❖EXERCISE 2:

1) Select from any table a number and determine whether it is within a given range (for example, between 1 and 10).

⇒ DELIMITER //

CREATE PROCEDURE CheckRange(IN input INT,OUT reslmsg VARCHAR(50))

-> BEGIN

-> IF input >= 1 AND input <= 10 THEN

-> SET reslmsg = CONCAT(input, 'is within range');

-> ELSE

-> SET reslmsg = CONCAT(input, 'is not within range');

-> END IF;

-> END //

DELIMITER ;

SET @reslmsg = '';

SELECT num INTO @input FROM tempp LIMIT 1;

CALL CheckRange(@input,@reslmsg);

SELECT @reslmsg as Result;

2) Select from any table three positive integers representing the sides of a triangle, and determine whether they form a valid triangle. Hint: In a triangle, the sum of any two sides must always be greater than the third side.

⇒ DELIMITER //

CREATE PROCEDURE checkTra(OUT resmsg VARCHAR(50))

```
    -> BEGIN
    -> DECLARE S1 INT;
    -> DECLARE S2 INT;
    -> DECLARE S3 INT;
    -> SELECT num,square,`cube` INTO S1,S2,S3 FROM
tempp LIMIT 1;
    -> IF S1 <= 0 OR S2 <=0 OR S3<= 0 THEN
    -> SET RESMSG = 'SIDES MUST BE POSITIVE';
    -> ELSE
    -> IF (S1 + S2 > S3) AND (S1 + S3 > S2) AND (S2 +
S3 > S1) THEN
    -> SET RESMSG = CONCAT(S1, ',' , S2, ',' ,S3, ' CAN
FORM TRIANGLE');
    -> ELSE
    -> SET RESMSG = CONCAT(S1, ',' , S2, ',' ,S3, '
CANNOT FORM TRIANGLE');
    -> END IF;
    -> END IF;
    -> END //
DELIMITER ;
SET @RESMSG = '';
CALL CheckTra(@RESMSG);
SELECT @RESMSG AS RESULT;
```

3) Check if a given a year is a leap year. The condition is:- year should be (divisible by 4 and not divisible by 100) or (divisible by 4 and divisible by 400.). The year should be Selected from some table.

⇒

```
DELIMITER //
-> CREATE PROCEDURE CHECKLEAP(OUT RESMSG
VARCHAR(50))
    -> BEGIN
    -> DECLARE SL INT;
    -> SELECT Cnum INTO SL FROM customers LIMIT 1;
```

```
    -> IF (SL % 4 = 0 AND SL % 100 != 0 ) OR (SL % 400 =0)
THEN
    -> SET RESMSG = CONCAT(SL, ' IS LEAP YEAR');
    -> ELSE
    -> SET RESMSG = CONCAT(SL, ' IS NOT LEAP YEAR');
    -> END IF;
    -> END //
DELIMITER ;
SET @RESMSG = '';
CALL CHECKLEAP(@RESMSG);
SELECT @RESMSG AS RESULT;
```

4) Write a program that Selects from any table two character strings. Your program should then determine if one character string exists inside another character string.

⇒

```
DELIMITER //
CREATE PROCEDURE CHECKSTRING( OUT RESMSG
VARCHAR(50))
    -> BEGIN
    -> DECLARE STR1 VARCHAR(100);
    -> DECLARE STR2 VARCHAR(100);
    -> SELECT Cname INTO STR1 FROM customers LIMIT 1;
    -> SELECT city INTO STR2 FROM customers LIMIT 1;
   -> IF STR1 LIKE CONCAT('%',STR2, '%') THEN
  -> SET RESMSG = CONCAT(STR2 , 'EXISTS INSIDE '
,STR1);
   -> ELSE
   -> SET RESMSG = CONCAT(STR2 , 'DOES NOT EXISTS
INSIDE ' ,STR1);
   -> END IF;
```

```
    -> END //
DELIMITER ;
SET @RESMSG = '';
CALL CHECKSTRING(@RESMSG);
SELECT @RESMSG AS RESULT;
```