**TASK 1**

**TABLE – EMPLOYEE/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.

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;

SELECT \* FROM `employee` order by salary DESC;

D] Get employee details from employee table whose first name contains ,,o”.

SELECT \* from employee WHERE name like '%o';

E] Get employee details from employee table whose joining month is “January”.

SELECT \* FROM employee WHERE month (JOINING\_DATE)='01';

F] Get department , total salary with respect to a department from employee table order by total salary descending.

SELECT DEPARTMENT,sum(SALARY) 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 MAX (salary) as max\_salary from employee order BY SALARY ASC;

H] select first\_name , incentive amount from employee and incentive table for those employees who have incentives and incentive amount greater than 3000.

SELECT FIRST\_NAME , INCENTIVE\_AMT FROM employee INNER JOIN incentives ON employee.EM\_ID = incentives.EMPLOYEE\_REF\_ID AND INCENTIVE\_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 , INCENTIVE\_AMT from employee RIGHT JOIN incentives on employee . EM\_ID = incentives . EMPLOYEE\_REF\_ID;

K] create view of employee table in which store first name , last name and salary only.

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) FIRST\_NAME , LAST\_NAME , SALARY  FROM employee;

L] create procedure to find out department wise highest salary.

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) DEPARTMENT, [MAX](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/aggregate-functions.html%23function_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/CUSTOMER/ORDER**

A] All order for more than $1000

SELECT\* from orderr where amt > 1000;

B] Names and Cities of all salespeople in London with commission above 0.10.

SELECT sname, city from sales\_person where comm > 0.10 [and](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/logical-operators.html%23operator_and) city = 'London';

C] All salespeople either in Barcelona or in London.

SELECT sname, city from sales\_person where city in (‘BARCELONA’,’LONDON’);

D] All salespeople with commission between 0.10 and 0.12.

(boundary values should be excluded)

SELECT sname, comm from sales\_person where comm > 0.10 and comm < 0.12;

E] All customers with NULL values in city column.

SELECT city from sales\_person WHERE city=null ;

F] All orders taken on oct 3rd and oct 4th 1994

SELECT \* from orderr where ODE in (‘1994-10-03’,’1994-10-04’);

G] All customers serviced by peel or motika

SELECT \* FROM sales\_person WHERE SNAME in ('peel','motika');

H] All customers whose names begin with a letter from A to B

SELECT cname 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 cname from customer where rating <= 100 OR city = ‘Rome’;

J] All order except those with 0 or NULL value in amt field.

[SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/8.0/en/select.html) AMT from orderr WHERE '0';

K] count the number of salespeople currently listing orders in the order table.

SELECT count (distinct SNO) from orderr;