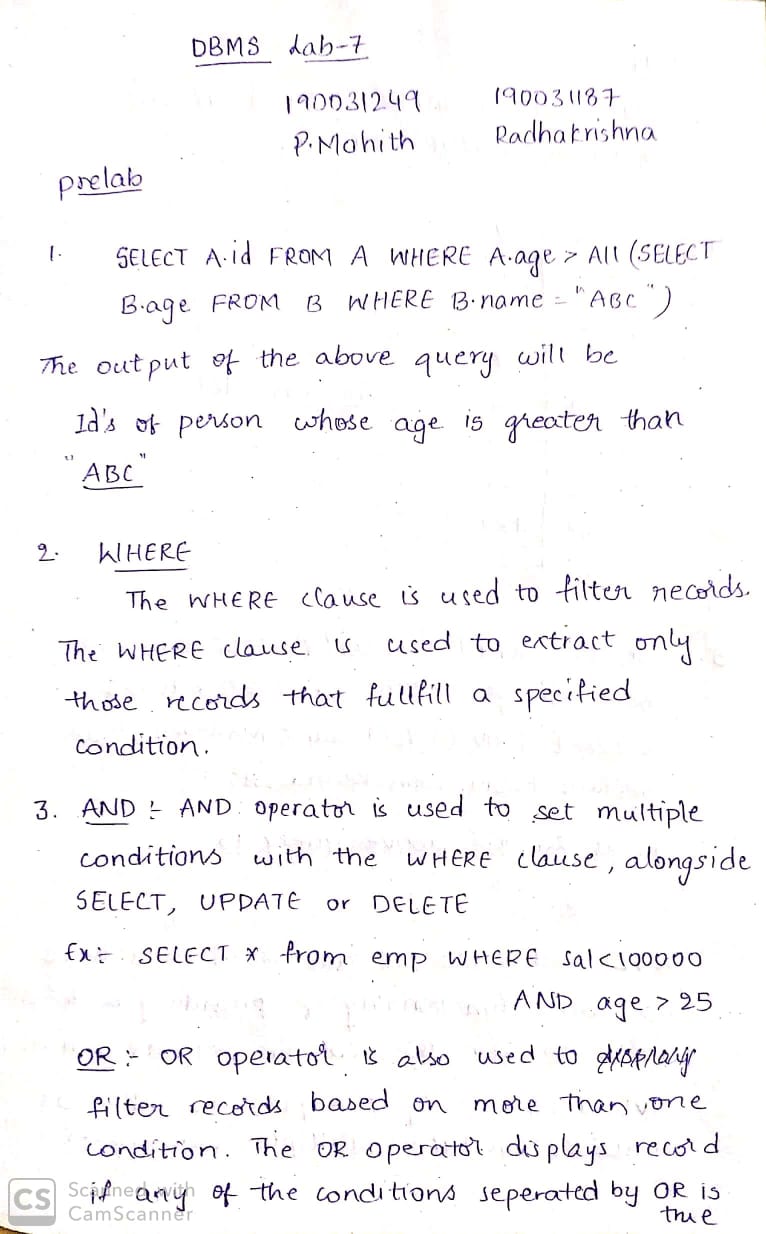
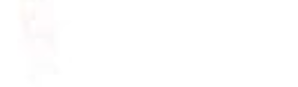
**DBMS PRACTICAL-7**

**PRE-LAB**

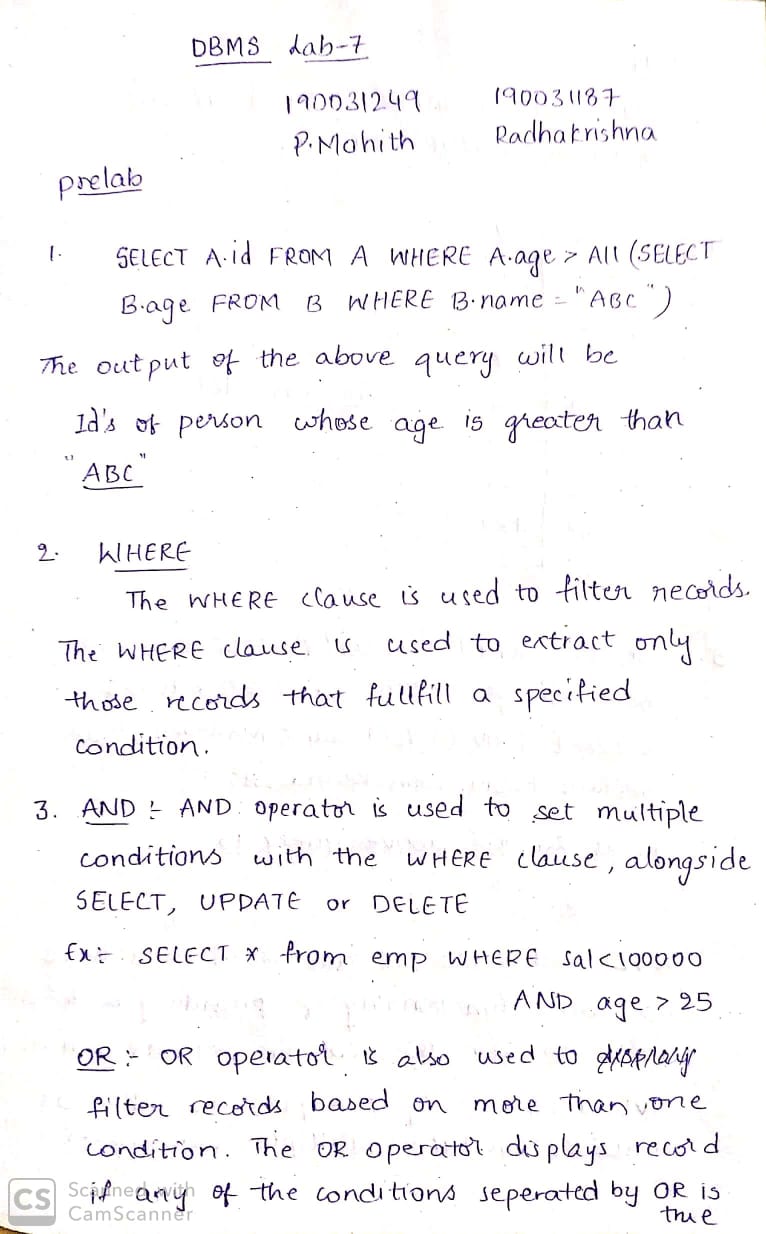
1. What is the output of following query

SELECT A.id FROM A WHERE A.age > ALL (SELECT B.age FROM B

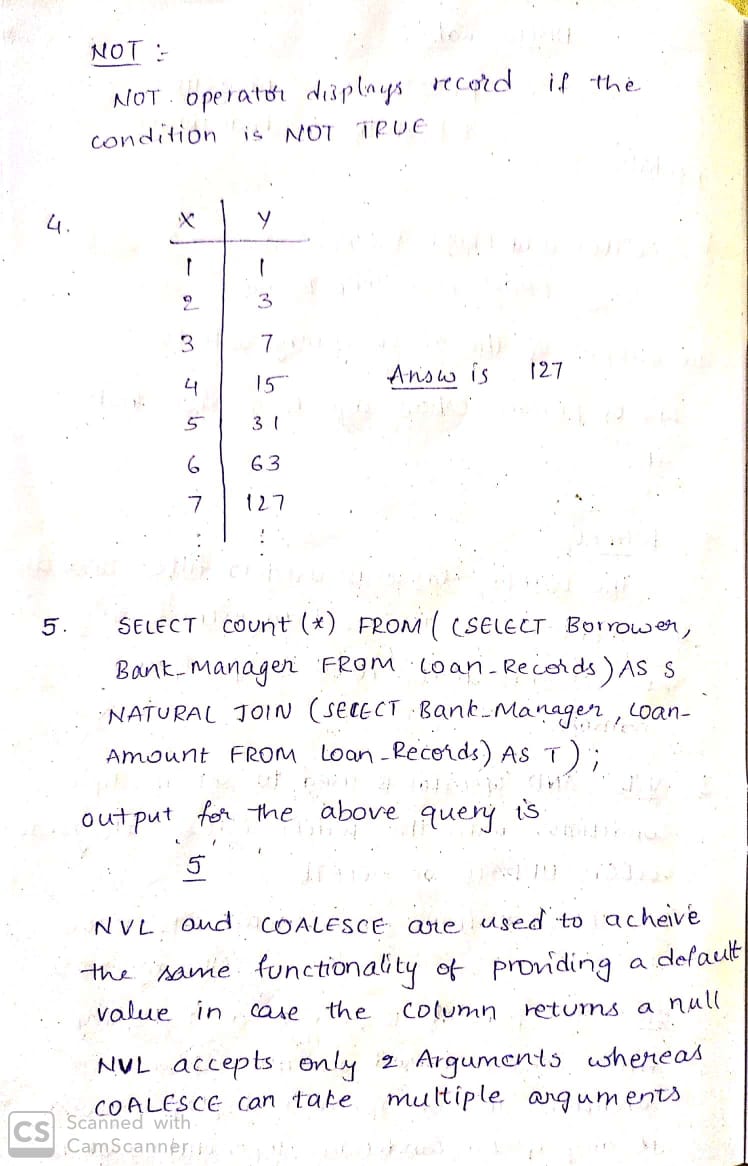
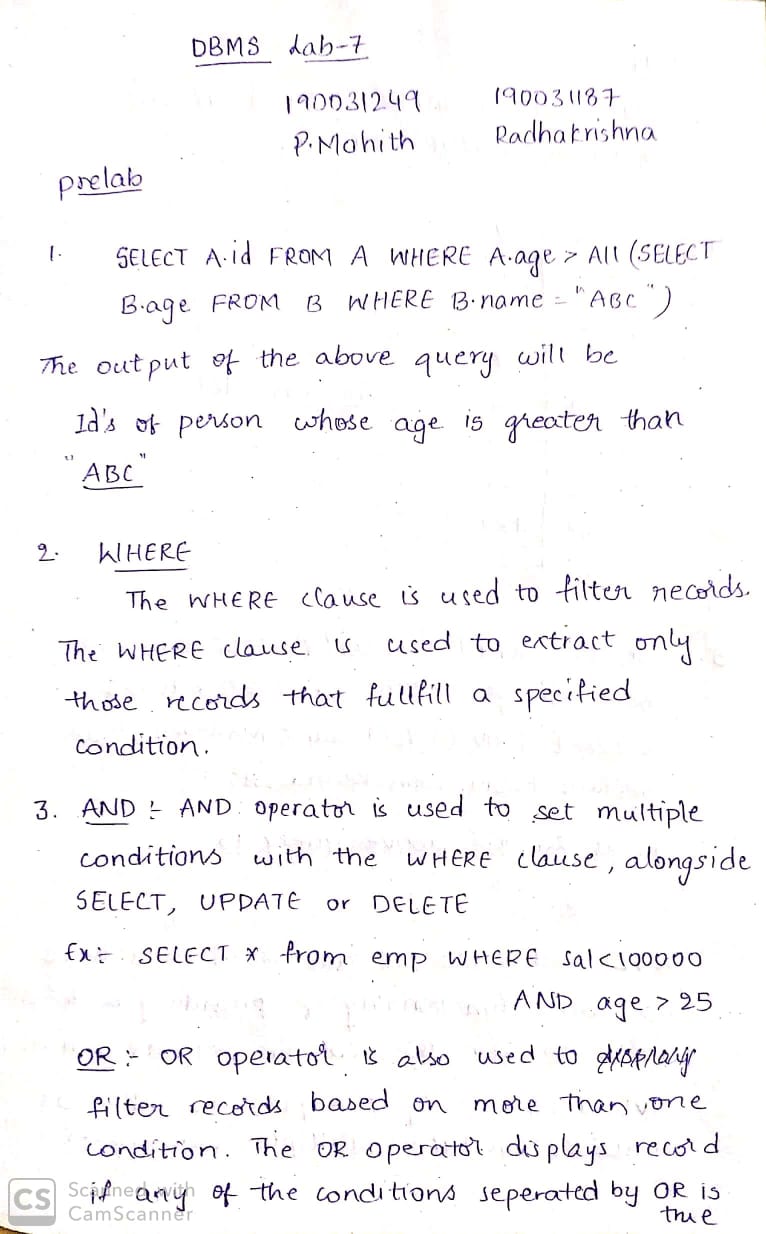
WHERE B. name = "ABC")

****

1. What is the purpose of WHERE Clause in mysql?

****

1. Differentiate between AND, OR and NOT operators in mysql

****

1. Database table by name Loan\_Records is given below.

**Borrower Bank\_Manager Loan\_Amount**

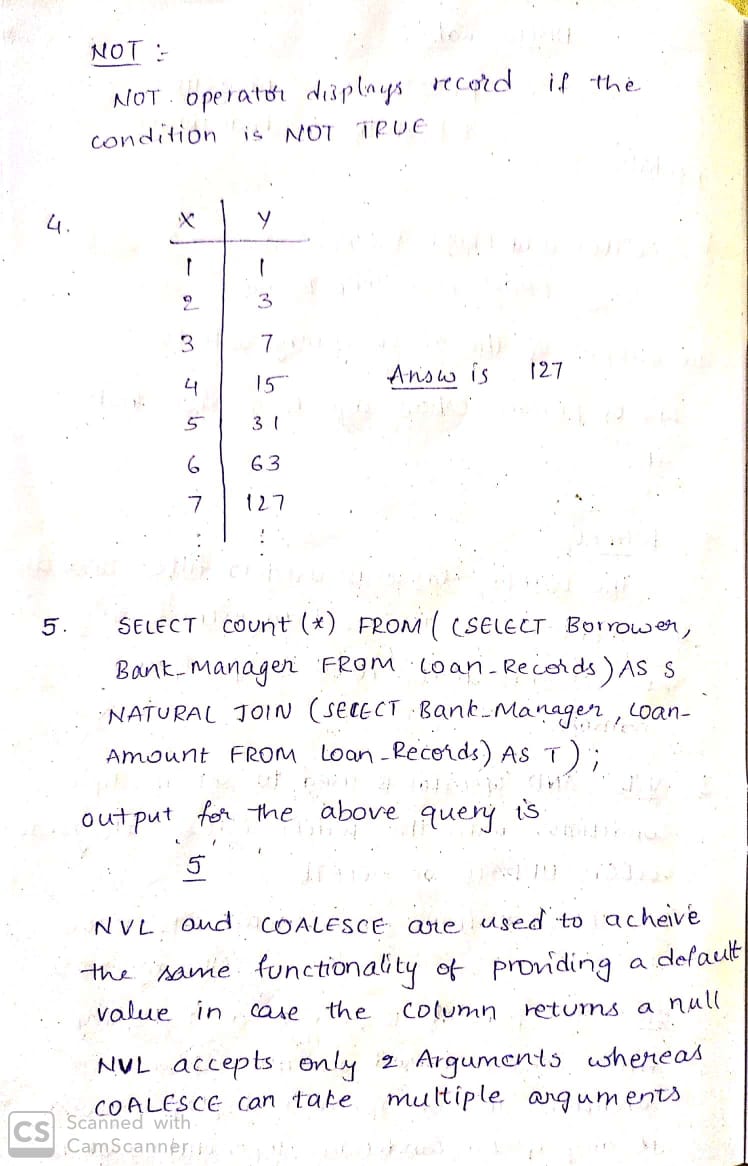
Ramesh Sunderajan 10000.00

Suresh Ramgopal 5000.00

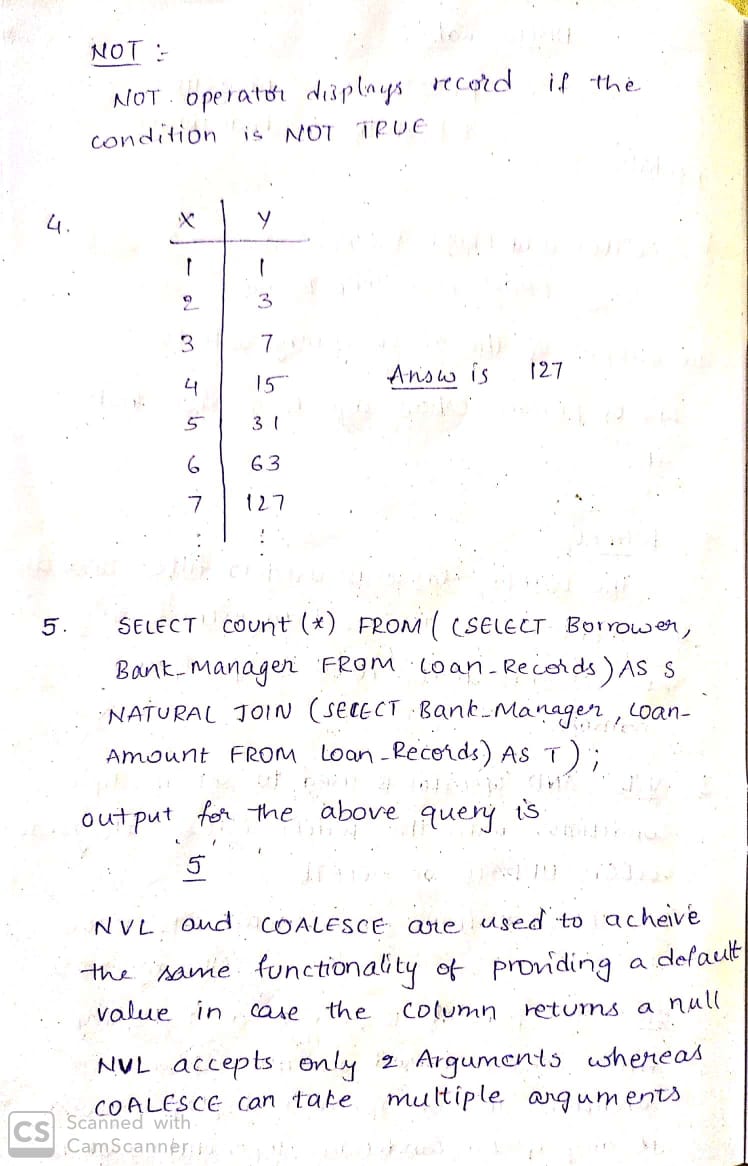
Mahesh Sunderajan 7000.00

SELECT Count(\*) FROM ( (SELECT Borrower, Bank\_Manager FROM Loan\_Records) AS S NATURAL JOIN (SELECT Bank\_Manager, Loan\_Amount FROM Loan\_Records) AS T );

What is the output of the following SQL query?

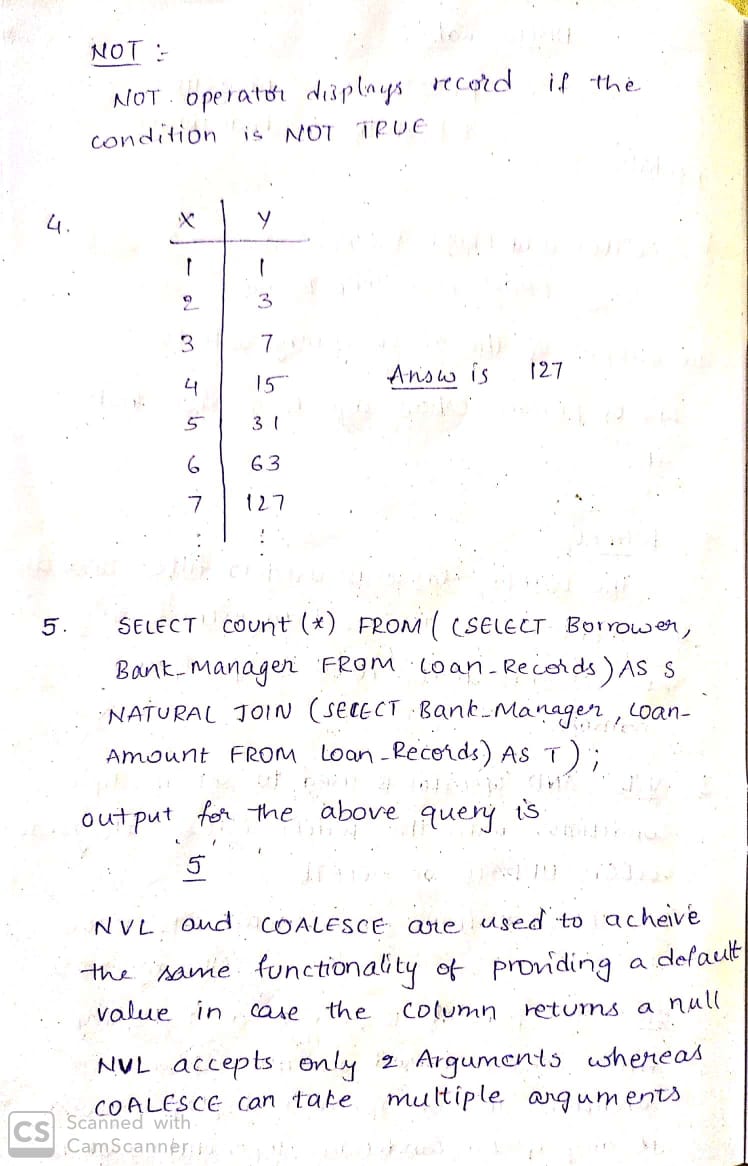
****

1. State the difference between (Null Value Function) nvl() & Coalesce().

****

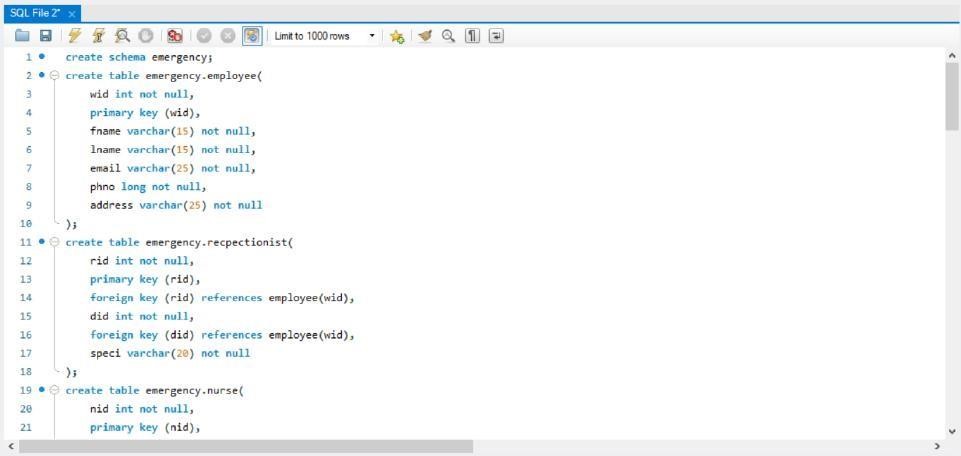
1. Consider a database table T containing two columns X and Y each of type integer. After the creation of the table, one record (X=1, Y=1) is inserted in the table. Let MX and My denote the respective maximum values of X and Y among all records in the table at any point in time. Using MX and MY, new records are inserted in the table 128 times with X and Y values being MX+1, 2\*MY+1 respectively. It may be noted that each time after the insertion, values of MX and MY change. What will be the output of the following SQL query after the steps mentioned above are carried out? [) Consider a database table T containing two columns X and Y each of type integer. After the creation of the table, one record (X=1, Y=1) is inserted in the table. Let MX and My denote the respective maximum values of X and Y among all records in the table at any point in time. Using MX and MY, new records are inserted in the table 128 times with X and Y values being MX+1, 2\*MY+1 respectively. It may be noted that each time after the insertion, values of MX and MY change. What will be the output of the following SQL query after the steps mentioned above are carried out?

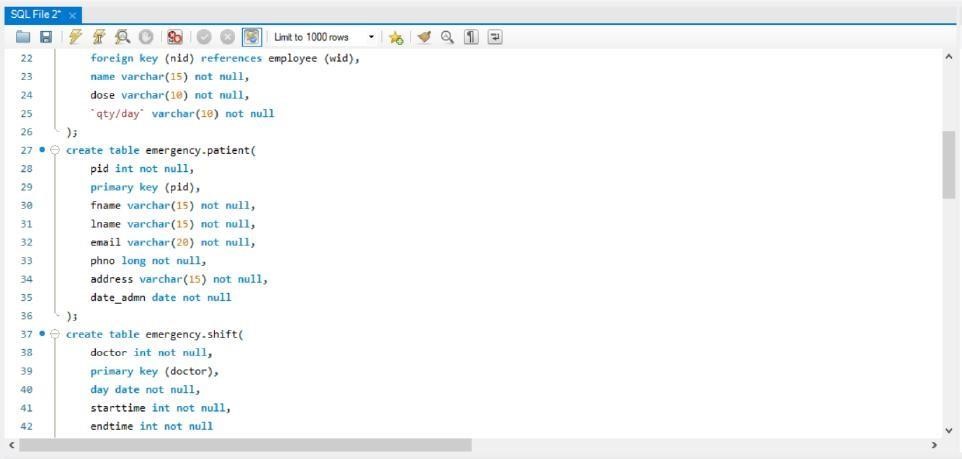
SELECT Y FROM T WHERE X=7;

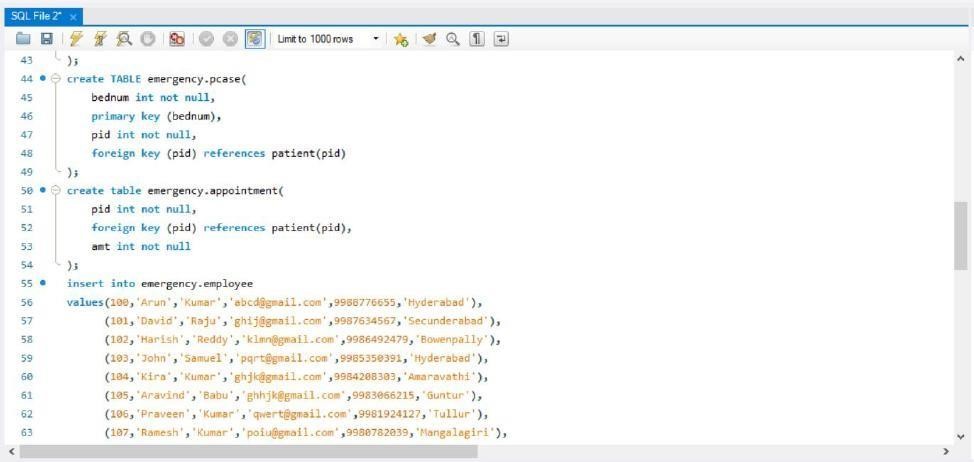
****

**Case Study 2 :EMERGENCY ROOM INFORMATION SYSTEM**

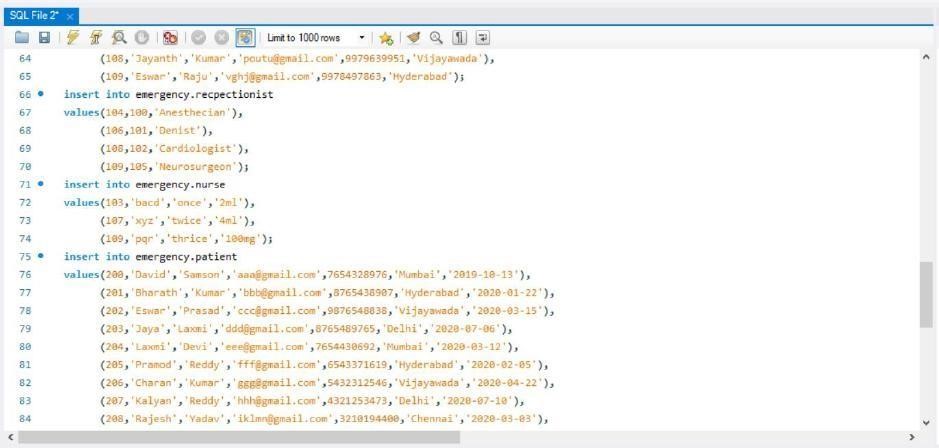
1) Create the database in MySQL and create the necessary tables for the given case study using appropriate keys and relationships between the tables

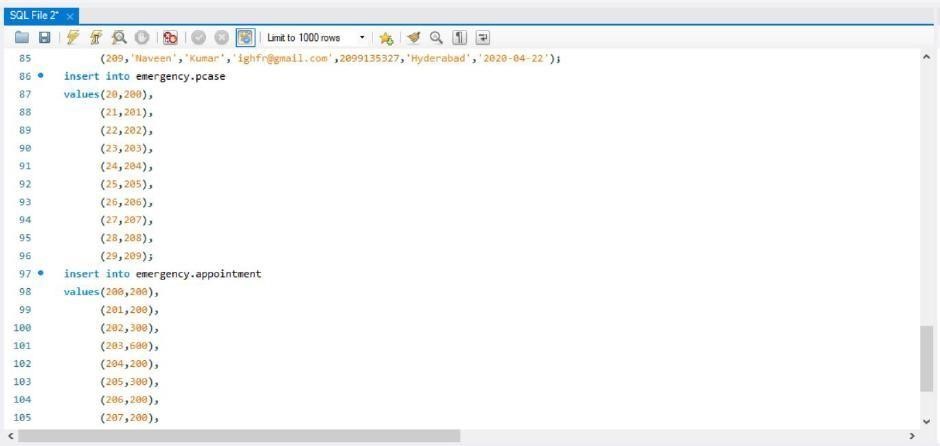


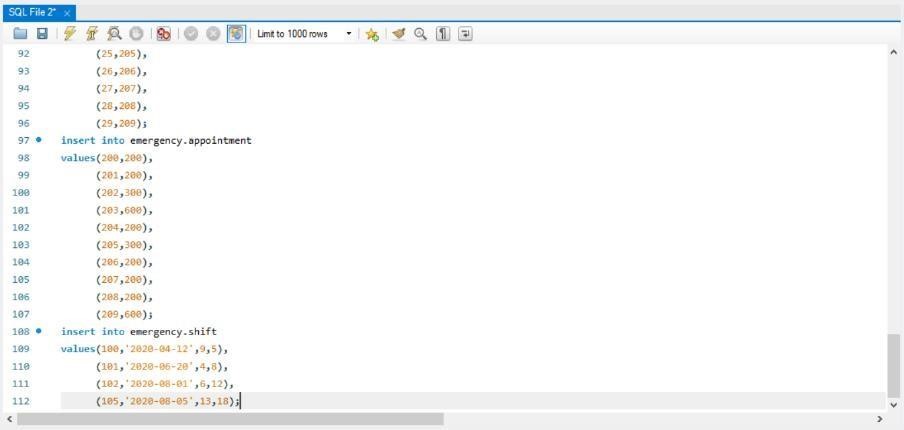




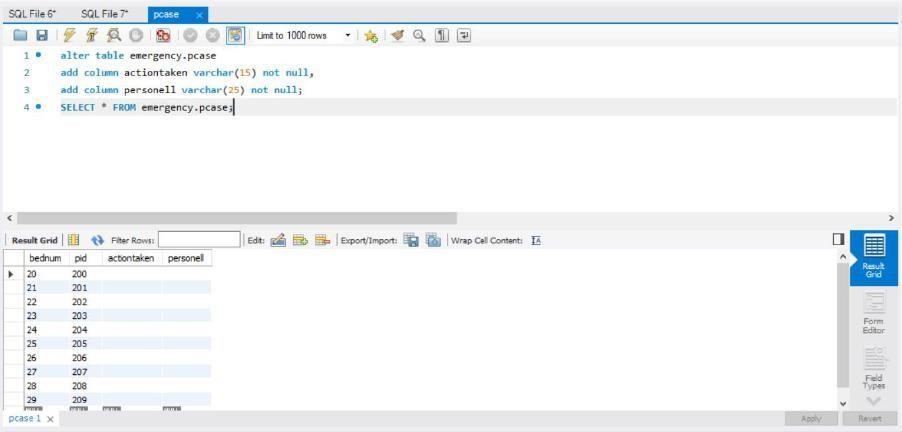
1. Insert atleast 10 records into every table that is implemented in the case study



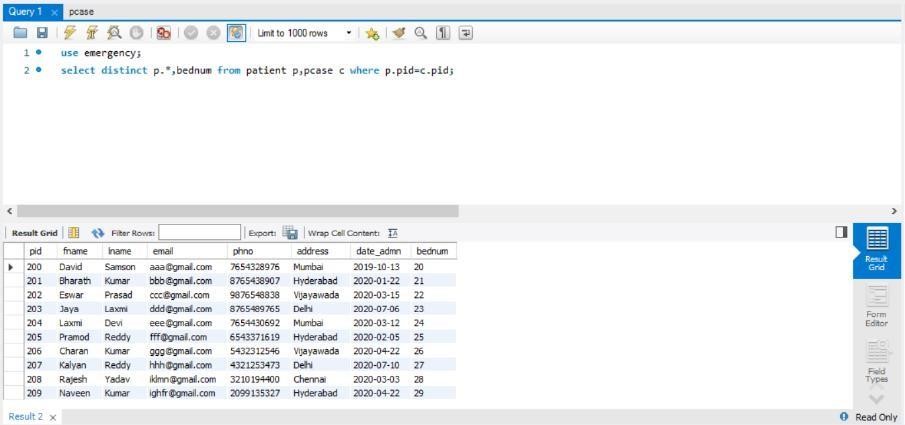




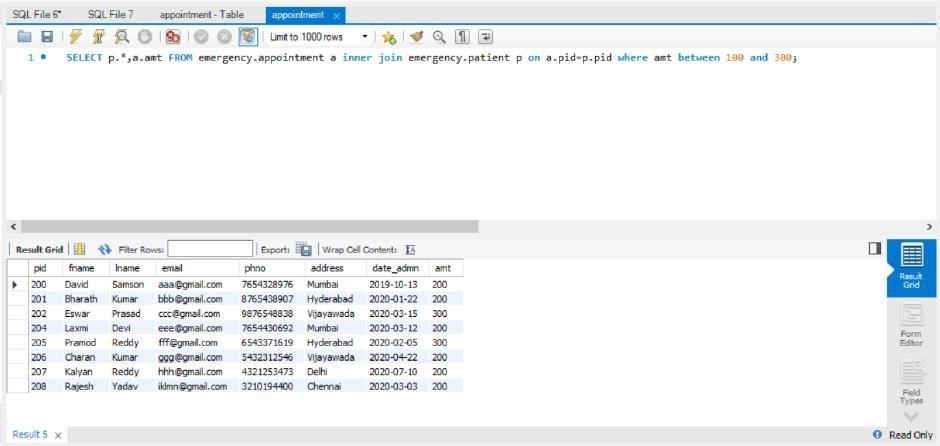
1. Write an MYSQL query to add columns actiontaken and personell to the table case?



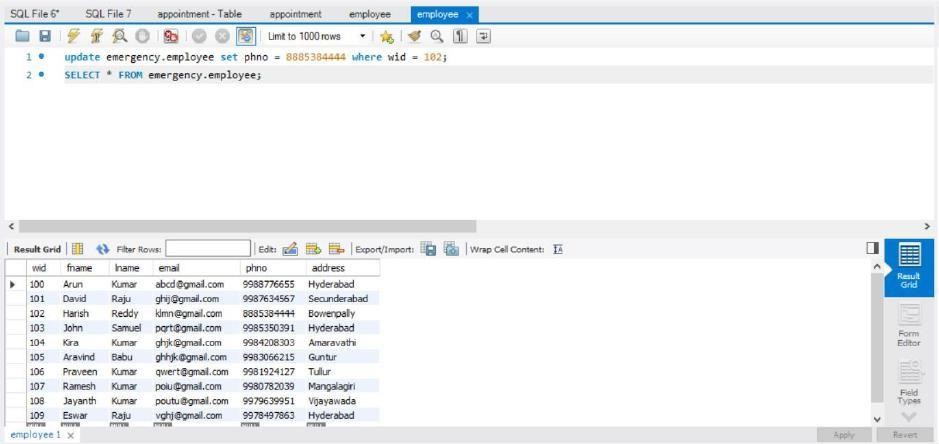
1. Write an MYSQL query to display who are patients required to stay in the hospital?



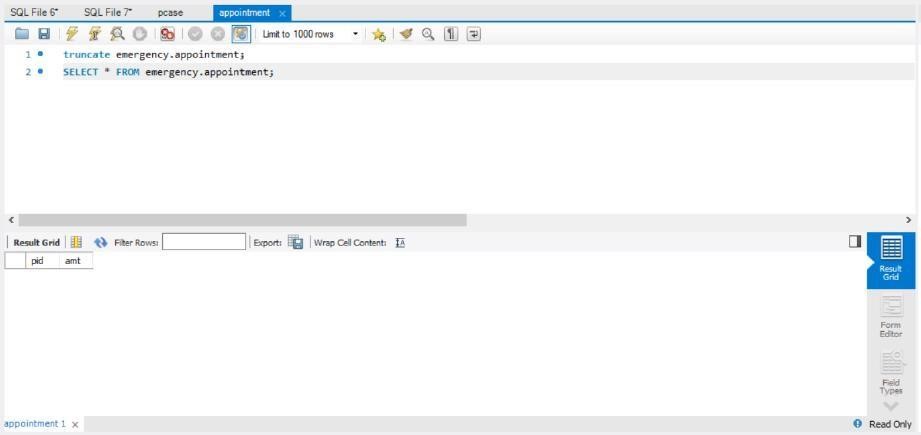
1. Write an MYSQL query to display who paid amount between 100 and 300 for the appointment



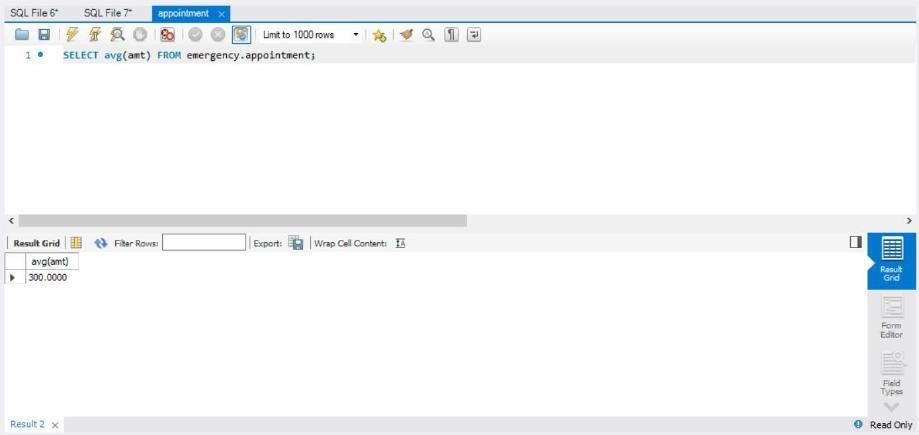
1. Write an MYSQLquery to update the value of phno of WID=102 of worker table



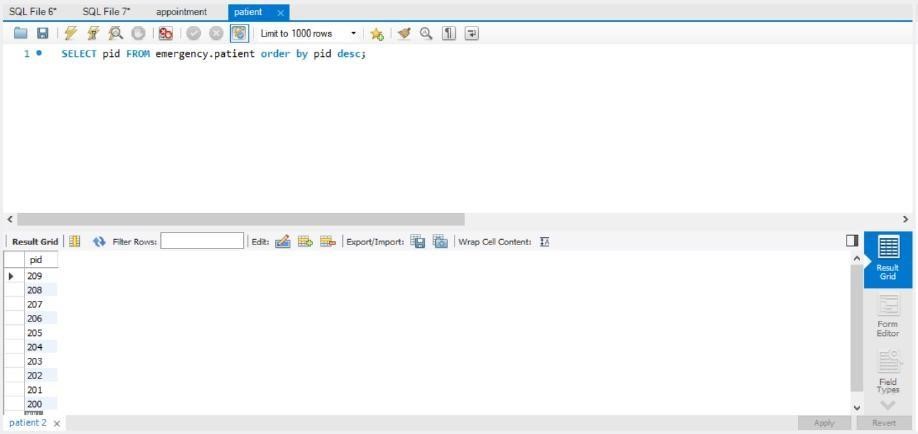
1. Write an MYSQL query to truncate the values of appointment table



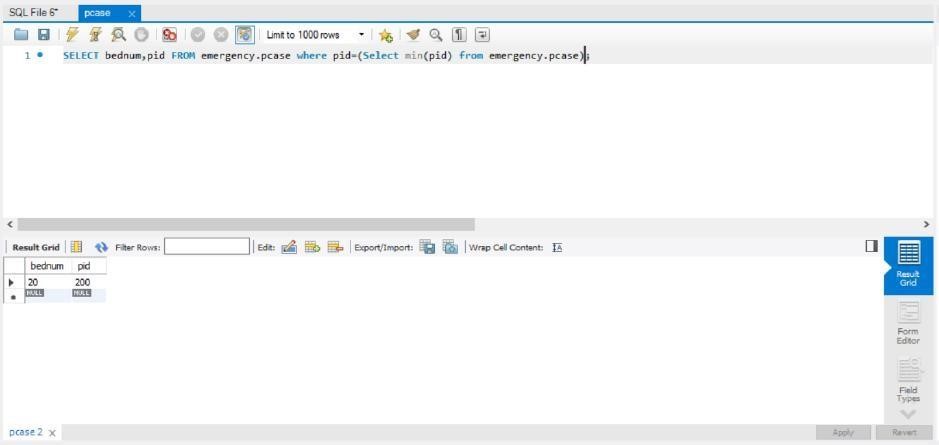
1. Write an MYSQL query to calculate average amount of appointment.



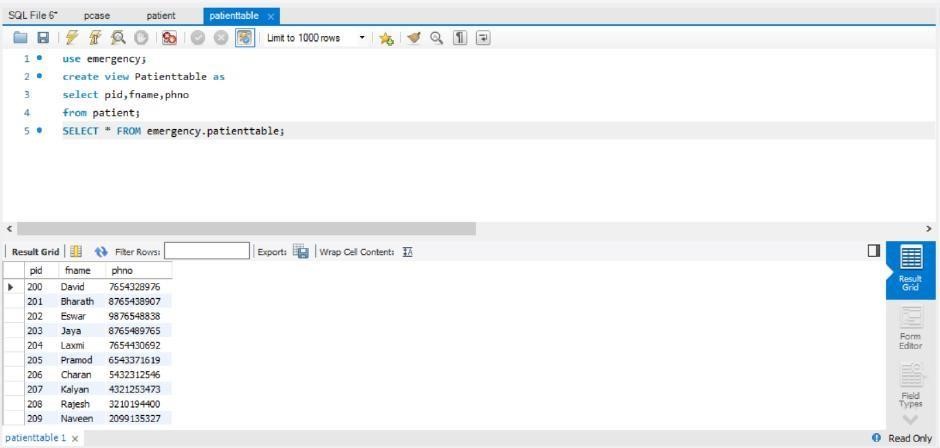
1. Write an MYSQL query to display patient number in descending order.



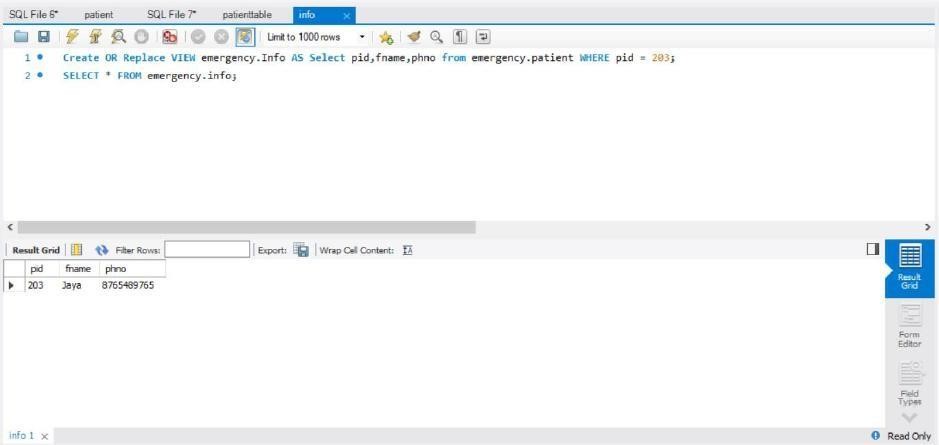
1. Write an MYSQL query to display bed number of patient with least patient id.



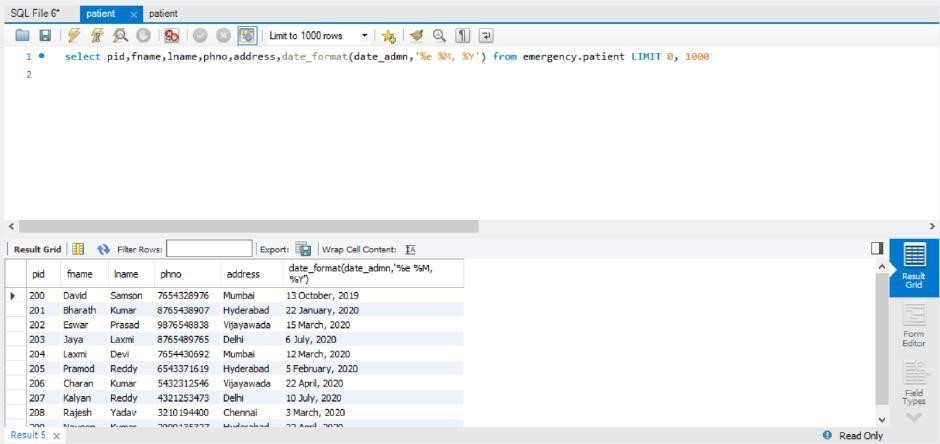
1. Write an MYSQL query to display bed number of patient with least patient id.



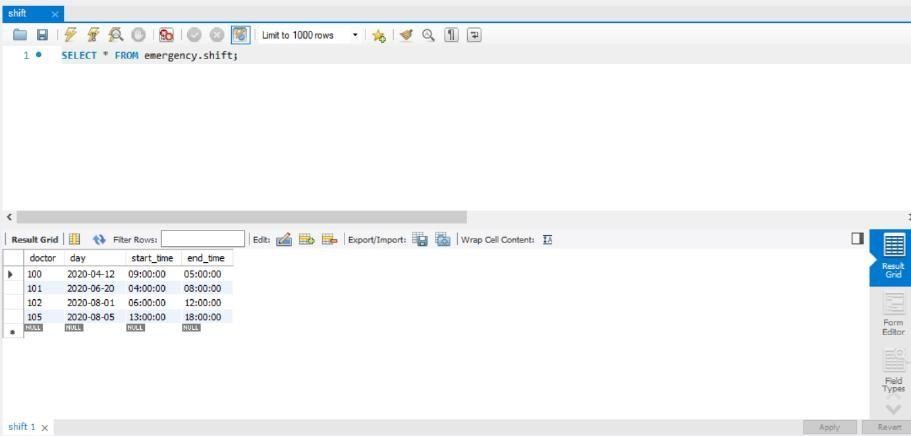
1. Write a MYSQL query to update view the Patientview where patientid>3 by using replace view statement



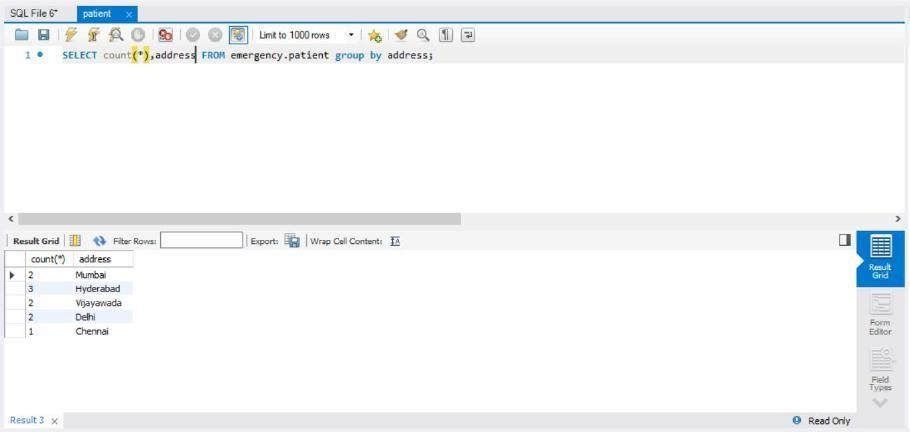
1. Display the patient date of joining into format “dd month yyyy’ for example as “24 March 2020”



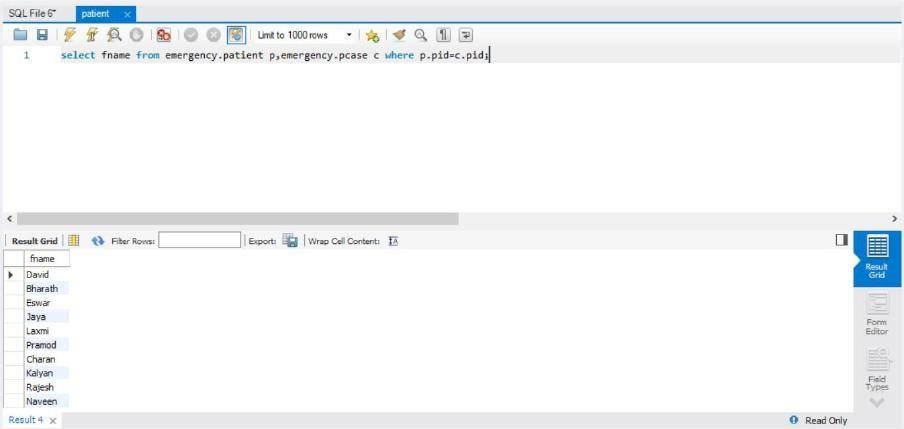
1. Display the patient joining time in 24 hour format. For eg: 3:30pm should be displayed as 15:30



1. Write a MYSQL query to display count of patients who come from same city.

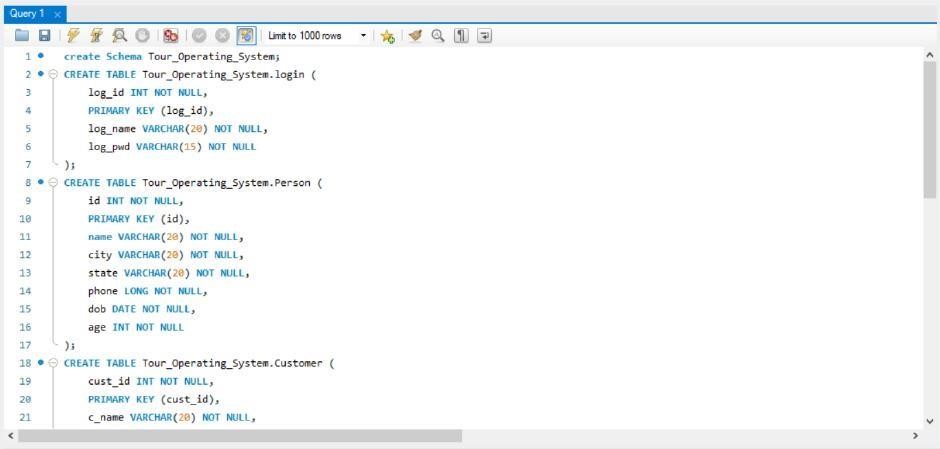


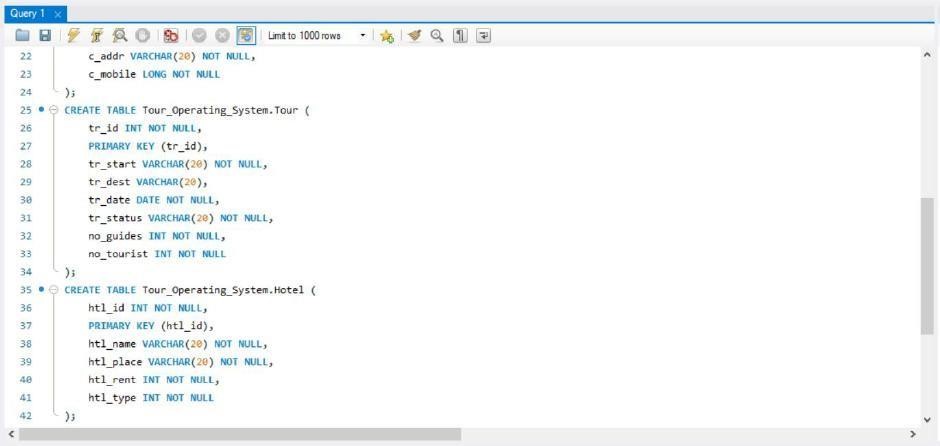
1. Write an MYSQLquery to first name of patient who admitted and allotted the bed



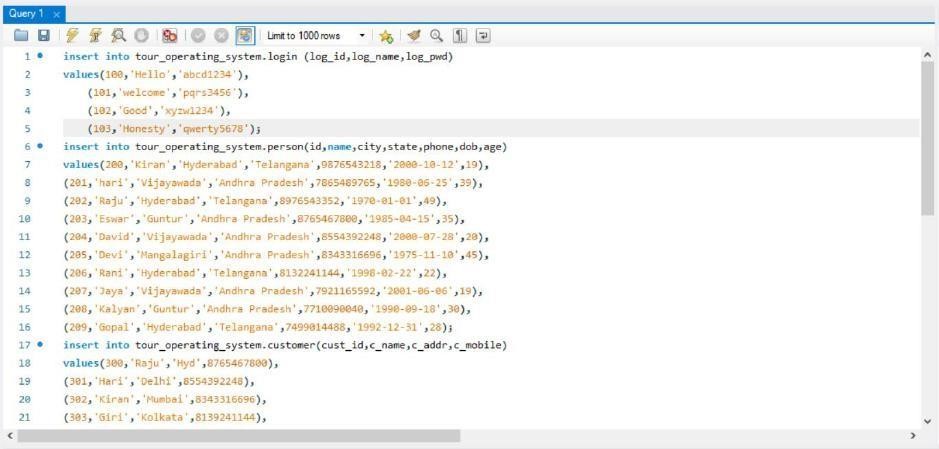
CASE STUDY 2: TOUR OPERATING SYSTEM

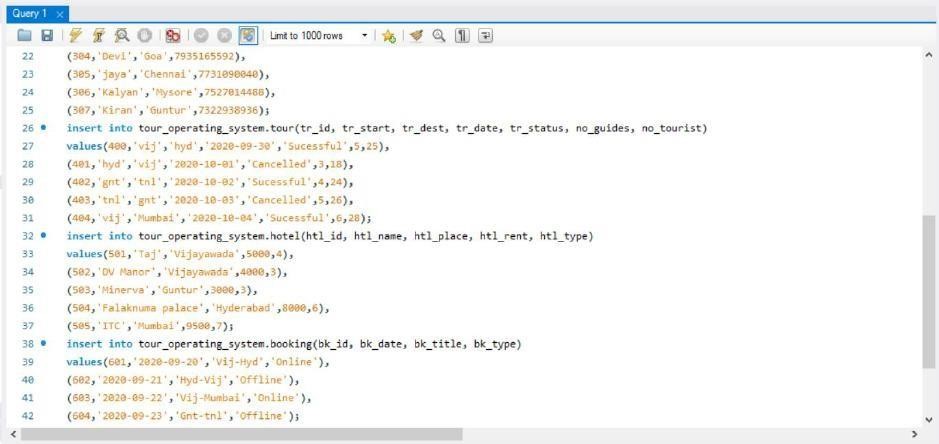
1. Create all the tables required with constraints and relationships between them



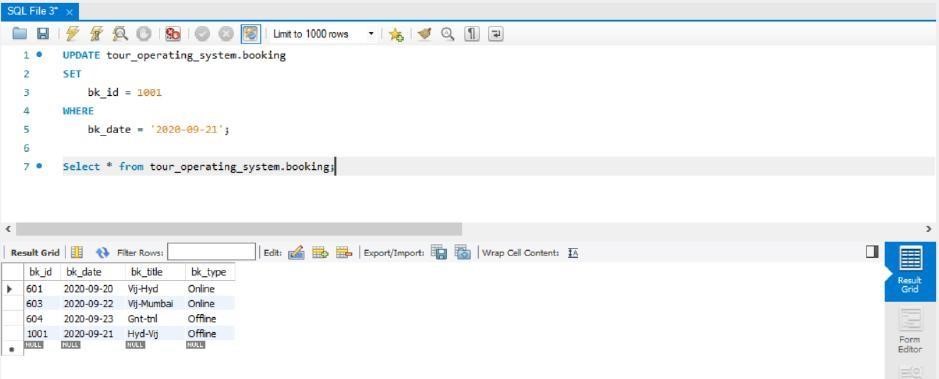


1. Insert atleast 10 records into the tables

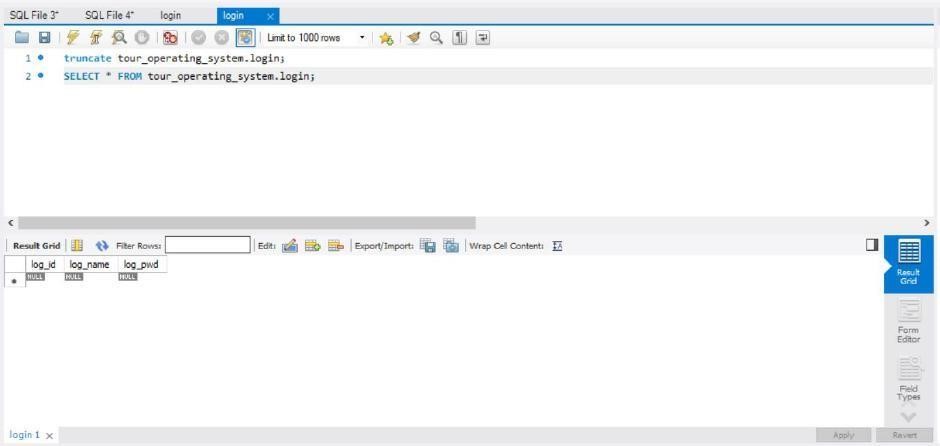




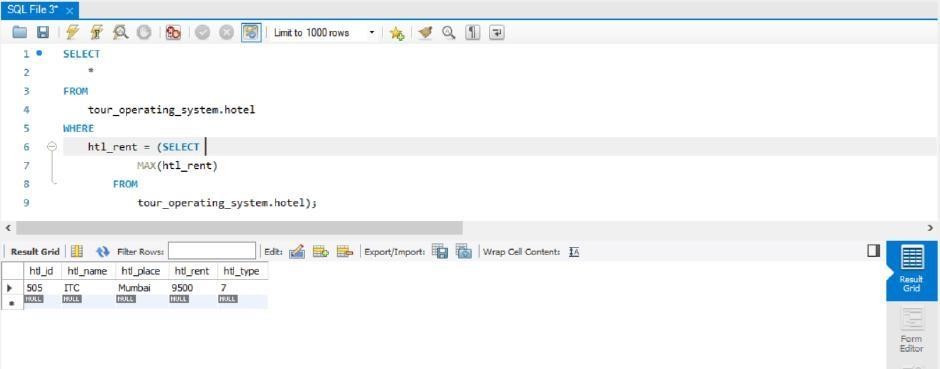
1. Write an SQL Query to Update the value of bk\_id=1001 in Booking table.



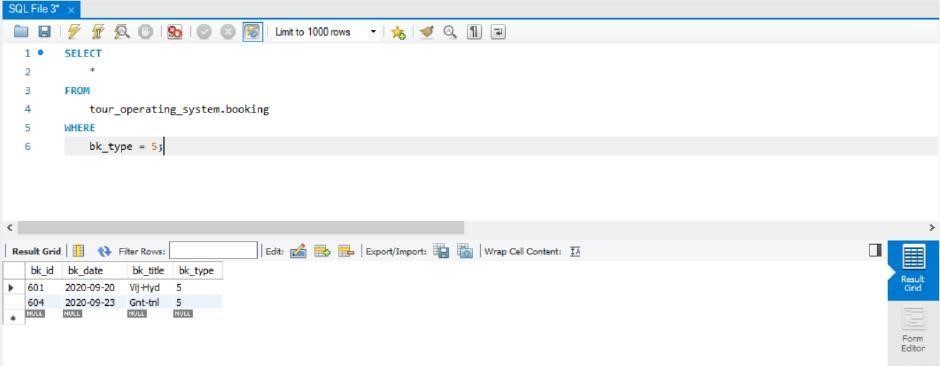
1. Write an SQL Query to Truncate the values of Login table.



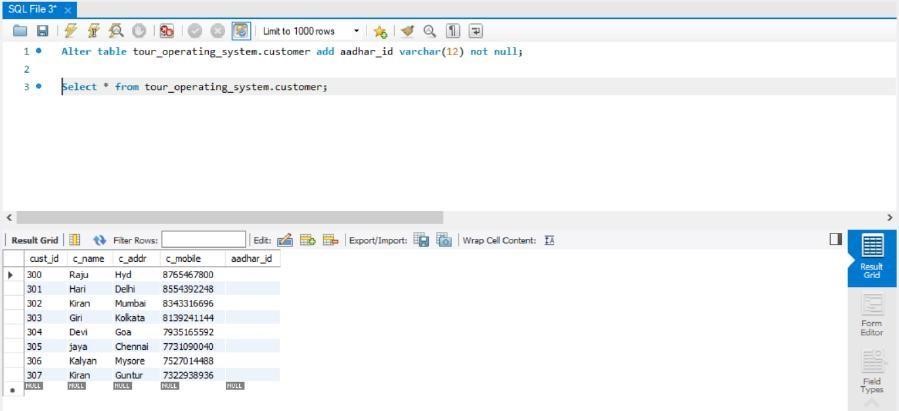
1. Which hotel rent is the highest amount for accommodations?



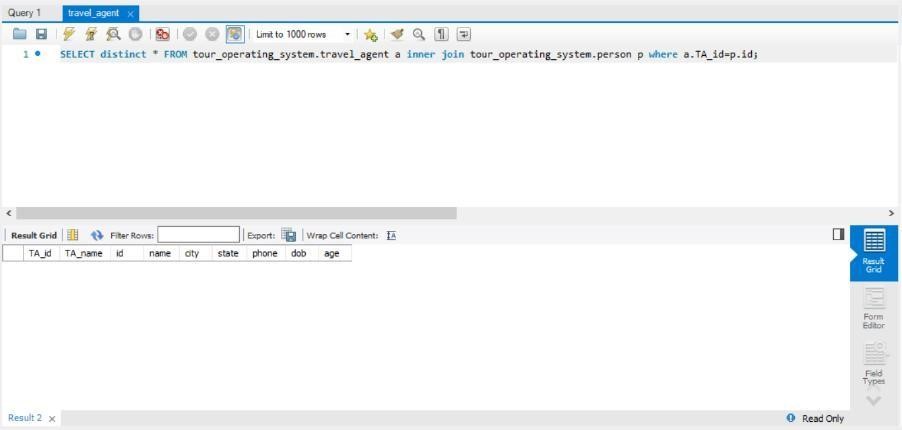
1. SQL Query to print the details of Booking information who have booked 5 star rated rooms.



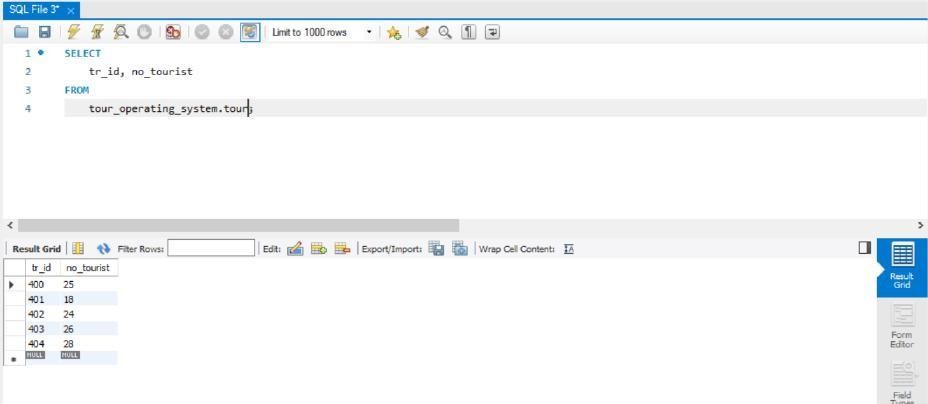
1. Insert a column named aadhar\_id of the customer.



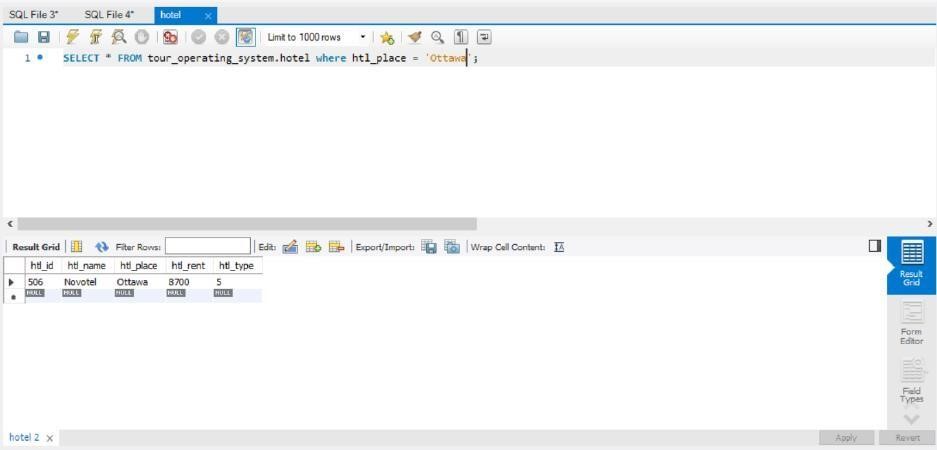
1. Display the persons whose TA\_id and role\_id is matched.



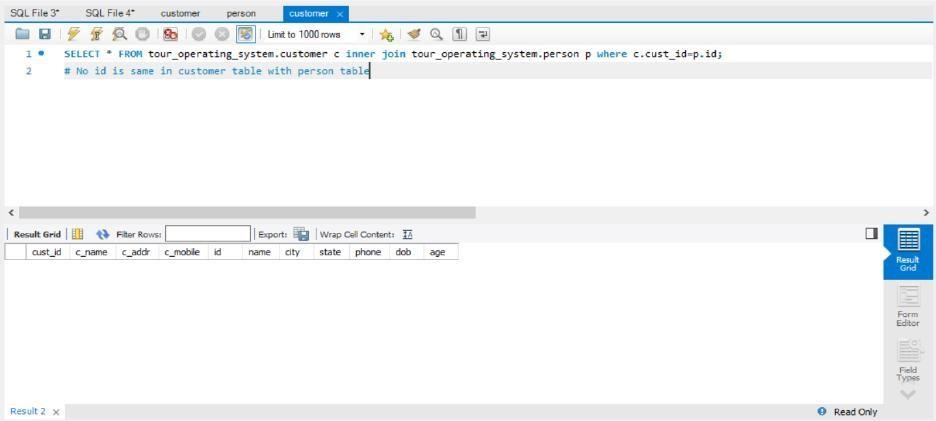
1. How many participants are travelling for a particular tour?



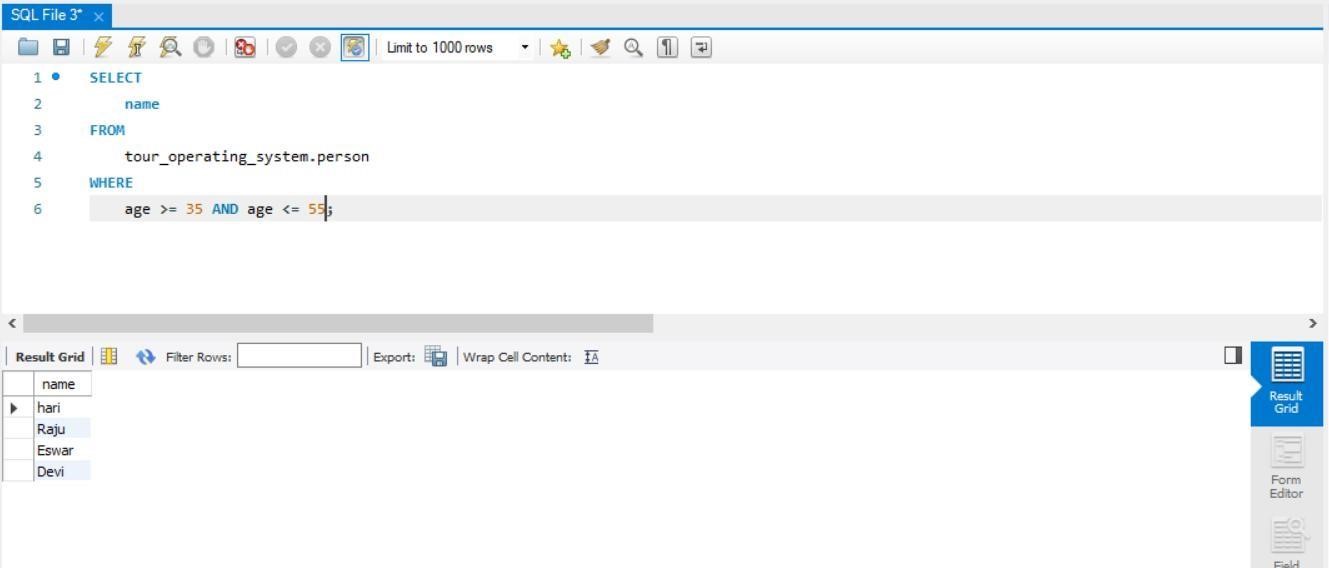
1. Search the names of the hotel in ottawa.



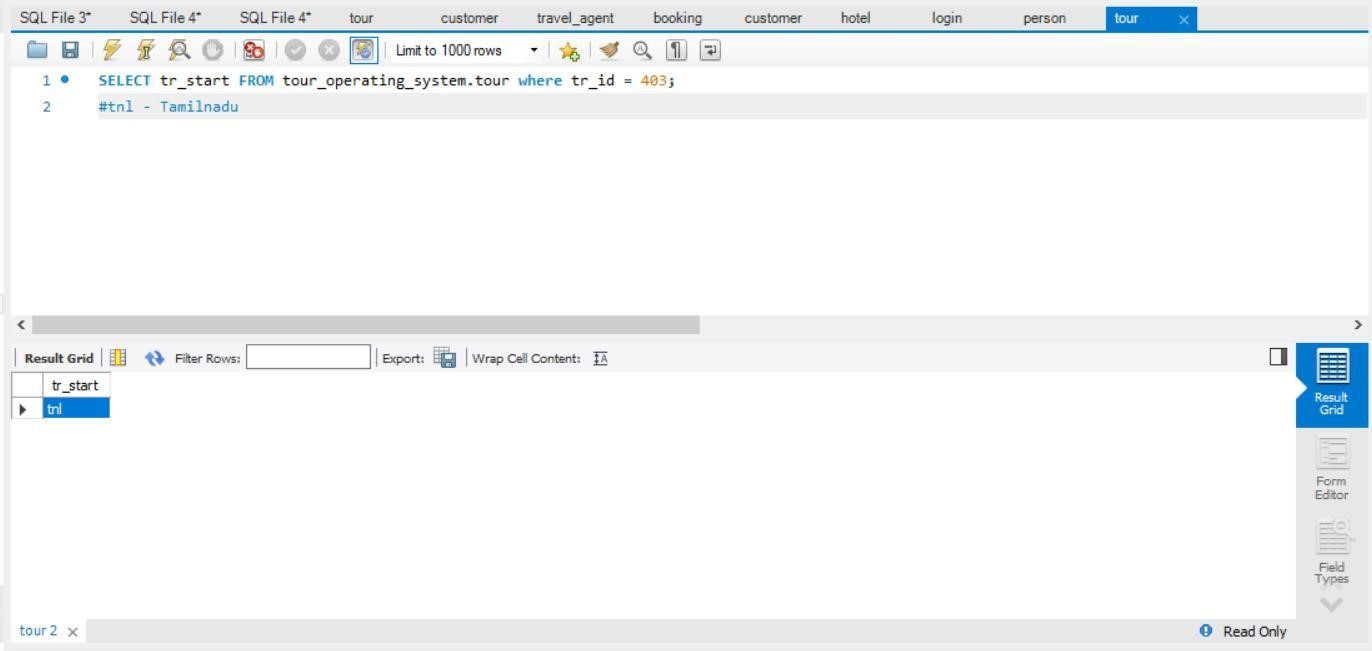
1. Extract the name the customer whose person id and customer id is equal using join.



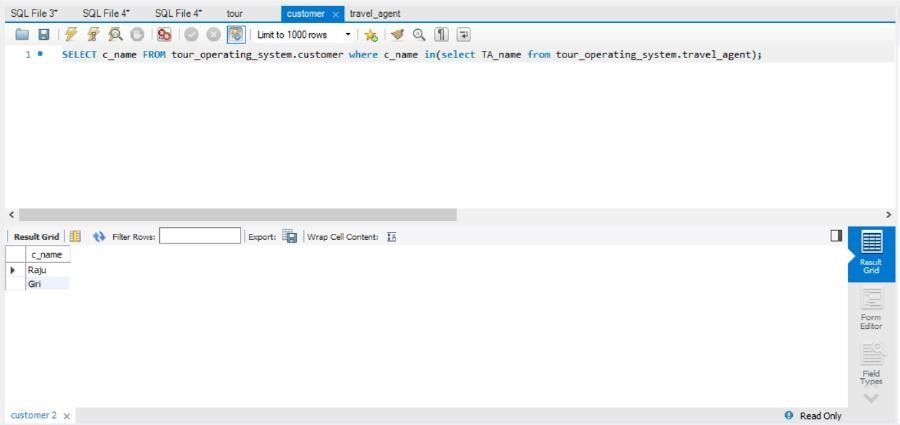
1. Display the name of the participants whose age is between 35 and 55;



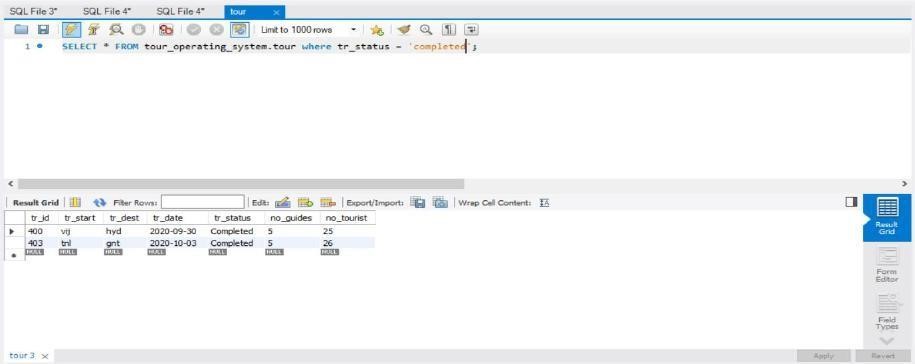
1. What is the origin of the travelling for tour id = 403



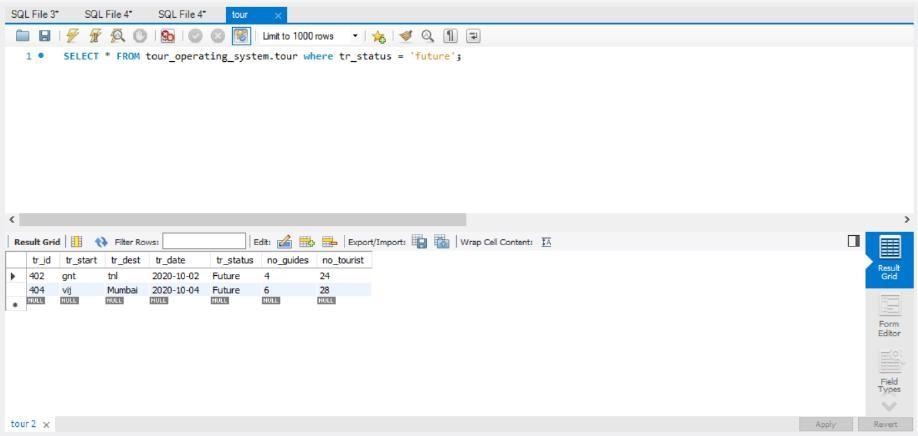
1. Select a person whose name is available in Customer and TravelAgent using nested query.



1. Select the past tour details.

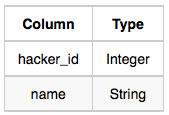


16)Select the future tour details.

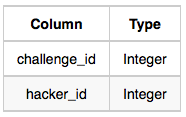


**POST-LAB**

1. Julia asked her students to create some coding challenges. Write a query to print the *hacker\_id*, *name*, and the total number of challenges created by each student. Sort your results by the total number of challenges in descending order. If more than one student created the same number of challenges, then sort the result by *hacker\_id*. If more than one student created the same number of challenges and the count is less than the maximum number of challenges created, then exclude those students from the result.

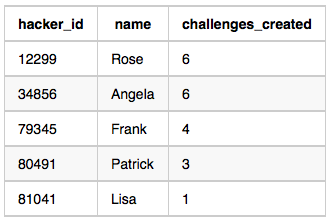
**Input Format**

The following tables contain challenge data:

* *Hackers:* The *hacker\_id* is the id of the hacker, and *name* is the name of the hacker.
* *Challenges:* The *challenge\_id* is the id of the challenge, and *hacker\_id* is the id of the student who created thechallenge

**Explanation**

For *Sample Case 1*, we can get the following details:

  
  
Students 12299 and 34856 both created 6 challenges. Because 6 is the maximum number of challenges created, these students are included in the result.

**ANS.**

SELECT c.hacker\_id, h.name, COUNT(c.challenge\_id) AS cnt FROM Hackers AS h JOIN Challenges AS c ON h.hacker\_id = c.hacker\_id GROUP BY c.hacker\_id, h.name HAVING

cnt = (SELECT COUNT(c1.challenge\_id) FROM Challenges AS c1 GROUP BY c1.hacker\_id ORDER BY COUNT(\*) DESC LIMIT 1) OR cnt NOT IN (SELECT COUNT(c2.challenge\_id) FROM Challenges AS c2 GROUP BY c2.hacker\_id HAVING c2.hacker\_id <> c.hacker\_id) ORDER BY cnt DESC, c.hacker\_id;

2) Query all columns for all American cities in the **CITY** table with populations larger than 100000. The **CountryCode** for America is USA.The **CITY** table is described as follows:



**ANS.**

**SELECT \* FROM CITY WHERE COUNTRYCODE = 'USA' AND POPULATION > 100000;**