

## Assignment No. 10

Q1) Write a procedure to find the factorial of the number.  
Hint: use only repeat loop

solution:

```
mysql> DELIMITER &&
mysql>
mysql> CREATE PROCEDURE CalculateFactorial(IN num INT, OUT factorial INT)
-> BEGIN
->     DECLARE temp INT;
->     DECLARE i INT;
->
->     SET factorial = 1;
->     SET i = 1;
->
->     REPEAT
->         SET temp = factorial * i;
->         SET factorial = temp;
->         SET i = i + 1;
->     UNTIL i > num END REPEAT;
->
-> END &&
```

Query OK, 0 rows affected (0.99 sec)

```
mysql>
mysql> DELIMITER ;
mysql> CALL CalculateFactorial(5, @result);
Query OK, 0 rows affected (0.17 sec)
```

```
mysql> SELECT @result;
```

```
+-----+
| @result |
+-----+
|      120 |
+-----+
```

1 row in set (0.00 sec)

```
mysql> CALL CalculateFactorial(5, @result);
Query OK, 0 rows affected (0.00 sec)
```

```
mysql> CALL CalculateFactorial(6, @result);
Query OK, 0 rows affected (0.00 sec)
```

```
mysql> SELECT @result;
```

```
+-----+
| @result |
+-----+
```

```
|      720 |
+-----+
1 row in set (0.00 sec)
```

2) Create a procedure to find the sum of digits of the number passed as parameter to it.  
Hint: use while loop

solution:

```
mysql> DELIMITER $$
mysql>
mysql> CREATE PROCEDURE CalculateDigitSum(IN num INT, OUT digit_sum INT)
-> BEGIN
->     DECLARE remainder INT;
->     DECLARE temp_num INT;
->
->     SET digit_sum = 0;
->     SET temp_num = num;
->
->     WHILE temp_num > 0 DO
->         SET remainder = temp_num % 10;
->         SET digit_sum = digit_sum + remainder;
->         SET temp_num = temp_num DIV 10;
->     END WHILE;
-> END $$
```

```
mysql> DELIMITER ;
mysql> CALL CalculateDigitSum(2467, @result);
Query OK, 0 rows affected (0.00 sec)
```

```
mysql> SELECT @result;
+-----+
| @result |
+-----+
|      19 |
+-----+
1 row in set (0.00 sec)
```

```
mysql> CALL CalculateDigitSum(45765, @result);
Query OK, 0 rows affected (0.00 sec)
```

```
mysql> SELECT @result;
+-----+
| @result |
+-----+
|      27 |
+-----+
1 row in set (0.00 sec)
```

Q3) Write a procedure to print the Fibonacci sequence using any kind of the loop.  
e.g. 0, 1, 1, 2, 3, 5, 8...

solution:

```
mysql> DELIMITER ##
mysql>
mysql> CREATE PROCEDURE FibonacciSequence(IN max_num INT, OUT fibonacci_sequence
VARCHAR(250))
-> BEGIN
->     DECLARE a INT DEFAULT 0;
->     DECLARE b INT DEFAULT 1;
->     DECLARE next_term INT;
->
->     SET fibonacci_sequence = '';
->
->     IF max_num >= 0 THEN
->         SET max_num = max_num - 1;
->         SET fibonacci_sequence = CONCAT(fibonacci_sequence, '0');
->     END IF;
->
->     WHILE max_num > 0 DO
->         SET next_term = a + b;
->         SET fibonacci_sequence = CONCAT(fibonacci_sequence, ', ',
next_term);
->         SET a = b;
->         SET b = next_term;
->         SET max_num = max_num - 1;
->     END WHILE;
->
-> END ##
```

Query OK, 0 rows affected (0.21 sec)

```
mysql>
mysql> DELIMITER ;
mysql> CALL FibonacciSequence(10,@fibonacci_sequence);
Query OK, 0 rows affected (0.04 sec)
```

```
mysql> SELECT @fibonacci_sequence;
+-----+
| @fibonacci_sequence |
+-----+
| 0, 1, 2, 3, 5, 8, 13, 21, 34, 55 |
+-----+
1 row in set (0.00 sec)
```

Q.4)Write a procedure to print all the odd and even numbers separately using same procedure.

solution;

```
mysql> DELIMITER @@
mysql>
mysql> CREATE PROCEDURE PrintOddEvenNumbers(IN max_num INT, OUT odd_numbers
VARCHAR(255), OUT even_numbers VARCHAR(255))
-> BEGIN
->     DECLARE current_num INT DEFAULT 1;
->
->     SET odd_numbers = '';
->     SET even_numbers = '';
->
->     WHILE current_num <= max_num DO
->         IF current_num % 2 <> 0 THEN
->             SET odd_numbers = CONCAT(odd_numbers, current_num, ', ');
->         ELSE
->             SET even_numbers = CONCAT(even_numbers, current_num, ', ');
->         END IF;
->         SET current_num = current_num + 1;
->     END WHILE;
->
-> END @@
```

Query OK, 0 rows affected (0.17 sec)

```
mysql>
mysql> DELIMITER ;
mysql> CALL PrintOddEvenNumbers(20, @odd_numbers, @even_numbers);
Query OK, 0 rows affected (0.00 sec)
```

```
mysql> SELECT @odd_numbers AS Odd_Numbers, @even_numbers AS Even_Numbers;
```

Odd_Numbers	Even_Numbers
1, 3, 5, 7, 9, 11, 13, 15, 17, 19,	2, 4, 6, 8, 10, 12, 14, 16, 18, 20,

1 row in set (0.00 sec)

