Queries on SQL * PLUS

Structure of Salespeople Table

SNUM	NOT NULL NUMBER(4) primary key
SNAME	VARCHAR2(15)
CITY	VARCHAR2(15)
COMM	NUMBER(10,2)

Structure of Customers table

CNUM	NOT NULL NUMBER(4) primary key
CNAME	VARCHAR2(15)
CITY	VARCHAR2(10)
RATING	NUMBER(3)
SNUM	NUMBER(4) foreign key

Structure of Orders table

ONUM	NOT NULL NUMBER(4) primary key
AMT	NUMBER(10,2)
ODATE	DATE
CNUM	NUMBER(4) foreign key
SNUM	NUMBER(4) foreign key

Salespeople Table

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	.1

Customers Table

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	PEREEIA	ROME	100	1004

Orders Table

ONUM	AMT	ODATE	CNUM	SNUM
3001	18.69	10-MAR-90	2008	1007
3003	767.19	10-MAR-90	2001	1001
3002	1900.1	10-MAR-90	2007	1004
3005	51.6	10-MAR-90	2003	1002
3006	1098.16	10-MAR-90	2008	1007
3009	1713.23	10-APR-90	2002	1003
3007	75.75	10-APR-90	2004	1002
3008	4723	10-MAY-90	2006	1001
3010	1309.95	10-JUN-90	2004	1002
3011	9891.88	10-JUN-90	2006	1001

1. Write a query that produces the order number, amount for all the rows in the ORDER table.

2. Write a query that produces the row from the Customer table with customer number 1001.

3. Write a query that produces the following columns from the salesperson table.
   CITY SNAME SNUM COMM

4. Write a query that produces the rating followed by the name of the employee in San Jose.

5. Write a query that will produce the snum values of all salespeople with orders currently in Orders table without any duplicates.

6. Write a query that will give you all the orders worth over $1000.

7. Write a query that will give the names and cities of all the salespeople in London with a commission above 0.10.

8. Write a query on the customer table that will exclude all the customers with a rating less than 100 unless they are located in Rome.

9. What will be the output of this query?

   SELECT * FROM ORDERS WHERE (AMT <1000 OR
   NOT(ODATE='10-MAR-1990' AND CNUM>2003));

10. What will be the output of this query?

    SELECT * FROM ORDERS WHERE NO(ODATA='10-MAR-90' OR SNUM >1006)
    AND AMT>=1500);

11. What is a simpler way to write this query ?

    SELECT SNUM,SNAME,CITY,COMM FROM SALESPEOPLE WHERE (COMM >.12 OR <.14);

12. Write two queries that will produce all the orders taken on October 3rd or 4th 1990.

13. Write a query that will select all the customers serviced by Peel or Motika.

14. Write a query that will produce a listing of the customers whose names begin with A or G.

15. Write a query that will select all Orders, which have zeros, or Null in the amount field.

16. Write a query that will display all the customers who neither stay in London or in San Jose.

17. Write a query to find the sum, average, max, min, count of all the orders from the orders table.

18. Write a query to find the sum of unique orders in the orders table.

19. Write a query to count the number of customers in the customer's table.

20. Write a query to count the number of unique customers in the orders table.

21. Write a query that counts all the Orders for the month of October.

22. Write a query that counts the number of different non-null city values in the Customer table.

23. Write a query that selects each customer's smallest order.

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

25. Write a query that counts the number of salespersons registering orders on a particular day.

26. Assume that all the salespersons have 12% commission write a query on the Orders table that will produce the order numbers, the sales person number, the amount and the commission for that order.

27. Write a query on the customer table that will find the highest rating in each city.

28. Write a query that will display the maximum order placed by every salesperson.

29. Write a query that will display the maximum order placed by every salesperson on 10 mar 90.

30. Write a query that lists the customers in the descending order of rating output the rating field first followed by the name and the number.

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

32. Write a query to display all the information, including the order details from the customers and salespeople, who don't belong to the same city.

33. Write a query that lists the order number followed by the customer who placed the order.

34. Write a query that lists orders with the names of the customers and the salesperson.

35. Write a query that produces all the customers who are serviced by salespersons with a commission of more that 12%.

36. Write a query that calculates the salesperson commission on each order by a customer with a rating of over 100.

37. Write a query to display all the customers in the same cities as those of Serres.

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

39. Write a query to display the names of the salespeople, who give service for the customers Clemens and Liu.

40. Write a query to display the information of all the orders placed by customer with cnum as 2001.(Use distinct clause with sub query).

41. Write a query to display the information of all the orders placed by salespeople who stay in London.

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

43. Write a query to find customers with a rating above San Jose's average.

44. Write a query to display the list of salespeople who stay in the same city as any customer.

45. Retrieve all the records of salespeople and customers who belong to the same city.

46. Select all the orders placed by Motika.

47. Retrieve all the information of the salespeople and their corresponding orders for those who stay in Barcelona.

48. Retrieve all the orders for snum = 1001.

49. Retrieve all the orders placed on 1st May 90 where the amount is greater than the average.

50. Retrieve all the orders placed by salesperson belonging to London.

51. Select all the information from customers whose rating is greater than the average rating of customers belonging to San Jose.

52. Select all the records whose rating is greater than any of the rating from Rome.

53. Select all the records whose amount is greater than any of the amount collected on 02-may-90.

54. Retrieve all the records whose amount is lesser than any amount for a customer in San Jose.

55. Select all the records whose rating is greater than all the ratings from Rome.

56. Retrieve all the salespeople and customers located in London (Using Union).

57. Display the highest and the lowest order bagged by each salesperson.

58. Assume there is a table called Multicust, with all the columns as Salespeople. Write a SQL statement that inserts all the salespeople with more than one customer into this table.

59. Create a view that shows the average and total orders for each salesperson after his or her name. Assume all names are unique.

60. Create a view that shows each salespersons with multiple customers.

61. Change the name of the salesperson to Gibson whose snum is equal to 1001.

62. Create a view with cnum cname , onum, and odate for all the matching records.

ORACLE ASSIGNMENT:(PERFORM ON ORACLE DEFAULT TABLE)

1)Write a query to display the following details (from table employee,department and salary_grade):

(empno ename dname sal grade)

2)Write a query to display Highest paid employee in each Deparment.

3)Write a query to display the empno,sal,comm from the employee table.If the comm is null then display
the message as "NOT APPLICABLE"

4)Write a query to display every 4th row.(4th,8th,.....and so on)

5)Write a query to display the employees who have joined first in their respective department.

6)write a query which displays empno,empname,mgrno and manager name.

7)write a query to select nth row from employee table.

8)write a query to display data as below :(select job , total sal for each job in each department)

Job	Dept10	Dept20	Dept30	Dept40
Analyst	Sum(sal)	sum(sal)	sum(sal)	sum(sal)
Clerk	....	.....	.....	.....
.....	.....	....	,....	.....

9)Write a query to make use of rollup(),cube(),merge,case statement.