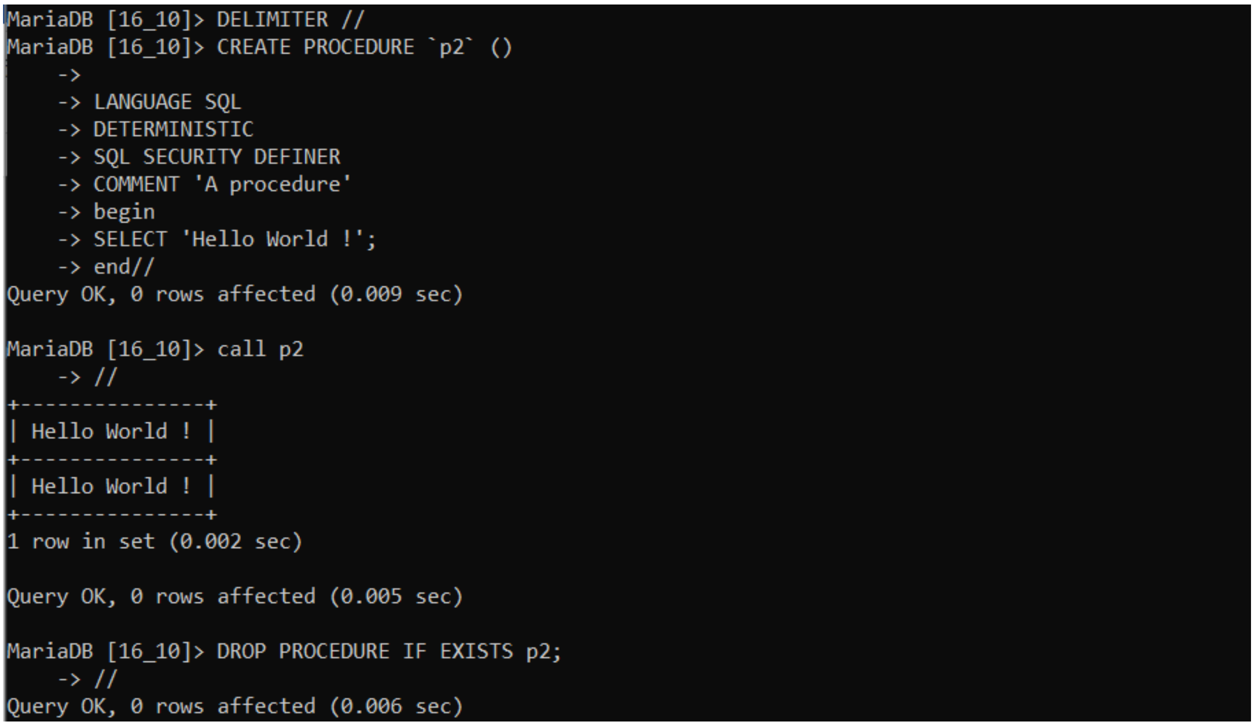
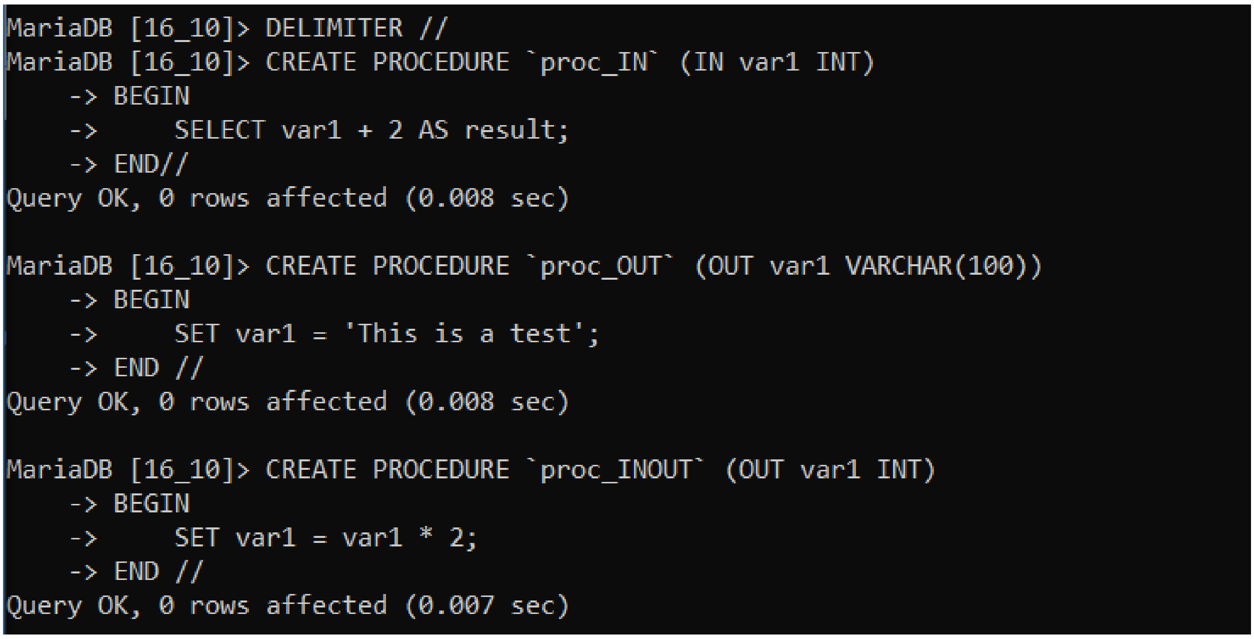
Отчет по хранимым процедурам и выполнение примеров из изученного материала

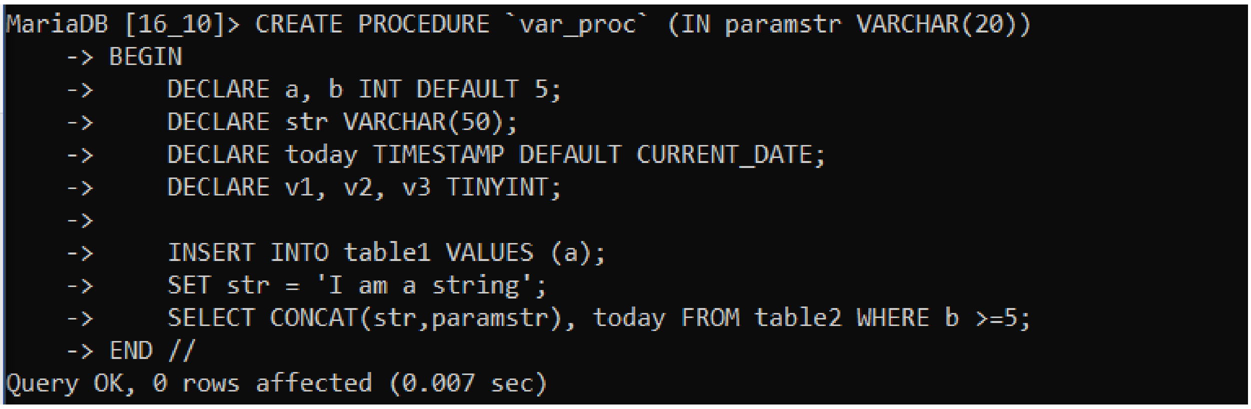
Выполнил Мамонов Антон 2ИСиП-19-1



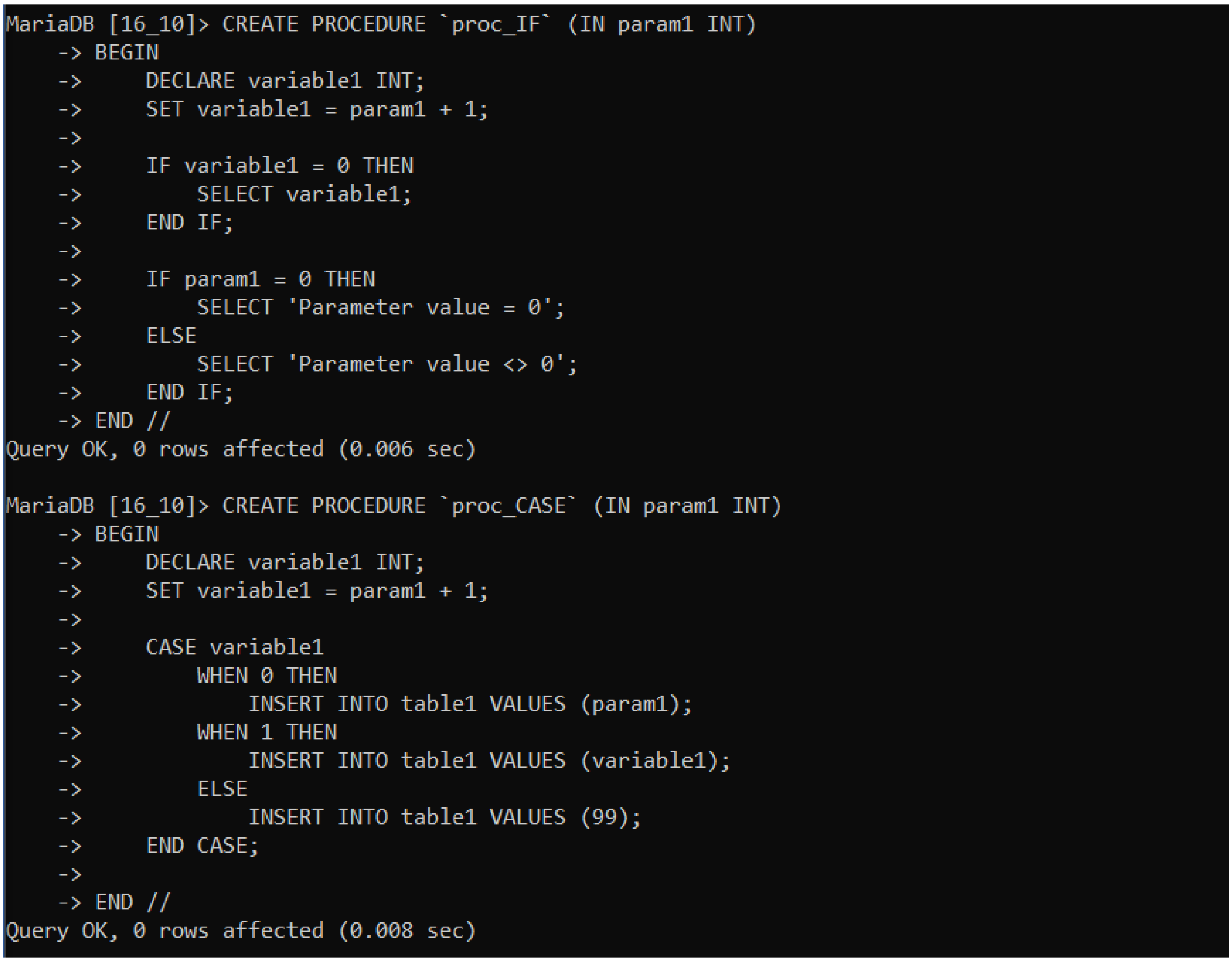
```
MariaDB [16_10]> DELIMITER //
MariaDB [16_10]> CREATE PROCEDURE `p2` ()
    ->
    -> LANGUAGE SQL
    -> DETERMINISTIC
    -> SQL SECURITY DEFINER
    -> COMMENT 'A procedure'
    -> begin
    -> SELECT 'Hello World !';
    -> end//
Query OK, 0 rows affected (0.009 sec)

MariaDB [16_10]> call p2
    -> //
+---------------+
| Hello World ! |
+---------------+
| Hello World ! |
+---------------+
1 row in set (0.002 sec)

Query OK, 0 rows affected (0.005 sec)

MariaDB [16_10]> DROP PROCEDURE IF EXISTS p2;
    -> //
Query OK, 0 rows affected (0.006 sec)
```



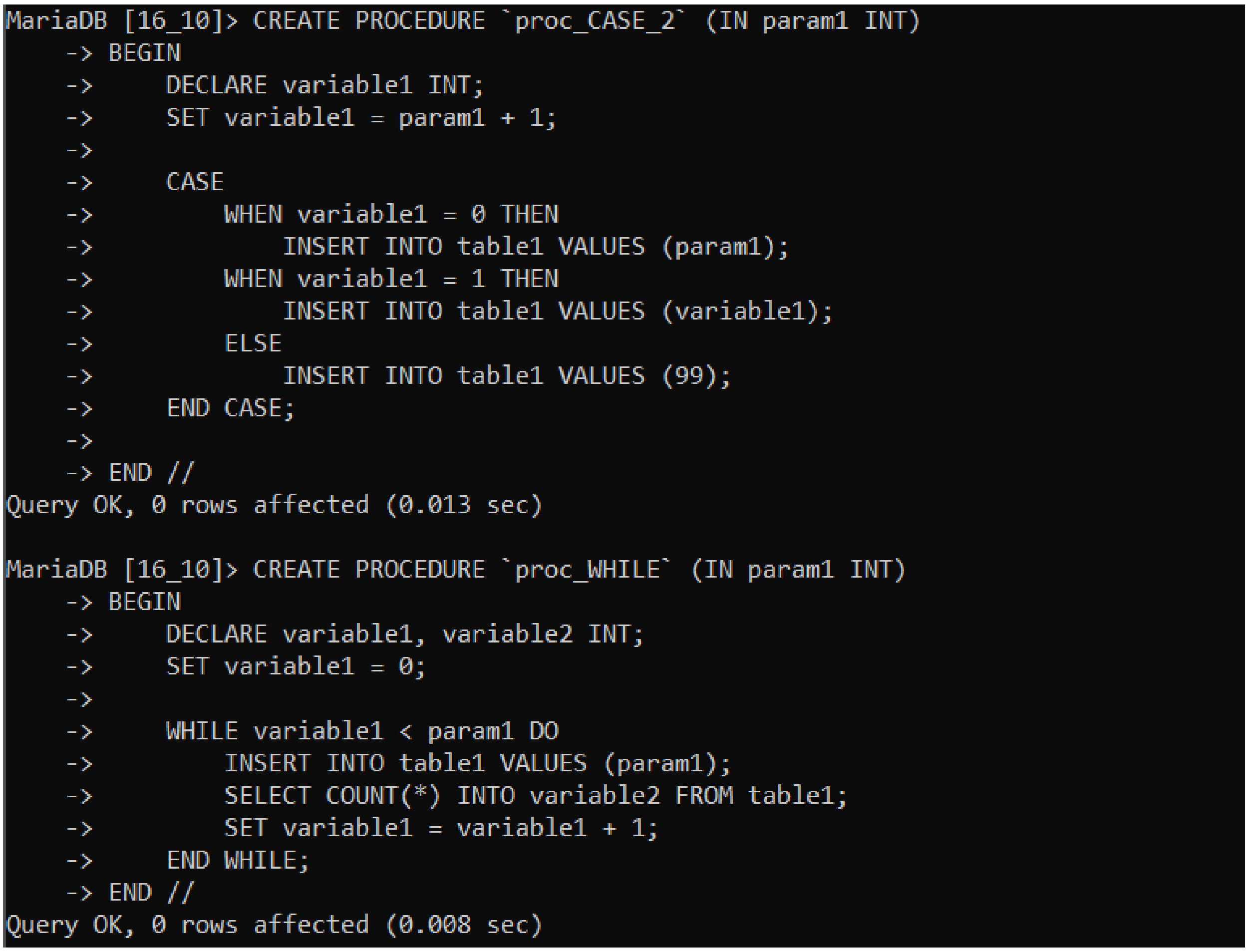
```
MariaDB [16_10]> DELIMITER //
MariaDB [16_10]> CREATE PROCEDURE `proc_IN` (IN var1 INT)
    -> BEGIN
    ->     SELECT var1 + 2 AS result;
    -> END//
Query OK, 0 rows affected (0.008 sec)

MariaDB [16_10]> CREATE PROCEDURE `proc_OUT` (OUT var1 VARCHAR(100))
    -> BEGIN
    ->     SET var1 = 'This is a test';
    -> END //
Query OK, 0 rows affected (0.008 sec)

MariaDB [16_10]> CREATE PROCEDURE `proc_INOUT` (OUT var1 INT)
    -> BEGIN
    ->     SET var1 = var1 * 2;
    -> END //
Query OK, 0 rows affected (0.007 sec)
```



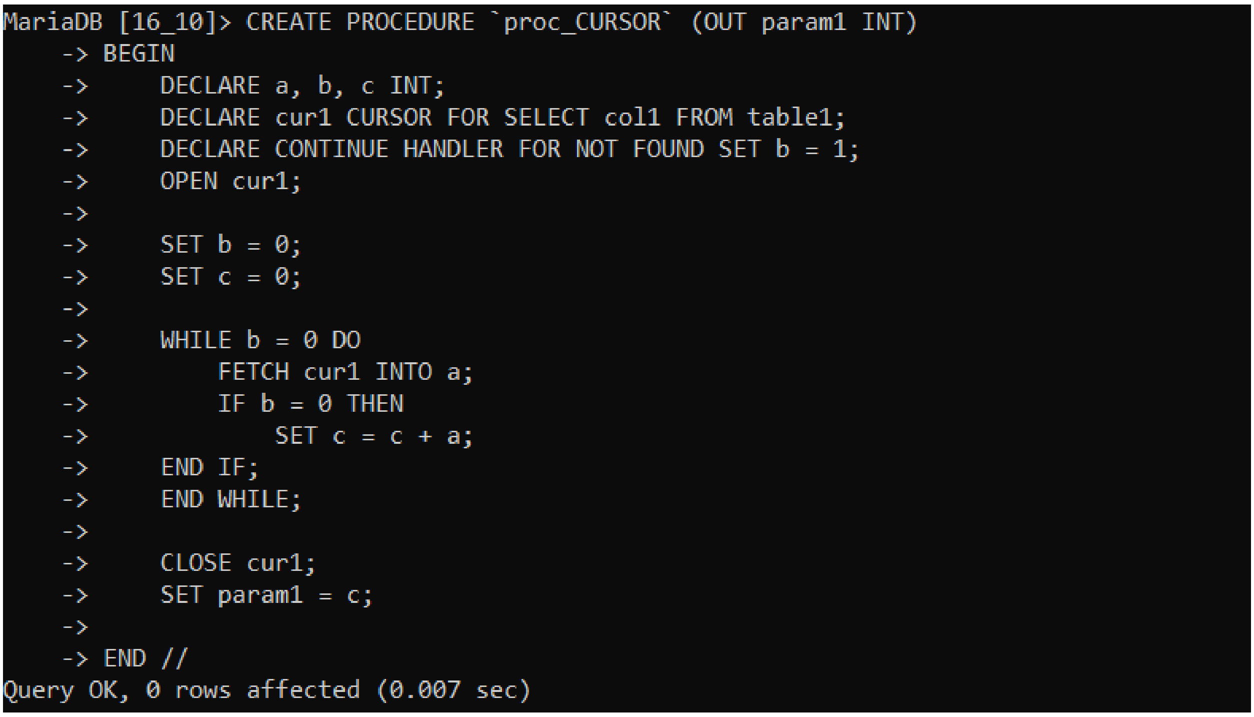
```
MariaDB [16_10]> CREATE PROCEDURE `var_proc` (IN paramstr VARCHAR(20))
    -> BEGIN
    ->     DECLARE a, b INT DEFAULT 5;
    ->     DECLARE str VARCHAR(50);
    ->     DECLARE today TIMESTAMP DEFAULT CURRENT_DATE;
    ->     DECLARE v1, v2, v3 TINYINT;
    ->
    ->     INSERT INTO table1 VALUES (a);
    ->     SET str = 'I am a string';
    ->     SELECT CONCAT(str,paramstr), today FROM table2 WHERE b >=5;
    -> END //
Query OK, 0 rows affected (0.007 sec)
```



```
MariaDB [16_10]> CREATE PROCEDURE `proc_IF` (IN param1 INT)
    -> BEGIN
    ->     DECLARE variable1 INT;
    ->     SET variable1 = param1 + 1;
    ->
    ->     IF variable1 = 0 THEN
    ->         SELECT variable1;
    ->     END IF;
    ->
    ->     IF param1 = 0 THEN
    ->         SELECT 'Parameter value = 0';
    ->     ELSE
    ->         SELECT 'Parameter value <> 0';
    ->     END IF;
    -> END //
Query OK, 0 rows affected (0.006 sec)

MariaDB [16_10]> CREATE PROCEDURE `proc_CASE` (IN param1 INT)
    -> BEGIN
    ->     DECLARE variable1 INT;
    ->     SET variable1 = param1 + 1;
    ->
    ->     CASE variable1
    ->         WHEN 0 THEN
    ->             INSERT INTO table1 VALUES (param1);
    ->         WHEN 1 THEN
    ->             INSERT INTO table1 VALUES (variable1);
    ->         ELSE
    ->             INSERT INTO table1 VALUES (99);
    ->     END CASE;
    ->
    -> END //
Query OK, 0 rows affected (0.008 sec)
```



```
MariaDB [16_10]> CREATE PROCEDURE `proc_CASE_2` (IN param1 INT)
    -> BEGIN
    ->     DECLARE variable1 INT;
    ->     SET variable1 = param1 + 1;
    ->
    ->     CASE
    ->         WHEN variable1 = 0 THEN
    ->             INSERT INTO table1 VALUES (param1);
    ->         WHEN variable1 = 1 THEN
    ->             INSERT INTO table1 VALUES (variable1);
    ->         ELSE
    ->             INSERT INTO table1 VALUES (99);
    ->     END CASE;
    ->
    -> END //
Query OK, 0 rows affected (0.013 sec)

MariaDB [16_10]> CREATE PROCEDURE `proc_WHILE` (IN param1 INT)
    -> BEGIN
    ->     DECLARE variable1, variable2 INT;
    ->     SET variable1 = 0;
    ->
    ->     WHILE variable1 < param1 DO
    ->         INSERT INTO table1 VALUES (param1);
    ->         SELECT COUNT(*) INTO variable2 FROM table1;
    ->         SET variable1 = variable1 + 1;
    ->     END WHILE;
    -> END //
Query OK, 0 rows affected (0.008 sec)
```



```
MariaDB [16_10]> CREATE PROCEDURE `proc_CURSOR` (OUT param1 INT)
    -> BEGIN
    ->     DECLARE a, b, c INT;
    ->     DECLARE cur1 CURSOR FOR SELECT col1 FROM table1;
    ->     DECLARE CONTINUE HANDLER FOR NOT FOUND SET b = 1;
    ->     OPEN cur1;
    ->
    ->     SET b = 0;
    ->     SET c = 0;
    ->
    ->     WHILE b = 0 DO
    ->         FETCH cur1 INTO a;
    ->         IF b = 0 THEN
    ->             SET c = c + a;
    ->     END IF;
    ->     END WHILE;
    ->
    ->     CLOSE cur1;
    ->     SET param1 = c;
    ->
    -> END //
Query OK, 0 rows affected (0.007 sec)
```