MySQL Labs

MySQL (Day3):

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)***
	delimiter $ *create function count_students_less_80(p_id integer)* *returns int(11)* *begin* *declare v_count_students int(11);* *set v_count_students = (select count(grade) as count* *from students_courses* *where course_id = p_id and* *grade < 80);* *return v_count_students;* *end$* *delimiter ;*
2	***Create stored procedure to display the names of the absence students of a certain courses.(Absent means has no grades)***
	delimiter$ *create procedure display_absent_students_numbers()* *begin* *select s.first_name , s.last_name , c.course_name* *from students s, courses c, students_courses sc* *where s.student_id = sc.student_id* *and c.course_id = sc.course_id* *and sc.grade is NULL;* *end$* *delimiter ;*
3	***Create stored procedure to calculate the average grades for certain course.***
	delimiter $ *create procedure count_course_average_grade(p_course_id integer)* *begin* *select sum(grade) , count(student_id) , sum(grade)/count(student_id) as average_grade*

	from students_courses *where p_course_id = course_id;* *end$* *delimiter ;*
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 ,*** ***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 TRIGGER updated_grades_log *AFTER UPDATE ON students_courses* *FOR EACH ROW* *INSERT INTO changes_table values(* *default,* *current_user(),* *'update',* *old.grade,* *new.grade,* *current_date()* *);* *update students_courses set grade = 55 where student_id=1 and course_id =1;* MariaDB [php]> select * from students_courses; +------------+-----------+-------+------------+ | student_id | course_id | grade | reg_date | +------------+-----------+-------+------------+ | 1 | 1 | 55 | 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 (0.001 sec) MariaDB [php]> select * from changes_table;; +----+----------------+--------+-----------+-----------+-------------+ | id | user | action | old_grade | new_grade | change_date | +----+----------------+--------+-----------+-----------+-------------+ | 1 | root@localhost | update | 80 | 55 | 2023-01-16 | +----+----------------+--------+-----------+-----------+-------------+ 1 row in set (0.000 sec)

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

	`Create event delete_changes`
	`On schedule every 5 minute`
	`Do`
	`Delete from changes_table;`