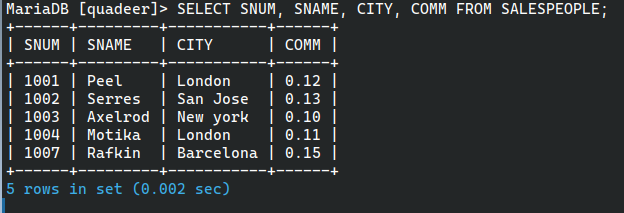
**SQL QUERIES WITH SCREENSHOTS**

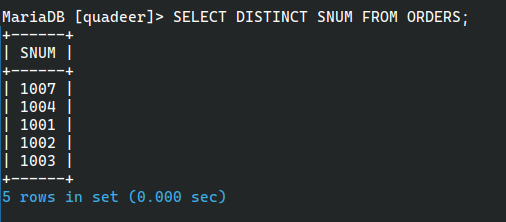
1. **Display snum,sname,city and comm of all salespeople.**

SELECT SNUM, SNAME, CITY, COMM FROM SALESPEOPLE;

****

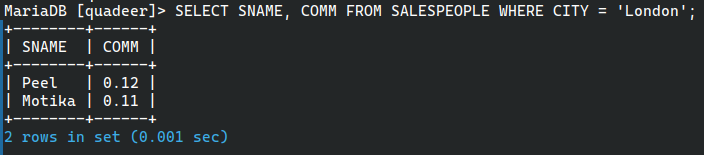
1. **Display all snum without duplicates from all orders**.

SELECT DISTINCT SNUM FROM ORDERS;



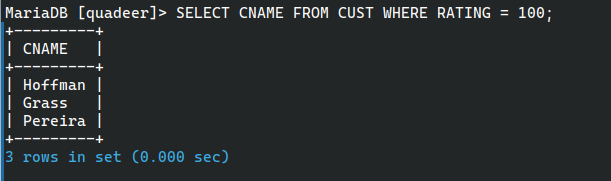
1. **Display names and commissions of all salespeople in london.**

SELECT SNAME, COMM FROM SALESPEOPLE WHERE CITY = 'London';

****

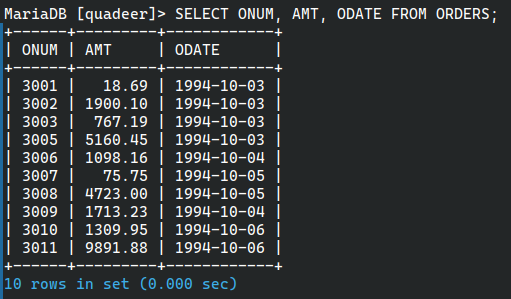
1. **All customers with rating of 100.**

SELECT CNAME FROM CUST WHERE RATING = 100;

****

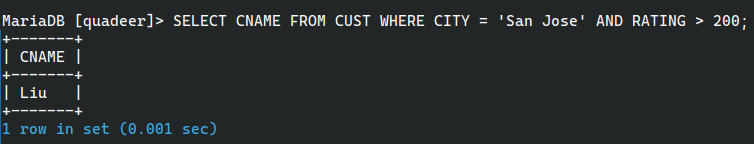
1. **Produce orderno, amount and date form all rows in the order table.**

SELECT ONUM, AMT, ODATE FROM ORDERS;

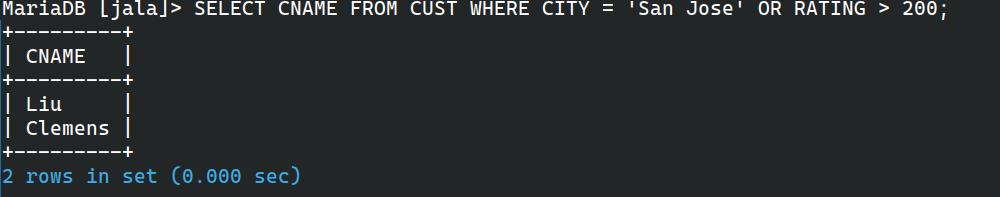
****

1. **All customers in San Jose, who have rating more than 200.**

SELECT CNAME FROM CUST WHERE CITY = 'San Jose' AND RATING > 200;

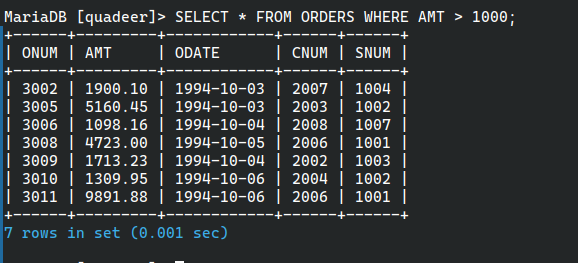
****

1. **All customers who were either located in San Jose or had a rating above 200.**

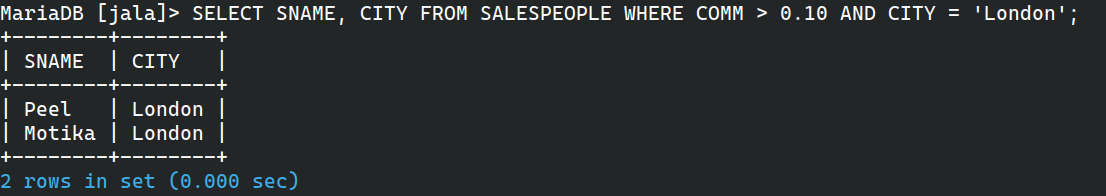


1. **All orders for more than $1000.**

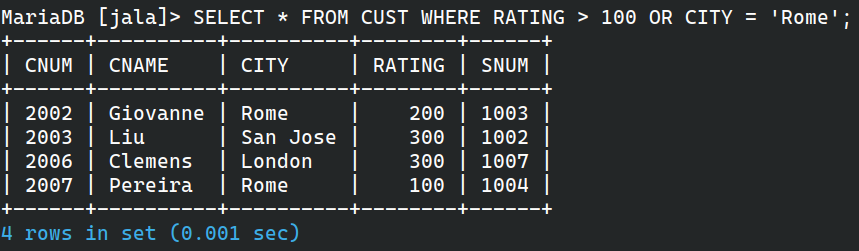
SELECT \* FROM ORDERS WHERE AMT > 1000;

****

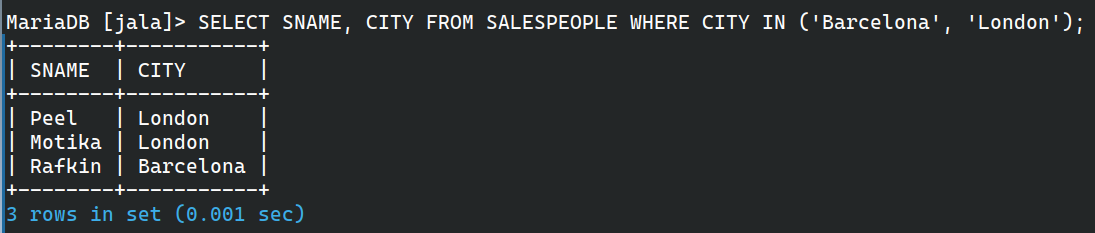
1. **Names and citires of all salespeople in london with commission above 0.10.**

****

1. **All customers excluding those with rating <= 100 unless they are located in Rome.**

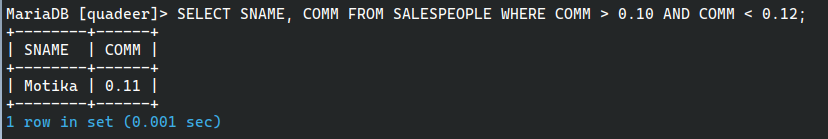
****

1. **All salespeople either in Barcelona or in london.**

****

1. **All salespeople with commission between 0.10 and 0.12. (Boundary values should be excluded)**

SELECT SNAME, COMM FROM SALESPEOPLE WHERE COMM > 0.10 AND COMM < 0.12;

****

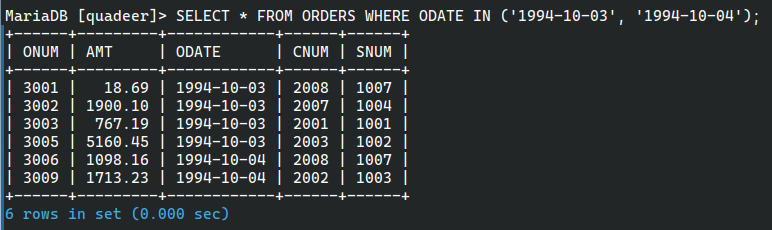
1. **All customers with NULL values in city column.**

SELECT CNAME FROM CUST WHERE CITY IS NULL;

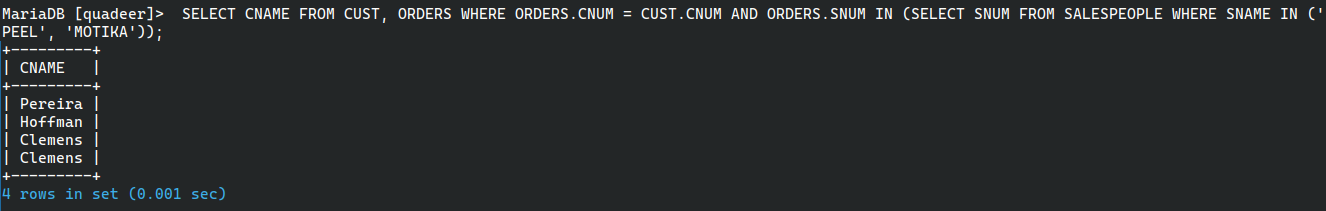


1. **All orders taken on Oct 3Rd and Oct 4th 1994.**

SELECT \* FROM ORDERS WHERE ODATE IN ('1994-10-03', '1994-10-04');

****

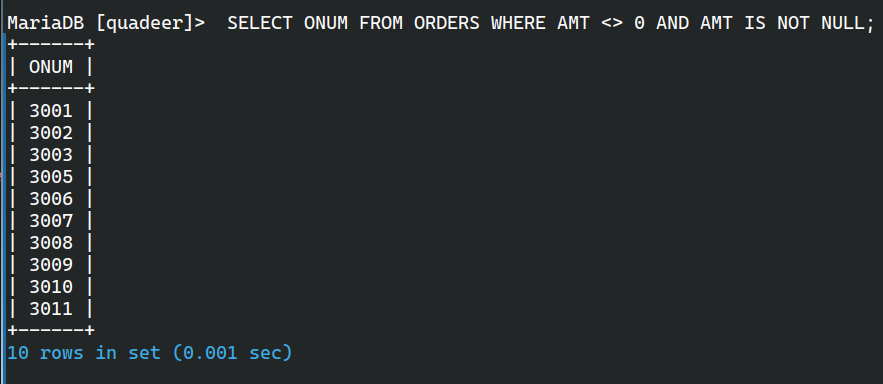
**15. ALL CUSTOMERS SERVICED BY PEEL OR MOTIKA** SELECT CNAME FROM CUST, ORDERS WHERE ORDERS.CNUM = CUST.CNUM AND ORDERS.SNUM IN (SELECT SNUM FROM SALESPEOPLE WHERE SNAME IN ('PEEL', 'MOTIKA'));



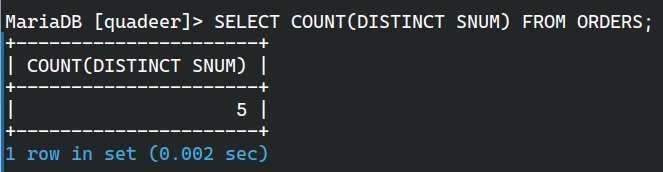
**16. ALL CUSTOMERS WHOSE NAMES BEGIN WITH A LETTER FROM A TO B** SELECT CNAME FROM CUST WHERE CNAME LIKE 'A%' OR CNAME LIKE 'B%';



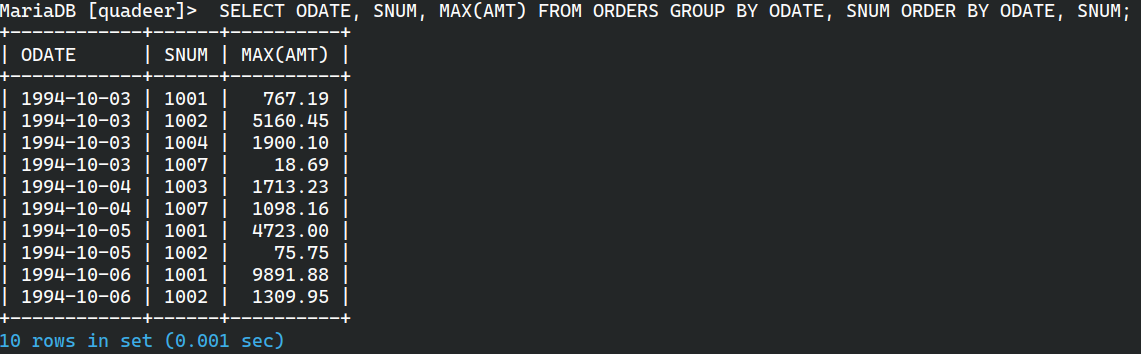
**17. ALL ORDERS EXCEPT THOSE WITH 0 OR NULL VALUE IN AMT FIELD** SELECT ONUM FROM ORDERS WHERE AMT <> 0 AND AMT IS NOT NULL;



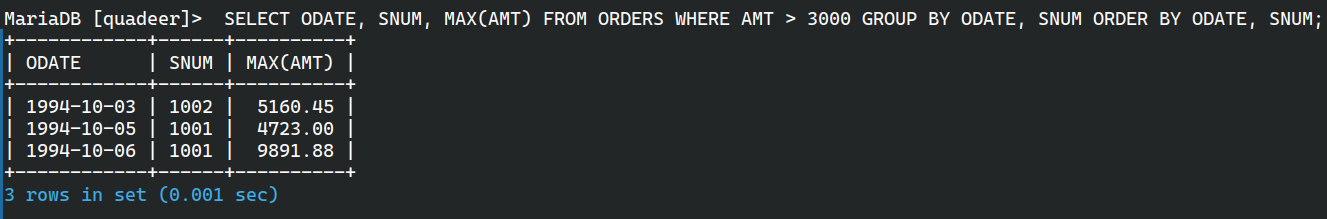
**18. COUNT THE NUMBER OF SALESPEOPLE CURRENTLY LISTING ORDERS IN THE ORDER TABLE** SELECT COUNT(DISTINCT SNUM) FROM ORDERS;



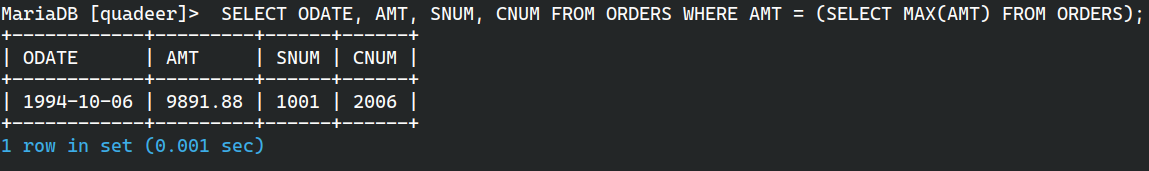
**19. LARGEST ORDER TAKEN BY EACH SALESPERSON, DATEWISE** SELECT ODATE, SNUM, MAX(AMT) FROM ORDERS GROUP BY ODATE, SNUM ORDER BY ODATE, SNUM;



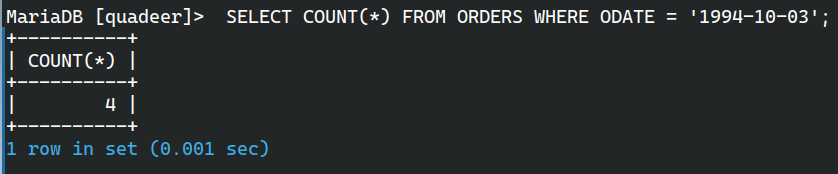
**20. LARGEST ORDER TAKEN BY EACH SALESPERSON WITH ORDER VALUE MORE THAN $3000** SELECT ODATE, SNUM, MAX(AMT) FROM ORDERS WHERE AMT > 3000 GROUP BY ODATE, SNUM ORDER BY ODATE, SNUM;



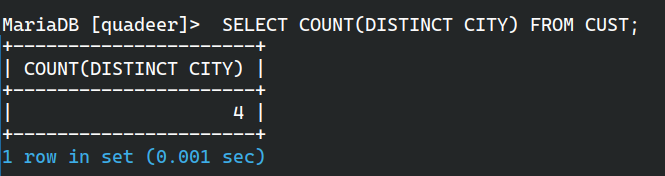
**21. WHICH DAY HAD THE HIGHEST TOTAL AMOUNT ORDERED** SELECT ODATE, AMT, SNUM, CNUM FROM ORDERS WHERE AMT = (SELECT MAX(AMT) FROM ORDERS);



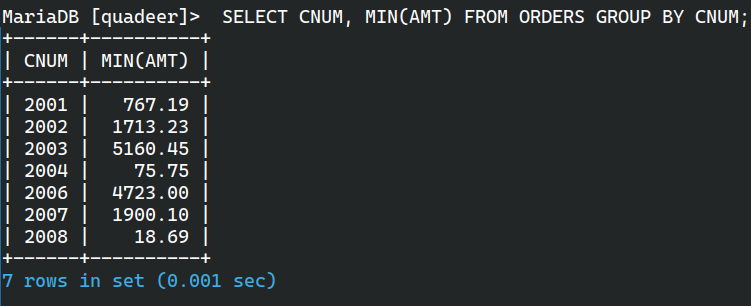
**22. COUNT ALL ORDERS FOR OCT 3RD** SELECT COUNT(\*) FROM ORDERS WHERE ODATE = '03-OCT-94';



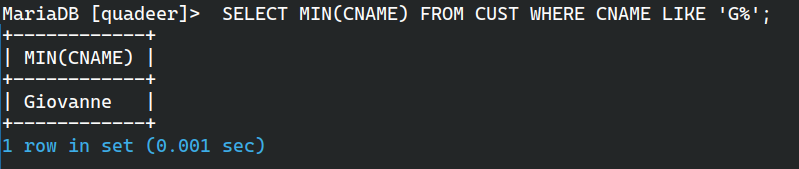
**23. COUNT THE NUMBER OF DIFFERENT NON NULL CITY VALUES IN CUSTOMERS TABLE** SELECT COUNT(DISTINCT CITY) FROM CUST;



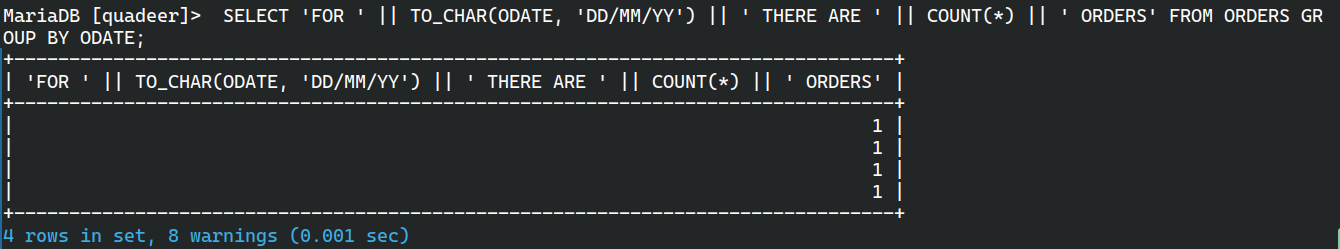
**24. SELECT EACH CUSTOMER’S SMALLEST ORDER** SELECT CNUM, MIN(AMT) FROM ORDERS GROUP BY CNUM;



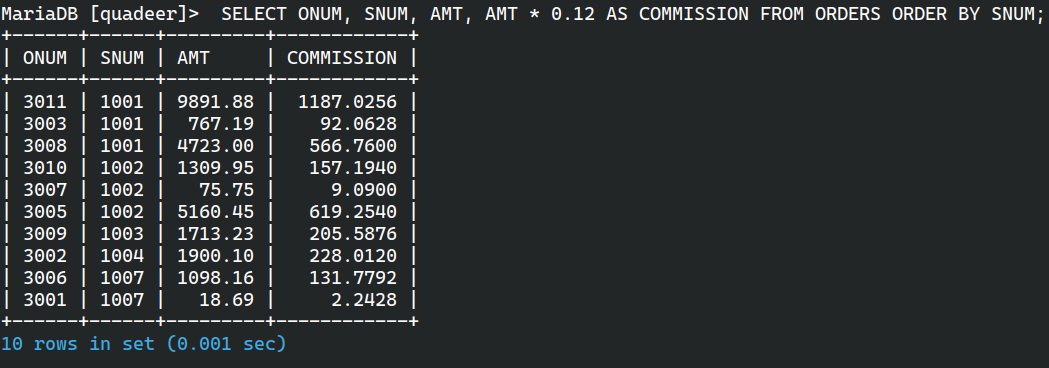
**25. FIRST CUSTOMER IN ALPHABETICAL ORDER WHOSE NAME BEGINS WITH G** SELECT MIN(CNAME) FROM CUST WHERE CNAME LIKE 'G%';



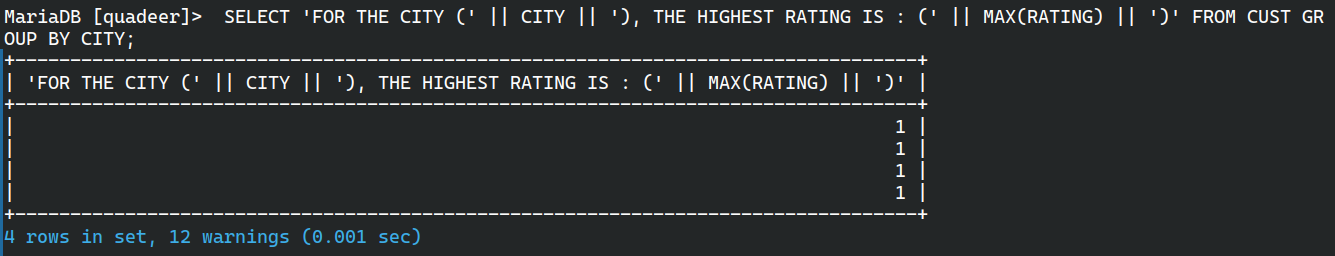
**26. GET THE OUTPUT LIKE “FOR DD/MM/YY THERE ARE \_\_\_ ORDERS”** SELECT 'FOR ' || TO\_CHAR(ODATE, 'DD/MM/YY') || ' THERE ARE ' || COUNT(\*) || ' ORDERS' FROM ORDERS GROUP BY ODATE;



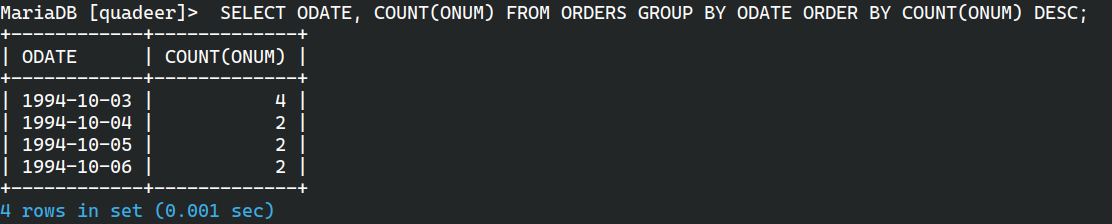
**27. ASSUME THAT EACH SALESPERSON HAS A 12% COMMISSION. PRODUCE ORDER NO., SALESPERSON NO., AND AMOUNT OF SALESPERSON’S COMMISSION FOR THAT ORDER** SELECT ONUM, SNUM, AMT, AMT \* 0.12 AS COMMISSION FROM ORDERS ORDER BY SNUM;



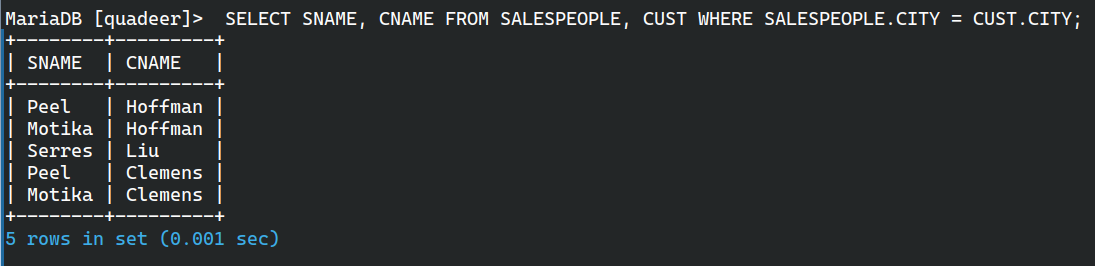
**28. FIND HIGHEST RATING IN EACH CITY. PUT THE OUTPUT IN THIS FORM. FOR THE CITY (CITY), THE HIGHEST RATING IS : (RATING)** SELECT 'FOR THE CITY (' || CITY || '), THE HIGHEST RATING IS : (' || MAX(RATING) || ')' FROM CUST GROUP BY CITY;



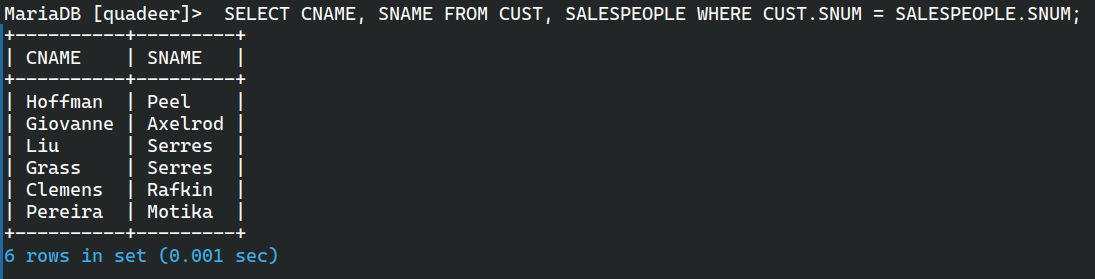
**29. DISPLAY THE TOTALS OF ORDERS FOR EACH DAY AND PLACE THE RESULTS IN DESCENDING ORDER** SELECT ODATE, COUNT(ONUM) FROM ORDERS GROUP BY ODATE ORDER BY COUNT(ONUM) DESC;



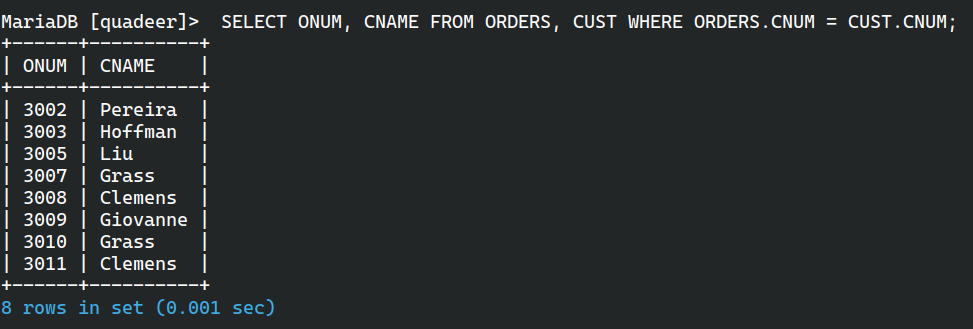
**30. ALL COMBINATIONS OF SALESPEOPLE AND CUSTOMERS WHO SHARED A CITY (IE SAME CITY)** SELECT SNAME, CNAME FROM SALESPEOPLE, CUST WHERE SALESPEOPLE.CITY = CUST.CITY;



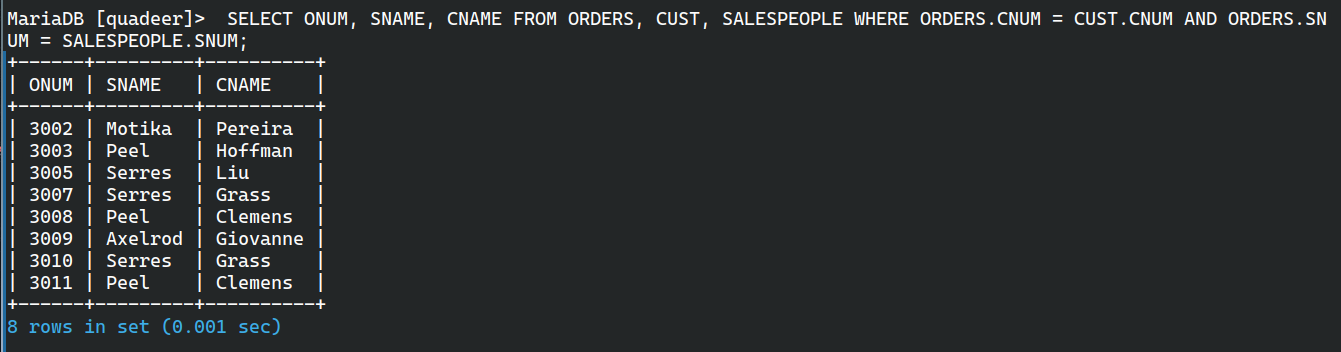
**31. NAME OF ALL CUSTOMERS MATCHED WITH THE SALESPEOPLE SERVING THEM** SELECT CNAME, SNAME FROM CUST, SALESPEOPLE WHERE CUST.SNUM = SALESPEOPLE.SNUM;



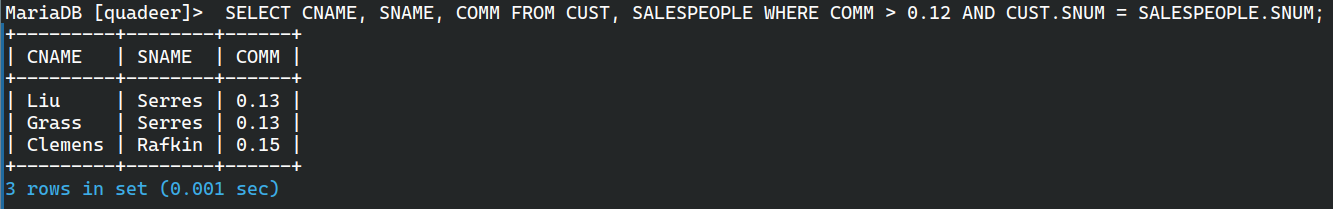
**32. LIST EACH ORDER NUMBER FOLLOWED BY THE NAME OF THE CUSTOMER WHO MADE THE ORDER** SELECT ONUM, CNAME FROM ORDERS, CUST WHERE ORDERS.CNUM = CUST.CNUM;



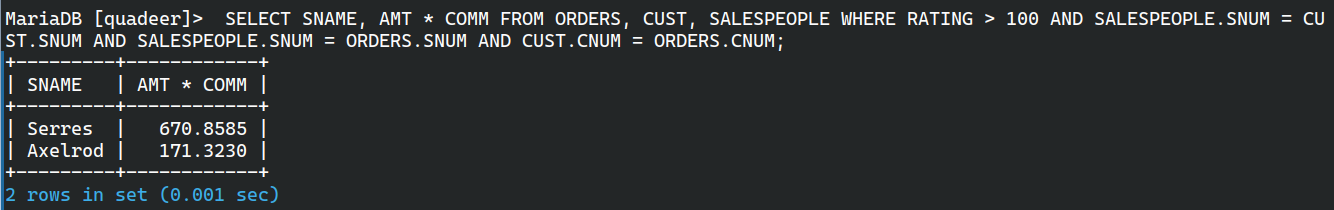
**33. NAMES OF SALESPERSON AND CUSTOMER FOR EACH ORDER AFTER THE ORDER NUMBER** SELECT ONUM, SNAME, CNAME FROM ORDERS, CUST, SALESPEOPLE WHERE ORDERS.CNUM = CUST.CNUM AND ORDERS.SNUM = SALESPEOPLE.SNUM;



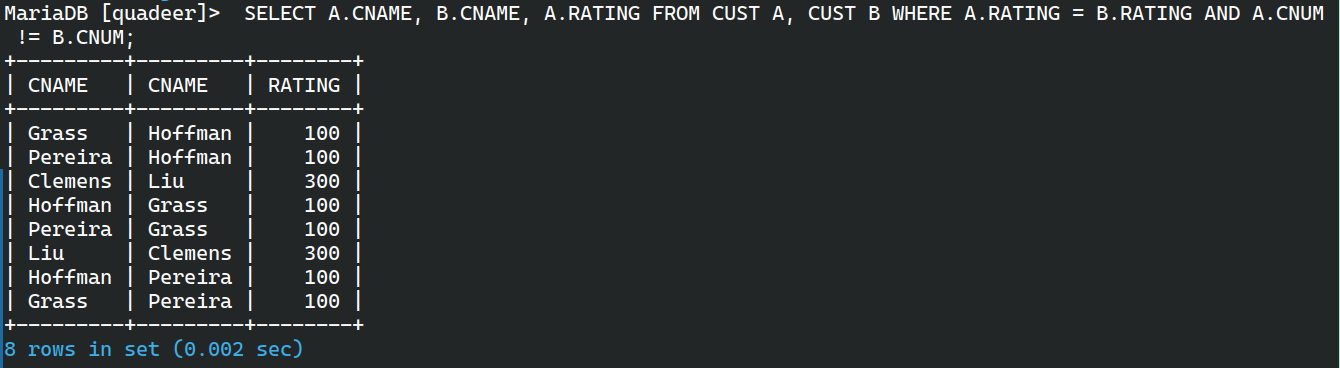
**34. PRODUCE ALL CUSTOMERS SERVICED BY SALESPEOPLE WITH A COMMISSION ABOVE 12%** SELECT CNAME, SNAME, COMM FROM CUST, SALESPEOPLE WHERE COMM > 0.12 AND CUST.SNUM = SALESPEOPLE.SNUM;



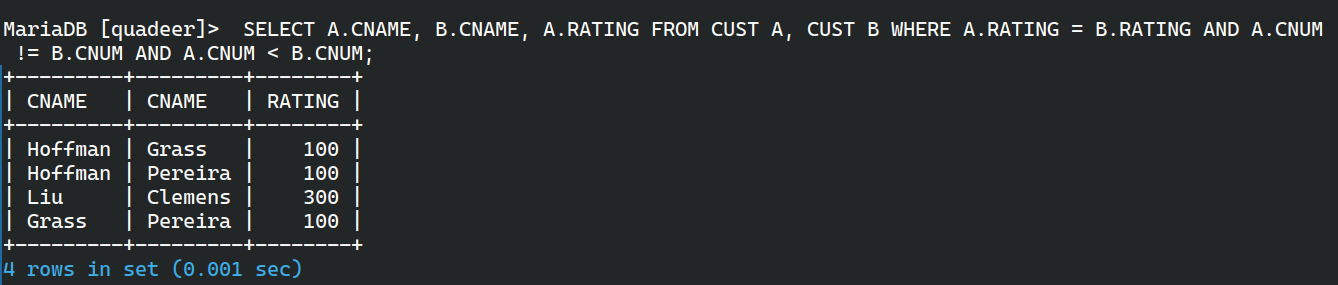
**35. CALCULATE THE AMOUNT OF THE SALESPERSON’S COMMISSION ON EACH ORDER WITH A RATING ABOVE 100** SELECT SNAME, AMT \* COMM FROM ORDERS, CUST, SALESPEOPLE WHERE RATING > 100 AND SALESPEOPLE.SNUM = CUST.SNUM AND SALESPEOPLE.SNUM = ORDERS.SNUM AND CUST.CNUM = ORDERS.CNUM;



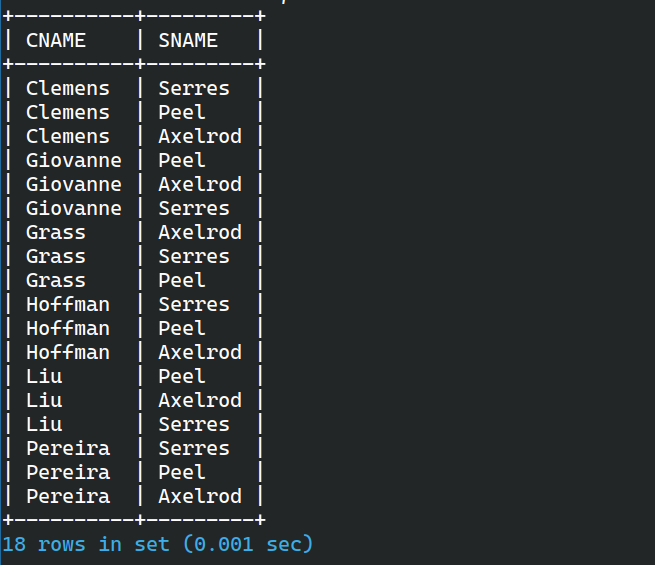
**36. FIND ALL PAIRS OF CUSTOMERS HAVING THE SAME RATING** SELECT A.CNAME, B.CNAME, A.RATING FROM CUST A, CUST B WHERE A.RATING = B.RATING AND A.CNUM != B.CNUM;



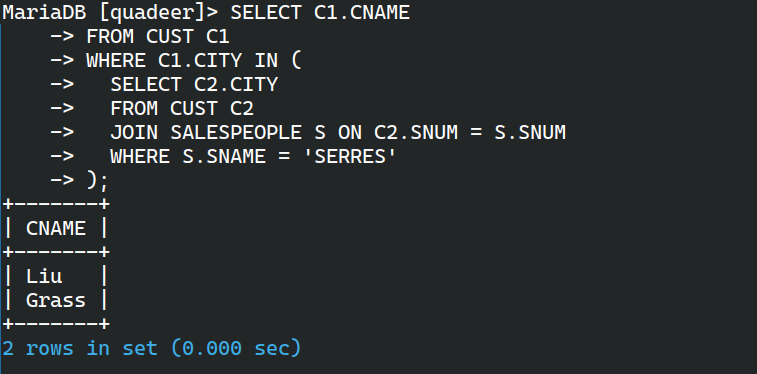
**37. FIND ALL PAIRS OF CUSTOMERS HAVING THE SAME RATING, EACH PAIR COMING ONCE ONLY** SELECT A.CNAME, B.CNAME, A.RATING FROM CUST A, CUST B WHERE A.RATING = B.RATING AND A.CNUM != B.CNUM AND A.CNUM < B.CNUM;



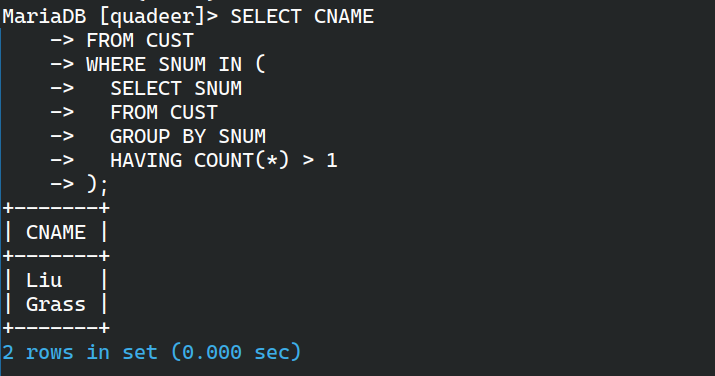
**38. POLICY IS TO ASSIGN THREE SALESPERSON TO EACH CUSTOMER. DISPLAY ALL SUCH COMBINATIONS** SELECT CNAME, SNAME FROM SALESPEOPLE, CUST WHERE SNAME IN (SELECT SNAME FROM SALESPEOPLE WHERE ROWNUM <= 3) ORDER BY CNAME;



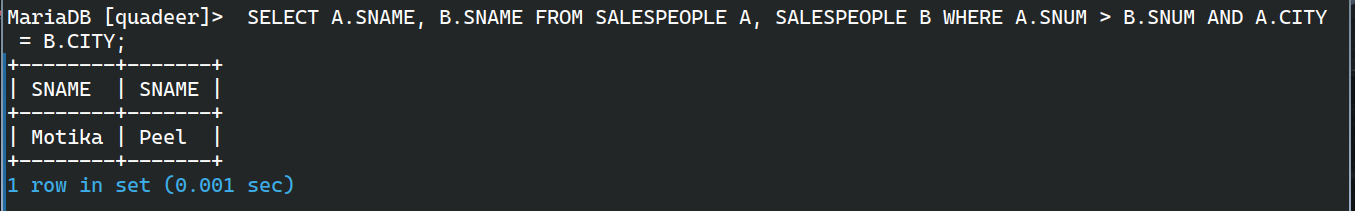
**39. DISPLAY ALL CUSTOMERS LOCATED IN CITIES WHERE SALESMAN SERRES HAS CUSTOMERS** SELECT CNAME FROM CUST WHERE CITY = (SELECT CITY FROM CUST, SALESPEOPLE WHERE CUST.SNUM = SALESPEOPLE.SNUM AND SNAME = 'SERRES'); SELECT CNAME FROM CUST WHERE CITY IN (SELECT CITY FROM CUST, ORDERS WHERE CUST.CNUM = ORDERS.CNUM AND ORDERS.SNUM IN (SELECT SNUM FROM SALESPEOPLE WHERE SNAME = 'SERRES'));



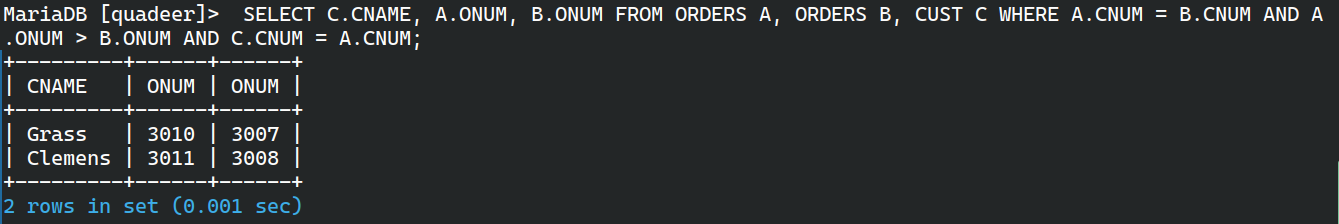
**40. FIND ALL PAIRS OF CUSTOMERS SERVED BY SINGLE SALESPERSON**SELECT CNAME FROM CUST WHERE SNUM IN (SELECT SNUM FROM CUST GROUP BY SNUM HAVING COUNT(\*) > 1);



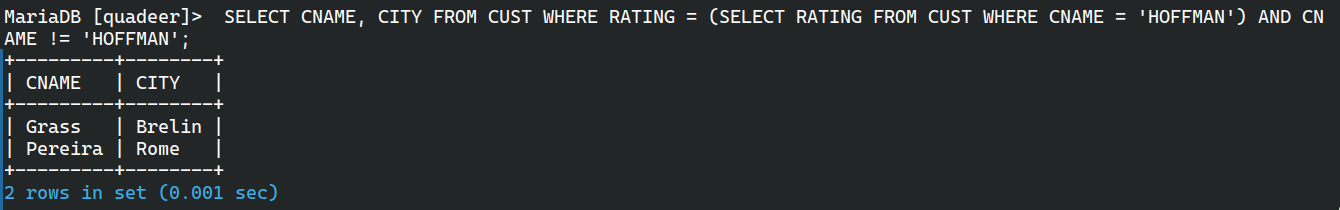
**41. PRODUCE ALL PAIRS OF SALESPEOPLE WHICH ARE LIVING IN THE SAME CITY. EXCLUDE COMBINATIONS OF SALESPEOPLE WITH THEMSELVES AS WELL AS DUPLICATES WITH THE ORDER REVERSED** SELECT A.SNAME, B.SNAME FROM SALESPEOPLE A, SALESPEOPLE B WHERE A.SNUM > B.SNUM AND A.CITY = B.CITY;



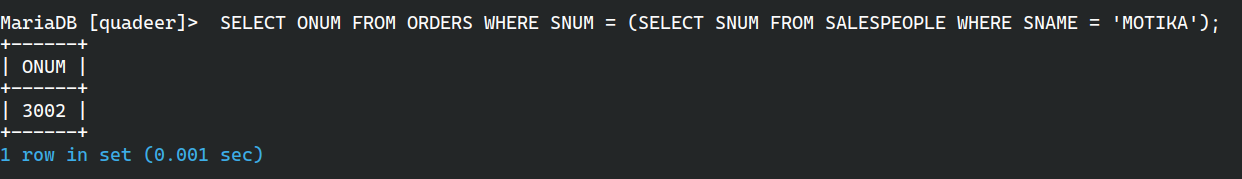
**42. PRODUCE ALL PAIRS OF ORDERS BY GIVEN CUSTOMER, NAMES THAT CUSTOMER AND ELIMINATES DUPLICATES** SELECT C.CNAME, A.ONUM, B.ONUM FROM ORDERS A, ORDERS B, CUST C WHERE A.CNUM = B.CNUM AND A.ONUM > B.ONUM AND C.CNUM = A.CNUM;



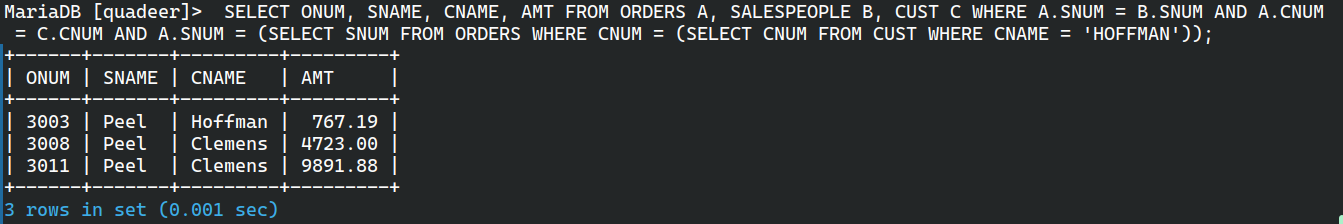
**43. PRODUCE NAMES AND CITIES OF ALL CUSTOMERS WITH THE SAME RATING AS HOFFMAN** SELECT CNAME, CITY FROM CUST WHERE RATING = (SELECT RATING FROM CUST WHERE CNAME = 'HOFFMAN') AND CNAME != 'HOFFMAN';



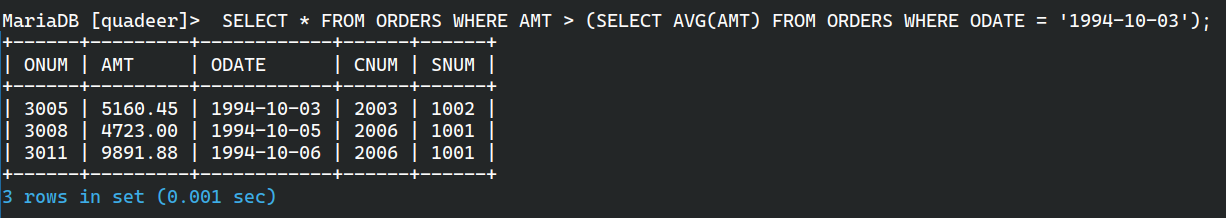
**44. EXTRACT ALL THE ORDERS OF MOTIKA** SELECT ONUM FROM ORDERS WHERE SNUM = (SELECT SNUM FROM SALESPEOPLE WHERE SNAME = 'MOTIKA');



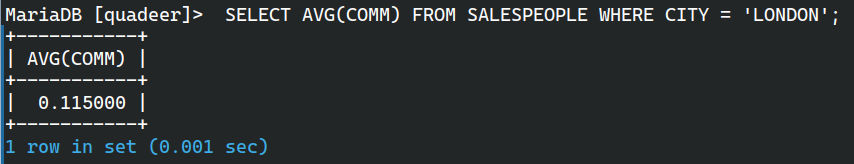
**45. ALL ORDERS CREDITED TO THE SAME SALESPERSON WHO SERVICES HOFFMAN** SELECT ONUM, SNAME, CNAME, AMT FROM ORDERS A, SALESPEOPLE B, CUST C WHERE A.SNUM = B.SNUM AND A.CNUM = C.CNUM AND A.SNUM = (SELECT SNUM FROM ORDERS WHERE CNUM = (SELECT CNUM FROM CUST WHERE CNAME = 'HOFFMAN'));



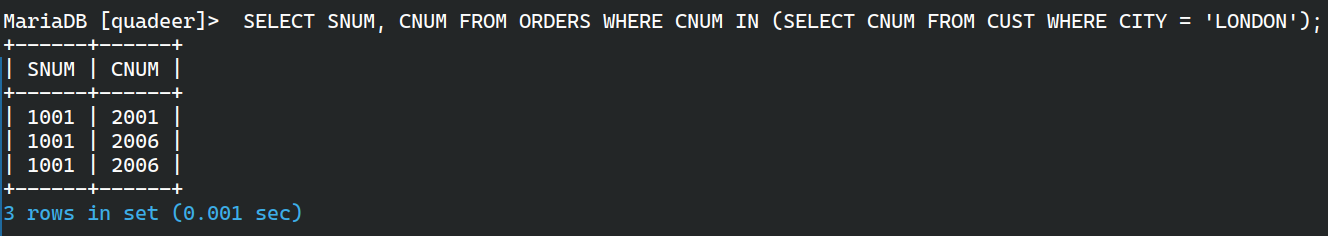
**46. ALL ORDERS THAT ARE GREATER THAN THE AVERAGE FOR OCT 4** SELECT \* FROM ORDERS WHERE AMT > (SELECT AVG(AMT) FROM ORDERS WHERE ODATE = '03-OCT-94');



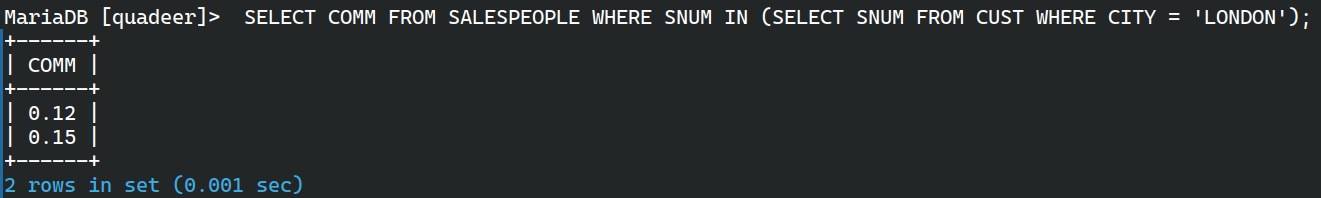
**47. FIND AVERAGE COMMISSION OF SALESPEOPLE IN LONDON** SELECT AVG(COMM) FROM SALESPEOPLE WHERE CITY = 'LONDON';



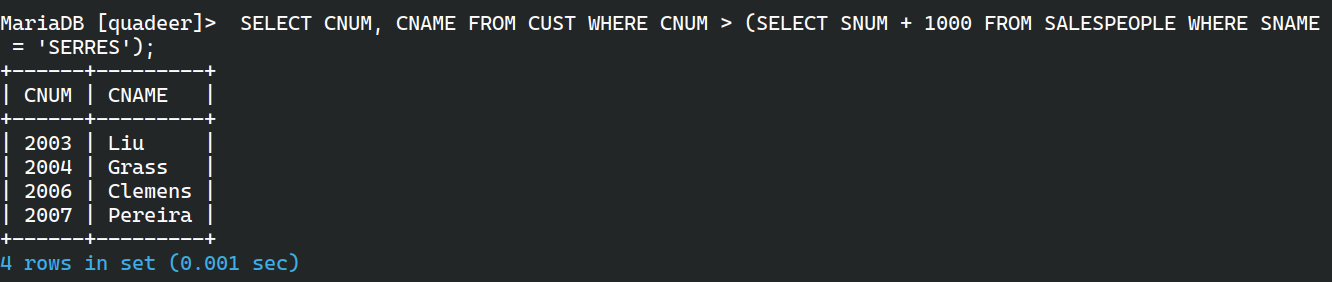
**48. FIND ALL ORDERS ATTRIBUTED TO SALESPEOPLE SERVICING CUSTOMERS IN LONDON** SELECT SNUM, CNUM FROM ORDERS WHERE CNUM IN (SELECT CNUM FROM CUST WHERE CITY = 'LONDON');



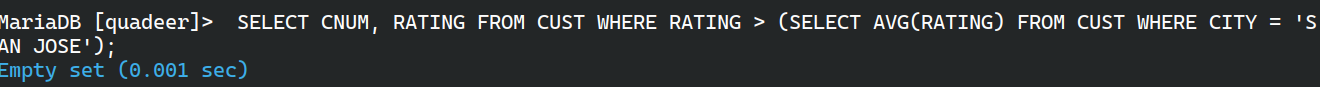
**49. EXTRACT COMMISSIONS OF ALL SALESPEOPLE SERVICING CUSTOMERS IN LONDON** SELECT COMM FROM SALESPEOPLE WHERE SNUM IN (SELECT SNUM FROM CUST WHERE CITY = 'LONDON');



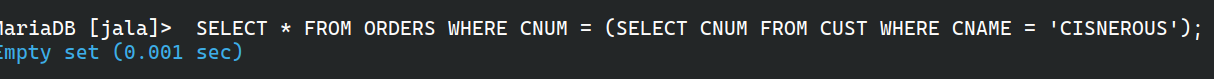
**50. FIND ALL CUSTOMERS WHOSE CNUM IS 1000 ABOVE THE SNUM OF SERRES** SELECT CNUM, CNAME FROM CUST WHERE CNUM > (SELECT SNUM + 1000 FROM SALESPEOPLE WHERE SNAME = 'SERRES');



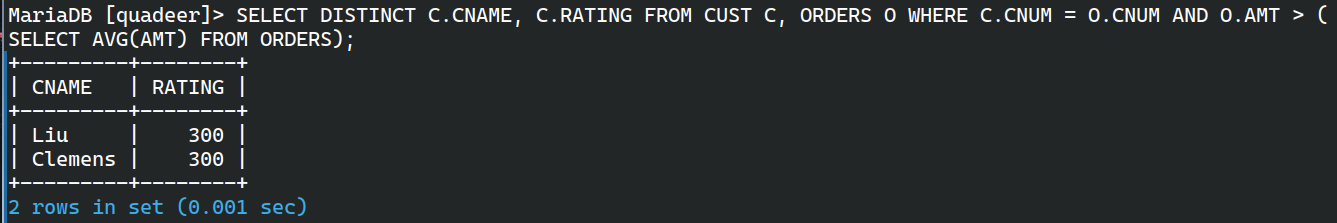
**51. COUNT THE CUSTOMERS WITH RATING ABOVE SAN JOSE’S AVERAGE** SELECT CNUM, RATING FROM CUST WHERE RATING > (SELECT AVG(RATING) FROM CUST WHERE CITY = 'SAN JOSE');



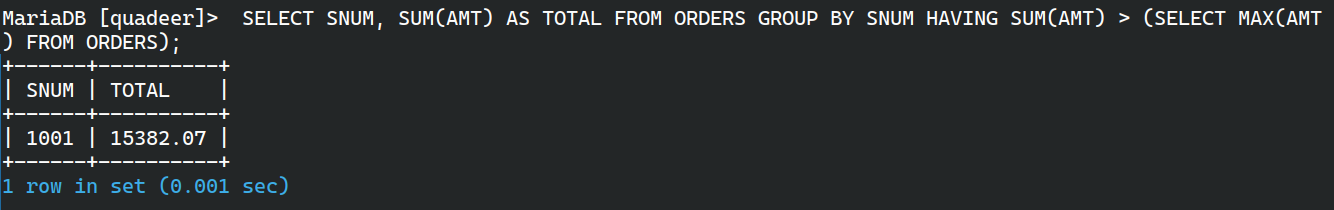
**52. OBTAIN ALL ORDERS FOR THE CUSTOMER NAMED CISNEROUS** SELECT \* FROM ORDERS WHERE CNUM = (SELECT CNUM FROM CUST WHERE CNAME = 'CISNEROUS');



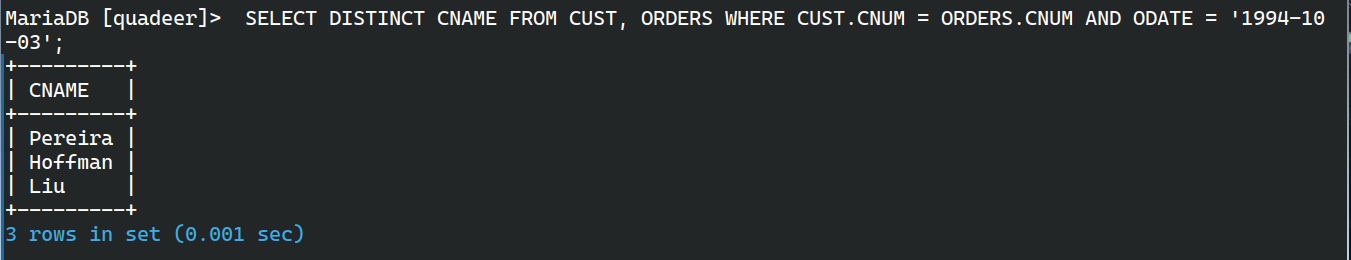
**53. PRODUCE THE NAMES AND RATING OF ALL CUSTOMERS WHO HAVE ABOVE AVERAGE ORDERS** SELECT DISTINCT C.CNAME, C.RATING FROM CUST C, ORDERS O WHERE C.CNUM = O.CNUM AND O.AMT > (SELECT AVG(AMT) FROM ORDERS);



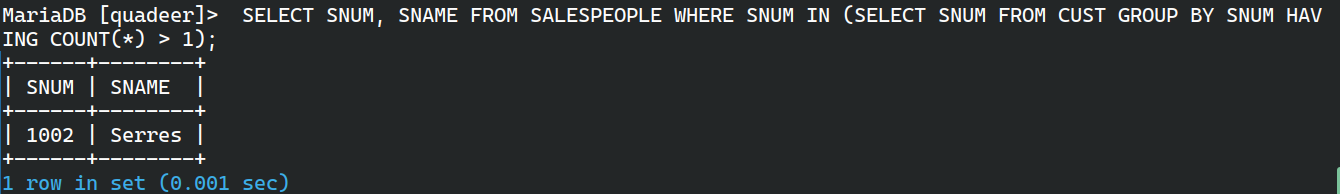
**54. FIND TOTAL AMOUNT IN ORDERS FOR EACH SALESPERSON FOR WHOM THIS TOTAL IS GREATER THAN THE AMOUNT OF THE LARGEST ORDER IN THE TABLE** SELECT SNUM, SUM(AMT) AS TOTAL FROM ORDERS GROUP BY SNUM HAVING SUM(AMT) > (SELECT MAX(AMT) FROM ORDERS);



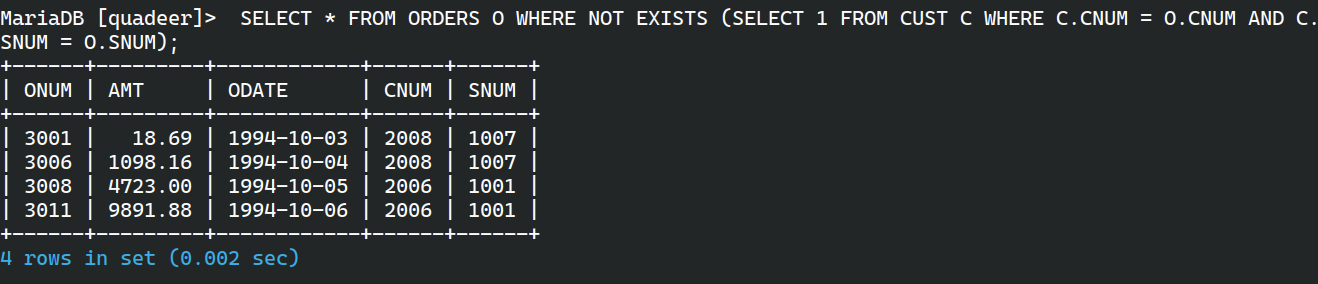
**55. FIND ALL CUSTOMERS WITH ORDER ON 3RD OCT** SELECT DISTINCT CNAME FROM CUST, ORDERS WHERE CUST.CNUM = ORDERS.CNUM AND ODATE = '03-OCT-94';



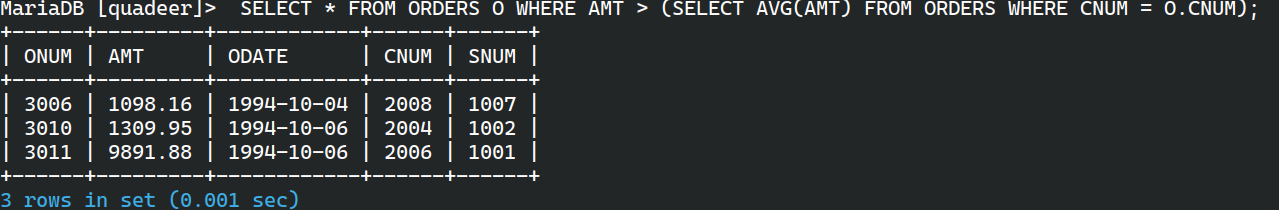
**56. FIND NAMES AND NUMBERS OF ALL SALESPERSON WHO HAVE MORE THAN ONE CUSTOMER** SELECT SNUM, SNAME FROM SALESPEOPLE WHERE SNUM IN (SELECT SNUM FROM CUST GROUP BY SNUM HAVING COUNT(\*) > 1);



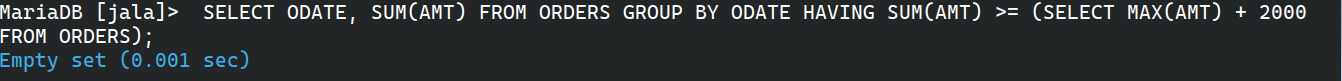
**57. CHECK IF THE CORRECT SALESPERSON WAS CREDITED WITH EACH SALE** SELECT \* FROM ORDERS O WHERE NOT EXISTS (SELECT 1 FROM CUST C WHERE C.CNUM = O.CNUM AND C.SNUM = O.SNUM);



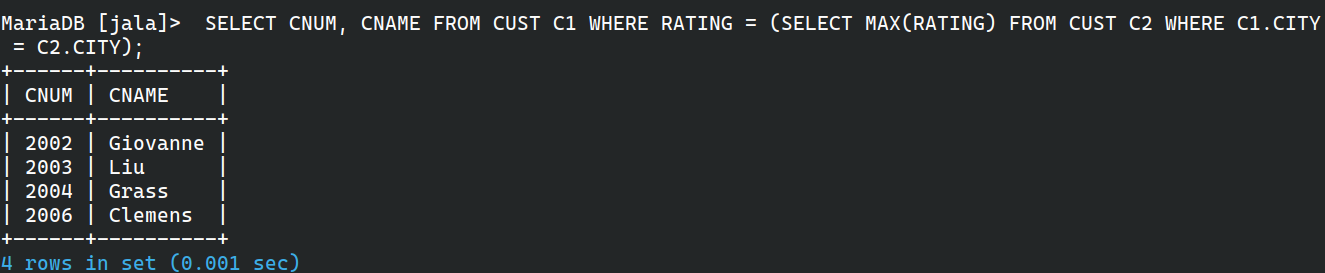
**58. FIND ALL ORDERS WITH ABOVE AVERAGE AMOUNTS FOR THEIR CUSTOMERS** SELECT \* FROM ORDERS O WHERE AMT > (SELECT AVG(AMT) FROM ORDERS WHERE CNUM = O.CNUM);

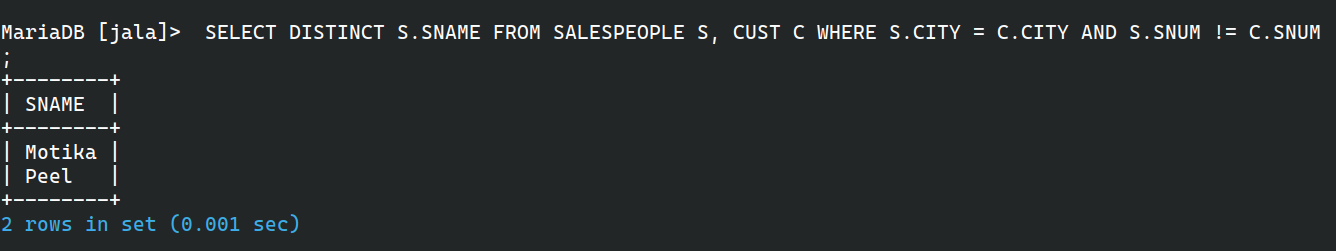


**59. FIND THE SUMS OF THE AMOUNTS FROM ORDER TABLE GROUPED BY DATE, ELIMINATING ALL THOSE DATES WHERE THE SUM WAS NOT AT LEAST 2000 ABOVE THE MAXIMUM AMOUNT** SELECT ODATE, SUM(AMT) FROM ORDERS GROUP BY ODATE HAVING SUM(AMT) >= (SELECT MAX(AMT) + 2000 FROM ORDERS);

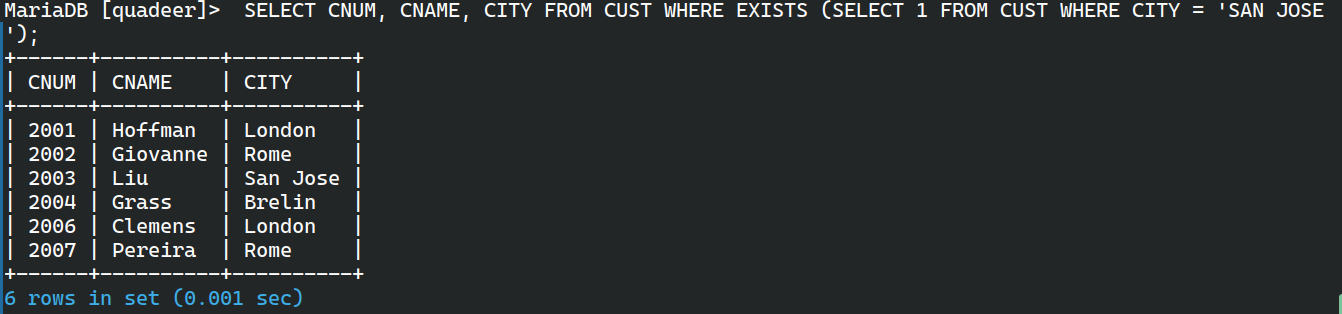


**60. FIND NAMES AND NUMBERS OF ALL CUSTOMERS WITH RATINGS EQUAL TO THE MAXIMUM FOR THEIR CITY** SELECT CNUM, CNAME FROM CUST C1 WHERE RATING = (SELECT MAX(RATING) FROM CUST C2 WHERE C1.CITY = C2.CITY);

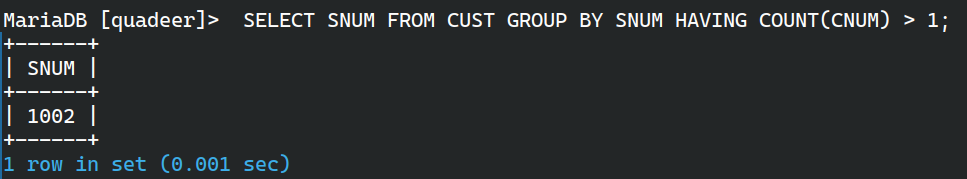


**61. FIND ALL SALESPEOPLE WHO HAVE CUSTOMERS IN THEIR CITIES WHO THEY DON’T SERVICE (JOIN & CORRELATED SUBQUERY)** SELECT DISTINCT S.SNAME FROM SALESPEOPLE S, CUST C WHERE S.CITY = C.CITY AND S.SNUM != C.SNUM;  


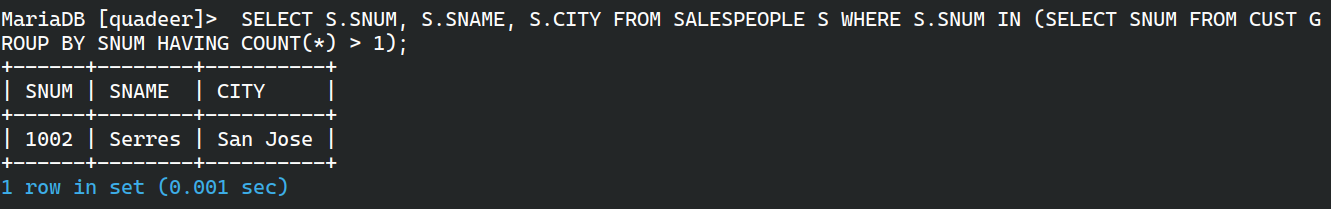
**62. EXTRACT CNUM, CNAME AND CITY FROM CUSTOMER TABLE IF AND ONLY IF ONE OR MORE OF THE CUSTOMERS ARE LOCATED IN SAN JOSE** SELECT CNUM, CNAME, CITY FROM CUST WHERE EXISTS (SELECT 1 FROM CUST WHERE CITY = 'SAN JOSE');



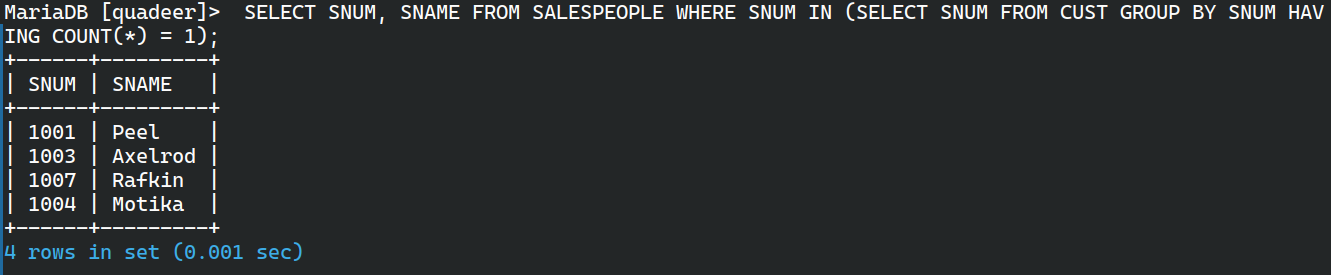
**63. FIND SALESPEOPLE NO. WHO HAVE MULTIPLE CUSTOMERS** SELECT SNUM FROM CUST GROUP BY SNUM HAVING COUNT(CNUM) > 1;



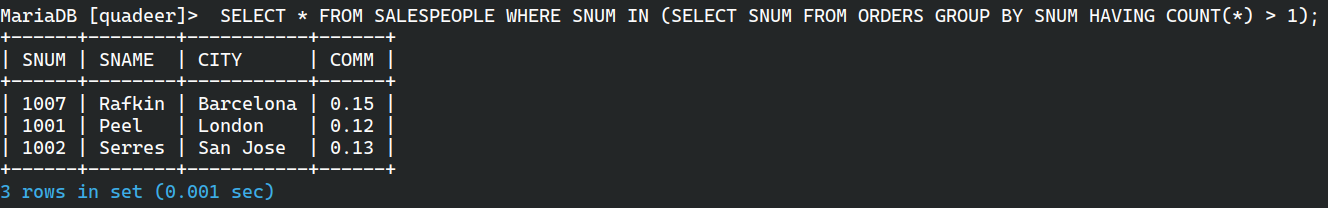
**64. FIND SALESPEOPLE NUMBER, NAME AND CITY WHO HAVE MULTIPLE CUSTOMERS** SELECT S.SNUM, S.SNAME, S.CITY FROM SALESPEOPLE S WHERE S.SNUM IN (SELECT SNUM FROM CUST GROUP BY SNUM HAVING COUNT(\*) > 1);



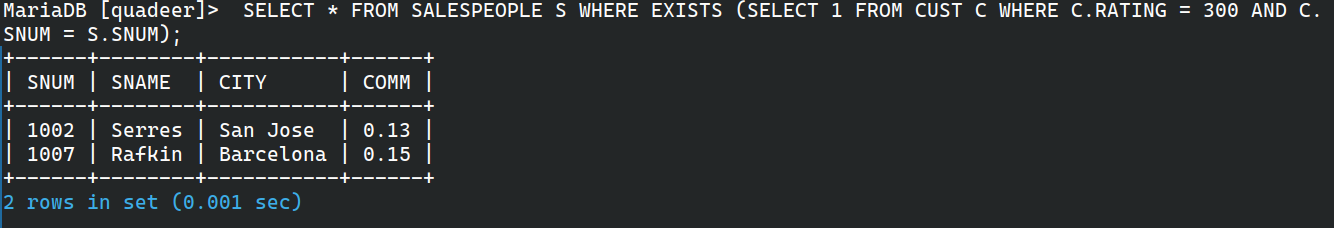
**65. FIND SALESPEOPLE WHO SERVE ONLY ONE CUSTOMER** SELECT SNUM, SNAME FROM SALESPEOPLE WHERE SNUM IN (SELECT SNUM FROM CUST GROUP BY SNUM HAVING COUNT(\*) = 1);



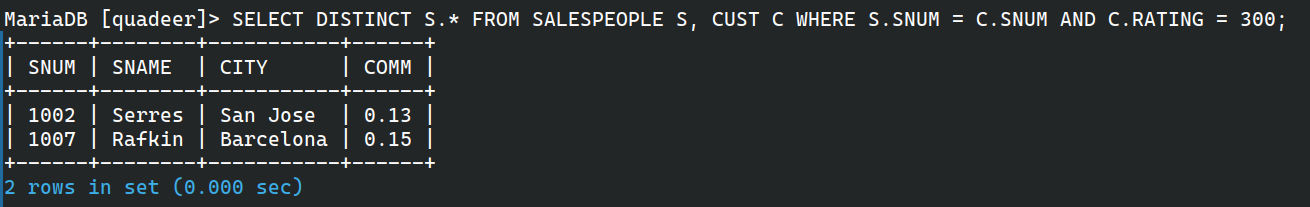
**66. EXTRACT ROWS OF ALL SALESPEOPLE WITH MORE THAN ONE CURRENT ORDER** SELECT \* FROM SALESPEOPLE WHERE SNUM IN (SELECT SNUM FROM ORDERS GROUP BY SNUM HAVING COUNT(\*) > 1);



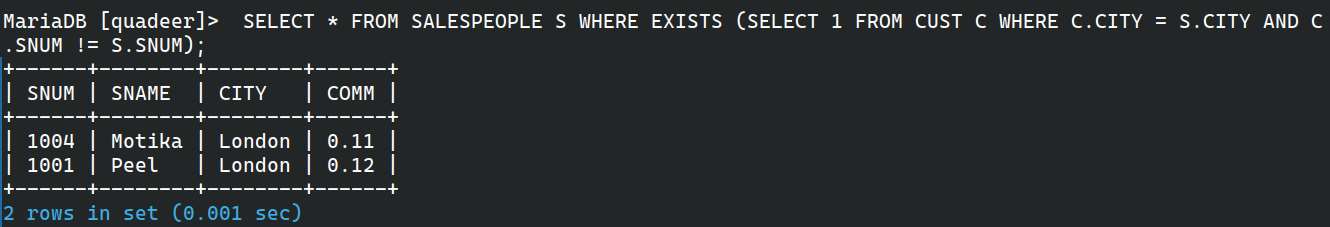
**67. FIND ALL SALESPEOPLE WHO HAVE CUSTOMERS WITH A RATING OF 300 (EXISTS)** SELECT \* FROM SALESPEOPLE S WHERE EXISTS (SELECT 1 FROM CUST C WHERE C.RATING = 300 AND C.SNUM = S.SNUM);



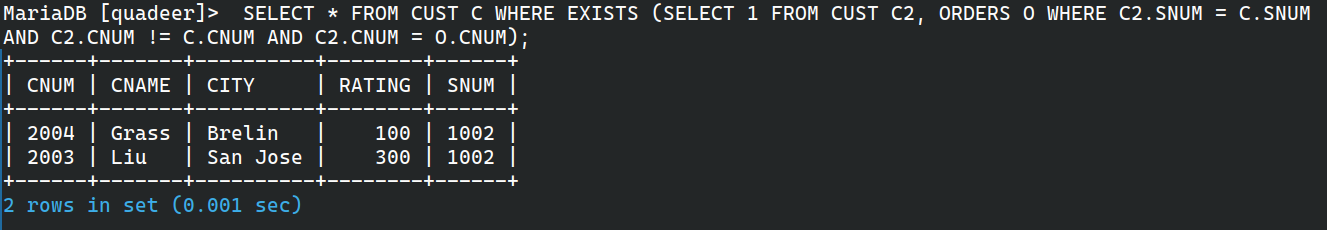
**68. FIND ALL SALESPEOPLE WHO HAVE CUSTOMERS WITH A RATING OF 300 (JOIN)** SELECT DISTINCT S.\* FROM SALESPEOPLE S, CUST C WHERE S.SNUM = C.SNUM AND C.RATING = 300;



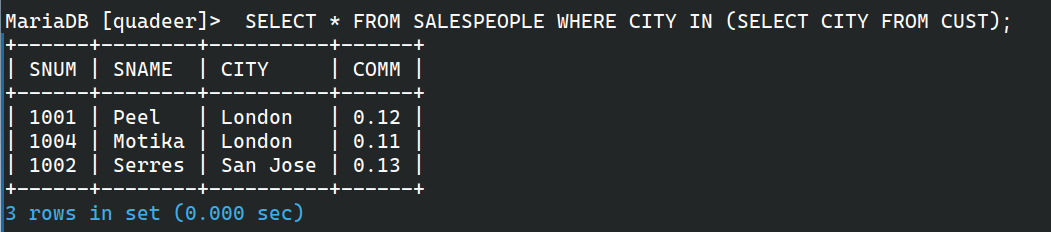
**69. SELECT ALL SALESPEOPLE WITH CUSTOMERS LOCATED IN THEIR CITIES WHO ARE NOT ASSIGNED TO THEM (EXISTS)** SELECT \* FROM SALESPEOPLE S WHERE EXISTS (SELECT 1 FROM CUST C WHERE C.CITY = S.CITY AND C.SNUM != S.SNUM);



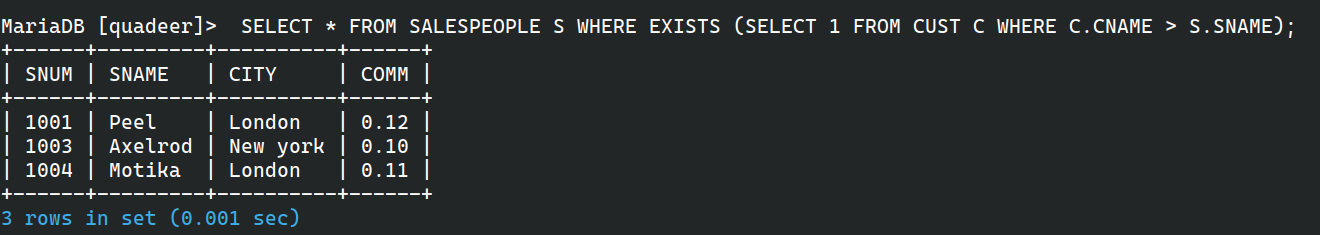
**70. EXTRACT FROM CUSTOMERS TABLE EVERY CUSTOMER ASSIGNED TO A SALESPERSON WHO CURRENTLY HAS AT LEAST ONE OTHER CUSTOMER WITH ORDERS** SELECT \* FROM CUST C WHERE EXISTS (SELECT 1 FROM CUST C2, ORDERS O WHERE C2.SNUM = C.SNUM AND C2.CNUM != C.CNUM AND C2.CNUM = O.CNUM);



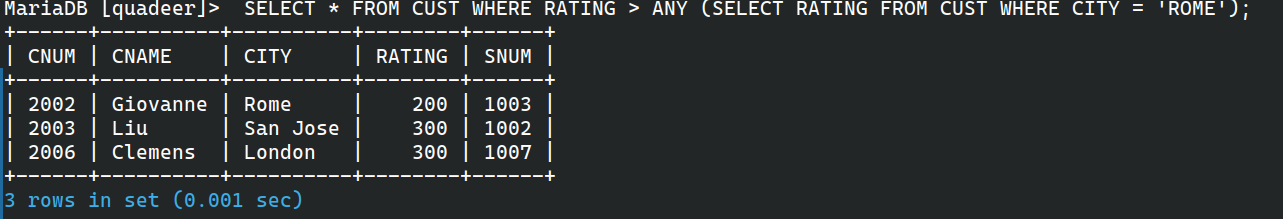
**71. FIND SALESPEOPLE WITH CUSTOMERS LOCATED IN THEIR CITIES (USING BOTH ANY AND IN)** SELECT \* FROM SALESPEOPLE WHERE CITY = ANY (SELECT CITY FROM CUST);  
 SELECT \* FROM SALESPEOPLE WHERE CITY IN (SELECT CITY FROM CUST);



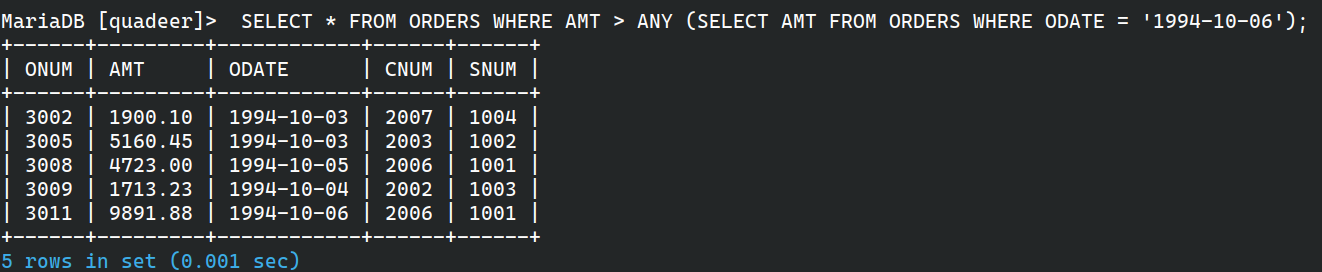
**72. FIND ALL SALESPEOPLE FOR WHOM THERE ARE CUSTOMERS THAT FOLLOW THEM IN ALPHABETICAL ORDER (ANY AND EXISTS)** SELECT \* FROM SALESPEOPLE S WHERE EXISTS (SELECT 1 FROM CUST C WHERE C.CNAME > S.SNAME);



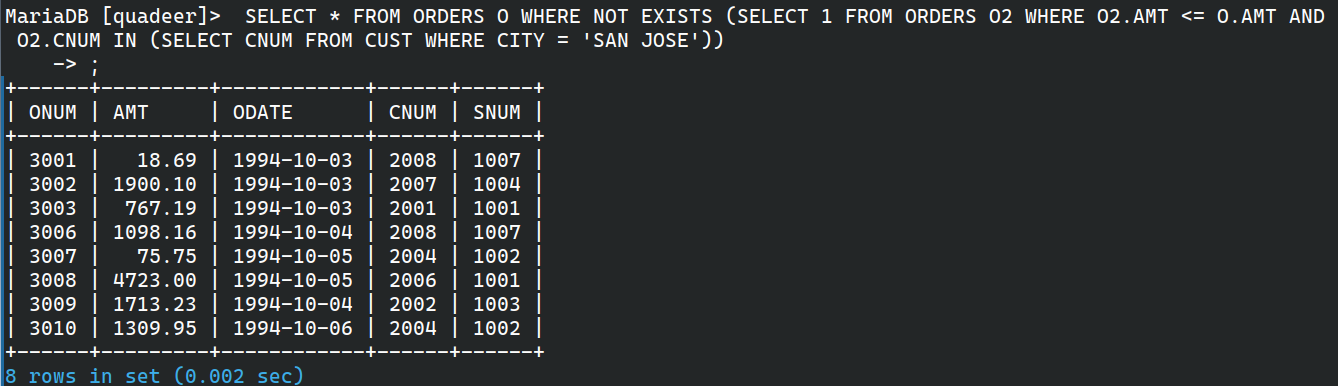
**73. SELECT CUSTOMERS WHO HAVE A GREATER RATING THAN ANY CUSTOMER IN ROME** SELECT \* FROM CUST WHERE RATING > ANY (SELECT RATING FROM CUST WHERE CITY = 'ROME');



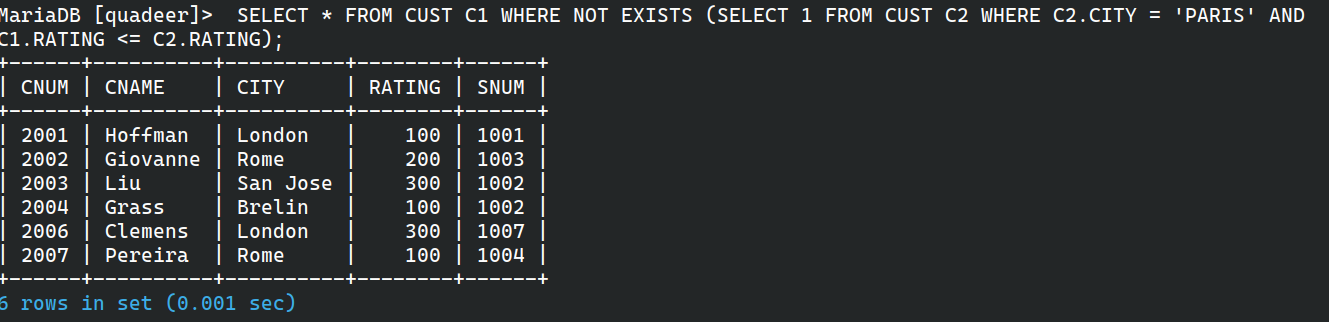
**74. SELECT ALL ORDERS THAT HAD AMOUNTS THAT WERE GREATER THAN AT LEAST ONE OF THE ORDERS FROM OCT 6TH** SELECT \* FROM ORDERS WHERE AMT > ANY (SELECT AMT FROM ORDERS WHERE ODATE = '06-OCT-94');



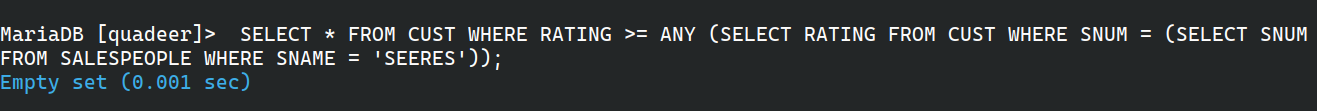
**75. FIND ALL ORDERS WITH AMOUNTS SMALLER THAN ANY AMOUNT FOR A CUSTOMER IN SAN JOSE (ANY AND WITHOUT ANY)** SELECT \* FROM ORDERS WHERE AMT < ANY (SELECT AMT FROM ORDERS WHERE CNUM IN (SELECT CNUM FROM CUST WHERE CITY = 'SAN JOSE'));  
 SELECT \* FROM ORDERS O WHERE NOT EXISTS (SELECT 1 FROM ORDERS O2 WHERE O2.AMT <= O.AMT AND O2.CNUM IN (SELECT CNUM FROM CUST WHERE CITY = 'SAN JOSE'));



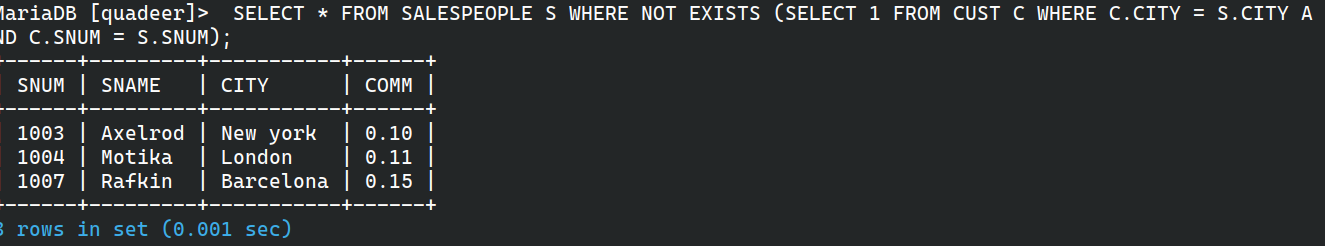
**76. SELECT CUSTOMERS WHOSE RATINGS ARE HIGHER THAN EVERY CUSTOMER IN PARIS (ALL AND NOT EXISTS)** SELECT \* FROM CUST WHERE RATING > ALL (SELECT RATING FROM CUST WHERE CITY = 'PARIS');  
 SELECT \* FROM CUST C1 WHERE NOT EXISTS (SELECT 1 FROM CUST C2 WHERE C2.CITY = 'PARIS' AND C1.RATING <= C2.RATING);



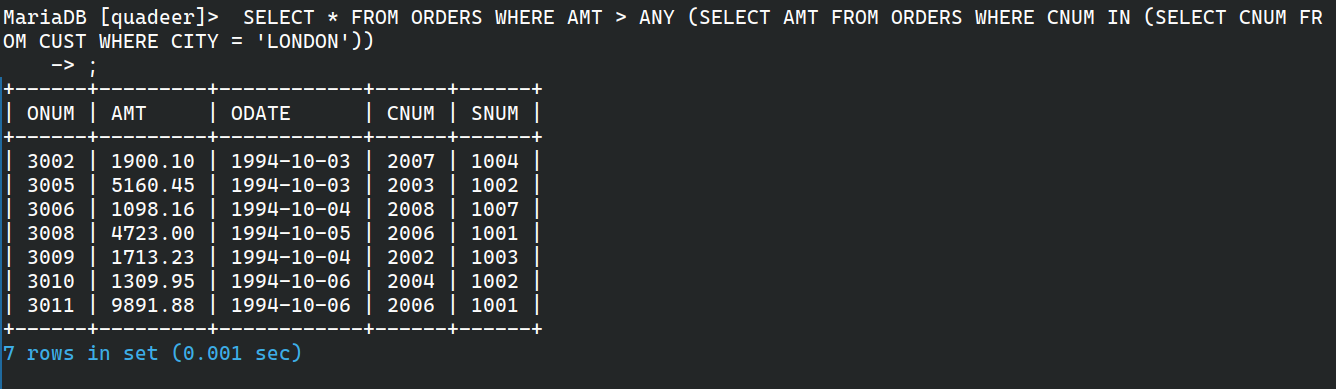
**77. SELECT CUSTOMERS WHOSE RATINGS ARE EQUAL TO OR GREATER THAN ANY OF THE SEERES** SELECT \* FROM CUST WHERE RATING >= ANY (SELECT RATING FROM CUST WHERE SNUM = (SELECT SNUM FROM SALESPEOPLE WHERE SNAME = 'SEERES'));



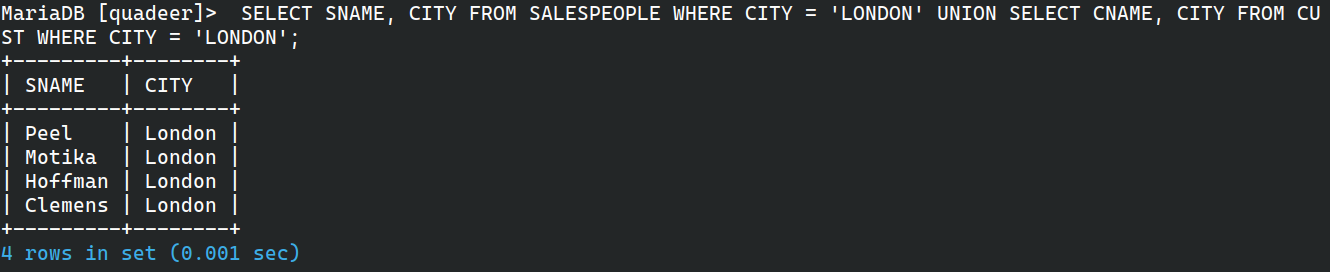
**78. FIND SALESPEOPLE WHO HAVE NO CUSTOMERS IN THEIR CITY (ANY AND ALL)** SELECT \* FROM SALESPEOPLE WHERE CITY != ANY (SELECT CITY FROM CUST WHERE CUST.SNUM = SALESPEOPLE.SNUM);  
 SELECT \* FROM SALESPEOPLE S WHERE NOT EXISTS (SELECT 1 FROM CUST C WHERE C.CITY = S.CITY AND C.SNUM = S.SNUM);



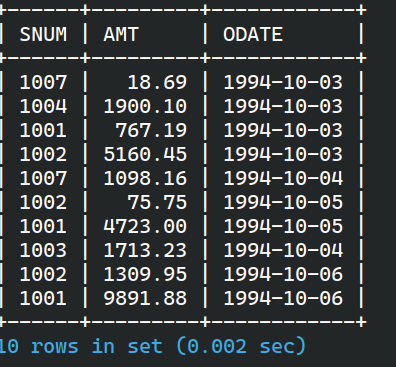
**79. FIND ALL ORDERS FOR AMOUNTS GREATER THAN ANY FOR CUSTOMERS IN LONDON** SELECT \* FROM ORDERS WHERE AMT > ANY (SELECT AMT FROM ORDERS WHERE CNUM IN (SELECT CNUM FROM CUST WHERE CITY = 'LONDON'));



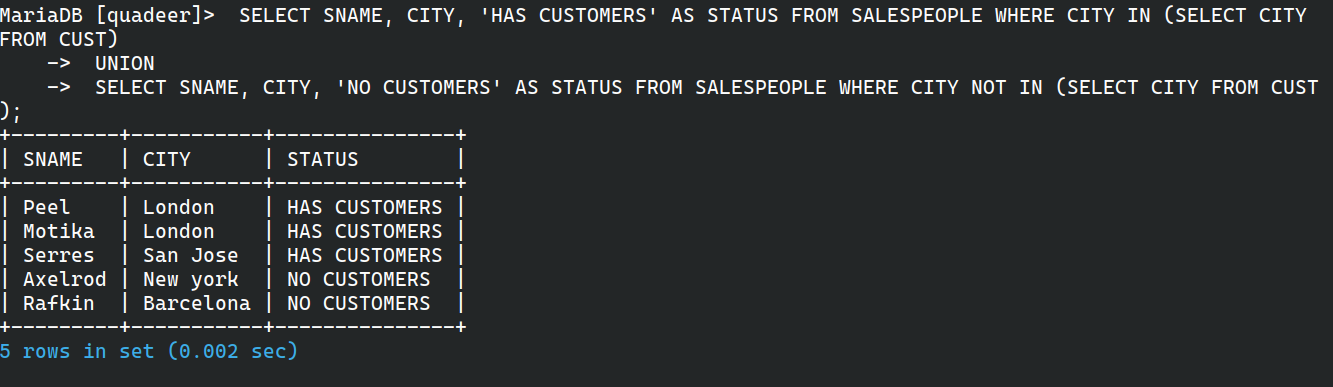
**80. FIND ALL SALESPEOPLE AND CUSTOMERS LOCATED IN LONDON** SELECT SNAME, CITY FROM SALESPEOPLE WHERE CITY = 'LONDON' UNION SELECT CNAME, CITY FROM CUST WHERE CITY = 'LONDON';



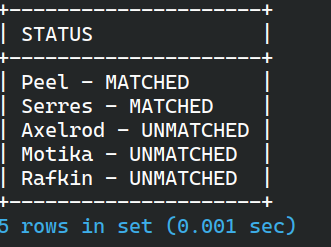
**81. FOR EVERY SALESPERSON, DATES ON WHICH HIGHEST AND LOWEST ORDERS WERE BROUGHT** SELECT SNUM, MAX(AMT), ODATE FROM ORDERS GROUP BY SNUM, ODATE HAVING AMT = MAX(AMT)  
 UNION  
 SELECT SNUM, MIN(AMT), ODATE FROM ORDERS GROUP BY SNUM, ODATE HAVING AMT = MIN(AMT);



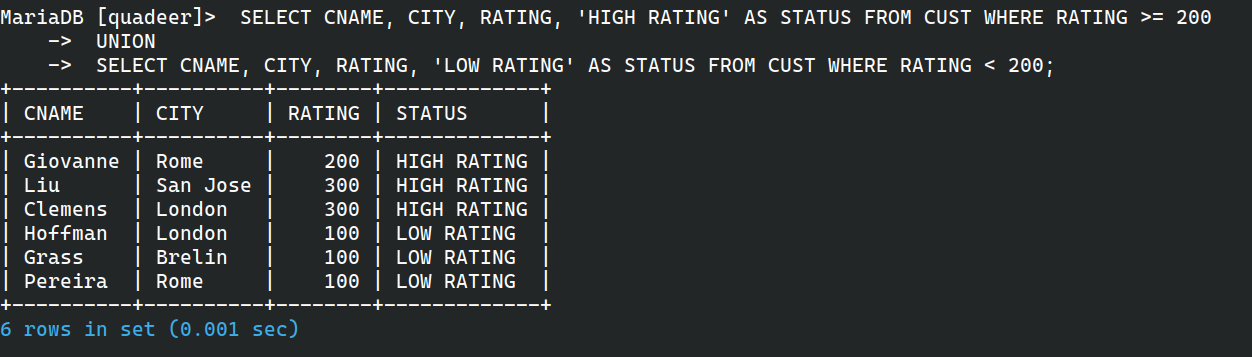
**82. LIST ALL SALESPEOPLE AND INDICATE THOSE WHO DON’T HAVE CUSTOMERS IN THEIR CITIES AND THOSE WHO DO** SELECT SNAME, CITY, 'HAS CUSTOMERS' AS STATUS FROM SALESPEOPLE WHERE CITY IN (SELECT CITY FROM CUST)  
 UNION  
 SELECT SNAME, CITY, 'NO CUSTOMERS' AS STATUS FROM SALESPEOPLE WHERE CITY NOT IN (SELECT CITY FROM CUST);



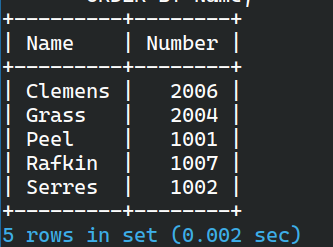
**83. APPEND STRINGS TO THE SELECTED FIELDS INDICATING WHETHER A SALESPERSON WAS MATCHED TO A CUSTOMER IN HIS CITY** SELECT SNAME || ' - ' || CASE WHEN CITY IN (SELECT CITY FROM CUST WHERE CUST.SNUM = SALESPEOPLE.SNUM) THEN 'MATCHED' ELSE 'UNMATCHED' END AS STATUS FROM SALESPEOPLE;



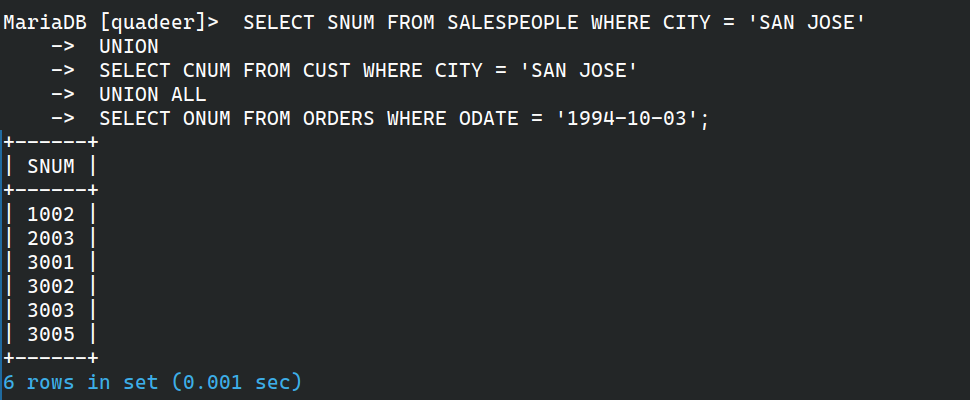
**84. CREATE A UNION OF TWO QUERIES THAT SHOW THE NAMES, CITIES AND RATINGS OF ALL CUSTOMERS, WITH 'HIGH RATING' OR 'LOW RATING'** SELECT CNAME, CITY, RATING, 'HIGH RATING' AS STATUS FROM CUST WHERE RATING >= 200  
 UNION  
 SELECT CNAME, CITY, RATING, 'LOW RATING' AS STATUS FROM CUST WHERE RATING < 200;



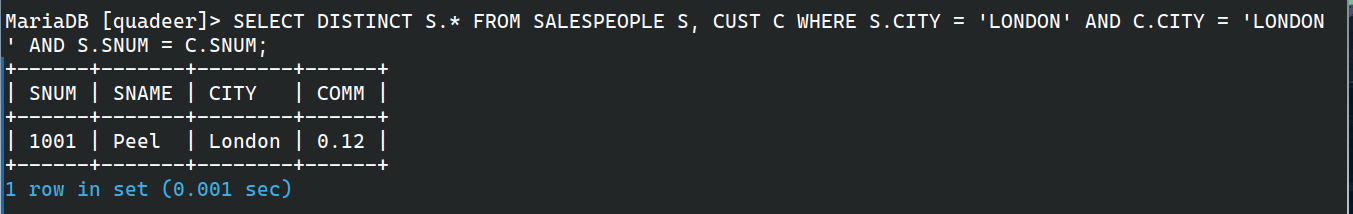
**85. NAME AND NUMBER OF EACH SALESPERSON AND CUSTOMER WITH MORE THAN ONE CURRENT ORDER, IN ALPHABETICAL ORDER** SELECT S.SNAME, S.SNUM FROM SALESPEOPLE S WHERE S.SNUM IN (SELECT SNUM FROM ORDERS GROUP BY SNUM HAVING COUNT(*) > 1)  
 UNION  
 SELECT C.CNAME, C.CNUM FROM CUST C WHERE C.CNUM IN (SELECT CNUM FROM ORDERS GROUP BY CNUM HAVING COUNT(*) > 1) ORDER BY 1;



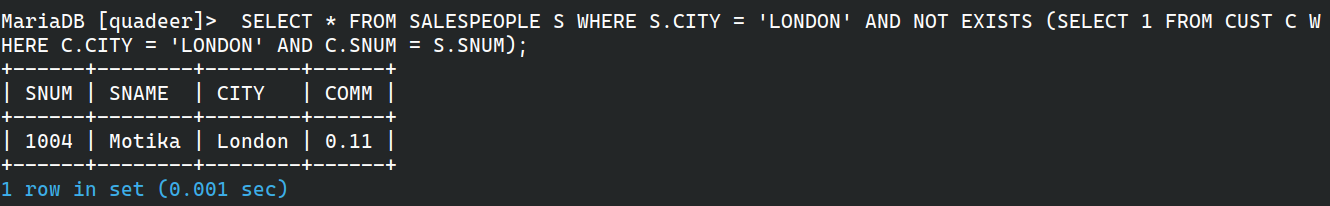
**86. UNION OF THREE QUERIES FOR SALESPEOPLE, CUSTOMERS, AND ORDERS IN SAN JOSE/OCT 3** SELECT SNUM FROM SALESPEOPLE WHERE CITY = 'SAN JOSE'  
 UNION  
 SELECT CNUM FROM CUST WHERE CITY = 'SAN JOSE'  
 UNION ALL  
 SELECT ONUM FROM ORDERS WHERE ODATE = '03-OCT-94';



**87. ALL SALESPEOPLE IN LONDON WITH AT LEAST ONE CUSTOMER THERE** SELECT DISTINCT S.\* FROM SALESPEOPLE S, CUST C WHERE S.CITY = 'LONDON' AND C.CITY = 'LONDON' AND S.SNUM = C.SNUM;



**88. ALL SALESPEOPLE IN LONDON WHO DID NOT HAVE CUSTOMERS THERE** SELECT \* FROM SALESPEOPLE S WHERE S.CITY = 'LONDON' AND NOT EXISTS (SELECT 1 FROM CUST C WHERE C.CITY = 'LONDON' AND C.SNUM = S.SNUM);



**89. SALESPEOPLE MATCHED TO CUSTOMERS INCLUDING THOSE WITHOUT CUSTOMERS (OUTER JOIN + UNION)** SELECT S.SNAME, C.CNAME FROM SALESPEOPLE S LEFT JOIN CUST C ON S.SNUM = C.SNUM  
 UNION  
 SELECT S.SNAME, NULL FROM SALESPEOPLE S WHERE S.SNUM NOT IN (SELECT DISTINCT SNUM FROM CUST);

