**MySQL Labs**

**MySQL (Day2):**

|  |  |
| --- | --- |
| ***1*** | ***Update students courses table, set the registration date value to “Today”;*** |
|  | UPDATE students\_coursed SET reg\_date = now(); |
| ***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(first\_name , ' ' , last\_name) as student\_name, grade, CASE  -> WHEN grade > 85 THEN 'Excellent'  -> WHEN grade > 75 THEN 'Very Good'  -> WHEN grade > 65 THEN 'Good'  -> WHEN grade > 55 THEN 'Pass'  -> ELSE 'Fail'  -> END AS grade\_desc  -> FROM students s, courses c, students\_courses sc  -> WHERE s.student\_id = sc.student\_id  -> AND c.course\_id = sc.course\_id; |
| ***4*** | ***Display the capitalized last name , and the grade , if he has no grade display the keyword absent. [using ifNULL function]*** |
|  | SELECT UCASE(last\_name), IFNULL(grade,'absent') as grade  -> FROM students s LEFT JOIN students\_courses sc  -> on s.student\_id = sc.student\_id; |
| ***5*** | ***Display students' names, course name along with their grades.*** |
|  | SELECT CONCAT(first\_name , ' ' , last\_name) as 'student\_name', course\_name, grade  -> FROM students s, courses c, students\_courses sc  -> WHERE s.student\_id = sc.student\_id  -> AND 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', MAX(sc.grade) as 'max', AVG(sc.grade) as 'Avg', COUNT(sc.student\_id) as 'studentsNum'  -> FROM students s, courses c, students\_courses sc  -> WHERE s.student\_id = sc.student\_id  -> AND c.course\_id = sc.course\_id  -> GROUP BY sc.course\_id; |
| ***7*** | ***Use subquery to display the names of the students who were born before student no 1.*** |
|  | SELECT CONCAT(first\_name, ' ', last\_name) as student\_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 ( 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  -> (student\_id, first\_name, last\_name, email, tel, gender, birth\_date)  -> VALUES (20, "Mahmoud", "Mohamed", "Mahmoud@gmail.com", "0122225222", "male", "2002-03-22"); |
| ***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 information\_schema.tables  -> WHERE TABLE\_NAME = 'students'; |
| ***16*** | ***Create a nonunique index on the foreign key column (COURSE\_ID) in the students\_courses table.*** |
|  | CREATE INDEX COURSE\_ID\_IDX  -> on students\_courses(course\_id); |
| ***17*** | ***Create a user with your name and give him the privilege to access the grades database*** |
|  | CREATE USER 'nada' IDENTIFIED BY 'NadaSaeed' ;  GRANT ALL ON grades.\* TO 'nada'; |
| ***18*** | ***Connect to mysql using the user you created and try to insert one record in the courses table.*** |
|  | INSERT INTO students  -> (`student-id`, `first-name`, `last-name`, email, tel, gender, `birth-date`)  -> VALUES (20, "Nourhan", "Saeed", "Nourhan@gmail.com", "0122225222", "female", "1995-02-22"); |
| ***19*** | ***Change your password.*** |
|  | ALTER USER nada IDENTIFIED BY "Nada98Sakr"; |
| ***20*** | ***Show your privileges.*** |
|  | SHOW GRANTS FOR nada; |