Queries

**1. List all the columns of the Salespeople table.**

desc salespeople;

**2. List all customers with a rating of 100.**

select \* from customer where rating=100;

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

select city from customer where city is null;

**4. Find the largest order taken by each salesperson on each date.**

select b.snum,a.odate,max(a.amt) from orders a, customer b group by snum, odate;

**5. Arrange the Orders table by descending customer number.**

select \* from orders order by cnum desc;

**6. Find which salespeople currently have orders in the Orders table.**

select a.onum, b.cname, c.sname from orders a, customer b, salespeople c where a.cnum=b.cnum and b.snum=c.snum;

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

select c.cnum, c.cname,s.snum, s.sname from customer c,salespeople s where c.snum=s.snum;

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

select b.sname, count(a.cnum) from customer a, salespeople b where a.snum=b.snum group by b.snum;

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

select count(a.onum), b.snum from orders a, customer b where a.cnum=b.cnum group by b.snum;

**10. 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 customer where city="San Jose";

**11. Match salespeople to customers according to what city they lived in.**

select a.cname, b.sname, b.city from customer a, salespeople b where a.snum=b.snum;

**12. Find the largest order taken by each salesperson.**

select b.snum, max(a.amt) from orders a, customer b where a.cnum=b.cnum group by b.snum;

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

select \* from customer where city="San Jose" having rating > 200;

**14. List the names and commissions of all salespeople in London.**

select sname, comm from salespeople where city = "London";

**15. List all the orders of salesperson Motika from the Orders table.**

select a.onum,c.sname from orders a, customer b, salespeople c where a.cnum=b.cnum and b.snum=c.snum and c.sname="Motika";

**16. Find all customers with orders on October 3.**

select \* from customer a, orders b where a.cnum=b.cnum having b.odate="1996-03-10";

**17. 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 sum(amt) as "Total", odate from orders group by odate having total>2000;

**18. 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="1996-06-10");

**19. Write a query that uses the EXISTS operator to extract all salespeople who have customers**

**with a rating of 300.**

select \* from salespeople where exists (select rating from customer where customer.snum=salespeople.snum and rating=300);

**20. Find all pairs of customers having the same rating.**

select c1.cname,c1.rating,c2.cname from customer c1 join customer c2 on c1.rating=c2.rating;

**21. Find all customers whose CNUM is 1000 above the SNUM of Serres.**

select \* from customer where cnum>(select (1000+snum) from salespeople where sname="Serres");

**22. Give the salespeople’s commissions as percentages instead of decimal numbers.**

select snum,sname,(100\*comm)+'%' as 'Percentage' from salespeople;

**23. Find the largest order taken by each salesperson on each date, eliminating those MAX orders**

**which are less than $3000.00 in value.**

select b.snum, max(a.amt) from orders a, customer b where a.cnum=b.cnum group by b.snum having max(a.amt)>=3000;

**24. List the largest orders for October 3, for each salesperson.**

select \* from customer a, orders b where a.cnum=b.cnum group by (snum) having b.odate="1996-03-10";

**25. Find all customers located in cities where Serres (SNUM 1002) has customers.**

select \* from customer where snum=(select snum from salespeople where sname="Serres");

**26. Select all customers with a rating above 200.00.**

select \* from customer where rating>200;

**27. Count the number of salespeople currently listing orders in the Orders table.**

select a.onum,b.cnum,count(c.snum) from orders a, customer b, salespeople c where a.cnum=b.cnum and b.snum=c.snum group by b.snum;

**28. 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 cname, comm from customer a, salespeople b where a.snum=b.snum and b.comm > 0.12;

**29. Find salespeople who have multiple customers.**

select a.cname,count(b.snum),b.sname from customer a join salespeople b where a.snum=b.snum group by b.snum having count(b.snum)>1;

**30. Find salespeople with customers located in their city.**

select a.cname,a.city,b.snum,b.sname,b.city from customer a join salespeople b where a.snum=b.snum and a.city=b.city;

**31. Find all salespeople whose name starts with ‘P’ and the fourth character is ‘l’.**

select \* from salespeople where sname like 'p%' and sname like '%%%l';

**32. 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 customer where cnum = (select cnum from customer where cname="Cisneros");

**33. Find the largest orders for Serres and Rifkin.**

select b.snum,a.onum,max(a.amt) from orders a, customer b where a.cnum=b.cnum group by snum having snum=(select snum from salespeople where sname="Serres") or snum=(select snum from salespeople where sname="Rifkin");

**34. Extract the Salespeople table in the following order : SNUM, SNAME, COMMISSION, CITY.**

select snum,sname,comm as commission, city from salespeople;

**35. Select all customers whose names fall in between ‘A’ and ‘G’ alphabetical range.**

select \* from customer where cname between 'a%' and 'G%';

**36. Select all the possible combinations of customers that you can assign.**

select a.onum,b.cnum,b.cname,c.snum,c.sname from orders a, customer b, salespeople c where a.cnum=b.cnum and b.snum=c.snum;

select \* from customer c1 join customer c2 on c1.cnum=c2.cnum;

**37. Select all orders that are greater than the average for October 4.**

select onum, odate,amt from orders where amt>(select avg(amt) from orders where odate='1996-04-10');

**38. Write a select command using a corelated subquery that selects the names and numbers of all**

**customers with ratings equal to the maximum for their city.**

select cname,city,max(rating) from customer group by city,cname;

**39. Write a query that totals the orders for each day and places the results in descending order.**

select count(onum),sum(amt),odate from orders group by odate;

**40. Write a select command that produces the rating followed by the name of each customer in**

**San Jose.**

select rating, cname from customer where city="San Jose";

**41. Find all orders with amounts smaller than any amount for a customer in San Jose.**

select amt from orders where amt < (select sum(a.amt) from orders a, customer b where a.cnum=b.cnum and city="San Jose");

**42. Find all orders with above average amounts for their customers.**

select \* from orders where amt > (select avg(amt) from orders);

**43. Write a query that selects the highest rating in each city.**

select city, max(rating) from customer group by city;

**44. 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 onum,cname,sname,(amt\*((100\*comm)/100)) as commission from orders a, customer b, salespeople c where a.cnum=b.cnum and b.snum=c.snum and b.rating>100;

**45. Count the customers with ratings above San Jose’s average.**

select count(cnum) from customer where rating>(select avg(rating) from customer where city="San Jose");

**46. Write a query that produces all pairs of salespeople with themselves as well as duplicate rows**

**with the order reversed.**

select \* from salespeople c1 join (select \* from salespeople order by snum desc) c2;

**47. Find all salespeople that are located in either Barcelona or London.**

select sname,snum,city from salespeople where city="Barcelona" or city="London";

**48. Find all salespeople with only one customer.**

select count(a.cnum),b.snum,b.sname from customer a,salespeople b where a.snum=b.snum group by snum having count(a.cnum)=1;

**49. Write a query that joins the Customer table to itself to find all pairs of customers served by a**

**single salesperson.**

select c1.cname,c2.cname,c2.city from customer c1 join customer c2;

**50. Write a query that will give you all orders for more than $1000.00**

select \* from orders where amt>1000;

**51. Write a query that lists each order number followed by the name of the customer who made**

**that order.**

select a.onum,b.cnum,b.cname from orders a,customer b where a.cnum=b.cnum;

**52. Write 2 queries that select all salespeople (by name and number) who have customers in their  
cities who they do not service, one using a join and one a corelated subquery. Which solution**

**is more elegant?**

select a.cname,b.snum,b.sname from customer a, salespeople b where a.snum=b.snum and a.city != b.city;

select a.cname,b.snum,b.sname from customer a join salespeople b on a.snum=b.snum where a.city != b.city;

**53. Write a query that selects all customers whose ratings are equal to or greater than ANY (in the**

**SQL sense) of Serres’?**

select a.cname,a.city,a.rating,b.snum,b.sname from customer a join salespeople b on a.snum=b.snum where a.rating>=(select max(a.rating) from customer a join salespeople b on a.snum=b.snum where b.sname="Serres");

**54. Write 2 queries that will produce all orders taken on October 3 or October 4.**

select \* from orders where odate="1996-03-10" or odate="1996-04-10";

**55. Write a query that produces all pairs of orders by a given customer. Name that customer and**

**eliminate duplicates.**

select a.onum,a.odate,b.cnum,b.cname from orders a join customer b on a.cnum=b.cnum;

**56. Find only those customers whose ratings are higher than every customer in Rome.**

select \* from customer where rating>=(select max(rating) from customer where city="Rome");

**57. Write a query on the Customers table whose output will exclude all customers with a rating <=**

**100.00, unless they are located in Rome.**

select \* from customer where rating>100 or city="Rome";

**58. Find all rows from the Customers table for which the salesperson number is 1001.**

select \* from customer where snum="1001";

**59. Find the total amount in Orders for each salesperson for whom this total is greater than the**

**amount of the largest order in the table.**

select b.snum,sum(a.amt) as 'total' from orders a join customer b on a.cnum=b.cnum group by b.snum having total>(select max(amt) from orders);

**60. Write a query that selects all orders save those with zeroes or NULLs in the amount field.**

select \* from orders where amt is null or amt=0;

**61. Produce all combinations of salespeople and customer names such that the former precedes**

**the latter alphabetically, and the latter has a rating of less than 200.**

select a.cname,b.sname from customer a join salespeople b on a.snum=b.snum where a.rating<200 order by a.cname;

**62. List all Salespeople’s names and the Commission they have earned.**

select a.onum,b.cname,c.sname,c.comm,a.amt,(c.comm\*a.amt) as 'total comm' from orders a, customer b, salespeople c where a.cnum=b.cnum and b.snum=c.snum;

**63. Write a query that produces the names and cities of all customers with the same rating as**

**Hoffman. Write the query using Hoffman’s CNUM rather than his rating, so that it would still be**

**usable if his rating changed.**

select cname,rating from customer where rating = (select rating from customer where cnum=(select cnum from customer where cname="Hoffman"));

**64. Find all salespeople for whom there are customers that follow them in alphabetical order.**

select b.sname,a.cname from customer a join salespeople b on a.snum=b.snum order by a.cname asc;

**65. Write a query that produces the names and ratings of all customers of all who have above**

**average orders.**

select b.cname,b.rating from orders a join customer b on a.cnum=b.cnum where a.amt>(select avg(amt) from orders);

**66. Find the SUM of all purchases from the Orders table.**

select sum(amt) from orders;

**67. Write a SELECT command that produces the order number, amount and date for all rows in**

**the order table.**

select onum,amt,odate from orders;

**68. Count the number of nonNULL rating fields in the Customers table (including repeats).**

select count(rating) from customer where rating is not null;

**69. Write a query that gives the names of both the salesperson and the customer for each order**

**after the order number.**

select a.onum,b.cname,c.sname from orders a join customer b on a.cnum=b.cnum join salespeople c on b.snum=c.snum;

**70. List the commissions of all salespeople servicing customers in London.**

select a.cname,b.comm,b.sname,a.city from customer a join salespeople b on a.snum=b.snum where a.city="London";

**71. Write a query using ANY or ALL that will find all salespeople who have no customers located in**

**their city.**

select a.cname,b.sname,a.city,b.city from customer a join salespeople b on a.snum=b.snum where a.city!=b.city;

**72. Write a query using the EXISTS operator that selects all salespeople with customers located in**

**their cities who are not assigned to them.**

select \* from salespeople where exists (select a.cname,b.sname,a.city,b.city from customer a join salespeople b on a.snum=b.snum where a.city!=b.city);

**73. Write a query that selects all customers serviced by Peel or Motika. (Hint : The SNUM field**

**relates the two tables to one another.)**

select a.cname,b.sname from customer a join salespeople b on a.snum=b.snum where b.sname="Peel" or b.sname="Motika";

**74. Count the number of salespeople registering orders for each day. (If a salesperson has more**

**than one order on a given day, he or she should be counted only once.)**

select count(a.onum),a.odate,b.snum from orders a, customer b where a.cnum=b.cnum group by odate;

**75. Find all orders attributed to salespeople in London.**

select a.onum,b.cname,c.sname from orders a join customer b on a.cnum=b.cnum join salespeople c on b.snum=c.snum where c.city="London";

**76. Find all orders by customers not located in the same cities as their salespeople.**

select a.onum,b.cname,c.sname from orders a join customer b on a.cnum=b.cnum join salespeople c on b.snum=c.snum where c.city!=b.city;

**77. Find all salespeople who have customers with more than one current order.**

select a.onum,b.cname,c.sname,count(b.cname) as 'count2' from orders a join customer b on a.cnum=b.cnum join salespeople c on b.snum=c.snum where c.city!=b.city group by c.sname having count2>1;

**78. Write a query that extracts from the Customers table every customer assigned to a**

**salesperson who currently has at least one other customer (besides the customer being**

**selected) with orders in the Orders table.**

**79. Write a query that selects all customers whose names begin with ‘C’.**

select \* from customer where cname like "C%";

**80. Write a query on the Customers table that will find the highest rating in each city. Put the output**

**in this form : for the city (city) the highest rating is : (rating).**

select concat('for the city',city),concat('the highest rating is',max(rating)) from customer group by city;

**81. Write a query that will produce the SNUM values of all salespeople with orders currently in the**

**Orders table (without any repeats).**

select a.onum,b.cname,b.snum from orders a join customer b on a.cnum=b.cnum group by snum;

**82. Write a query that lists customers in descending order of rating. Output the rating field first,**

**followed by the customer’s names and numbers.**

select rating,cname,cnum from customer order by rating desc;

**83. Find the average commission for salespeople in London.**

select avg(comm) from salespeople where city="London";

**84. Find all orders credited to the same salesperson who services Hoffman (CNUM 2001).**

select a.onum,b.cnum,b.snum from orders a join customer b on a.cnum=b.cnum where b.cnum=(select cnum from customer where cname="Hoffman");

**85. Find all salespeople whose commission is in between 0.10 and 0.12 (both inclusive).**

select \* from salespeople where comm between 0.10 and 0.12;

**86. Write a query that will give you the names and cities of all salespeople in London with a**

commission above 0.10.

select sname,city from salespeople where city="London" and comm>0.10;

**87. What will be the output from the following query?**

SELECT \* FROM ORDERS  
 where (amt < 1000 OR NOT (odate = 10/03/1996 AND cnum >

2003));

+------+---------+------------+------+

| onum | amt | odate | cnum |

+------+---------+------------+------+

| 3001 | 18.69 | 1996-03-10 | 2008 |

| 3002 | 1900.10 | 1996-03-10 | 2007 |

| 3003 | 767.19 | 1996-03-10 | 2001 |

| 3005 | 5160.45 | 1996-03-10 | 2003 |

| 3006 | 1098.45 | 1996-03-10 | 2008 |

| 3007 | 75.75 | 1996-04-10 | 2002 |

| 3008 | 4723.00 | 1996-05-10 | 2006 |

| 3009 | 1713.23 | 1996-04-10 | 2002 |

| 3010 | 1309.95 | 1996-06-10 | 2004 |

| 3011 | 9891.88 | 1996-06-10 | 2006 |

+------+---------+------------+------+

**88. Write a query that selects each customer’s smallest order.**

select min(amt),onum,cnum from orders group by cnum;

**89. Write a query that selects the first customer in alphabetical order whose name begins with G.**

select \* from customer where cname like 'G%';

**90. Write a query that counts the number of different nonNULL city values in the Customers table.**

select count(city) from customer where city is not null;

**91. Find the average amount from the Orders table.**

select avg(amt) from orders;

**92. What would be the output from the following query?**

**SELECT \* FROM ORDERS**

**WHERE NOT (odate = 10/03/96 OR snum > 1006) AND amt >=**

**1500);**

ERROR 1054 (42S22): Unknown column 'snum' in 'where clause'

As there is no join present for orders-customer-salespeople

**93. Find all customers who are not located in San Jose and whose rating is above 200.**

select \* from customer where city!="San Jose" and rating>200;

**94. Give a simpler way to write this query :**

**SELECT snum, sname city, comm FROM salespeople**

**WHERE (comm > + 0.12 OR comm < 0.14);**

select \* from salespeople where (comm>0.12 or comm<0.14);

**95. Evaluate the following query :**

**SELECT \* FROM orders**

**WHERE NOT ((odate = 10/03/96 AND snum > 1002) OR amt > 2000.00);**

ERROR 1054 (42S22): Unknown column 'snum' in 'where clause'

As there is no join present for orders-customer-salespeople

**96. Which salespersons attend to customers not in the city they have been assigned to?**

select a.cname,b.sname,a.city,b.city from customer a join salespeople b on a.snum=b.snum where a.city!=b.city;

**97. Which salespeople get commission greater than 0.11 are serving customers rated less than**

**250?**

select b.snum,b.sname,b.comm,b.city from customer a join salespeople b on a.snum=b.snum where b.comm>0.11 and a.rating<250;

**98. Which salespeople have been assigned to the same city but get different commission**

**percentages?**

**99. Which salesperson has earned the most by way of commission?**

select c.sname,max(c.comm\*a.amt) from orders a join customer b on a.cnum=b.cnum join salespeople c on b.snum=c.snum;

**100.Does the customer who has placed the maximum number of orders have the maximum rating?**

true

**101.Has the customer who has spent the largest amount of money been given the highest rating?**

false

**102.List all customers in descending order of customer rating.**

select \* from customer order by rating desc;

**103.On which days has Hoffman placed orders?**

select odate from orders where cnum=(select cnum from customer where cname="Hoffman");

**104.Do all salespeople have different commissions?**

True

**105.Which salespeople have no orders between 10/03/1996 and 10/05/1996?**

select onum,odate,cname,snum from orders a join customer b on a.cnum=b.cnum where odate between '1996-03-10' and '1996-05-10';

**106.How many salespersons have succeeded in getting orders?**

select count(sname),onum,cname from orders a join customer b on a.cnum=b.cnum join salespeople c on b.snum=c.snum;

10

**107.How many customers have placed orders?**

select \* from orders group by cnum; 7

**108.On which date has each salesperson booked an order of maximum value?**

select odate,max(amt) from orders group by odate order by max(amt) desc limit 1;

**109.Who is the most successful salesperson?**

select (comm\*amt),sname from orders a join customer b on a.cnum=b.cnum join salespeople c on b.snum=c.snum order by (comm\*amt) desc limit 1;

Peel

**110.Who is the worst customer with respect to the company?**

select cname,rating from customer order by rating asc limit 1;

Hoffman

**111.Are all customers not having placed orders greater than 200 totally been serviced by**

**salespersons Peel or Serres?**

select amt,sname from orders a join customer b on a.cnum=b.cnum join salespeople c on b.snum=c.snum where amt>200 and (sname="Peel" or sname="Serres");

No

**112.Which customers have the same rating?**

select rating,cname from customer order by rating asc;

**113.Find all orders greater than the average for October 4th.**

select amt,odate from orders where amt>(select avg(amt) from orders where odate="1996-04-10");

**114.Which customers have above average orders?**

select amt,odate,cnum from orders where amt>(select avg(amt) from orders);

**115.List all customers with ratings above San Jose’s average.**

select rating,city from customer where rating>(select avg(rating) from customer where city="San Jose");

**116.Select the total amount in orders for each salesperson for whom the total is greater than the**

**amount of the largest order in the table.**

select sum(amt), snum from orders a join customer b on a.cnum=b.cnum group by snum having sum(amt)>(select max(amt) from orders);

**117.Give names and numbers of all salespersons who have more than one customer.**

select sname,b.snum,count(a.cname) from customer a join salespeople b on a.snum=b.snum group by sname having count(a.cname)>1;

**118.Select all salespersons by name and number who have customers in their city whom they**

**don’t service.**

select sname,a.cname from customer a join salespeople b on a.snum=b.snum where a.city!=b.city;

**119.Which customers’ rating should be lowered?**

select (comm\*amt),rating,sname from orders a join customer b on a.cnum=b.cnum join salespeople c on b.snum=c.snum order by (comm\*amt) asc, rating asc;

Rifkin

**120.Is there a case for assigning a salesperson to Berlin?**

select \* from customer where city="Berlin";

Since 1 customer is from berlin we can assign one salesperson there.

**121.Is there any evidence linking the performance of a salesperson to the commission that he or**

**she is being paid?**

**122.Does the total amount in orders by customer in Rome and London exceed the commission**

**paid to salespersons in London and New York by more than 5 times?**

**123.Which is the date, order number, amt and city for each salesperson (by name) for the**

**maximum order he has obtained?**

select odate,onum,c.city,sname,max(amt) from orders a join customer b on a.cnum=b.cnum join salespeople c on b.snum=c.snum group by c.snum;

**124.Which salesperson(s) should be fired?**

select (comm\*amt),sname from orders a join customer b on a.cnum=b.cnum join salespeople c on b.snum=c.snum order by (comm\*amt) asc limit 1;

Rifkin

1**25.What is the total income for the company?**

select sum(a.amt-(comm\*amt)) from orders a join customer b on a.cnum=b.cnum join salespeople c on b.snum=c.snum;

23416.2096