*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)*** |
|  | drop function if exists count\_std;  delimiter $  CREATE FUNCTION count\_std(p\_id integer)  RETURNS int(11)  BEGIN  declare x int ;  set x= (select count(student\_id) from students\_courses where grade < 80 and course\_id=p\_id) ;  RETURN x;  END$  delimiter ; |
| *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 absnt\_std;  delimiter $  CREATE PROCEDURE absnt\_std(p\_id integer)  BEGIN  Select concat(first\_name,'',last\_name) as Fullname  from students s,students\_courses sc  where s.student\_id=sc.student\_id  and grade IS null  and course\_id=p\_id;END$  delimiter ; |
| *3* | ***Create stored procedure to calculate the average grades for certain course.*** |
|  | drop procedure if exists avg\_grade;  delimiter $  CREATE PROCEDURE avg\_grade(p\_id integer)  BEGIN  Select avg(grade) as AverageGrade  from students\_courses  where course\_id=p\_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 TABLE IF NOT EXISTS `changed\_grade` (  `change\_id` INT(11) NOT NULL AUTO\_INCREMENT,  `user` VARCHAR(45) NULL,  `action` VARCHAR(45) NULL,  `old\_grade` int(11) NULL,  `new\_grade` int(11) NULL,  `change\_date` date,  PRIMARY KEY (`change\_id`)  );  /\*Table creation\*/  CREATE TRIGGER std\_grade\_log  AFTER Update ON students\_courses  FOR EACH ROW  INSERT INTO changed\_grade (username,`action`,old\_grade ,new\_grade,change\_date)  VALUES (user(),'update',OLD.grade,NEW.grade,now()); |
| *5* | ***Create event to delete the changes tables every 5 minute*** |
|  | CREATE EVENT delete\_changes  ON SCHEDULE EVERY 5 minute  DO  DELETE FROM changed\_grade; |