROM STUD\_MARKS;

+------+-------------+

| NAME | TOTAL\_MARKS |

+------+-------------+

| AA | 1400 |

| BB | 950 |

| CC | 1099 |

| DD | 750 |

| EE | 850 |

+------+-------------+

5 rows in set (0.00 sec)

mysql> SELECT \*FROM RESULT; +------+------+---------------------+

| ROLL | NAME | CLASS |

+------+------+---------------------+

| 1 | AA | DISTINCTION |

| 2 | BB | FIRST CLASS |

| 3 | CC | DISTINCTION |

| 4 | DD | NA |

| 5 | EE | HIGHER SECOND CLASS |

+------+------+---------------------+ 5 rows in set (0.00 sec)