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 low\_grades (f\_id integer)  RETURNS integer  BEGIN  RETURN  (SELECT count(student\_id)  FROM students\_courses  WHERE course\_id = f\_id AND grade<80);  END$  DELIMITER ;  SELECT low\_grades(3); |
| *2* | ***Create stored procedure to display the names of the absence students of a certain courses.(Absent means has no grades)*** |
|  | DELIMITER $  CREATE PROCEDURE course\_absent\_students\_names (f\_course\_id integer)  BEGIN  SELECT DISTINCT (CONCAT(first\_name, ' ', last\_name)) AS full\_name  FROM students S INNER JOIN students\_courses SC  ON S.student\_id = SC.student\_id  WHERE course\_id=f\_course\_id AND grade IS NULL;  END$  DELIMITER ;  CALL course\_absent\_students\_names(1); |
| *3* | ***Create stored procedure to calculate the average grades for certain course.*** |
|  | DELIMITER $  CREATE PROCEDURE average (f\_course\_name char(30))  BEGIN  SELECT AVG(grade)  FROM students\_courses SC INNER JOIN courses C  ON SC.course\_id = C.course\_id AND course\_name = f\_course\_name;  END$  DELIMITER ;  *CALL average(‘c’);* |
| *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 grades\_change\_trigger  AFTER UPDATE  ON students\_courses  FOR EACH ROW  REPLACE INTO change\_table  VALUES (old.student\_id, current\_user(), "update", OLD.grade, NEW.grade, current\_date());  UPDATE students\_courses  SET grade=13  WHERE student\_id=1 AND course\_id=1;  UPDATE students\_courses  SET grade=15  WHERE student\_id=1 AND course\_id=1;  **select \* from change\_table;** |
| *5* | ***Create event to delete the changes tables every 5 minute*** |
|  | create event delete\_changes  on schedule every 5 minute  do  delete from change\_table; |