**MySQL Labs**

**MySQL (Day2):**

|  |  |
| --- | --- |
| ***1*** | ***Update students courses table, set the registration date value to “Today”;*** |
|  | ***update students\_courses set reg\_date = curdate();*** |
| ***2*** | ***Display the registration date in the following format:***  ***Day, month/ year*** |
|  | ***select date\_format(reg\_date, '%d , %m / %y') as DateFormat from students\_courses;*** |
| ***3*** | ***Display the full name (first, last) of the student with his grade.***  ***if his garde is greater than 85% Excellent, from 75% to 85% Very good, from 65% to 75% Good and from 55% to 65% pass otherwise will be graded as failed.*** |
|  | **select first\_name, last\_name, grade,**  **CASE**  **WHEN grade > 85 THEN "Excellent"**  **WHEN grade < 85 && sc.grade > 75 THEN "very good"**  **WHEN grade < 75 && sc.grade > 65 THEN "good"**  **WHEN grade < 65 && sc.grade > 55 THEN "pass"**  **ELSE "Fail"**  **END**  **from students s**  **left join students\_courses sc on s.student\_id = sc.student\_id;** |
| ***4*** | ***Display the capitalized last name , and the grade , if he has no grade display the keyword absent. [using ifNULL function]*** |
|  | ***select upper(last\_name), ifnull(grade, "absent")***  ***from students e***  ***left join students\_courses c***  ***on e.student\_id = c.student\_id;*** |
| ***5*** | ***Display students' names, course name along with their grades.*** |
|  | ***select s.first\_name , s.last\_name, c.course\_name, sc.grade***  ***from students s, courses c, students\_courses sc***  ***where s.student\_id = sc.student\_id && c.course\_id = sc.course\_id;*** |
| ***6*** | ***For each course, display the course name, min grade, max grade, average grade, number of attended students.*** |
|  | **select c.course\_name, min(sc.grade) as "Min Grade", max(sc.grade) as "Max Grade" ,avg(sc.grade) as "Average Grade", count(sc.student\_id) as "Number of students"**  **from courses c, students\_courses sc**  **where c.course\_id = sc.course\_id**  **group by c.course\_name;** |
| ***7*** | ***Use subquery to display the names of the students who were born before student no 1.*** |
|  | ***select first\_name, last\_name from students***  ***where birth\_date <***  ***(select birth\_date from students where student\_id = 1);*** |
| ***8*** | ***Use subquery to display the data of all the courses with a credit hour similar to MySQL's credit hours*** |
|  | ***select \* from courses***  ***where credit\_hour =***  ***(select credit\_hour from courses where course\_name = "MySQL");*** |
| ***10*** | ***Create a view called female\_students\_vu to display all the female students*** |
|  | ***create view female\_students\_vu as***  ***select \* from students where gender = "female";*** |
| ***11*** | ***Try to insert a male student through your view*** |
|  | ***insert into female\_students\_vu values(10, "ziad", "bebo", null, null, "male", curdate());*** |
| ***12*** | ***Select all the data from your view and then from the students table*** |
|  | ***select \* from female\_students\_vu;***  ***select \* from students;*** |
| ***13*** | ***Prevent the ability to insert another male student through you view*** |
|  | ***alter view female\_students\_vu as***  ***select \* from students where gender = "female"***  ***with check option;*** |
| ***14*** | ***Use the information schema to display the table name , schema and the updatability of the female\_students\_vu view*** |
|  | ***select TABLE\_NAME, TABLE\_SCHEMA, is\_updatable from views where table\_name = "female\_students\_vu";*** |
| ***15*** | ***Use the information schema to display the create time, table\_rows, auto\_increment, and the comments on the students table.*** |
|  | ***select create\_time, table\_rows, auto\_increment, table\_comment from tables where table\_name = "students";*** |
| ***16*** | ***Create a nonunique index on the foreign key column (COURSE\_ID) in the students\_courses table.*** |
|  | ***create index myIndex on students\_courses (course\_id);*** |
| ***17*** | ***Create a user with your name and give him the privilege to access the grades database*** |
|  | ***create user "shehab";***  ***grant all on grades.\* to "shehab";*** |
| ***18*** | ***Connect to mysql using the user you created and try to insert one record in the courses table.*** |
|  | ***mysql -u shehab***  ***use grades;***  ***insert into courses values(10, "Red Hat",7);*** |
| ***19*** | ***Change your password.*** |
|  | ***alter user "shehab" identified by "test123";*** |
| ***20*** | ***Show your privileges.*** |
|  | ***show grants; // from shehab***  ***show grants for "shehab"; // from root*** |