What is Database?

A database is a separate application that stores a collection of data. Each database has one or more distinct APIs for creating, accessing, managing, searching and replicating the data it holds.

Other kinds of data stores can be used, such as files on the file system or large hash tables in memory, but data fetching and writing would not be so fast and easy with those types of systems.

So nowadays, we use relational database management systems (RDBMS) to store and manage huge volume of data. This is called relational database because all the data is stored into different tables and relations are established using primary keys or other keys known as foreign keys.

**A Relational DataBase Management System (RDBMS) is a software that**:

Enables you to implement a database with tables, columns and indexes.

Guarantees the Referential Integrity between rows of various tables.

Updates the indexes automatically.

Interprets an SQL query and combines information from various tables.

**RDBMS Terminology**:

Before we proceed to explain MySQL database system, let's revise few definitions related to database.

Database: A database is a collection of tables, with related data.

Table: A table is a matrix with data. A table in a database looks like a simple spreadsheet.

Column: One column (data element) contains data of one and the same kind, for example the column postcode.

Row: A row (= tuple, entry or record) is a group of related data, for example the data of one subscription.

Redundancy: Storing data twice, redundantly to make the system faster.

Primary Key: A primary key is unique. A key value can not occur twice in one table. With a key, you can find at most one row.

Foreign Key: A foreign key is the linking pin between two tables.

--------------------------------------------------------------------------------------------------------------------------------------

create database retail;

use retail;

CREATE TABLE IF NOT EXISTS salespeople

( snum INT NOT NULL,

sname VARCHAR(30) NOT NULL,

city VARCHAR(30) NOT NULL,

comm DECIMAL(4,2) NOT NULL,

PRIMARY KEY (snum) );

INSERT INTO salespeople VALUES (1001, 'Peel', 'London', 0.12);

Salespeople SNUM SNAME CITY COMM

1001 Peel London .12 1002 Serres San Jose .13

1004 Motika London .11 1007 Rifkin Barcelona .15

1003 AxelRod New York .10

1005 Fran London .26

CREATE TABLE IF NOT EXISTS customer

( cnum INT NOT NULL,

cname VARCHAR(30) NOT NULL,

city VARCHAR(30) NOT NULL,

ratingint not null,

snumint NOT NULL,

PRIMARY KEY (cnum),

FOREIGN KEY (snum) REFERENCES salespeople(snum) );

INSERT INTO customer VALUES (2001, 'Hoffman', 'London',100, 1001);

Customers CNUM CNAME CITY RATING SNUM

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

CREATE TABLE IF NOT EXISTS orders

( onum INT NOT NULL,

amt DECIMAL(7,2) NOT NULL,

odate Date NOT NULL,

cnumint NOT NULL,

PRIMARY KEY (onum),

FOREIGN KEY (cnum) REFERENCES customer(cnum) );

INSERT INTO orders VALUES (3001, 18.69, '1996-03-10', 2008);

Orders ONUM AMT ODATE CNUM

3001 18.69 10/03/96 2008

3003 767.19 10/03/96 2001

3002 1900.10 10/03/96 2007

3005 5160.45 10/03/96 2003

3006 1098.16 10/03/96 2008

3009 1713.23 10/04/96 2002

3007 75.75 10/04/96 2002

3008 4723.00 10/05/96 2006

3010 1309.95 10/06/96 2004

3011 9891.88 10/06/96 2006

show tables ;

describe orders;

Update clause UPDATE table\_name SET field1=new-value1, field2=new-value2 [WHERE Clause]

Delete clause DELETE FROM table\_name [WHERE Clause]

Queries

1.       List all the columns of the Salespeople table.

 describe 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 \* from customer having city = null  
  
select \* from customer where city = null

4.       Find the largest order taken by each salesperson on each date.   
  
selecta.snum, o.onum, max(o.amt), o.odate from customer a, orders o where a.cnum=o.cnum group by snum, odate;

5.       Arrange the Orders table by descending customer number.

select \* from orders order by cnumdesc

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

selects.sname,c.cname, o.onum  from customer c, orders o, salespeople s where s.snum=c.snum and c.cnum=o.cnum

7.       List names of all customers matched with the salespeople serving them.   
  
selectc.cname,c.cnum, s.sname, s.snum from customer c,salespeople s where s.snum=c.snum

8.       Find the names and numbers of all salespeople who had more than one customer.   
  
selects.snum, s.sname, count(c.cname) as Customers\_count from customer c,salespeople s where s.snum=c.snum group by snum having  Customers\_count> 1

9.       Count the orders of each of the salespeople and output the results in descending order.   
  
selectc.snum,count(o.onum) as OrderCount from customer c, orders o where  c.cnum=o.cnum                    group by c.snum order by OrderCountdesc

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.  
  
selects.snum, s.sname, s.city as Salesperson\_City, c.cname, c.city as Customer\_City from salespeople s, customer c where s.snum=c.snum and s.city=c.city order by snum

12.   Find the largest order taken by each salesperson.   
  
selectc.snum, max(o.amt) from customer c, orders o where c.cnum= o.cnum group by c.snum

13.   Find customers in San Jose who have a rating above 200.   
  
selectc.cname, c.rating from customer c where city='San Jose' having rating > 200

14.   List the names and commissions of all salespeople in London.   
  
selects.sname, s.com from salespeople s where city='London'

15.   List all the orders of salesperson Motika from the Orders table.   
  
select \* from orders where cnum in (select cnum from customer where snum=(select s.snum from salespeople s where s.sname='Motika'))

                                    OR

      select o.\* from orders o, customer c, salespeople s where c.cnum=o.cnum and s.snum=c.snum and s.sname="motika";

16.   Find all customers with orders on October 3.   
  
selectc.cname,o.cnum,o.odate,o.onum from customer c, orders o where c.cnum=o.cnum and o.odate like '%-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.   
  
selecto.odate, sum(o.amt) as Total from orders o 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 (select \* from orders o where o.odate like '%-06-10' ) A  where amt> (select  min(amt) from orders o where o.odate like '%-06-10' )

                                          OR

   select \* from orders  where amt> all (select amt from  orders where odate like '%-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 customer a where exists(select \* from salespeople b where a.snum=b.snum) and a.rating ="300";

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

selectc.rating, group\_concat(c.cname) from customer c group by c.rating;

                                                OR

selectc.cname, c.rating, c1.cname from customer c join  customer c1 on c.rating=c1.rating and c.cname!=c1.cname

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

select \* from customer c where c.cnum> ((select snum from salespeople where sname='Serres') + 1000)

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

selectsnum,sname, CONCAT((com\*100),'%') 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.

selects.snum, s.sname,max(o.amt) as MaxOrder, o.odate from salespeople s, customer c,orders o where s.snum=c.snum and c.cnum=o.cnum group by s.snum, o.odate having MaxOrder> 3000

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

selects.snum, s.sname,max(o.amt) as MaxOrder, o.odate from salespeople s, customer c,orders o where s.snum=c.snum and c.cnum=o.cnum and  o.odate like '%-03-10' group  by s.snum

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

select \* from customer c where c.city in (select c.city from salespeople s, customer c where c.snum=s.snum and  sname='Serres')

26. Select all customers with a rating above 200.00.

select \* from customer c having c.rating> 200

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

select count(distinct(c.snum)) from customer c, orders o where c.cnum= o.cnum

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.

selectc.cnum, c.cname, s.sname, s.snum, s.com from customer c, salespeople s where c.snum=s.snum and s.com > 0.12

29. Find salespeople who have multiple customers.

selects.snum, count(c.cnum) as 'No of Customer' from  salespeople s, customer c where c.snum=s.snum group by s.snum having count(c.cnum) > 1

30. Find salespeople with customers located in their city.

selects.snum, s.sname, s.city as Sales\_City, c.cnum, c.cname, c.city as Cust\_City from  salespeople s, customer c where c.snum=s.snum and s.city=c.city

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

select \* from salespeople where sname like 'P\_\_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 orders  where cnum=(select cnum from customer where cname='Cisneros')

33. Find the largest orders for Serres and Rifkin.

selects.sname, max(o.amt) from salespeople s, customer c, orders o  where s.snum=c.snum and c.cnum=o.cnum and s.sname in ('Serres','Rifkin') group by s.snum

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

selects.snum as SNUM, s.sname as SNAME,s.com as COMMISSION, s.city as CITY from salespeople

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

select \* from customer C where  substr(C.cname,1,1) between  'A' and 'G'

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

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

select \* from orders where odate like '%-04-10' having amt> (select avg(amt) from orders);

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 a.cnum, a.cname from customer a where a.rating = (  select max(rating) from customer b where a.city = b.city);

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

selects.odate, sum(s.amt) as Amount from orders s group by s.odate order by Amount desc

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

selectconcat(c.rating,', ', c.cname) as Customer  from customer c where c.city='San Jose'

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

select \* from orders o where amt< (select max(amt) from customer c, orders o where c.city='San Jose' and c.cnum=o.cnum)

                                                            OR

Select \* from orders where amt< any ( selectamt from orders o, customer c where city = 'San Jose' and  o.cnum = c.cnum);

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

select \* from orders a where amt> (  select avg(amt) from orders b where a.cnum = b.cnum group by cnum);

                                                            OR

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

selectc.city, max(c.rating) from customer c group by c.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.

selects.snum, s.sname,c.cname, o.amt, s.com\*o.amt as Commission  from salespeople s, customer c, orders o where s.snum=c.snum and c.cnum=o.cnum and c.rating> 100

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

select count(\*) from customer c where c.rating> (select avg(c1.rating) from customer c1 where c1.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 a,salespeople b order by a.snum,b.snumdesc;

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

select \* from salespeople s where s.city in ('Barcelona','London')

48. Find all salespeople with only one customer.

selects.snum, s.sname, count(c.cnum) as Counts from salespeople s, customer c where s.snum= c.snum group by s.snum having Counts = 1

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

selecta.snum, concat(a.cname,',',b.cname) as Customer\_Pair from customer a, customer b where  a.snum=b.snum and a.cname!=b.cname order by a.snum;

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

select \* from orders where amt> 1000

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.

selecto.onum, c.cname , o.cnum,  o.amt, o.odate from customer c, orders o where c.cnum=o.cnum order by o.onum

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 distinct(s.snum),s.sname from salespeople s, customer c where s.city=c.city and s.snum!=c.snum     -----------Using Join

Select s.snum, s.sname from salespeople s where exists ( selectc.cnum  from customer c where s.city = c.city and   s.snum != c.snum);

53. Write a query that selects all customers whose ratings are equal to or greater than ANY (in the SQL sense) of Serres’?

select \* from customer c where c.rating>= Any( select c1.rating from customer c1 where snum=(select snum from salespeople where sname='Serres'))

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

select \* from orders o where odate like '%-04-10' or  '%-03-10'  ---- only  from 04

select \* from orders o where odate like '%-04-10' or   odate like '%-03-10'

                                                OR

select \* from orders o where odate like '%-04-10' union  select \* from orders o where odate like '%-03-10'

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

select o1.cnum,c.cname, concat(o1.onum,',',o2.onum) as Order\_Pair from orders o1, orders o2, customer c where o1.cnum=o2.cnum and o1.onum!=o2.onum and o1.cnum=c.cnum order by o1.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.

selectc.cname, c.rating, c.city from customer c where rating >100 or city='Rome'

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

select c.\*  from customer c where c.snum=1001

59. Find the total amount in Orders for each salesperson for whom this total is greater than theamount of the largest order in the table.

selects.snum, s.sname, sum(o.amt) as Total from salespeople s, customer c, orders o where s.snum= c.snum and c.cnum= o.cnum group by s.snum having Total >  (select max(amt)  from orders)

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

select \* from orders where amt =! null

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.

selecta.sname,b.cname from salespeople a join customer b on a.snum=b.snum and b.rating> 200 order by sname,cname;

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

selects.sname, s.comm \* (sum(o.amt)) as Commission from salespeople s, customer c, orders o where s.snum=c.snum and c.cnum=o.cnum group by s.snum

63. Write a query that produces the names and cities of all customers with the same rating asHoffman. Write the query using Hoffman’s CNUM rather than his rating, so that it would still beusable if his rating changed.

selectc.cname, c.city from customer c where rating = (select rating from customer where cname='Hoffman') and c.cname!='Hoffman'

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

Select sname from salespeople where sname< any ( selectcname from customer   where salespeople.snum = customer.snum);

                                                                          OR

Select sname from salespeople where exists ( selectcname from customer where salespeople.snum = customer .snum and  salespeople.sname<customer.cname);

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

average orders.

selectc.cnum, c.cname from customer c, orders o where o.amt> (select avg(amt) from orders)  and c.cnum=o.cnum

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 inthe order table.

selecto.onum, o.amt, o.odate from orders o

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

select count(\*) from customer where rating =! null

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

after the order number.

selects.sname, c.cname, o.onum from salespeople s join customer c using (snum) join orders o using (cnum)

OR

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

selects.sname, s.comm, c.cname from salespeople s, customer c where s.snum=c.snum and c.city='London'

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

select \* from salespeople s where city !=Any (select city from customer c where s.snum=c.snum )

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 s.snum as SNUM, s.sname, c.cname, c.cnum as CSNUM, s.city as Scity, c.city as CCity  from salespeople s, customer c where exists (select \* from customer cc where s.city=cc.city) and s.snum!=c.snum and s.city=c.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.)

selectcnum,cname from customer where snum=any(select snum from salespeople where sname="peel" or 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.)

selecto.odate, count(distinct(s.snum)) from salespeople s, customer c, orders o where s.snum=c.snum and c.cnum= o.cnum group by o.odate

75. Find all orders attributed to salespeople in London.

selects.sname, s.city, o.onum,o.odate, o.amt, c.cname from salespeople s, customer c, orders o where s.snum=c.snum and c.cnum= o.cnum and s.city='London'

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

select c.cname, c.cnum, c.city as Customer\_City, s.snum, s.city as Salesperson\_City,  o.onum, o.odate, o.amt from salespeople s, customer c, orders o where s.snum=c.snum and c.cnum= o.cnum and s.city!=c.city

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

selects.snum, s.sname, count(c.cnum) as Customer\_Count from salespeople s, customer c, orders o where s.snum=c.snum and c.cnum= o.cnum group by s.snum having Customer\_Count> 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.

selects.snum, s.sname, count(c.cnum) as Customer\_Count from salespeople s, customer c, orders o where s.snum=c.snum and c.cnum= o.cnum group by s.snum having Customer\_Count> 1

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).

selectc.city, max(c.rating) from customer c group by c.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 distinct(s.snum) from salespeople s, customer c, orders o where s.snum=c.snum and c.cnum=o.cnum

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.

selectc.rating, c.cname, c.cnum from customer c order by c.ratingdesc

83. Find the average commission for salespeople in London.

selectavg(comm) from salespeople where city='London'

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

select o.\* from orders o where o.cnum in (select cnum from customer where snum=(select snum 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 \* from salespeople where comm> .10  and city='London'

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));

All records from order except date is 03-oct 1990 with cnum>2003 or orders  
where amt is less then 1000

# 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, 1098.16, 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 o.\*, min(amt) from orders o group by cnum

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

Select min(cname) 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(distinct(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);

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);

---------------------------------------------------------------------------------------------------------------------------------

95. Evaluate the following query :

SELECT \* FROM orders

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

**Error Code: 1054. Unknown column 'snum' in 'where clause'      0.000 sec**

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

selects.sname, s.city as Salesperson\_City, c.cname, c.city as Customer\_city  from salespeople s, customer c where s.snum=c.snum and s.city!=c.city

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

250?

selects.sname,s.comm,c.cname, c.rating  from salespeople s, customer c where s.snum=c.snum and s.comm> 0.11 and c.rating> 250

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

percentages?

select s1.sname,s1.city, s1.comm, s2.sname, s2.city, s2.comm from salespeople s1, salespeople s2 where s1.city=s2.city and s1.snum!=s2.snum and s1.comm!=s2.comm

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

selecta.snum, a.sname, max(a.Commission) from (select s.snum,s.sname, s.comm\*(sum(o.amt)) as Commission from salespeople s, customer c, orders o where s.snum=c.snum and c.cnum = o.cnum group by s.snum) a

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

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

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?

selectc.cname, o.odate from customer c, orders o where c.cnum=o.cnum and  c.cname='Hoffman'

104.Do all salespeople have different commissions?

selecta.sname,a.comm, b.sname, b.comm from salespeople a, salespeople b where a.comm=b.comm and a.snum!=b.snum

If Rowcount is ZERO means all salespeoples have different commission.

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

selects.sname,o.onum, o.odate from salespeople s, customer c, orders o where s.snum=c.snum and  c.cnum=o.cnum and  o.odate not between '1996-03-10' and '1996-05-10'

106.How many salespersons have succeeded in getting orders?

select count(distinct(s.snum)) from salespeople s, customer c, orders o where s.snum=c.snum and  c.cnum=o.cnum

107.How many customers have placed orders?

select count(distinct(c.cnum)) from customer c, orders o where c.cnum=o.cnum

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

selects.sname, max(o.amt), o.odate from salespeople s, customer c, orders o where s.snum=c.snum and  c.cnum=o.cnum group by s.snum

109.Who is the most successful salesperson?

selects.sname, (s.comm \* sum(o.amt)) as Income from salespeople s, customer c, orders o where s.snum=c.snum and c.cnum=o.cnum group by  s.snum order by Income desc

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

selectc.cname, o.cnum, sum(o.amt) as Total\_Amount from customer c, orders o where c.cnum=o.cnum group by o.cnum order by Total\_Amount limit 1

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

salespersons Peel or Serres?

112.Which customers have the same rating?

select c1.cname, c1.rating, c2.cname from customer c1, customer c2 where c1.rating = c2.rating and c1.cname!=c2.cname

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

select \* from orders where amt> (select avg(amt) from orders where odate='1996-04-10')

114.Which customers have above average orders?

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

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

select \* 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 s.sname, sum(o.amt) as Total from salespeople s, customer c, orders o where s.snum=c.snum and c.cnum=o.cnum  group by s.sname having  Total > (select max(amt) from orders)

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

selects.sname,s.snum,  count(c.cname) as Customer\_count from salespeople s, customer c where s.snum=c.snum  group by s.sname having Customer\_count> 1

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

don’t service.

 selects.snum, s.sname from salespeople s, customer c where s.city=c.city and s.snum!=c.snum

119.Which customers’ rating should be lowered?

selectc.cnum, c.cname, sum(o.amt) as Total from customer c, orders o where c.cnum=o.cnum group by c.cnum order by Total limit 1

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

selects.snum, s.sname, c.cname from salespeople s, customer c where s.snum=c.snum and c.city='Berlin'

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

she is being paid?

No

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?

select sum(amt) from orders a,customer b where a.cnum=b.cnum and (b.city="london" or "rome");

select (sum(a.amt)\*(c.com))\*5 as comm\_paid from orders a,customerb,salespeople c where a.cnum=b.cnum and b.snum=c.snum and (c.city="london" or c.city="new york");

---------------------------------------------------------------------------------------------------------------------------------