

**COURSE TITLE: DATABASE SYSTEMS LAB COURSE CODE: BCSE302P**

**EXPERIMENT NO: 6**

SQL JOINS

**NAME:** PRIYANSHU SONI **REGISTER NO:** 21BRS1629 **SLOT:** L21+L22

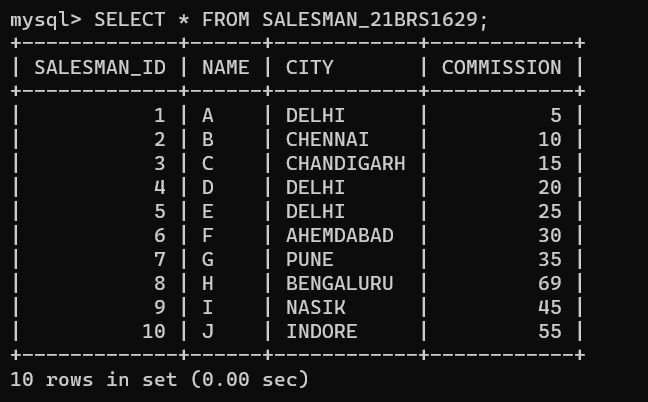
**CLASS NUMBER:** CH2022232501096 **DATE:** 15-APRIL, 2023 **FACULTY NAME:** L. JANI ANBARASI

Create three tables: salesman, customer and orders.

* SALESMAN( )

## CODE:

CREATE TABLE SALESMAN\_21BRS1629(SALESMAN\_ID INT PRIMARY KEY, NAME VARCHAR(15), CITY VARCHAR(15), COMMISSION INT);



* CUSTOMER(CUSTOMER\_ID, CUST\_NAME, CITY, GRADE, SALESMAN\_ID)

## CODE:

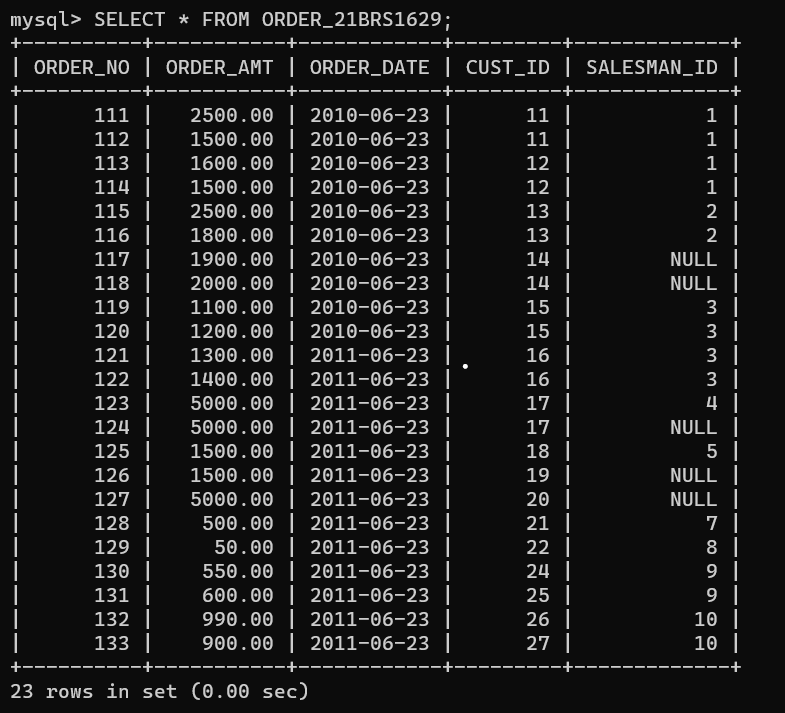
CREATE TABLE CUSTOMER\_21BRS1629(CUSTOMER\_ID INT PRIMARY KEY, CUST\_NAME VARCHAR(15), CITY VARCHAR(15), GRADE INT, SALESMAN\_ID INT,FOREIGN KEY (SALESMAN\_ID) REFERENCES SALESMAN\_21BRS1629(SALESMAN\_ID));



* ORDERS(ORDER\_NO, ORDER\_AMT, ORDER\_DATE, CUST\_ID, SALESMAN\_ID)

## CODE:

CREATE TABLE ORDER\_21BRS1629(ORDER\_NO INT PRIMARY KEY, ORDER\_AMT DECIMAL(10,2), ORDER\_DATE DATE, CUST\_ID INT, SALESMAN\_ID INT, FOREIGN KEY (SALESMAN\_ID) REFERENCES SALESMAN\_21BRS1629 (SALESMAN\_ID) ,FOREIGN KEY(CUST\_ID) REFERENCES CUSTOMER\_21BRS1629(CUSTOMER\_ID) );

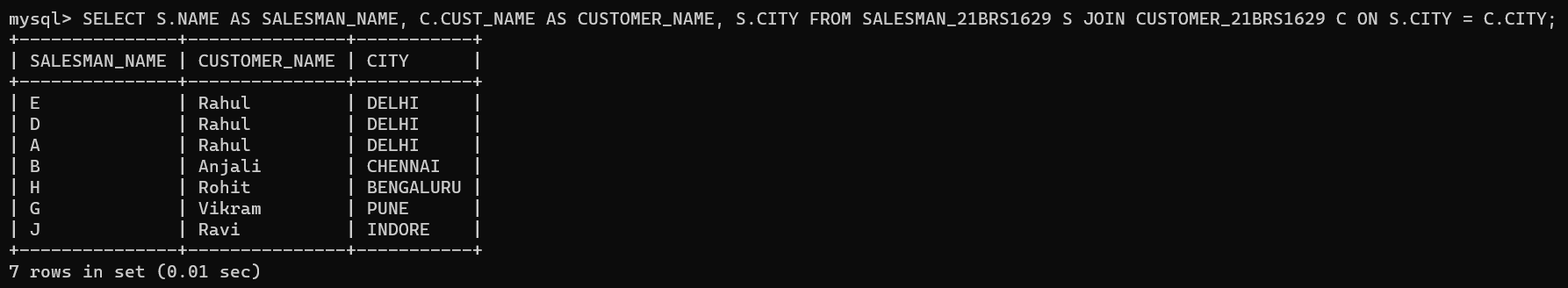


# Prepare a list with salesman name, customer name and their cities for the salesmen and customer who belongs to the same city.

**CODE:**

SELECT S.NAME AS SALESMAN\_NAME, C.CUST\_NAME AS CUSTOMER\_NAME, S.CITY FROM SALESMAN\_21BRS1629 S JOIN CUSTOMER\_21BRS1629 C ON S.CITY = C.CITY;

**OUTPUT**:



1. **Make a list with order no, purchase amount, customer name and their cities for those orders which order amount between 500 and 2000.**

# CODE:

SELECT O.ORDER\_NO, O.ORDER\_AMT, C.CUST\_NAME, C.CITY FROM ORDER\_21BRS1629 O JOIN CUSTOMER\_21BRS1629 C ON O.CUST\_ID = C.CUSTOMER\_ID WHERE O.ORDER\_AMT BETWEEN 500 AND 2000;

**OUTPUT**:

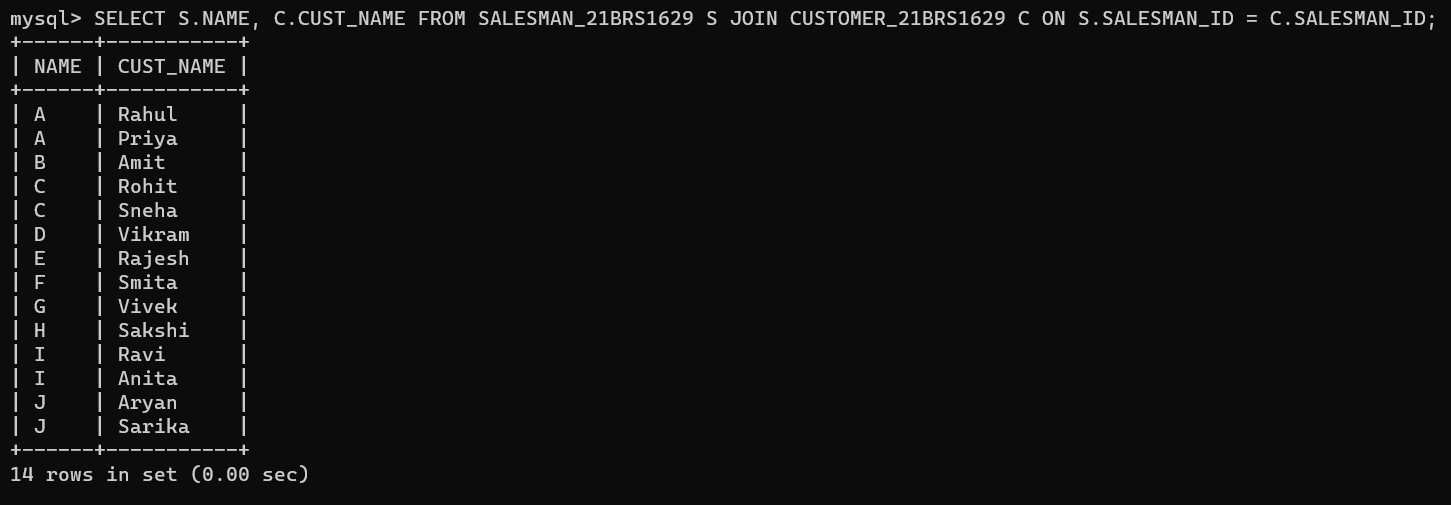


1. **Know which salesman is working for which customer.**

# CODE:

SELECT S.NAME, C.CUST\_NAME FROM SALESMAN\_21BRS1629 S JOIN CUSTOMER\_21BRS1629 C ON S.SALESMAN\_ID = C.SALESMAN\_ID;

**OUTPUT**:



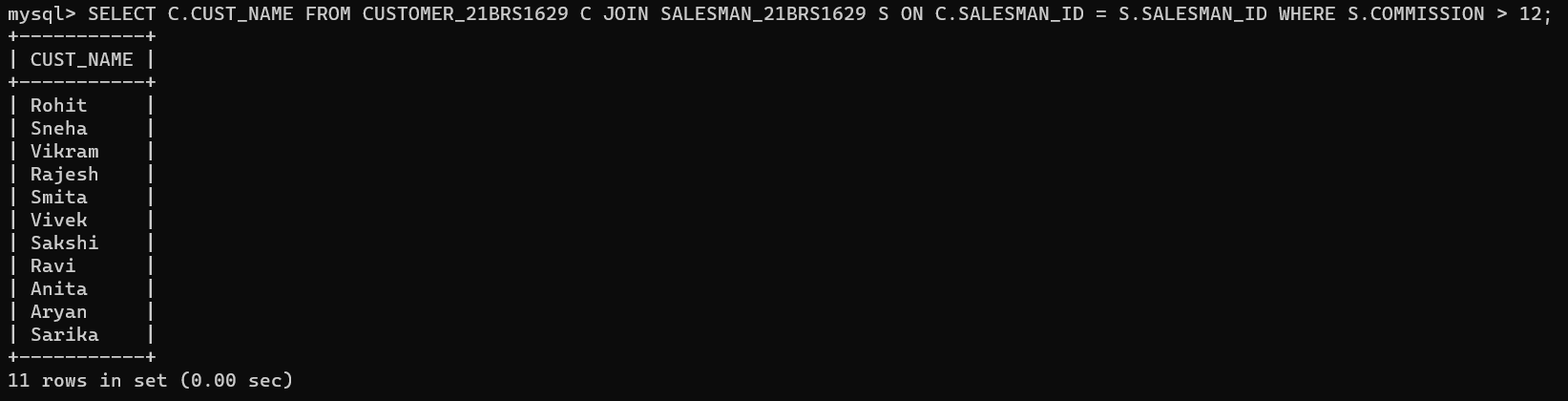
1. **Find the list of customers who appointed a salesman for their jobs who gets a commission from the company is more than 12%.**

# CODE:

SELECT C.CUST\_NAME FROM CUSTOMER\_21BRS1629 C JOIN SALESMAN

\_21BRS1629 S ON C.SALESMAN\_ID = S.SALESMAN\_ID WHERE S.COMMISSION > 12;

**OUTPUT**:



1. **Find the list of customers who appointed a salesman for their jobs who does not live in the same city where their customer lives, and gets a commission is above 12%.**

# CODE:

SELECT C.CUST\_NAME FROM CUSTOMER\_21BRS1629 C JOIN SALESMAN\_21BRS1629 S ON C.SALESMAN\_ID = S.SALESMAN\_ID WHERE S.CITY != C.CITY AND S.COMMISSION > 12;

**OUTPUT**:

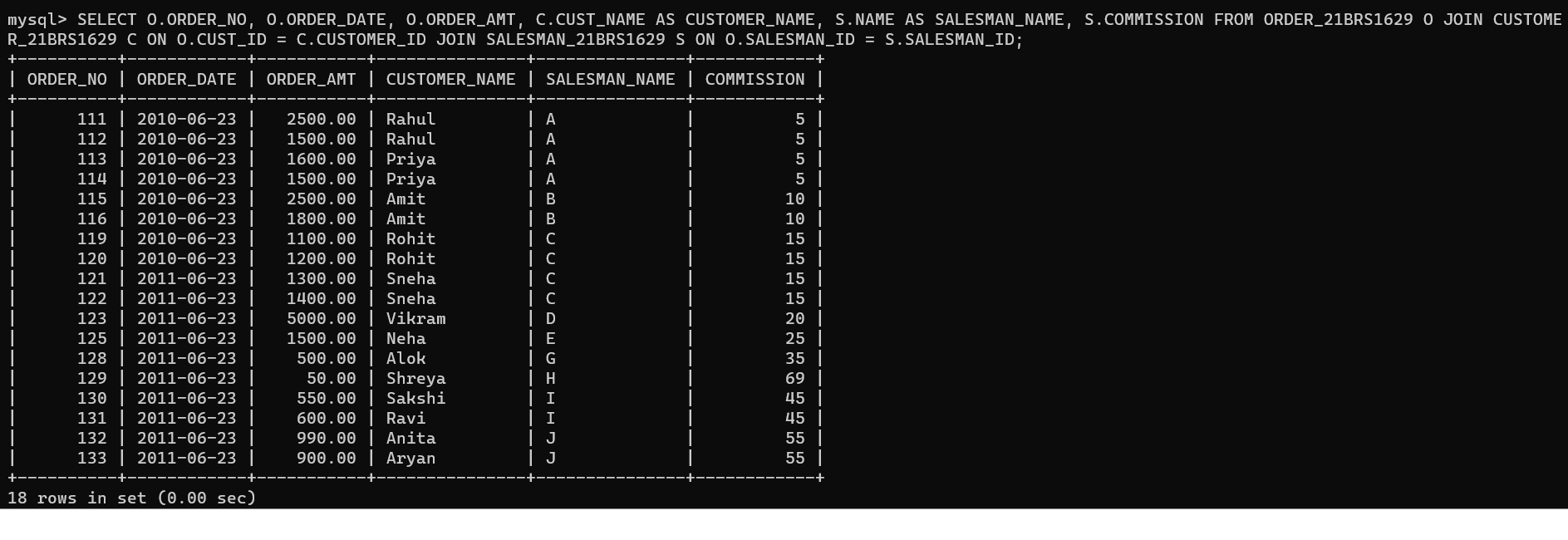


1. **Find the details of an order i.e. order number, order date, amount of order, which customer gives the order and which salesman works for that customer and how much commission he gets for an order.**

# CODE:

SELECT O.ORDER\_NO, O.ORDER\_DATE, O.ORDER\_AMT, C.CUST\_NAME AS CUSTOMER\_NAME, S.NAME AS SALESMAN\_NAME, S.COMMISSION FROM ORDER\_21BRS1629 O JOIN CUSTOMER\_21BRS1629 C ON O.CUST\_ID = C.CUSTOMER\_ID JOIN SALESMAN\_21BRS1629 S ON O.SALESMAN\_ID = S.SALESMAN\_ID;

**OUTPUT**:

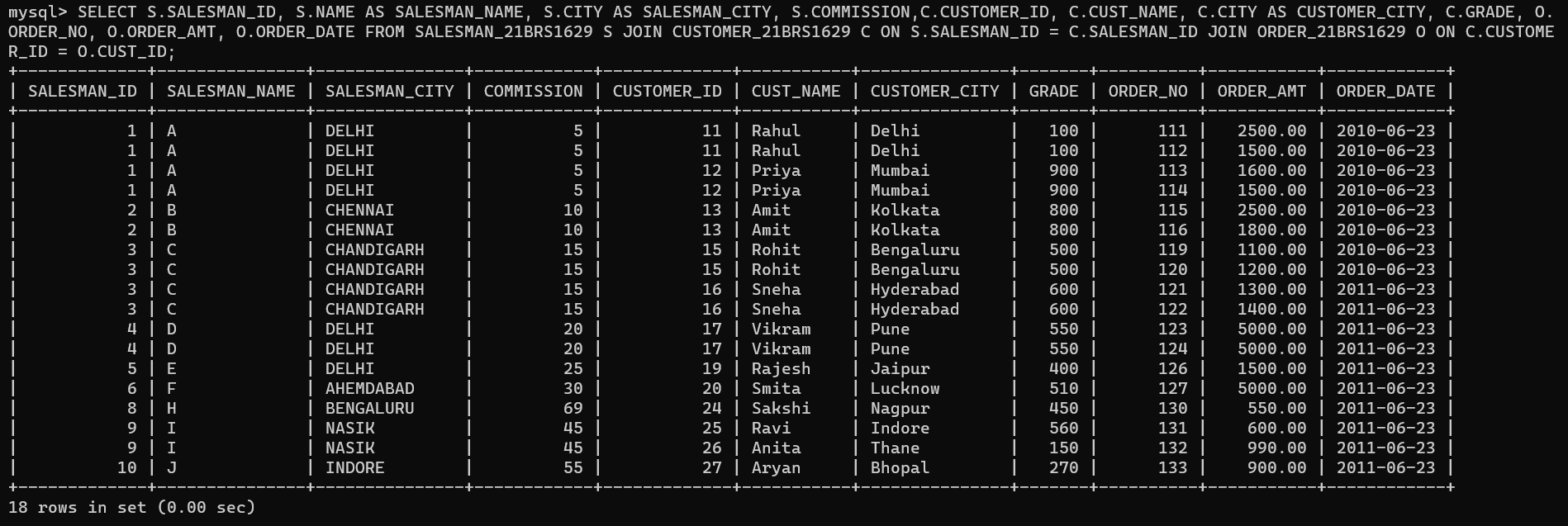


1. **Make a join on the tables: salesman, customer and orders in such a form that the same column of each table will appear once and only the relational rows will come.**

# CODE:

SELECT S.SALESMAN\_ID, S.NAME AS SALESMAN\_NAME, S.CITY AS SALESMAN\_CITY, S.COMMISSION,C.CUSTOMER\_ID, C.CUST\_NAME, C.CITY AS CUSTOMER\_CITY, C.GRADE, O.ORDER\_NO, O.ORDER\_AMT, O.ORDER\_DATE FROM SALESMAN\_21BRS1629 S JOIN CUSTOMER\_21BRS1629 C ON S.SALESMAN\_ID = C.SALESMAN\_ID JOIN ORDER\_21BRS1629 O ON C.CUSTOMER\_ID = O.CUST\_ID;

**OUTPUT**:



1. **Make a list in ascending order for the customer who works either through a salesman or by own.**

# CODE:

SELECT C.CUST\_NAME FROM CUSTOMER\_21BRS1629 C LEFT JOIN SALESMAN\_21BRS1629 S ON C.SALESMAN\_ID = S.SALESMAN\_ID WHERE C.SALESMAN\_ID IS NOT NULL OR C.SALESMAN\_ID IS NULL ORDER BY C.CUST\_NAME ASC;

**OUTPUT**:

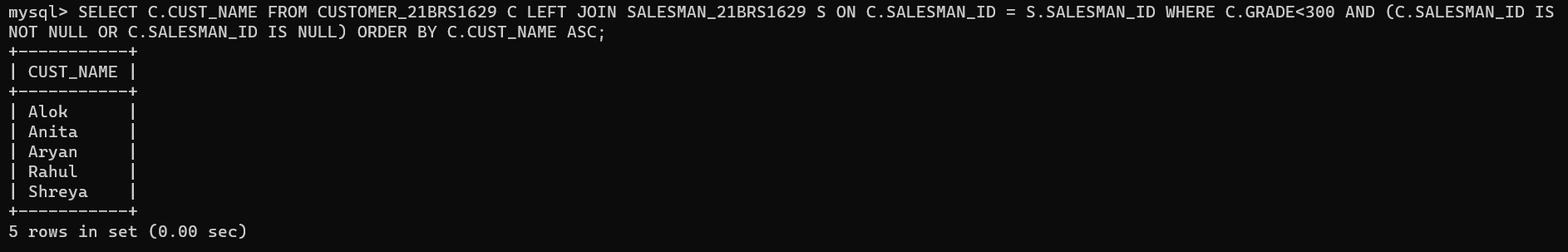


1. **Make a list in ascending order for the customer who holds a grade less than 300 and works either through a salesman or by own.**

# CODE:

SELECT C.CUST\_NAME FROM CUSTOMER\_21BRS1629 C LEFT JOIN SALESMAN\_21BRS1629 S ON C.SALESMAN\_ID = S.SALESMAN\_ID WHERE C.GRADE<300 AND (C.SALESMAN\_ID IS NOT NULL OR C.SALESMAN\_ID IS NULL) ORDER BY C.CUST\_NAME ASC;

**OUTPUT**:

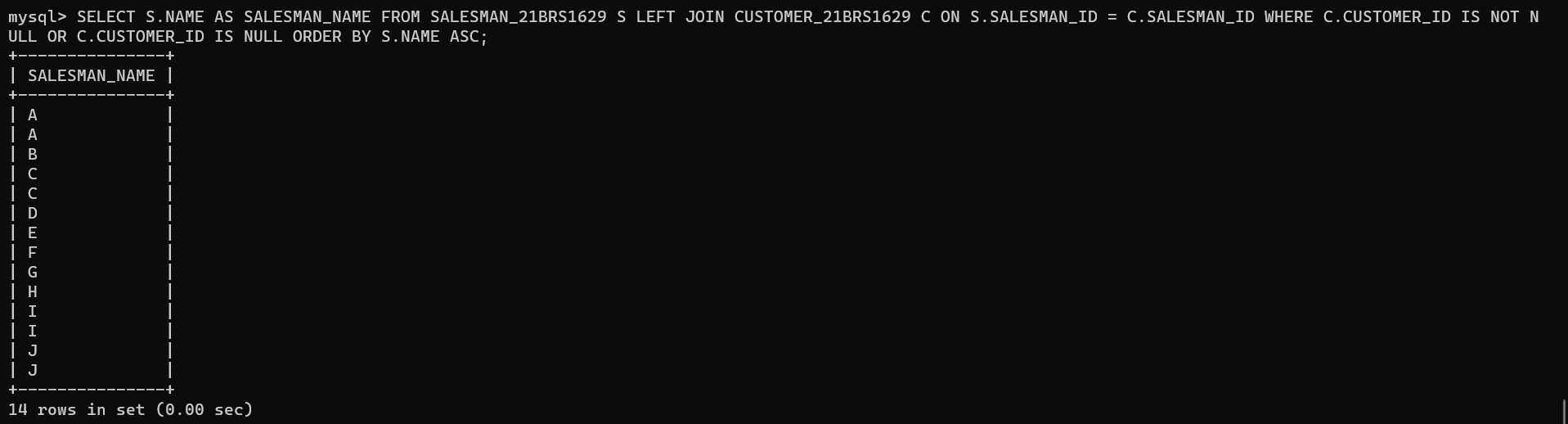


1. **Make a list in ascending order for the salesmen who work either for one or more customer or not yet join under any of the customers.**

# CODE:

SELECT S.NAME AS SALESMAN\_NAME FROM SALESMAN\_21BRS1629 S LEFT JOIN CUSTOMER\_21BRS1629 C ON S.SALESMAN\_ID = C.SALESMAN\_ID WHERE C.CUSTOMER\_ID IS NOT NULL OR C.CUSTOMER\_ID IS NULL ORDER BY S.NAME ASC

**OUTPUT**:

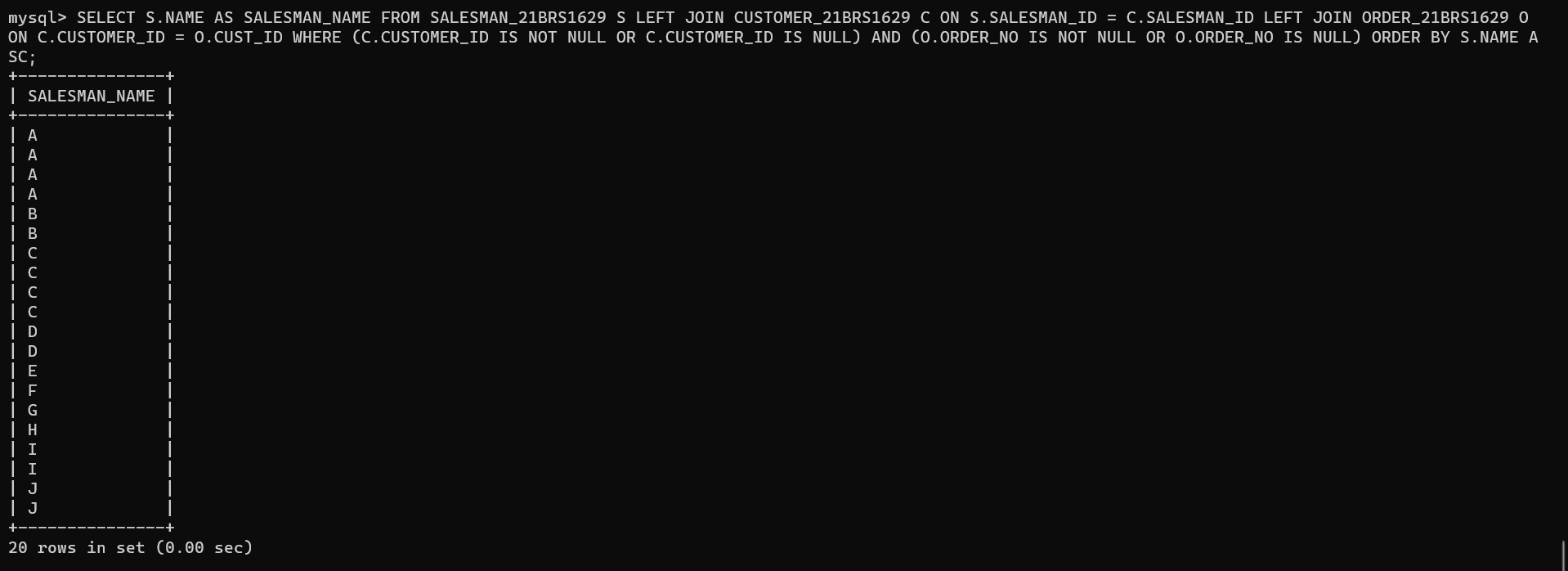


1. **Make a list for the salesmen who work either for one or more customer or not yet join under any of the customers who placed either one or more orders or no order to their supplier.**

# CODE:

SELECT S.NAME AS SALESMAN\_NAME FROM SALESMAN\_21BRS1629 S LEFT JOIN CUSTOMER\_21BRS1629 C ON S.SALESMAN\_ID = C.SALESMAN\_ID LEFT JOIN ORDER\_21BRS1629 O ON C.CUSTOMER\_ID = O.CUST\_ID WHERE (C.CUSTOMER\_ID IS NOT NULL OR C.CUSTOMER\_ID IS NULL) AND (O.ORDER\_NO IS NOT NULL OR O.ORDER\_NO IS NULL) ORDER BY S.NAME ASC;

**OUTPUT**:



1. **Make a Cartesian product between salesman and customer i.e. each salesman will appear for all customers and vice versa.**

# CODE:

SELECT S.NAME AS SALESMAN\_NAME, C.CUST\_NAME FROM SALESMAN\_21BRS1629 S, CUSTOMER\_21BRS1629 C;

**OUTPUT**:

