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_below_grade(course_id_param INT)
RETURNS INT
BEGIN
    DECLARE student_count INT;
    SELECT COUNT(*)
    INTO student_count
    FROM students_courses
    WHERE course_id = course_id_param AND grade < 80;
    RETURN student_count;
END $
DELIMITER ;

SELECT count_students_below_grade(1) AS 'Students Below 80';
``` |
| 2 | *Create stored procedure to display the names of the absence students of a certain courses.(Absent means has no grades)* |
| | ```
DELIMITER $

 CREATE PROCEDURE absen_Students(student_id INT(11))

BEGIN

  SELECT s.first_name

  FROM students s

  JOIN students_courses sc ON s.student_id = sc.student_id
``` |

	WHERE s.student_id = student_Id *AND sc.grade IS NULL;* *END $* *DELIMITER ;* *CALL bsen_Students(1);*
3	**Create stored procedure to calculate the average grades for certain course.**
	Delimeter $ *Create procedure calculate_avg (course_id_param int)* *Begin* *Select avg(grade)* *From students_courses* *Where course_id=course_id_param* *End $* *Delimeter ;*
4	**Create trigger to keep track the changes(updates) of the grades in the studnets_courses table** **(create <u>changes table</u> with the following fields:** **id int primary key ,** **user varchar(30),** **action varchar(40),** **old_grade int,** **new_grade int,** **change_date date).** **Test the trigger by updating grade int the "Students_courses" table** **Confirm that the row is added in the" change_table"**
	create trigger table_changes AFTER UPDATE ON students_courses FOR EACH ROW INSERT INTO changes_table (id, user, action, old_grade, new_grade, change_date)VALUES(OLD.student_id ,USER(), 'update', OLD.grade, NEW.grade, NOW())$
5	**Create event to delete the changes tables every 5 minute**
	```create event delete_changes on schedule every 5 minute do delete from changes;```

6	**Create a user with your name and give him the privilege to access the grades database**
| | ```
create user 'hoda' identified by 'hoda22';
GRANt all ON grades.* to hoda identified by 'hoda22';
``` |
| *7* | **Connect to mysql using the user you created and try to insert one record in the courses table.** |
| | ```
mysql -u hoda -p
use grades
insert into `courses` (`course_id`, `course_name`,
`credit_hour`) values('24','database','1');
``` |
| *8* | **Change your password.** |
| | ```
Set password =password('hoda23');
``` |
| *9* | **Show your privileges.** |
| | ```
show grants;
``` |