❖EXERCISE 5:

1) Write a stored function to take three parameters, the sides of a triangle. The sides of the triangle should be accepted from the user. The function should return a Boolean value:- true if the triangle is valid, false otherwise. A triangle is valid if the length of each side is less than the sum of the lengths of the other two sides. Check if the dimensions entered can form a valid triangle.

⇒ DELIMITER //

```
CREATE FUNCTION IsValidTriangle(
    side1 DECIMAL(10, 2),
    side2 DECIMAL(10, 2),
    side3 DECIMAL(10, 2)
)
RETURNS BOOLEAN
DETERMINISTIC
BEGIN
    -- Check if any side is less than or equal to zero
    IF (side1 <= 0 OR side2 <= 0 OR side3 <= 0) THEN
        RETURN FALSE;
    END IF;

    -- Check the triangle inequality conditions
    IF (side1 < side2 + side3 AND
        side2 < side1 + side3 AND
        side3 < side1 + side2) THEN
```

```
        RETURN TRUE;
    ELSE
        RETURN FALSE;
    END IF;
END //
DELIMITER ;
SELECT IsValidTriangle(3, 4, 5);  -- Returns TRUE
SELECT IsValidTriangle(1, 2, 3);  -- Returns FALSE
```

2) Write a function that generates a random number between 1 and 10. Use any logic of your choice to achieve this.

⇒

```
DELIMITER //
CREATE FUNCTION GenerateRandomNumber()
RETURNS INT
DETERMINISTIC
BEGIN
    RETURN FLOOR(1 + (RAND() * 10));
END //
DELIMITER ;
SELECT GenerateRandomNumber();
```

3) Create a function that accepts a string of n characters and exchanges the first character with the last, the second with the next – to – last, and so forth until n exchanges have been made. What will the final string look like? Write the function to verify your conclusion.

⇒

```
DELIMITER //
CREATE FUNCTION ExchangeCharacters(inputString
VARCHAR(255))
```

```
RETURNS VARCHAR(255)
DETERMINISTIC
BEGIN
    DECLARE resultString VARCHAR(255);
    DECLARE strLength INT;
    DECLARE i INT;
    SET strLength = LENGTH(inputString);

    SET resultString = inputString;
    SET i = 1;
    WHILE i <= strLength / 2 DO
        SET resultString = CONCAT(
            LEFT(resultString, i - 1),
            SUBSTRING(resultString, strLength - i + 1, 1),
SUBSTRING(resultString, i + 1, strLength - (i * 2)),
            SUBSTRING(resultString, i, 1),
SUBSTRING(resultString, strLength - i + 2)                );
        SET i = i + 1;
    END WHILE;

    RETURN resultString;
END //
DELIMITER ;
SELECT ExchangeCharacters('abcdefg');
SELECT ExchangeCharacters('hello');
```