

UNIT - 3

1. What is SQL?
2. Explain different types (category) of SQL commands.
3. Explain following commands. Explain its purpose, syntax with all options available.
 - a. Create table
 - b. Insert
 - c. Update
 - d. Drop table
 - e. Drop view
 - h. Create view
 - i. Delete
 - j. Select
 - k. Alter table
4. What are data types?
5. Explain various data types that oracle supports and its characteristics.
6. What is JOIN? Explain Outer Join.
7. What is sub query? Why it requires?
8. Explain or write in brief about
 - f. Table
 - b. View
 - c. Sequence
 - d. Index
9. What do you mean by TPL? List the command of TPL categories.
10. What type of constraints we can define in oracle?
11. Explain each constraint supported by oracle.
12. Explain the following operators
 - g. In
 - b. Between
 - c. Like
 - d. Is null
 - e. Distinct
 - f. Exist
13. What are the aggregate functions?
14. When can you make a union between queries?
15. Is join view possible? if yes give example of that.
16. Explain when can you change records in view?
17. Explain what COMMIT and ROLLBACK command does?
18. write following query statements for following statements (Please Consider tables given at the end of assignment)
 1. Write a SELECT command that produces the order number, amount and date for all rows in the order table.
 2. Write a query that produces all rows form the customer table for which the salesperson's number is 1001.
 3. Write queries that produce the salesperson table with the columns in the following order: city, name, snum, comm.
 4. Write a SELECT command that produces the rating followed by the name of each customer in San Jose.
 5. Write a query that will produce the snum values of all salespeople with orders currently in the orders table without any repeats.
 6. Write a query that will give you all orders for more than \$1,000.
 7. Write a query that will give you the names and cities of all salespeople in London with a commission above 10.

8. Write a query on the Customers table whose output will exclude all customers with a rating ≤ 100 , unless they are located in Rome.
9. What will be the output from the following Query?

```
SELECT * FROM ORDERS WHERE
(AMT < 1000 OR NOT (ODATE=10/03/1990 AND CNUM>20003));
```

10. What will be the output of the following query?

```
SELECT * FROM ORDERS
WHERE NOT ((ODATE=10/03/1990 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 COMM<.14);
```

12. Write two queries that will produce all orders taken on October 3rd or 4th, 1990.
13. Write a query that select all of the customers serviced by Peel or Motika.
14. Write a query that will produce all of the customers whose names begin with a letters from A to G.
15. Write a query that selects all customers whose name beginning with C.
16. Write a query that select all orders save those with zeroes or NULLs in the amt (amount) field.
17. Write a query that counts all orders for October 3.
18. Write a query that count the numbers of different non-NULL city value in the Customers Table.
19. Write a query that selects each customer's smallest order.
20. Write a query that selects the first customers, in alphabetical order, whose name begins with G.
21. Write a query that selects the highest rating in each city.
22. Write a query that counts 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.)
23. Assume each salespeople has a 12% commission. Write a query on the orders table that will produce the order number, the salesperson number, and the amount of the salesperson's commission for that order.
24. Write a query on the customers table that will find the highest rating in each city. Put the output in thisform:

for the city (city), the highest rating is (rating).
25. Write a query that list customers in descending order of rating. Output the rating field first, followed by the customer's name and number.
26. Write a query that totals the orders for each day and place that results in descending orders.
27. Write a query that lists each orders number followed by the name of the customer who made the order.

28. Write a query that gives the names of both the salesperson and the customer for each order after the order number.
29. Write a query that produces all customers serviced by salespeople with a commission above 12%. Output the customer's name, the salesperson's name, and the sales person's rate of commission.
30. Write a query that calculates the amount of the salesperson's commission on each order by a customer with a rating above 100.
31. Write a query that produces all pairs of salespeople who are living in the same city. Exclude combinations of salespeople with themselves as well as duplicate rows with the order reversed.
32. Write a query that produces all pairs of orders by a given customers, names that customer, and eliminates duplicates, as above.
33. 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 rating, so that it would still be usable if his rating changed.
34. Write a query that uses a subquery to obtain all orders for the customer named Cisneros. Assume you do not know his customer number (cnum).
35. Write a query that produces the names and ratings of all customers who have above average orders.
36. Write a query that selects the total amount in orders for each salesperson for whom this total is greater than the amount of the largest order in the table.
37. 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.
38. Write two 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 correlated sub query. Which solution is more elegant?
39. Write a query that use the exists operator to extract all salespeople who have customers with a rating of 300.
40. How could you have solved the above problem with a join?
41. Write a query using the exists operator that selects all salespeople with customers located in their cities who are not assigned to them.
42. 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.
43. Write a query that selects all customers whose rating are equal to or greater than any (in SQL sense) of Serres'
44. What would be the output of the above command?
45. Write a query using ANY or ALL that will find all sales people who have no customers located in their city?
46. Write a query that selects all orders for amounts greater than any (in the usual sense) for the customers in London.
47. Write above query using MAX.
48. Create a union of two queries that shows the names, cities and ratings of all customers. Those with a rating of 200 or greater will also have the words "High Rating" with the others will have the words "Low Rating"

49. Write a command that produce the name and number of each salesperson and each customer with more than one current order. Put the result in alphabetical order.
50. form a union of three queries. Have the first select the snums of all salespeople in San Jose; the second, the cnums of all customers in San Jone; and the third the onnums of all orders on October 3. Retain duplicates between the last two queries but eliminate any redundancies between either of them and the first.

Table that are used in queries

Table : Salespeople	
Snum	Primary key (Salesperson number)
Sname	Name of sales person
City	Location of sales person
Comm.	The salesperson's commission on orders in decimal form
Table : customer	
Cnum	Primary key (Customer Number)
Cname	The name of customer
City	City of customer
Rating	A numeric code indicating level of performance given this customer
Snum	The number of salesperson assign to this customer
Table : orders	
Onum	Primary key (Order Number)
Amt	The amount of the purchase
Odate	The date of purchase
Cnum	Reference to the table customer (The number of the customer making the purchase)
Snum	Reference to the table salesperson (The number of the salesperson credited with sale)

Data for above three tables :

Table Salespeople

SNUM	SNAME	CITY	COMM
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

Table 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

Table Orders

Onum	Amt	Odate	Cnum	Snum
3001	18.69	10/03/1990	2008	1007
3003	767.19	10/03/1990	2001	1001
3002	1900.19	10/03/1990	2007	1004
3005	5160.45	10/03/1990	2003	1002
3006	1098.16	10/03/1990	2008	1007
3009	1713.23	10/04/1990	2002	1003
3007	75.75	10/04/1990	2004	1002
3008	4723.00	10/05/1990	2006	1001
3010	1309.95	10/06/1990	2004	1002
3011	9891.88	10/06/1990	2006	1001

