Name: Dhaval Shah

Sap id.: 40778230008

Roll No.: L006

MSc Data Science and AI

Practical 2: Subquery-join operations on Relational Schema

Enter password: \*\*\*\*

Welcome to the MySQL monitor. Commands end with ; or \g.

Your MySQL connection id is 10

Server version: 8.0.34 MySQL Community Server - GPL

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

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> use mscds;

Database changed

1) USING (practical 1)

Q1. Count the customers with grades above New York average.

mysql> SELECT grade, COUNT(\*) FROM customer GROUP BY grade HAVING grade > (SELECT

-> AVG(grade) FROM customer WHERE city = 'New York');

+-------+----------+

| grade | COUNT(\*) |

+-------+----------+

| 200 | 3 |

| 300 | 2 |

+-------+----------+

2 rows in set (0.20 sec)

Q2. Find the name and numbers of all salesmen who had more than one customer

mysql> select salesman\_id,name from salesmanes a where 1<(select count(\*) from customer where salesman\_id=a.salesman\_id);

+-------------+------------+

| salesman\_id | name |

+-------------+------------+

| 5001 | James Hooq |

| 5002 | Nail Knite |

| 5003 | Lauson Hen |

| 5005 | Pit Alex |

| 5006 | Mc Lyon |

| 5007 | Paul Adam |

+-------------+------------+

6 rows in set (0.07 sec)

mysql> select salesman\_id,name from salesmanes a where 1<(select count(\*) from customer where salesman\_id=a.salesman\_id);

+-------------+------------+

| salesman\_id | name |

+-------------+------------+

| 5001 | James Hooq |

| 5002 | Nail Knite |

| 5003 | Lauson Hen |

| 5005 | Pit Alex |

| 5006 | Mc Lyon |

| 5007 | Paul Adam |

+-------------+------------+

6 rows in set (0.06 sec)

Q3))Demonstrate the DELETE operation by removing salesman with id 1000. All his orders

must also be deleted

mysql> delete from salesmanes where salesman\_id=1000;

Query OK, 0 rows affected (0.06 sec)

Q2. Design ERD for the following schema and execute the following Queries on it:

Consider the schema for Movie Database:

ACTOR (Act\_id, Act\_Name, Act\_Gender)

DIRECTOR (Dir\_id, Dir\_Name, Dir\_Phone)

MOVIES (Mov\_id, Mov\_Title, Mov\_Year, Mov\_Lang, Dir\_id)

MOVIE\_CAST (Act\_id, Mov\_id, Role)

RATING (Mov\_id, Rev\_Stars)

mysql> create table Actor(act\_id integer primary key,act\_name varchar(100),act\_gender varchar(10));

Query OK, 0 rows affected (0.14 sec)

mysql> create table Director(dir\_id integer primary key,dir\_name varchar(200),dir\_phone varchar(100));

Query OK, 0 rows affected (0.03 sec)

mysql> create table Movies(mov\_id integer primary key,mov\_title varchar(255),mov\_year year,mov\_lang varchar(100),dir\_id int, foreign key (dir\_id) references Director(dir\_id));

Query OK, 0 rows affected (0.09 sec)

mysql> create table Movie\_cast (act\_id int,foreign key (act\_id) references

-> Actor(act\_id), mov\_id int, foreign key(mov\_id) references Movies(mov\_id),role

-> varchar(100), primary key(act\_id,mov\_id) );

Query OK, 0 rows affected (0.08 sec)

mysql> create table Rating(mov\_id integer primary key , foreign key(mov\_id)

-> references Movies(mov\_id),rev\_stars integer);

Query OK, 0 rows affected (0.02 sec)

mysql> insert into Actor values(301, 'anuska','f'),(302,'PRABHAS','M'),(303,'PUNITH','M'),(304,'jermy','M');

Query OK, 4 rows affected (0.06 sec)

Records: 4 Duplicates: 0 Warnings: 0

mysql> insert into director values(60, 'rajamouli',8751611001), (61,'HITCHCOCK', 7766138911), (62,'FARAN', 9986776531), (63,'STEVEN SPIELBERG', 8989776530);

Query OK, 4 rows affected (0.06 sec)

Records: 4 Duplicates: 0 Warnings: 0

mysql> insert into movies values(1001,'BAHUBALI-2', 2017, 'TELAGU', 60), (1002,'BAHUBALI-2', 2015, 'TELAGU', 60),(1003,'AKASH', 2008, 'KANNADA', 61),(1004,'WAR HORSE', 2011, 'ENGLISH', 63);

Query OK, 4 rows affected (0.01 sec)

Records: 4 Duplicates: 0 Warnings: 0

mysql> INSERT INTO MOVIE\_CAST VALUES (301, 1002, 'HEROINE'),(301, 1001, 'HEROINE'),(303, 1003, 'HERO'),(303, 1002, 'guest'),(304,1004,'hero');

Query OK, 5 rows affected (0.06 sec)

Records: 5 Duplicates: 0 Warnings: 0

mysql> INSERT INTO RATING VALUES (1001, 4),(1002, 2),(1003, 5),(1004, 4);

Query OK, 4 rows affected (0.01 sec)

Records: 4 Duplicates: 0 Warnings: 0

Empty set (0.05 sec)

Q1). List the titles of all movies directed by ‘Hitchcock

mysql> select mov\_title from movies where dir\_id in(select dir\_id from director where dir\_name='hitchcock');

+-----------+

| mov\_title |

+-----------+

| AKASH |

+-----------+

1 row in set (0.00 sec)

Q2). Find the movie names where one or more actors acted in two or more movies.

mysql> select mov\_title from movies m,movie\_cast mv where m.mov\_id=mv.mov\_id and act\_id in(select act\_id from movie\_cast group by act\_id having count(act\_id)>1) group by mov\_title having count(\*)>1;

+------------+

| mov\_title |

+------------+

| BAHUBALI-2 |

+------------+

1 row in set (0.05 sec)

Q3). List all actors who acted in a movie before 2000 and also in a movie after

2015 (use JOIN operation)

mysql> select a.act\_name,c.mov\_title,c.mov\_year from actor a,movie\_cast b,movies c where a.act\_id=b.act\_id and b.mov\_id=c.mov\_id and c.mov\_year not between 2000 and 2015;

+----------+------------+----------+

| act\_name | mov\_title | mov\_year |

+----------+------------+----------+

| anuska | BAHUBALI-2 | 2017 |

+----------+------------+----------+

1 row in set (0.05 sec)

Q4). Find the title of movies and number of stars for each movie that has at least one

rating and find the highest number of stars that movie received. Sort the result by

movie title

mysql> select mov\_title,max(rev\_stars) from movies inner join rating using(mov\_id) group by mov\_title having max(rev\_stars)>0 order by mov\_title;

+------------+----------------+

| mov\_title | max(rev\_stars) |

+------------+----------------+

| AKASH | 5 |

| BAHUBALI-2 | 4 |

| WAR HORSE | 4 |

+------------+----------------+

3 rows in set (0.06 sec)

Q5). Update rating of all movies directed by ‘Steven Spielberg’ to 5

mysql> update rating set rev\_stars=5 where mov\_id in(select mov\_id from movies where dir\_id in (select dir\_id from director where dir\_name='STEVEN SPIELBERG'));

Query OK, 1 row affected (0.06 sec)

Rows matched: 1 Changed: 1 Warnings: 0

mysql> select \* from rating;

+--------+-----------+

| mov\_id | rev\_stars |

+--------+-----------+

| 1001 | 4 |

| 1002 | 2 |

| 1003 | 5 |

| 1004 | 5 |

+--------+-----------+

4 rows in set (0.00 sec)

3. Design ERD for the following schema and execute the following Queries on it:

Q1). Find the names of students who took only four-credit courses.

mysql> select name from students where stno IN (select stno from grades where cno IN (select cno from courses where cr=4));

+------------------+

| name |

+------------------+

| Edwards P. David |

| Mixon Leaths |

| Pierce Richard |

| Prior Lorraine |

| Rawlings Jerry |

| Lewis Jerry |

+------------------+

6 rows in set (0.07 sec)

Q2).Find the names of students who took no four-credit courses.

mysql> select distinct s.name from students s where s.stno NOT IN (select distinct g.stno from grades g join courses c on g.cno=c.cno where c.cr=4);

+----------------+

| name |

+----------------+

| Grogan A. Mary |

| McLane Sandy |

| Novak Roland |

+----------------+

3 rows in set (0.05 sec)

Q3). Find the names of students who took cs210 or cs310

mysql> select name from students where stno IN (select stno from grades where cno=210 or cno=310);

+------------------+

| name |

+------------------+

| Edwards P. David |

| Mixon Leaths |

| Pierce Richard |

| Prior Lorraine |

| Lewis Jerry |

+------------------+

5 rows in set (0.00 sec)

Q5). Find names of all students who have a cs210 grade higher than the highest grade

given in cs310 and did not take any course with Prof. Evans

mysql> select cno from grades group by cno having count(distinct stno)>=2;

+------+

| cno |

+------+

| 110 |

| 210 |

| 240 |

| 310 |

| 410 |

+------+

5 rows in set (0.05 sec)

mysql> select cno from grades group by cno having count(distinct stno)>=3;

+------+

| cno |

+------+

| 110 |

| 210 |

| 240 |

+------+

3 rows in set (0.00 sec)

Q6). Find the names of students who obtained the highest grade in cs210.

mysql> select name from students where stno in (select stno from grades where cno=210 and grade=(select max(grade) from grades where cno=210));

+------------------+

| name |

+------------------+

| Edwards P. David |

| Pierce Richard |

+------------------+

2 rows in set (0.00 sec)

Q7. Find course numbers for courses that enroll exactly two students;

mysql> select distinct cno from grades where empno in (select empno from instructors where ranks='Assist. Prof');

+------+

| cno |

+------+

| 410 |

| 310 |

+------+

2 rows in set (0.00 sec)

Q8.Find the lowest grade of a student who took a course during the spring of 2003.

mysql> select name from students where stno in (select stno from grades where cno=210 and grade=(select max(grade) from grades where cno=210));

+------------------+

| name |

+------------------+

| Edwards P. David |

| Pierce Richard |

+------------------+

2 rows in set (0.00 sec)

Q11).Find the names of students who have failed all their courses (failing is defined as a grade less than 60).

mysql> select name from students where stno in (select stno from grades where grade<60);

+------------------+

| name |

+------------------+

| Edwards P. David |

| Lewis Jerry |

+------------------+

2 rows in set (0.00 sec)

Q12).Find the highest grade of a student who never took cs110.

mysql> select max(grade) from grades where cno!=110;

+------------+

| max(grade) |

+------------+

| 100 |

+------------+

1 row in set (0.00 sec)

Q13)Find the names of students who do not have an advisor.

mysql> select name from students where stno not in (select stno from advising);

Empty set (0.00 sec)

Q14).Find names of courses taken by students who do not live in Massachusetts (MA).

mysql> select cno from grades where cno in (select cno from courses where stno in(select stno from students where state!='MA'));

+------+

| cno |

+------+

| 410 |

| 240 |

| 410 |

| 210 |

+------+

4 rows in set (0.00 sec)