* **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;

1. 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;