**MySQL Lab 2**

**( Ahmed Ashraf Ibrahim )**

**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,’%a,%e %M %Y ’) as RegDate**  **-> 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 Full\_Name ,**  **course\_name c , grade**  **, CASE**  **when grade >= 85 then "Excellent"**  **when grade < 85 and grade >= 75 then "Very good"**  **when grade < 75 and grade >= 65 then "Good"**  **when grade < 65 and grade >= 55 then "Pass"**  **Else "failed"**  **End as Remarks**  **FROM students s , courses c , students\_courses s\_c**  **WHERE s.student\_id = s\_c.student\_id**  **AND c.course\_id = s\_c.course\_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 s) left join (students\_courses s\_c)**  **on (s.student\_id = s\_c.student\_id)** |
| **5** | **Display students' names, course name along with their grades.** |
|  | **SELECT concat (first\_name ,' ', last\_name) as Full\_Name ,**  **course\_name c , grade**  **FROM students s , courses c , students\_courses s\_c**  **WHERE s.student\_id = s\_c.student\_id**  **AND c.course\_id = s\_c.course\_id ;** |
| **6** | **For each course, display the course name, min grade, max grade, average grade, number of attended students.** |
|  | **SELECT course\_name , min(grade) as lower , max(grade) as higher ,**  **avg(grade) ,**  **Count(s\_c.student\_id) as Num\_Of\_Students**  **From courses c , students\_courses s\_c**  **WHERE c.course\_id = s\_c.course\_id**  **Group By course\_name ;** |
| **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 credit\_hour = (Select credit\_hour from courses**  **WHERE course\_name like "MySQL")**  **AND course\_name != “** |
| **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" ;**  **-> Select \* from female\_students\_vu " ;** |
| **11** | **Try to insert a male student through your view** |
|  | **Insert into female\_students\_vu**  **Values(6, "ALI", "ALAA", "1111", "ALI@gm", "male", "2000-09-01") ;** |
| **12** | **Select all the data from your view and then from the students table** |
|  |  |
| **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 ;**  **Insert into female\_students\_vu**  **Values(7, "Ahmed", "Ashraf", "1234", "ahmed@gm", "male", "1997-09-20") ;** |
| **14** | **Use the information schema to display the table name , schema and the updatability of the female\_students\_vu view** |
|  | **SELECT table\_name , is\_updatable , table\_schema**  **FROM views**  **Where**  **table\_name = "female\_students\_vu" ;**  **OR**  **SELECT \* FROM INFORMATION\_SCHEMA.VIEWS**  **WHERE TABLE\_NAME = "female\_students\_vu"**  **AND TABLE\_SCHEMA = "php"\G ;** |
| **15** | **Use the information schema to display the create time, table\_rows, auto\_increment, and the comments on the students table.** |
|  | **SELECT TABLE\_ROWS,TABLE\_COMMENT,**  **CREATE\_TIME,AUTO\_INCREMENT**  **FROM INFORMATION\_SCHEMA.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.** |
|  | **Create index COURSE\_ID**  **ON steudents\_courses (course\_ID) ;**  **>> SHOW INDEX from students\_courses\G ;** |
| **17** | **Create a user with your name and give him the privilege to access the grades database** |
|  | **Create user ‘ ahmed09ashraf ’ @localhost IDENTIFIED BY ‘123’**  **GRANT ALL ON grades.\***  **-> TO "ahmed09ashraf" @"localhost"**  **-> WITH GRANT OPTION ;** |
| **18** | **Connect to mysql using the user you created and try to insert one record in the courses table.** |
|  | **mysql -u ahmed09ashraf -p**  **Enter password: \*\*\***  **INSERT INTO courses**  **VALUES (7 , "NodeJS" , "5") ;** |
| **19** | **Change your password.** |
|  | **SET PASSWORD For 'ahmed09ashraf' @'localhost' = password('111')** |
| **20** | **Show your privileges.** |
|  | **SHOW GRANTS FOR CURRENT\_USER() ;** |