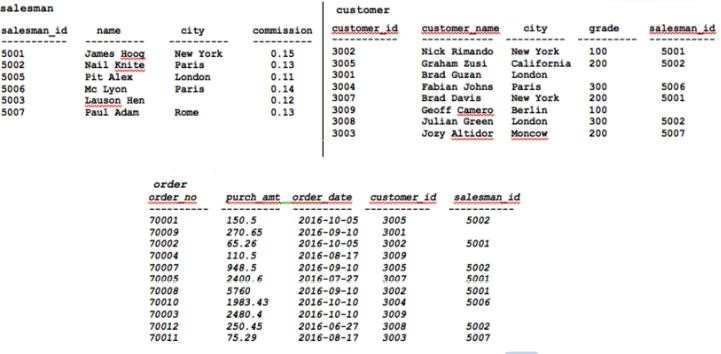
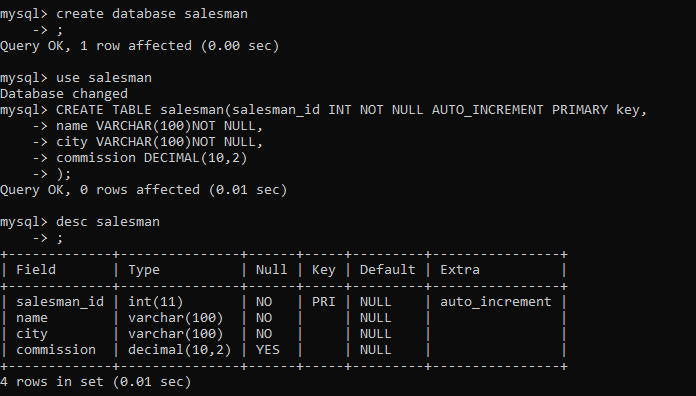
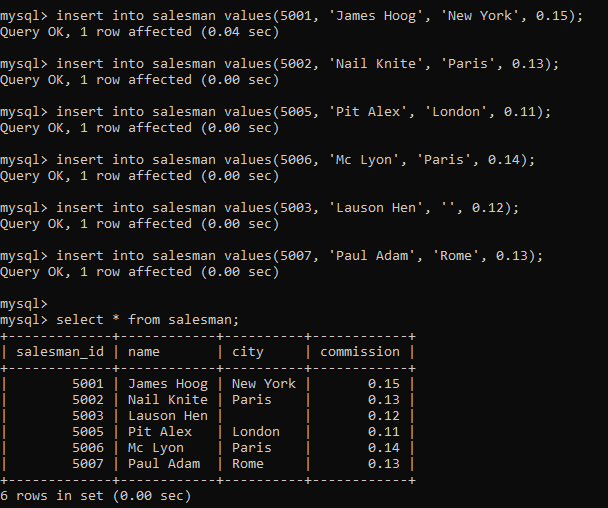
**PRACTICAL N0 – 1**

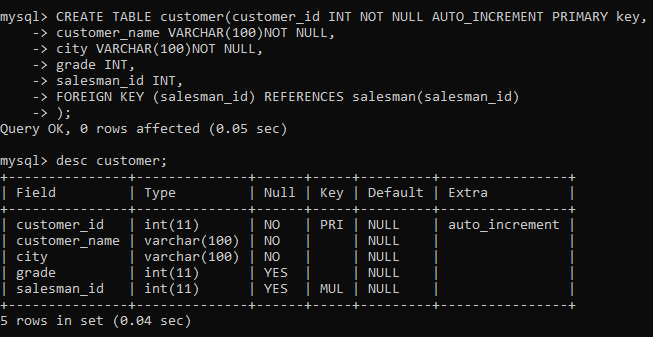
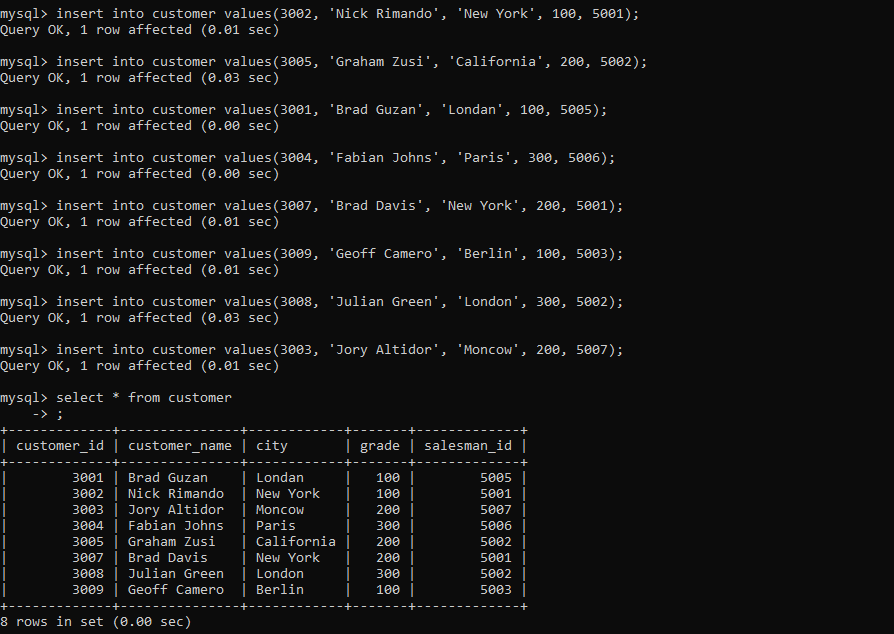
**Aim:** DDL operations on Relational Schema

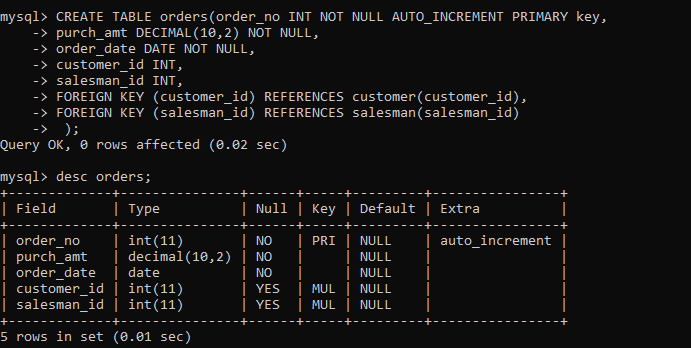
Design the following schema and execute the following queries on it:

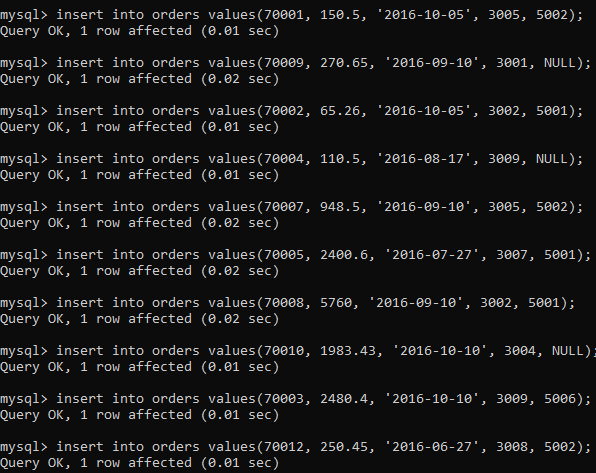


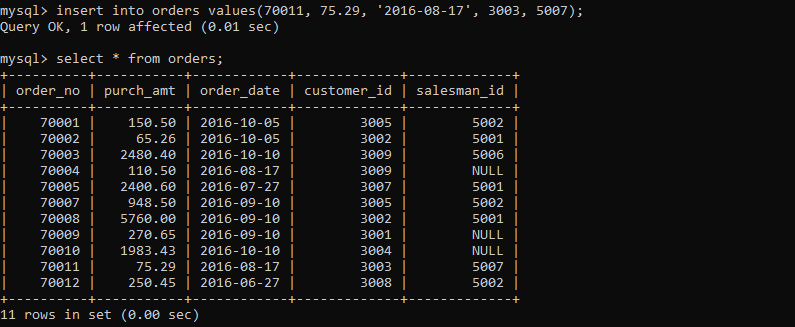
**Output:**

****

****

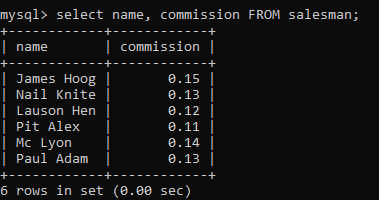
****

****

****

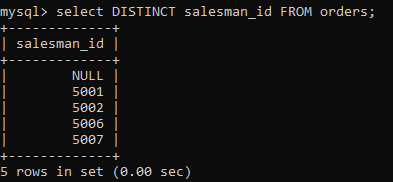
1. Display name and commission for all the salesmen.

**Output:**

****

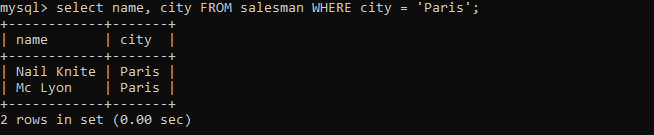
1. Retrieve salesman id of all salesmen from orders table without any repeats.

**Output:**

****

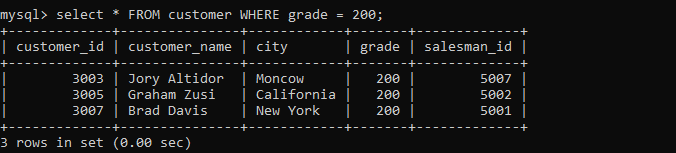
1. Display names and city of salesman, who belongs to the city of Paris.

**Output:**

****

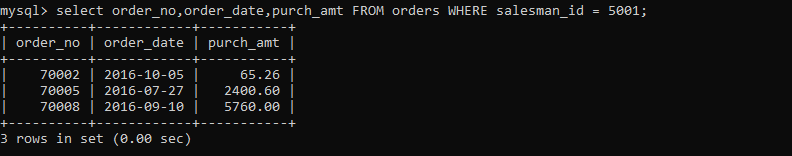
1. Display all the information for those customers with a grade of 200.

**Output:**



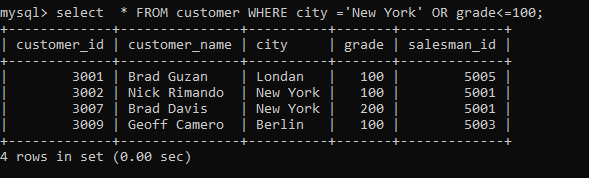
1. Display the order number, order date and the purchase amount for order(s) which will be delivered by the salesman with ID 5001

**Output:**



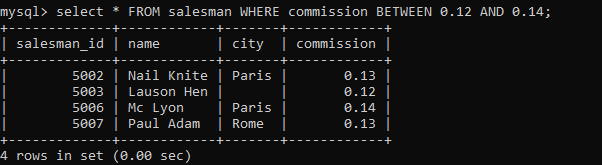
1. Display all the customers, who are either belongs to the city New York or not had a grade above 100.

**Output:**



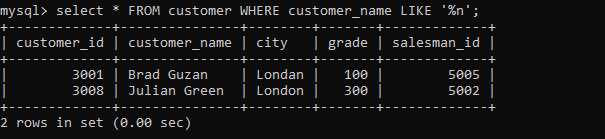
1. Find those salesmen with all information who gets the commission within a range of 0.12 and 0.14.

**Output:**



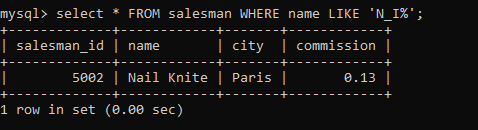
1. Find all those customers with all information whose names are ending with the letter 'n'.

**Output:**



1. Find those salesmen with all information whose name containing the 1st character is 'N' and the 4th character is 'l' and rests may be any character.

**Output:**



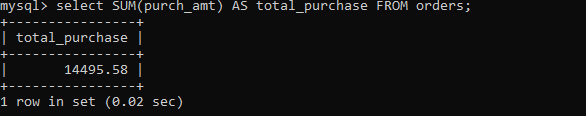
1. Find that customer with all information who does not get any grade except NULL.

**Output:**



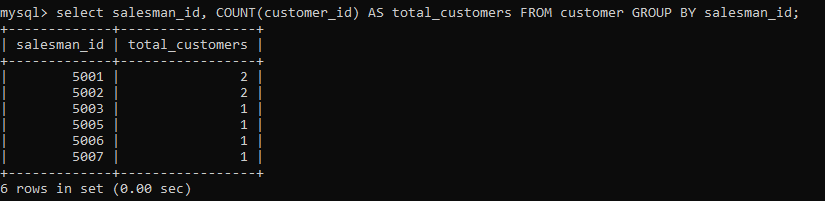
1. Find the total purchase amount of all orders.

**Output:**



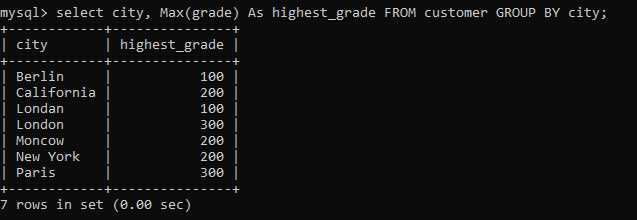
1. Find the number of salesman currently listing for all of their customers.

**Output:**



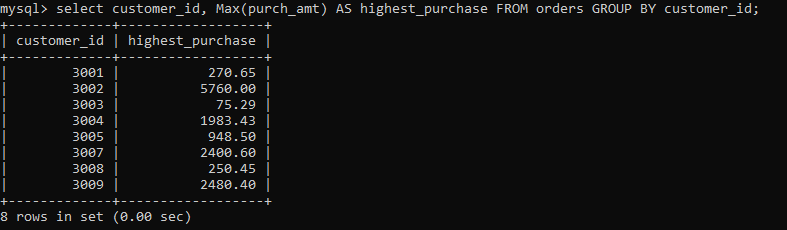
1. Find the highest grade for each of the cities of the customers.

**Output:**



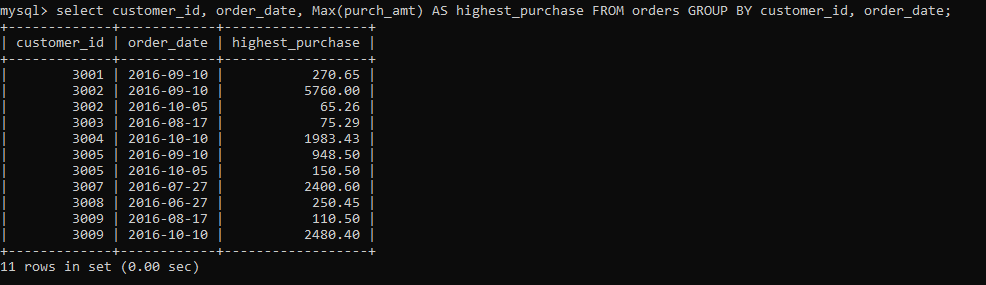
1. Find the highest purchase amount ordered by each customer with their ID and highest purchase amount.

**Output:**



1. Find the highest purchase amount ordered by each customer on a particular date with their ID, order date and highest purchase amount.

**Output:**



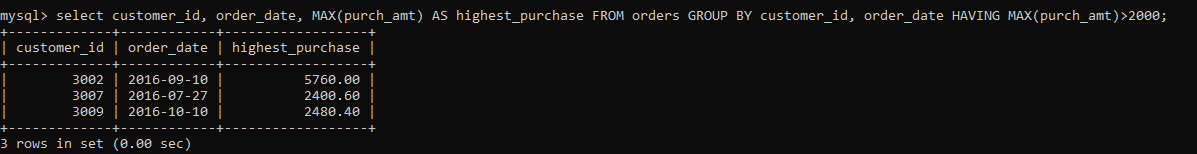
1. Find the highest purchase amount on a date '2012-08-17' for each salesman with their ID.

**Output:**



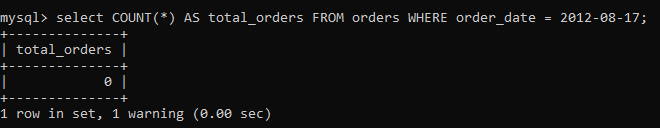
1. Find the highest purchase amount with their customer ID and order date, for only those customers who have the highest purchase amount in a day is more than 2000.

**Output:**



1. Write a SQL statement that counts all orders for a date August 17th, 2012.

**Output:**



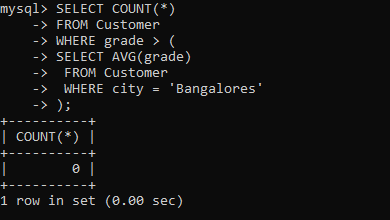
**PRACTICAL N0 – 2**

**Aim:** Subquery-join operations on Relational Schema

**USING (practical 1)**

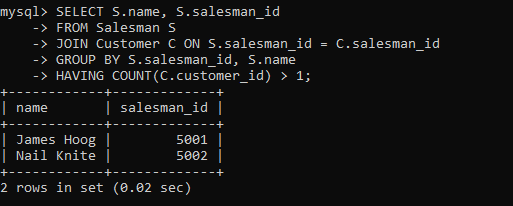
1. **Count the customers with grades above Bangalore’s average.**

**Output:**



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

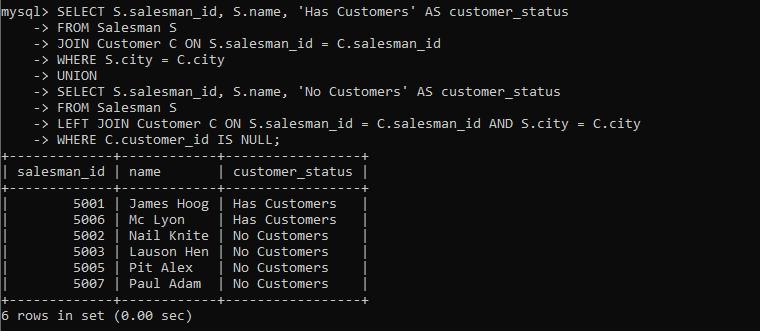
**Output:**



1. **List all salesmen and indicate those who have and don’t have**

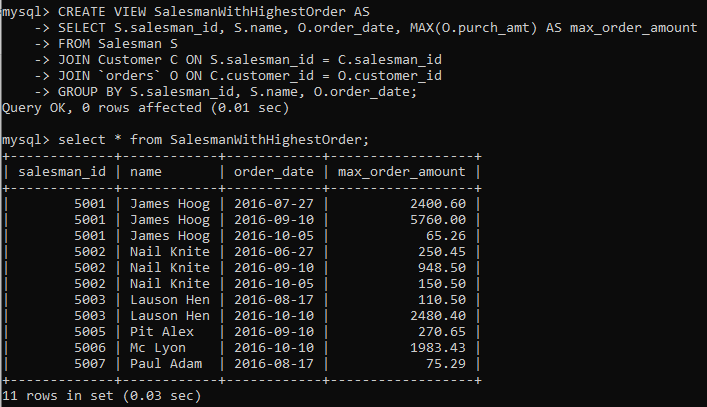
**customers in their cities (Use UNION operation.)**

**Output:**



1. **Create a view that finds the salesman who has the customer with the highest order of a day.**

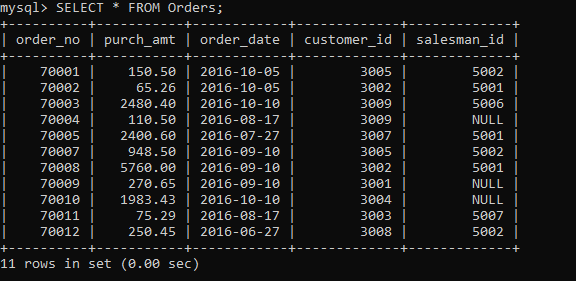
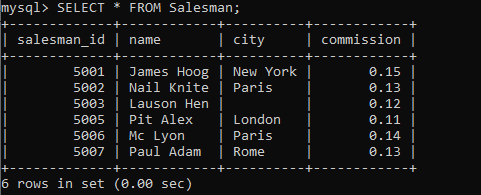
**Output:**



1. **Demonstrate the DELETE operation by removing salesman with id 1000. All his orders must also be deleted**

**Output:**



****

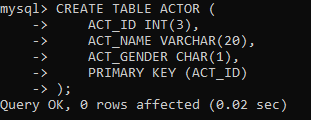
**2. Design ERD for the following schema and execute the following Queries on it:**

**Consider the schema for Movie Database: ACTOR (Act\_id, Act\_Name, Act\_Gender) DIRECTOR (Dir\_id, Dir\_Name, Dir\_Phone)**

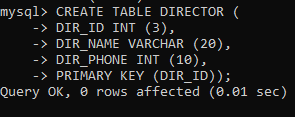
**MOVIES (Mov\_id, Mov\_Title, Mov\_Year, Mov\_Lang, Dir\_id) MOVIE\_CAST (Act\_id, Mov\_id, Role)**

**RATING (Mov\_id, Rev\_Stars)**

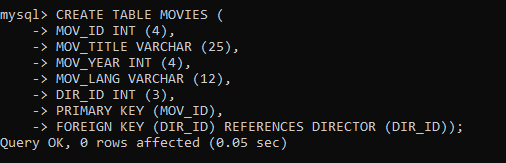
**Output:**



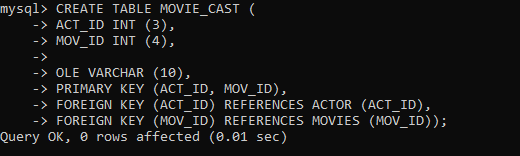
**Output:**



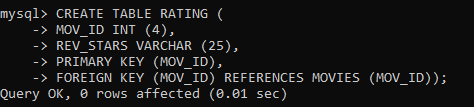
**Output:**



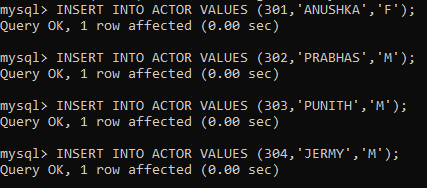
**Output:**



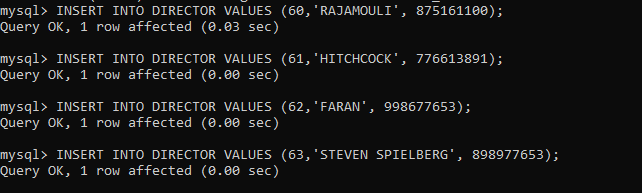
**Output:**



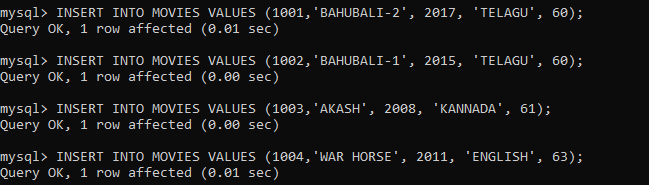
**Output:**



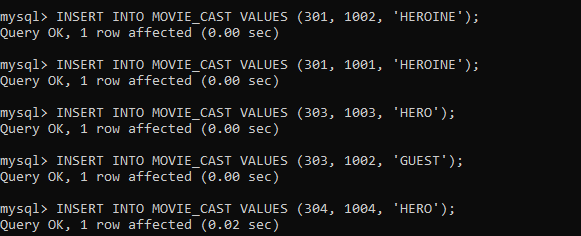
**Output:**



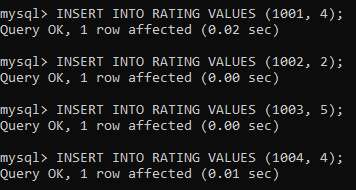
**Output:**



**Output:**



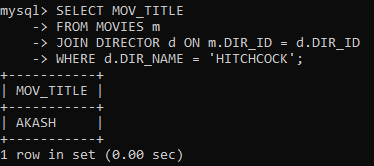
**Output:**



Write SQL queries to

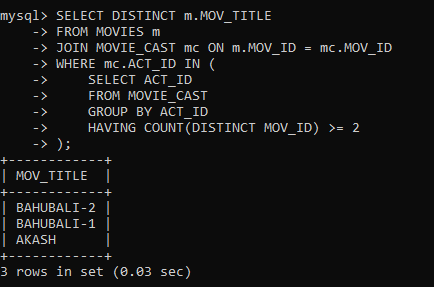
1. **List the titles of all movies directed by ‘Hitchcock’.**

**Output:**



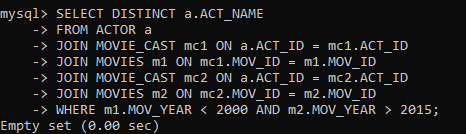
1. **Find the movie names where one or more actors acted in two or more movies.**

**Output:**



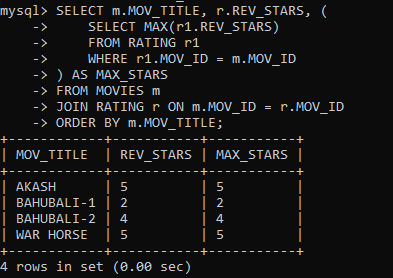
1. **List all actors who acted in a movie before 2000 and also in a movie after 2015 (use JOIN operation).**

**Output:**



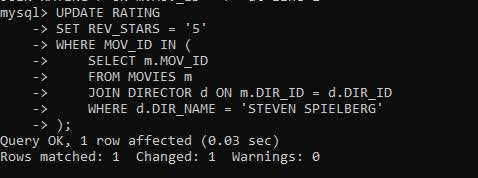
1. **Find the title of movies and number of stars for each movie that has at least one rating and find the highest number of stars that movie received. Sort the result by movie title.**

**Output:**



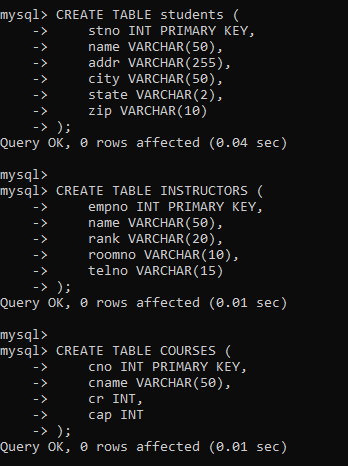
1. **Update rating of all movies directed by ‘Steven Spielberg’ to 5.**

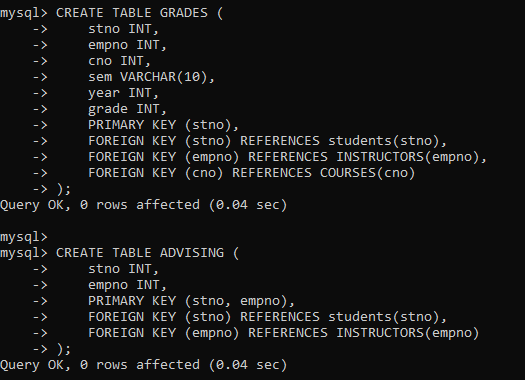
**Output:**



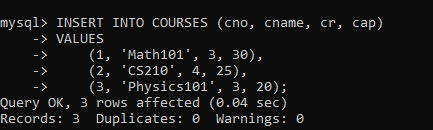
**3. Design ERD for the following schema and execute the following Queries on it:**

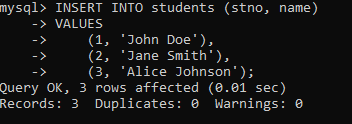
**Output:**

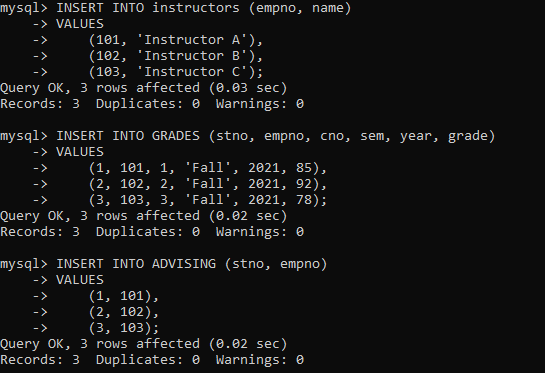


****

**Output:**



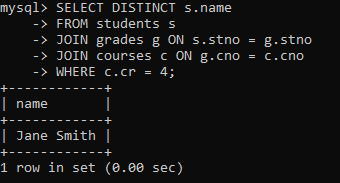
****



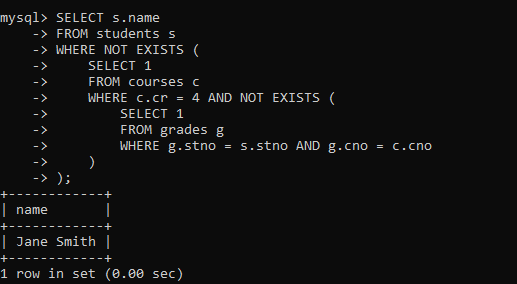
**For odd rollnumbers(any 10 )**

1. **Find the names of students who took some four-credit courses.**

**Output:**

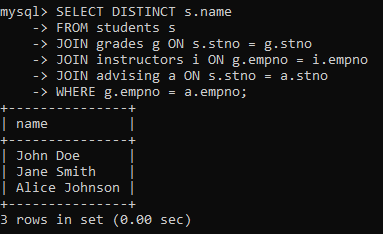


1. **Find the names of students who took every four-credit course. Output:**

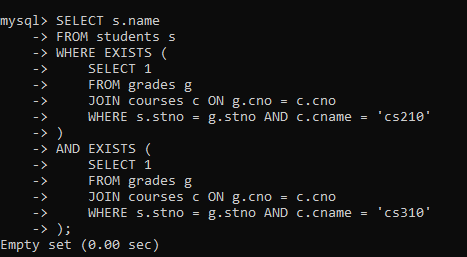


1. **Find the names of students who took a course with an instructor who is also their advisor.**

**Output:**

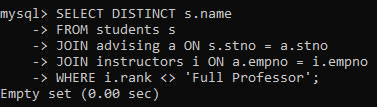


1. **Find the names of students who took cs210 and cs310. Output:**



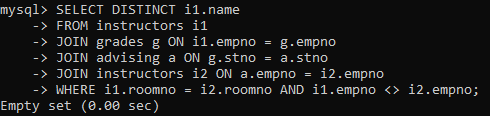
1. **Find the names of all students whose advisor is not a full professor.**

**Output:**

****

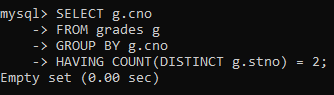
1. **Find instructors who taught students who are advised by another instructor who shares the same room.**

**Output:**



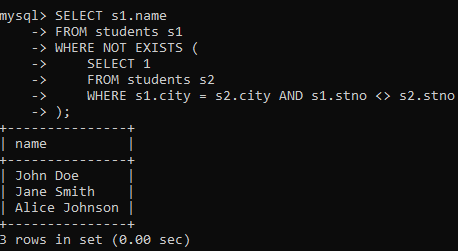
1. **Find course numbers for courses that enroll exactly two students**

**Output:**



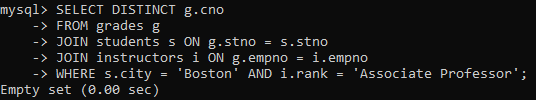
**ODD ROLL NUMBERS**

1. **Find the names of all students for whom no other student lives in the same city.**



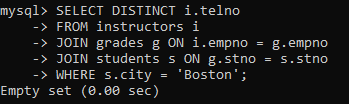
1. **Find course numbers of courses taken by students who live in Boston and which are taught by an associate professor.**

**Output:**

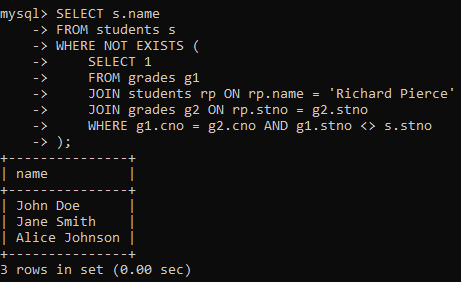


1. **Find the telephone numbers of instructors who teach a course taken by any student who lives in Boston.**

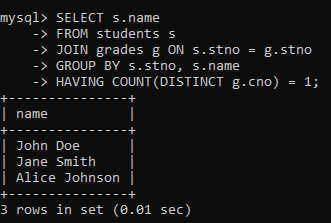
**Output:**



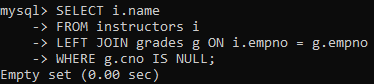
1. **Find names of students who took every course taken by Richard Pierce.**



1. **Find the names of students who took only one course. Output:**



1. **Find the names of instructors who teach no course. Output:**



1. **Find the names of the instructors who taught only one course during the spring semester of 2001.**

**Output:**

