MySQL Labs

MySQL (Day2):

1	***Update students courses table, set the registration date value to "Today";***

	UPDATE students_courses
	Set reg_date = "2022-01-04";
	
	Or
	
	UPDATE students_courses
	Set reg_date = CURRENT_DATE();
	
	`mysql> select * from students_courses`
	`    -> ;`
	`+------------+-----------+-------+------------+`
	`| student_id | course_id | grade | reg_date |`
	`+------------+-----------+-------+------------+`
	`| 1 | 1 | 80 | 2022-01-04 |`
	`| 1 | 2 | 90 | 2022-01-04 |`
	`| 1 | 3 | 100 | 2022-01-04 |`
	`| 2 | 2 | 99 | 2022-01-04 |`
	`| 2 | 3 | 80 | 2022-01-04 |`
	`| 3 | 4 | 70 | 2022-01-04 |`
	`+------------+-----------+-------+------------+`
	`6 rows in set (0.01 sec)`

2	***Display the registration date in the following format:*** ***Day, month/ year***

	Select dayofmonth(reg_date), month(reg_date), year(reg_date)
	From students_courses;
	
	`mysql> Select dayofmonth(reg_date), month(reg_date), year(reg_date)`
	`    -> From students_courses;`
	`+----------------------+-----------------+----------------+`
	`| dayofmonth(reg_date) | month(reg_date) | year(reg_date) |`
	`+----------------------+-----------------+----------------+`
	`| 4 | 1 | 2022 |`
	`| 4 | 1 | 2022 |`
	`| 4 | 1 | 2022 |`
	`| 4 | 1 | 2022 |`
	`| 4 | 1 | 2022 |`
	`| 4 | 1 | 2022 |`
	`+----------------------+-----------------+----------------+`
	`6 rows in set (0.01 sec)`

3	***Display the <u>full name (first, last)</u> of the student with <u>his grade</u>.*** ***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(s.first_name," ",s.last_name)as student_name, **c.course_name, e.grade, case** **when e.grade >= 85 then "Excellent"** **when e.grade < 85 and e.grade >= 75 then "Very good"** **when e.grade < 75 and e.grade >= 65 then "*Good*"** **when e.grade < 65 and e.grade >= 55 then "Pass"** **Else "*failed*"** **End as grade_** **From courses c, students_courses e, students s** **Where e.course_id = c.course_id** **AND e.student_id = s.student_id;**
4	***Display the <u>capitalized last name</u> , and the <u>grade</u> , if he has no grade display the keyword <u>absent</u>. [using ifNULL function]***
	Select ucase(s.last_name), c.course_name, IFNULL(e.grade, "Absent") **From (courses c, students s) left join (students_courses e)** **on e.course_id = c.course_id** **AND e.student_id = s.student_id** **Order by s.last_name;**

```
mysql> Select concat(s.first_name," ",s.last_name)as student_name, c.course_name,  e.grade, case
    -> when e.grade >= 85 then "Excellent"
    -> when e.grade < 85 and e.grade >= 75 then "Very good"
    -> when e.grade < 75 and e.grade >= 65 then "Good"
    -> when e.grade < 65 and e.grade >= 55 then "Pass"
    -> Else "failed"
    -> End as grade_
    -> From courses c, students_courses e, students s
    -> Where e.course_id = c.course_id
    -> AND e.student_id = s.student_id;
+---------------+-------------+-------+-----------+
| student_name  | course_name | grade | grade_    |
+---------------+-------------+-------+-----------+
| Ahmed Aly     | Database    |    80 | Very good |
| Ahmed Aly     | C           |    90 | Excellent |
| Ahmed Aly     | Network     |   100 | Excellent |
| Ahmed Ibrahim | C           |    99 | Excellent |
| Ahmed Ibrahim | Network     |    80 | Very good |
| Ahmed  Ossama | OS          |    70 | Good      |
+---------------+-------------+-------+-----------+
6 rows in set (0.00 sec)
```

```
mysql> Select ucase(s.last_name), c.course_name, IFNULL(e.grade, "Absent")
    -> From (courses c, students s) left join (students_courses e)
    -> on e.course_id = c.course_id
    -> AND e.student_id = s.student_id
    -> Order by s.last_name
    -> ;
+--------------------+-------------+---------------------------+
| ucase(s.last_name) | course_name | IFNULL(e.grade, "Absent") |
+--------------------+-------------+---------------------------+
| ALY                | Database    | 80                        |
| ALY                | C           | 90                        |
| ALY                | Network     | 100                       |
| ALY                | OS          | Absent                    |
| ALY                | MySQL       | Absent                    |
| ALY                | Java        | Absent                    |
| IBRAHIM            | Database    | Absent                    |
| IBRAHIM            | C           | 99                        |
| IBRAHIM            | Network     | 80                        |
| IBRAHIM            | OS          | Absent                    |
| IBRAHIM            | MySQL       | Absent                    |
| IBRAHIM            | Java        | Absent                    |
| KHALED             | Database    | Absent                    |
| KHALED             | C           | Absent                    |
| KHALED             | Network     | Absent                    |
| KHALED             | OS          | Absent                    |
| KHALED             | MySQL       | Absent                    |
| KHALED             | Java        | Absent                    |
| KHALIL             | Database    | Absent                    |
| KHALIL             | C           | Absent                    |
| KHALIL             | Network     | Absent                    |
| KHALIL             | OS          | Absent                    |
| KHALIL             | MySQL       | Absent                    |
| KHALIL             | Java        | Absent                    |
| OSSAMA             | Database    | Absent                    |
| OSSAMA             | C           | Absent                    |
| OSSAMA             | Network     | Absent                    |
| OSSAMA             | OS          | 70                        |
| OSSAMA             | MySQL       | Absent                    |
| OSSAMA             | Java        | Absent                    |
+--------------------+-------------+---------------------------+
```

5 ***Display <u>students' names</u>, <u>course name</u> along with their grades.***

```
Select concat(s.first_name," ",s.last_name)as student_name,
c.course_name, e.grade
From courses c, students_courses e, students s
Where e.course_id = c.course_id
AND e.student_id = s.student_id;
```

```
mysql> Select concat(s.first_name," ",s.last_name)as student_name, c.course_name, e.grade
    -> From courses c, students_courses e, students s
    -> Where e.course_id = c.course_id
    -> AND e.student_id = s.student_id;
+---------------+-------------+-------+
| student_name  | course_name | grade |
+---------------+-------------+-------+
| Ahmed Aly     | Database    |    80 |
| Ahmed Aly     | C           |    90 |
| Ahmed Aly     | Network     |   100 |
| Ahmed Ibrahim | C           |    99 |
| Ahmed Ibrahim | Network     |    80 |
| Ahmed  Ossama | OS          |    70 |
+---------------+-------------+-------+
```

6 ***For each course, display the course name, min grade, max grade, average grade, number of attended students.***

Select c.course_name, min(e.grade), max(e.grade), avg(e.grade), count(e.student_id)
From courses c, students_courses e
Where e.course_id = c.course_id

Group by c.course_name;

```
mysql> Select c.course_name, min(e.grade), max(e.grade), avg(e.grade), count(e.student_id)
    -> From courses c, students_courses e
    -> Where e.course_id = c.course_id
    -> Group by c.course_name;
+-------------+--------------+--------------+--------------+---------------------+
| course_name | min(e.grade) | max(e.grade) | avg(e.grade) | count(e.student_id) |
+-------------+--------------+--------------+--------------+---------------------+
| Database    |           80 |           80 |      80.0000 |                   1 |
| C           |           90 |           99 |      94.5000 |                   2 |
| Network     |           80 |          100 |      90.0000 |                   2 |
| OS          |           70 |           70 |      70.0000 |                   1 |
+-------------+--------------+--------------+--------------+---------------------+
4 rows in set (0.01 sec)
```

7 ***Use subquery to display the names of the students who were born before student no 1.***

Select concat(s.first_name," ",s.last_name)as student_name
From students s
Where birth_date < (select birth_date From students where student_id = 1);

```
mysql> Select concat(s.first_name," ",s.last_name)as student_name
    -> From students s
    -> Where birth_date < (select birth_date From students where student_id = 1);
+---------------+
| student_name  |
+---------------+
| Ahmed Ibrahim |
| Hoda Khaled   |
+---------------+
```

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"));

```
mysql> Select *
    -> From courses
    -> Where credit_hour < (select credit_hour from courses where course_name like ("MySQL"));
+-----------+-------------+-------------+
| course_id | course_name | credit_hour |
+-----------+-------------+-------------+
|         3 | Network     |           1 |
|         4 | OS          |           1 |
+-----------+-------------+-------------+
2 rows in set (0.01 sec)
```

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";**

```
mysql> CREATE VIEW female_students_vu
    -> AS
    -> SELECT * FROM students where gender = "female";
Query OK, 0 rows affected (0.13 sec)

mysql> select * from female_students_vu;
+------------+------------+-----------+------+-------+--------+------------+
| student_id | first_name | last_name | tel  | email | gender | birth_date |
+------------+------------+-----------+------+-------+--------+------------+
|          4 | Hoda       | Khaled    | NULL | NULL  | female | 1991-09-01 |
|          5 | Mona       | Khalil    | NULL | NULL  | female | 1992-10-01 |
+------------+------------+-----------+------+-------+--------+------------+
2 rows in set (0.02 sec)
```

11	***Try to <u>insert a male student</u> through your view***
	Insert into female_students_vu ***Values(6, "Mahmoud", "Kamal", NULL, NULL, "male", "1991-09-01");***
12	***Select all the data from your view and then from the students table***

```
mysql> select * from female_students_vu;
+------------+------------+-----------+------+-------+--------+------------+
| student_id | first_name | last_name | tel  | email | gender | birth_date |
+------------+------------+-----------+------+-------+--------+------------+
|          4 | Hoda       | Khaled    | NULL | NULL  | female | 1991-09-01 |
|          5 | Mona       | Khalil    | NULL | NULL  | female | 1992-10-01 |
+------------+------------+-----------+------+-------+--------+------------+
2 rows in set (0.01 sec)

mysql> select * from students;
+------------+------------+-----------+------+-------+--------+------------+
| student_id | first_name | last_name | tel  | email | gender | birth_date |
+------------+------------+-----------+------+-------+--------+------------+
|          1 | Ahmed      | Aly       | NULL | NULL  | male   | 1991-10-01 |
|          2 | Ahmed      | Ibrahim   | NULL | NULL  | male   | 1991-09-01 |
|          3 | Ahmed      | Ossama    | NULL | NULL  | male   | 1992-10-01 |
|          4 | Hoda       | Khaled    | NULL | NULL  | female | 1991-09-01 |
|          5 | Mona       | Khalil    | NULL | NULL  | female | 1992-10-01 |
|          6 | Mahmoud    | Kamal     | NULL | NULL  | male   | 1991-09-01 |
+------------+------------+-----------+------+-------+--------+------------+
6 rows in set (0.00 sec)
```

13 ***Prevent the ability to insert another male student through you view***

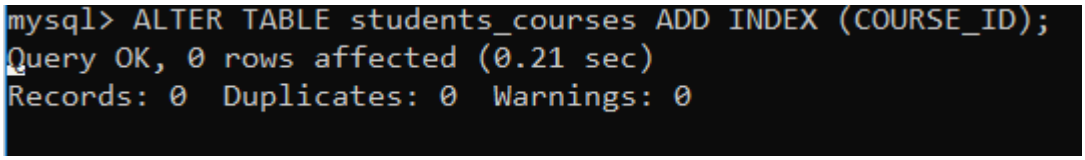
alter view *female_students_vu*
AS
*SELECT * FROM students where gender = “female”*
WITH CHECK OPTION;
.
.
.
Insert into female_students_vu
Values(7, “Mahmoud”, “Kamal”, NULL, NULL, “male”, “1991-09-01”);

```
mysql> alter view female_students_vu
    -> AS
    -> SELECT * FROM students where gender = "female"
    -> WITH CHECK OPTION;
Query OK, 0 rows affected (0.04 sec)

mysql> Insert into female_students_vu
    -> Values(7, "Mahmoud", "Kamal", NULL, NULL, "male", "1991-09-01");
ERROR 1369 (HY000): CHECK OPTION failed 'os42.female_students_vu'
```

14 ***Use the information schema to display the table name , schema and the updatability of the female_students_vu view***

USE INFORMATION_SCHEMA;
*SELECT * FROM INFORMATION_SCHEMA.VIEWS*
WHERE TABLE_NAME = “female_students_vu”
AND TABLE_SCHEMA = “OS42”\G

	Or ***SELECT TABLE_NAME,TABLE_SCHEMA,VIEW_DEFINITION FROM INFORMATION_SCHEMA.VIEWS WHERE TABLE_SCHEMA = 'OS42' AND 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 TABLE_ROWS,TABLE_COMMENT,CREATE_TIME,AUTO_INCREMENT FROM INFORMATION_SCHEMA.Tables WHERE TABLE_SCHEMA = 'OS42' AND TABLE_NAME = 'students';
16	***Create a nonunique index on the foreign key column (COURSE_ID) in the students_courses table.***
	ALTER TABLE students_courses ADD INDEX (COURSE_ID);  ***SHOW INDEX from students_courses;***
17	***Create a user with your name and give him the privilege to access the grades database***
	CREATE USER "open_source"@"localhost" ***IDENTIFIED BY "os123";*** ***GRANT ALL PRIVILEGES ON *.*** ***TO "open_source"@"localhost"*** ***WITH GRANT OPTION;***

```
C:\Program Files\MySQL\MySQL Server 8.0\bin>mysql -u open_source -p
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 31
Server version: 8.0.19 MySQL Community Server - GPL

Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| grades             |
| information_schema |
| mysql              |
```

18 ***Connect to mysql using the user you created and try to insert one record in the courses table.***

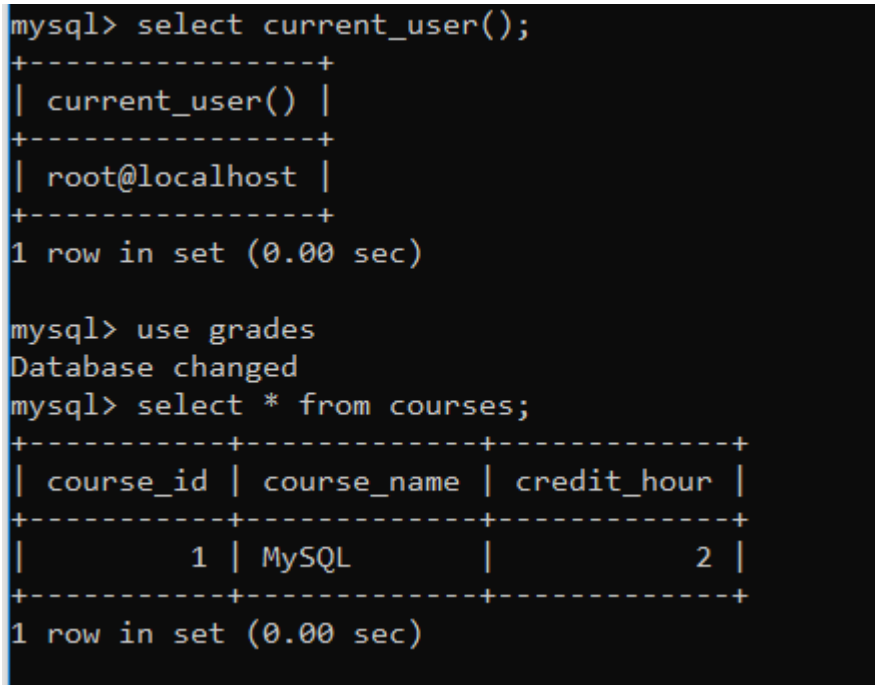
"open_source"@"localhost"

```
mysql> use grades
Database changed
mysql> select * from courses;
Empty set (0.03 sec)

mysql> insert into courses values(1,"MySQL",2);
Query OK, 1 row affected (0.01 sec)

mysql> select * from courses;
+-----------+-------------+-------------+
| course_id | course_name | credit_hour |
+-----------+-------------+-------------+
|         1 | MySQL       |           2 |
+-----------+-------------+-------------+
1 row in set (0.00 sec)

mysql> CURRENT_USER();
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to y
 for the right syntax to use near 'CURRENT_USER()' at line 1
mysql>  CURRENT_USER();
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to y
 for the right syntax to use near 'CURRENT_USER()' at line 1
mysql> select current_user();
+-----------------------+
| current_user()        |
+-----------------------+
| open_source@localhost |
+-----------------------+
```

	"root"@"localhost"
	
19	***Change your password.***
	SET PASSWORD FOR "open_source"@"localhost" = "iti";
20	***Show your privileges.***
	SHOW GRANTS FOR CURRENT_USER(); ***SHOW GRANTS FOR "open_source"@"localhost";***