

CREATE TABLE Salespeople (

SNUM INT PRIMARY KEY,

SNAME VARCHAR(50) NOT NULL,

CITY VARCHAR(50),

COMM DECIMAL(3, 2)

);

CREATE TABLE Customers (

CNUM INT PRIMARY KEY,

CNAME VARCHAR(50) NOT NULL,

CITY VARCHAR(50),

RATING INT,

SNUM INT,

FOREIGN KEY (SNUM) REFERENCES Salespeople(SNUM)

);

CREATE TABLE Orders (

ONUM INT PRIMARY KEY,

AMT DECIMAL(10, 2),

ODATE DATE,

CNUM INT,

SNUM INT,

FOREIGN KEY (CNUM) REFERENCES Customers(CNUM),

FOREIGN KEY (SNUM) REFERENCES Salespeople(SNUM)

);

INSERT INTO Salespeople (SNUM, SNAME, CITY, COMM) VALUES

(1001, 'Peel', 'London', 0.12),

(1002, 'Serres', 'San Jose', 0.13),

(1004, 'Motika', 'London', 0.11),

(1007, 'Rifkin', 'Barcelona', 0.15),

(1003, 'AxelRod', 'New York', 0.10),

(1005, 'Fran', 'London', 0.26);

INSERT INTO Customers (CNUM, CNAME, CITY, RATING, SNUM) VALUES

(2001, 'Hoffman', 'London', 100, 1001),

(2002, 'Giovanni', 'Rome', 200, 1003),

(2003, 'Liu', 'San Jose', 200, 1002),

(2004, 'Grass', 'Berlin', 300, 1002),

(2006, 'Clemens', 'London', 100, 1001),

(2008, 'Cisneros', 'San Jose', 300, 1007),

(2007, 'Pereira', 'Rome', 100, 1004);

INSERT INTO Orders (ONUM, AMT, ODATE, CNUM, SNUM) VALUES

(3001, 18.69, '1996-10-03', 2008, 1007),

(3003, 767.19, '1996-10-03', 2001, 1001),

(3002, 1900.10, '1996-10-03', 2007, 1004),

(3005, 5160.45, '1996-10-03', 2003, 1002),

(3006, 1098.16, '1996-10-03', 2008, 1007),

(3009, 1713.23, '1996-10-04', 2002, 1003),

(3007, 75.75, '1996-10-04', 2002, 1003),

(3008, 4723.00, '1996-10-05', 2006, 1001),

(3010, 1309.95, '1996-10-06', 2004, 1002),

(3011, 9891.88, '1996-10-06', 2006, 1001);

1. List all the columns of the Salespeople table.

Select \* from Salespeople;

1. List all customers with a rating of 100.

Select \* from Customers where rating = 100;

1. Find all records in the Customer table with NULL values in the city column.

Select city from customers where city is null;

1. Find the largest order taken by each salesperson on each date.

Select snum, max(amt), odate from salesperson group by snum, odate order by odate, snum;

1. Arrange the Orders table by descending customer number.

Select \* from orders order by cnum desc;

1. Find which salespeople currently have orders in the Orders table.

Select distinct o.snum, s.sname from orders o left join salespeople s on o.snum = s.snum;

1. List names of all customers matched with the salespeople serving them.

( List names of all customers and their salesperson’s name serving them.)

Select cname, sname from customers c join salespeople s on c.snum = s.snum;

1. Find the names and numbers of all salespeople who had more than one customer.

Select s.snum, s.sname

from salespeople s

join (

select snum

from customers

group by snum

having count(cnum) > 1

) as more\_cus

On s.snum = more\_cus.snum;

1. Count the orders of each of the salespeople and output the results in descending order.

Select s.snum,s.sname, count(onum)

from salespeople s

join orders o

on s.snum = o.snum

group by o.snum

order by o.snum;

1. List the Customer table if and only if one or more of the customers in the Customer table are located in San Jose. select \* from customers

where exists

(

select 1 from customers where city = 'San Jose'

);

1. Match salespeople to customers according to what city they lived in.

Select s.sname, c.cname, c.city from customers c join salespeople s on s.city = c.city;

1. Find the largest order taken by each salesperson.

Select s.snum, s.sname , max(o.amt) from salespeople s join orders o on s.snum = o.snum group by s.snum;

1. Find customers in San Jose who have a rating above 200.

Select cname from customers where city = 'San Jose' and rating > 200;

1. List the names and commissions of all salespeople in London.

Select sname, comm from salespeople where city = 'London';

1. List all the orders of salesperson Motika from the Orders table.

Select onum, amt,odate from orders o join salespeople s on s.snum = o.snum where s.sname = 'Motika';

1. Find all customers with orders on October 3.

Select cname from customers c join orders o on c.cnum = o.cnum where month( odate ) = 10 and dayofmonth(odate) = 3;

1. give the sums of the amounts from the orders table, grouped by date, eliminating all those dates where the sum was not at least 2000.00 above the max amount  
   select odate, sum(amt) as total\_amount from orders group by odate having sum(amt) >= (max(amt) + 2000.00) order by odate;
2. select all orders that had amounts that were greater than at least one of the orders from october 6  
   select \* from orders where amt > (select min(amt) from orders where odate = '1996-10-06');
3. write a query that uses the exists operator to extract all salespeople who have customers with a rating of 300  
   select s.snum, s.sname from salespeople s where exists (select 1 from customers c where c.snum = s.snum and c.rating = 300);
4. find all pairs of customers having the same rating  
   select c1.cname as customer1, c2.cname as customer2, c1.rating from customers c1 join customers c2 on c1.rating = c2.rating and c1.cnum < c2.cnum;
5. find all customers whose cnum is 1000 above the snum of serres  
   select \* from customers where cnum = (select snum from salespeople where sname = 'serres') + 1000;
6. give the salespeople’s commissions as percentages instead of decimal numbers  
   select sname, concat(comm \* 100, '%') as commission\_percentage from salespeople;
7. find the largest order taken by each salesperson on each date, eliminating those max orders which are less than 3000.00 in value  
   select snum, max(amt) as largest\_order, odate from orders group by snum, odate having max(amt) >= 3000.00 order by odate, snum;
8. list the largest orders for october 3, for each salesperson  
   select snum, max(amt) as largest\_order from orders where odate = '1996-10-03' group by snum;
9. find all customers located in cities where serres (snum 1002) has customers  
   select \* from customers where city in (select city from customers where snum = (select snum from salespeople where sname = 'serres'));
10. select all customers with a rating above 200.00  
    select \* from customers where rating > 200;
11. count the number of salespeople currently listing orders in the orders table  
    select count(distinct snum) as salespeople\_with\_orders from orders;
12. write a query that produces all customers serviced by salespeople with a commission above 12%. output the customer’s name and the salesperson’s rate of commission  
    select c.cname, s.comm from customers c join salespeople s on c.snum = s.snum where s.comm > 0.12;
13. find salespeople who have multiple customers  
    select s.snum, s.sname from salespeople s join customers c on s.snum = c.snum group by s.snum, s.sname having count(c.cnum) > 1;
14. find salespeople with customers located in their city  
    select distinct s.snum, s.sname from salespeople s join customers c on s.snum = c.snum where s.city = c.city;
15. find all salespeople whose name starts with ‘p’ and the fourth character is ‘l’  
    select \* from salespeople where sname like 'p\_\_l%';
16. write a query that uses a subquery to obtain all orders for the customer named cisneros. assume you do not know his customer number  
    select \* from orders where cnum = (select cnum from customers where cname = 'cisneros');
17. find the largest orders for serres and rifkin  
    select o.snum, s.sname, max(o.amt) as largest\_order from orders o join salespeople s on o.snum = s.snum where s.sname in ('serres', 'rifkin') group by o.snum, s.sname;
18. extract the salespeople table in the following order: snum, sname, commission, city  
    select snum, sname, comm as commission, city from salespeople;
19. select all customers whose names fall in between ‘a’ and ‘g’ alphabetical range  
    select \* from customers where cname >= 'a' and cname < 'h';
20. select all the possible combinations of customers that you can assign  
    select c1.cname as customer1, c2.cname as customer2 from customers c1 cross join customers c2;
21. select all orders that are greater than the average for october 4  
    select \* from orders where amt > (select avg(amt) from orders where odate = '1996-10-04');
22. write a select command using a correlated subquery that selects the names and numbers of all customers with ratings equal to the maximum for their city  
    select c1.cnum, c1.cname, c1.rating, c1.city from customers c1 where c1.rating = (select max(c2.rating) from customers c2 where c2.city = c1.city);
23. write a query that totals the orders for each day and places the results in descending order  
    select odate, sum(amt) as daily\_total from orders group by odate order by daily\_total desc;
24. write a select command that produces the rating followed by the name of each customer in san jose  
    select rating, cname from customers where city = 'san jose';
25. find all orders with amounts smaller than any amount for a customer in san jose  
    select \* from orders where amt < (select min(o.amt) from orders o join customers c on o.cnum = c.cnum where c.city = 'san jose');

select \* from orders where amt <any (select amt from orders o , customers c where o.cnum = c.cnum and c.city = 'san jose');

1. find all orders with above average amounts for their customers  
   select o1.onum, o1.amt, o1.cnum from orders o1 where o1.amt > (select avg(o2.amt) from orders o2 where o2.cnum = o1.cnum);
2. write a query that selects the highest rating in each city  
   select city, max(rating) as highest\_rating from customers group by city;
3. write a query that calculates the amount of the salesperson’s commission on each order by a customer with a rating above 100.00  
   select o.onum, o.amt, c.cname as customer\_name, s.sname as salesperson, (o.amt \* s.comm) as commission\_amount from orders o join customers c on o.cnum = c.cnum join salespeople s on o.snum = s.snum where c.rating > 100;
4. count the customers with ratings above san jose’s average  
   select count(cnum) as customers\_above\_sanjose\_avg from customers where rating > (select avg(rating) from customers where city = 'san jose');
5. write a query that produces all pairs of salespeople with themselves as well as duplicate rows with the order reversed  
   select s1.sname as salesperson1, s2.sname as salesperson2 from salespeople s1 cross join salespeople s2;
6. find all salespeople that are located in either barcelona or london  
   select \* from salespeople where city in ('barcelona', 'london');
7. find all salespeople with only one customer  
   select s.snum, s.sname from salespeople s join customers c on s.snum = c.snum group by s.snum, s.sname having count(c.cnum) = 1;
8. write a query that joins the customer table to itself to find all pairs of customers served by a single salesperson  
   select c1.cname as customer1, c2.cname as customer2, c1.snum as salesperson\_id from customers c1 join customers c2 on c1.snum = c2.snum where c1.cnum < c2.cnum;
9. write a query that will give you all orders for more than 1000.00  
   select \* from orders where amt > 1000.00;

Top of Form

Bottom of Form