**Topic – SQL Task-1**

1.Table = employee and incentives

a) Get First\_Name from employee table using alias name “Employee Name”.

SELECT First\_Name AS 'Employee Name' FROM employee;

b) Get FIRST\_NAME, Joining year, Joining Month and Joining Date from employee Table.

Answer SELECT First\_Name,JOINING\_DATE FROM employee;

c) Get all employee details from the employee table order by First Name Ascending And Salary descending?

SELECT \*

FROM employee

ORDER BY FIRST\_NAME ASC, SALARY DESC;

d) Get employee details from employee table whose first name contains „o‟.

SELECT \*

FROM employee

WHERE FIRST\_NAME LIKE '%o%';

e) Get employee details from employee table whose joining month is “January”.

SELECT \* FROM employee WHERE JOINING\_DATE='2013-01-01'

f) Get department, total salary with respect to a department from employee table Order By total salary descending.

SELECT DISTINCT DEPARTMENT, SUM(SALARY) AS Total\_Salary

FROM employee

GROUP by DEPARTMENT

ORDER BY Total\_Salary DESC;

g) Get department wise maximum salary from employee table order by salary Ascending?

SELECT DEPARTMENT, MAX(SALARY)

FROM employee

GROUP BY DEPARTMENT

ORDER BY SALARY ASC;

h) Select first\_name, incentive amount from employee and incentives table for those Employees who have incentives and incentive amount greater than 3000

SELECT FIRST\_NAME, INSENTIVE\_AMT

FROM employee

INNER JOIN incentives WHERE INSENTIVE\_AMT > '3000';

i) Select 2nd Highest salary from employee table.

SELECT MAX(SALARY)

FROM employee

WHERE SALARY < (SELECT MAX(SALARY) FROM employee);

j) Select first\_name, incentive amount from employee and incentives table for all Employees who got incentives using left join.

SELECT FIRST\_NAME, INSENTIVE\_AMT

FROM employee

LEFT JOIN incentives

ON employee.FIRST\_NAME = incentives.INSENTIVE\_AMT;

k) Create View OF Employee table in which store first name, last name and salary only.

CREATE VIEW EmployeeInfo AS

SELECT first\_name, last\_name, salary

FROM Employee;

l) Create Procedure to find out department wise highest salary.

SELECT DEPARTMENT, FIRST\_NAME, SALARY

FROM employee

WHERE (DEPARTMENT, SALARY) IN ( SELECT DEPARTMENT, MAX(SALARY)

FROM employee

GROUP BY DEPARTMENT

);

m) Create after Insert trigger on Employee table which insert records in view table.

**Task-2:**

TABLE = SALES PERSON,ORDER,CUSTOMER

a) All orders for more than $1000.

SELECT AMT FROM order WHERE AMT>1000;

b) Names and cities of all salespeople in London with commission above 0.10.

SELECT SNAME,CITY,COMM FROM sales person WHERE CITY='LONDON' AND COMM>'O.10';

c) All salespeople either in Barcelona or in London.

SELECT \* FROM sales person WHERE CITY= 'BERCELONA' OR CITY= 'LONDON';

d) All salespeople with commission between 0.10 and 0.12. (Boundary values should be excluded).

SELECT \* FROM sales person WHERE COMM BETWEEN 0.10 AND 0.12;

e) All customers with NULL values in city column.

SELECT \* FROM customer WHERE CITY IS NULL;

f)All orders taken on Oct 3Rd and Oct 4th 1994.

SELECT \* FROM order

WHERE ODE = '1994-10-03' OR ODE = '1994-10-04';

g) All customers serviced by peel or Motika.

SELECT \* FROM sales person WHERE SNO='1001' OR SNO='1004';

h) All customers whose names begin with a letter from A to B

SELECT \* FROM customer WHERE CNAME LIKE 'A%' OR CNAME LIKE 'B%';

i) All customers excluding those with rating <= 100 unless they are located in Rome.

SELECT \* FROM customer WHERE RATING <= '100' OR CITY='ROME';

j) All orders except those with 0 or NULL value in amt field.

SELECT \* FROM order WHERE amt IS NOT NULL ;

k) Count the number of salespeople currently listing orders in the order table.

SELECT SNAME, COUNT(ORDER.SNO) FROM sales person, order WHERE sales person.SNO = order.SNO;