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 $**  **CREATE FUNCTION numOfStudents(p\_courseID integer)**  **RETURNS INT**  **BEGIN**  **Return(Select count(\*)**  **From students\_courses**  **Where course\_id = p\_courseID**  **And grade < 80**  **);**  **END$**  **Delimiter ;**  **Graphical user interface, text  Description automatically generated** |
| *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 displayNameOfAbsent;**  **Delimiter $**  **CREATE procedure displayNameOfAbsent (p\_courseID integer)**  **BEGIN**  **select concat (first\_name ," ",last\_name) as fullname**  **from students\_courses sc, students s**  **where p\_courseID = sc.course\_id**  **and sc.grade is NULL**  **and s.student\_id = sc.student\_id ;**  **END$**  **Delimiter ;**  *Text  Description automatically generated* |
| *3* | ***Create stored procedure to calculate the average grades for certain course.*** |
|  | **Delimiter $**  **CREATE procedure calcAvrGrades(p\_courseID integer)**  **BEGIN**  **select AVG(grade)**  **from students\_courses**  **where p\_courseID = course\_id;**  **END$**  **Delimiter ;**  **Graphical user interface, text  Description automatically generated** |
| *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 `changes\_Table`(**  **id int primary key ,**  **user varchar(30),**  **action varchar(40),**  **old\_grade int,**  **new\_grade int,**  **change\_date date**  **);**  **CREATE TRIGGER keepTrackUpdates**  **After UPDATE on students\_courses**  **FOR EACH ROW**  **INSERT INTO changes\_Table (id, user,action,old\_grade,new\_grade,change\_date)**  **Values (OLD.student\_id, USER(), "UPDATE", OLD.grade, NEW.grade ,now());**  **Text  Description automatically generated** |
| *5* | ***Create event to delete the changes tables every 5 minute*** |
|  | **CREATE EVENT delete\_changes**  **ON SCHEDULE EVERY MINUTE**  **DO**  **DELETE FROM changes\_Table;**  Text  Description automatically generated |