**MySQL Labs**

**MySQL (Day2):**

|  |  |
| --- | --- |
| ***1*** | ***Update students courses table, set the registration date value to “Today”;*** |
|  | ***update students\_courses***  ***set reg\_date = current\_date();*** |
| ***2*** | ***Display the registration date in the following format:***  ***Day, month/ year*** |
|  | ***Select DATE\_FORMAT(reg\_date, '%d , %M / %Y' )***  ***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 concat\_ws(' ', s.first\_name, s.last\_name) as fullName,**  **if( cs.grade >= 85,**  **"Excellent", if(cs.grade >= 75,**  **"Very good", if(cs.grade >= 65,**  **"Good",if(cs.grade >= 55,**  **"Pass", "Fail"))) )**  **as grade**  **from students s inner join students\_courses cs**  **on (s.student\_id = cs.student\_id);**  **select concat\_ws(' ', s.first\_name, s.last\_name) as fullName,**  **case**  **when cs.grade >= 85 then "Excellent"**  **when cs.grade >= 75 then "Very good"**  **when cs.grade >= 65 then "Good"**  **when cs.grade >= 55 then "Pass"**  **else "Fail"**  **end as grade**  **from students s, students\_courses cs**  **where s.student\_id = cs.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(s.last\_name), ifnull(grade, "absent")***  ***from students s left outer join students\_courses cs***  ***on (s.student\_id = cs.student\_id);*** |
| ***5*** | ***Display students' names, course name along with their grades.*** |
|  | ***select s.first\_name, c.course\_name, sc.grade***  ***from students s, courses c, students\_courses sc***  ***where***  ***s.student\_id = sc.student\_id***  ***and***  ***c.course\_id = sc.course\_id;***  ***select s.first\_name, c.course\_name, sc.grade***  ***from students s***  ***inner join students\_courses sc***  ***on s.student\_id = sc.student\_id***  ***inner join courses c***  ***on 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), max(sc.grade), avg(sc.grade), count(s.first\_name)**  **from courses c, students\_courses sc, students s**  **where**  **c.course\_id = sc.course\_id**  **and**  **s.student\_id = sc.student\_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 from students***  ***where birth\_date < (***  ***select birth\_date from students***  ***limit 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'***  ***);***  ***--***  ***(without mysql)***  ***select \* from courses***  ***where credit\_hour = (***  ***select credit\_hour from courses***  ***where course\_name = 'mysql'***  ***)***  ***and 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 (student\_id, first\_name, gender) values (11, 'hema', 'male');*** |
| ***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*** |
|  | ***create or replace 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*** |
|  | ***USE information\_schema;***  ***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\_schema = 'php' AND table\_name = 'students';*** |
| ***16*** | ***Create a nonunique index on the foreign key column (COURSE\_ID) in the students\_courses table.*** |
|  | ***USE php;***  ***CREATE INDEX my\_new\_index***  ***ON students\_courses(course\_id);*** |
| ***17*** | ***Create a user with your name and give him the privilege to access the grades database*** |
|  | ***CREATE USER 'edriso'@'localhost' IDENTIFIED BY 'asd123';***  ***GRANT ALL PRIVILEGES ON grades.\* TO 'edriso'@'localhost';*** |
| ***18*** | ***Connect to mysql using the user you created and try to insert one record in the courses table.*** |
|  | ***mysql -u edriso -p***  ***->123***  ***insert into courses values(11, 'mysql is awesome', 3);*** |
| ***19*** | ***Change your password.*** |
|  | ***ALTER USER 'edriso'@'localhost' IDENTIFIED BY '12345';*** |
| ***20*** | ***Show your privileges.*** |
|  | ***show grants;*** |