LAB 3 “MySQL”

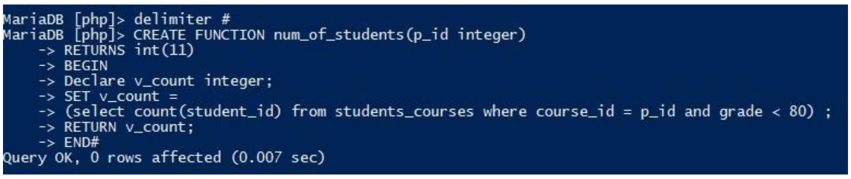
(Ahmed Ashraf Ibrahim)

MySQL Labs

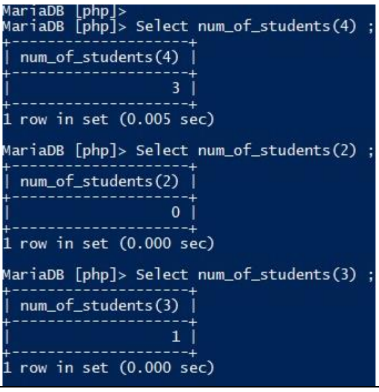
insert into students_courses
values
(1,4,60,NULL), (2,1,NULL,NULL), (2,4,75,NULL),
(3,1,NULL,NULL), (3,2,NULL,NULL), (3,3,75,NULL);

1 *Create function to calculate the number of students who get grade less than 80 in a certain exam (course id will be sent as a parameter)*

```
drop function if exists num_of_students;

delimiter #
CREATE FUNCTION num_of_students(p_id integer)
RETURNS int(11)

 BEGIN
   Declare v_count integer;
   SET v_count =
   (select count(student_id) from students_courses
    where course_id = p_id    and grade < 80) ;
   RETURN v_count;
 END#

delimiter ;
```



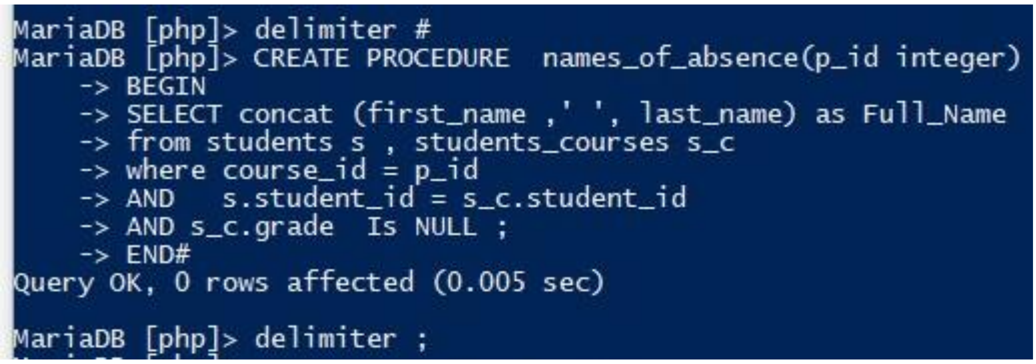
```
MariaDB [php]> delimiter #
MariaDB [php]> CREATE FUNCTION num_of_students(p_id integer)
    -> RETURNS int(11)
    -> BEGIN
    -> Declare v_count integer;
    -> SET v_count =
    -> (select count(student_id) from students_courses where course_id = p_id and grade < 80) ;
    -> RETURN v_count;
    -> END#
Query OK, 0 rows affected (0.007 sec)
```



```
MariaDB [php]>
MariaDB [php]> Select num_of_students(4) ;
+--------------------+
| num_of_students(4) |
+--------------------+
|                  3 |
+--------------------+
1 row in set (0.005 sec)

MariaDB [php]> Select num_of_students(2) ;
+--------------------+
| num_of_students(2) |
+--------------------+
|                  0 |
+--------------------+
1 row in set (0.000 sec)

MariaDB [php]> Select num_of_students(3) ;
+--------------------+
| num_of_students(3) |
+--------------------+
|                  1 |
+--------------------+
1 row in set (0.000 sec)
```

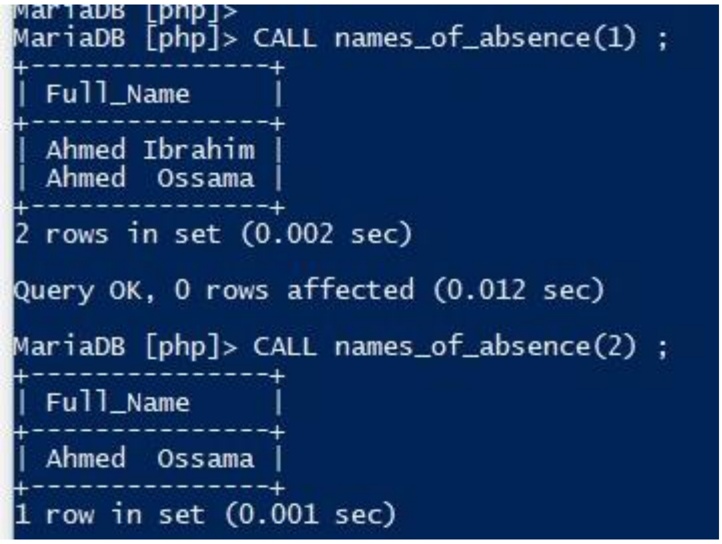
2 *Create stored procedure to display the names of the absence students of a certain courses.(Absent means has no grades)*

```
drop procedure if exists names_of_absence;
delimiter #
CREATE PROCEDURE  names_of_absence(p_id integer)
BEGIN
SELECT concat (first_name ,' ', last_name) as Full_Name
from students s , students_courses s_c
where course_id = p_id
AND   s.student_id = s_c.student_id
AND s_c.grade  Is NULL ;
END#
delimiter ;
```



```
MariaDB [php]> delimiter #
MariaDB [php]> CREATE PROCEDURE  names_of_absence(p_id integer)
    -> BEGIN
    -> SELECT concat (first_name ,' ', last_name) as Full_Name
    -> from students s , students_courses s_c
    -> where course_id = p_id
    -> AND   s.student_id = s_c.student_id
    -> AND s_c.grade  Is NULL ;
    -> END#
Query OK, 0 rows affected (0.005 sec)

MariaDB [php]> delimiter ;
```

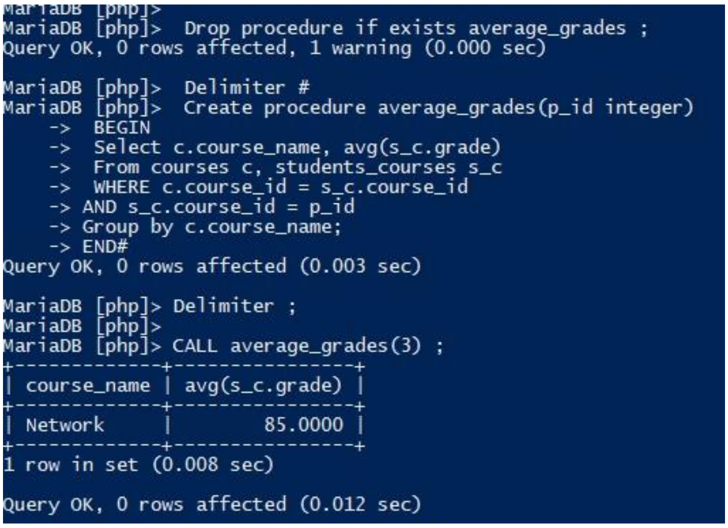
```
CALL names_of_absence(1) ;
```



```
MariaDB [php]>
MariaDB [php]> CALL names_of_absence(1) ;
+---------------+
| Full_Name     |
+---------------+
| Ahmed Ibrahim |
| Ahmed  Ossama |
+---------------+
2 rows in set (0.002 sec)

Query OK, 0 rows affected (0.012 sec)

MariaDB [php]> CALL names_of_absence(2) ;
+---------------+
| Full_Name     |
+---------------+
| Ahmed  Ossama |
+---------------+
1 row in set (0.001 sec)
```

3 ***Create stored procedure to calculate the average grades for certain course.***

```
Drop procedure if exists average_grades ;

Delimiter #
Create procedure average_grades(p_id integer)

BEGIN
 Select c.course_name, avg(s_c.grade)
 From courses c, students_courses s_c
 WHERE c.course_id = s_c.course_id
 AND s_c.course_id = p_id
 Group by c.course_name;
END#


Delimiter ;
```

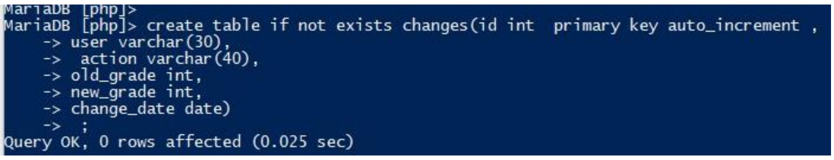


```
MariaDB [php]>
MariaDB [php]>  Drop procedure if exists average_grades ;
Query OK, 0 rows affected, 1 warning (0.000 sec)

MariaDB [php]>  Delimiter #
MariaDB [php]>  Create procedure average_grades(p_id integer)
    ->  BEGIN
    ->  Select c.course_name, avg(s_c.grade)
    ->  From courses c, students_courses s_c
    ->  WHERE c.course_id = s_c.course_id
    -> AND s_c.course_id = p_id
    -> Group by c.course_name;
    -> END#
Query OK, 0 rows affected (0.003 sec)

MariaDB [php]> Delimiter ;
MariaDB [php]>
MariaDB [php]> CALL average_grades(3) ;
+-------------+----------------+
| course_name | avg(s_c.grade) |
+-------------+----------------+
| Network     |        85.0000 |
+-------------+----------------+
1 row in set (0.008 sec)

Query OK, 0 rows affected (0.012 sec)
```

4 *Create trigger to keep track the changes(updates) of the grades in the studnets_courses table*
(create changes table with the following fields:
id int primary key auto_increment , user varchar(30) , action varchar(40), old_grade int, new_grade int, change_date date).

Test the trigger by updating grade int the "Students_courses" table
Confirm that the row is added in the" change_table"

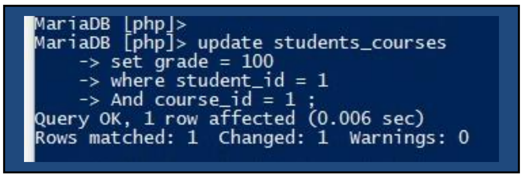
```
create table if not exists changes(id int  primary key
auto_increment ,
    -> user varchar(30) , action varchar(40), old_grade int,
    -> new_grade int, change_date date) ;
```



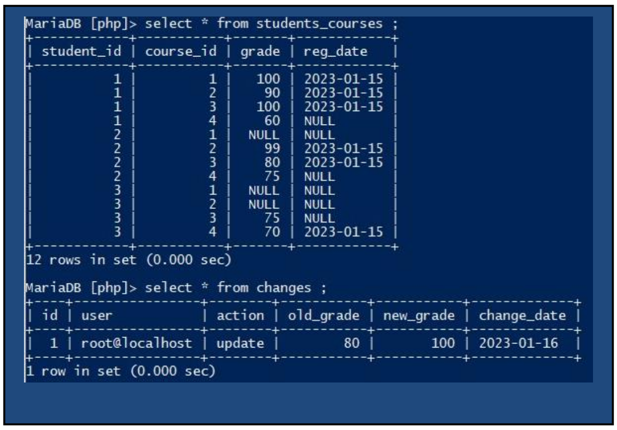
```
MariaDB [php]>
MariaDB [php]> create table if not exists changes(id int  primary key auto_increment ,
    -> user varchar(30),
    ->  action varchar(40),
    -> old_grade int,
    -> new_grade int,
    -> change_date date)
    ->  ;
Query OK, 0 rows affected (0.025 sec)
```

```
> delimiter #
> CREATE TRIGGER update_grade
    -> AFTER Update ON students_courses
    -> FOR EACH ROW
    -> INSERT INTO changes(user, action, old_grade,
         new_grade,  change_date)

    -> VALUES (current_user(), "update" , OLD.grade,
         NEW.grade , current_date());
    -> END#

> delimiter ;
```

```
update students_courses
   -> set grade = 100
   -> where student_id = 1
   -> And course_id = 1 ;
```



```
MariaDB [php]>
MariaDB [php]> update students_courses
    -> set grade = 100
    -> where student_id = 1
    -> And course_id = 1 ;
Query OK, 1 row affected (0.006 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

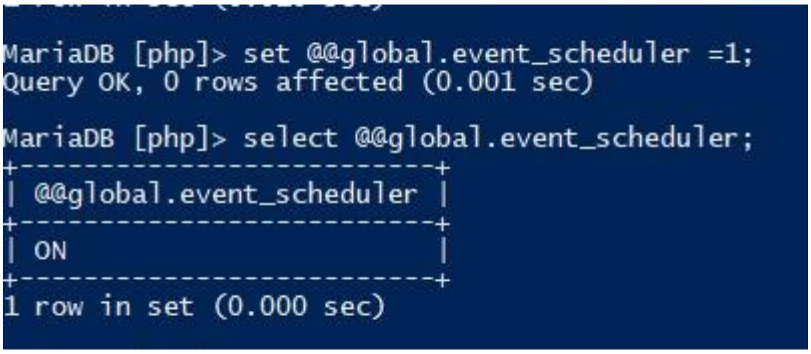


```
MariaDB [php]> select * from students_courses ;
+------------+-----------+-------+------------+
| student_id | course_id | grade | reg_date   |
+------------+-----------+-------+------------+
|          1 |         1 |   100 | 2023-01-15 |
|          1 |         2 |    90 | 2023-01-15 |
|          1 |         3 |   100 | 2023-01-15 |
|          1 |         4 |    60 | NULL       |
|          2 |         1 |  NULL | NULL       |
|          2 |         2 |    99 | 2023-01-15 |
|          2 |         3 |    80 | 2023-01-15 |
|          2 |         4 |    75 | NULL       |
|          3 |         1 |  NULL | NULL       |
|          3 |         2 |  NULL | NULL       |
|          3 |         3 |    75 | NULL       |
|          3 |         4 |    70 | 2023-01-15 |
+------------+-----------+-------+------------+
12 rows in set (0.000 sec)

MariaDB [php]> select * from changes ;
+----+----------------+--------+-----------+-----------+-------------+
| id | user           | action | old_grade | new_grade | change_date |
+----+----------------+--------+-----------+-----------+-------------+
|  1 | root@localhost | update |        80 |       100 | 2023-01-16  |
+----+----------------+--------+-----------+-----------+-------------+
1 row in set (0.000 sec)
```

5	*Create event to delete the changes tables every 5 minute*

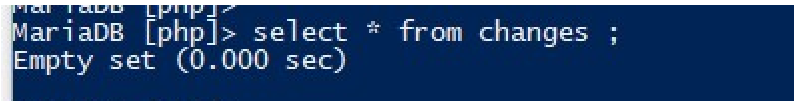
set @@global.event_scheduler =1;

select @@global.event_scheduler;



```
MariaDB [php]> set @@global.event_scheduler =1;
Query OK, 0 rows affected (0.001 sec)

MariaDB [php]> select @@global.event_scheduler;
+--------------------------+
| @@global.event_scheduler |
+--------------------------+
| ON                       |
+--------------------------+
1 row in set (0.000 sec)
```

CREATE EVENT delete_changes
ON SCHEDULE EVERY 5 MINUTES
DO
DELETE FROM changes



```
MariaDB [php]>
MariaDB [php]> select * from changes ;
Empty set (0.000 sec)
```