

# 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)
	<pre>create function num_student_less_80(c_id integer) returns int(11) begin declare v_count integer; set v_count = (select count(grade) from students_courses where grade&lt;80 and course_id=c_id); return v_count; end /</pre>
2	Create stored procedure to display the names of the absence students of a certain courses.(Absence means has no grades)
	<pre>create procedure display_absent_students(c_id integer) begin select concat(s.first_name,' ',s.last_name) as full_name from students s,students_courses c where s.student_id = c.student_id and c.course_id = c_id and grade is NULL; end /</pre>
3	Create stored procedure to calculate the average grades for certain course.
	<pre>create procedure avg_garde(c_id integer) begin select avg(grade) as grade_avg from students_courses where course_id = c_id; end /</pre>
4	Create trigger to keep track the changes(updates) of the grades in the students_courses table ( create <u>changes table</u> with the following fields: id int primary key , user varchar(30), action varchar(40), old_grade int, new_grade int, change_date date).

	<p><i>Test the trigger by updating grade int the “Students_courses” table</i></p> <p><i>Confirm that the row is added in the” change_table”</i></p>
	<pre> create trigger grade_track_change after update on students_courses for each row if (OLD.grade &lt;&gt; NEW.grade) then insert into changes(id,user,action,old_grade,new_grade,change_date) values(OLD.student_id,current_user(),"Update",OLD.grade,NEW.grade,now()) end if / </pre>
◦	<p><i>Create event to delete the changes tables every 5 minute</i></p>
	<pre> create event delete_changes on schedule every 5 minute do delete from changes; </pre>