**Assignment Module-1 MYSQL**

**Task-1**

1. **Get First\_Name from employee table using alias name “Employee Name”.**

**Ans:-** select first\_name as Employee\_name from employee;

1. **Get FIRST\_NAME, Joining year, Joining Month and Joining Date from employee table.**

**Ans:-** select first\_name,date(joining\_date) as Joining\_Date,month(joining\_date) as Joining\_Month,year(joining\_date) as Joining\_Year from employee;

1. **Get all employee details from the employee table order by First Name Ascending And Salary descending?**

**Ans:-** select \* from employee ORDER by salary DESC,first\_name;

1. **Get employee details from employee table whose first name contains „o‟.**

**Ans:-** SELECT \* FROM employee WHERE first\_name LIKE "%O%";

1. **Get employee details from employee table whose joining month is “January”.**

**Ans:-** SELECT \* FROM employee WHERE MONTH(joining\_date)=1;

**f) Get department, total salary with respect to a department from employee table**

**Order By total salary descending.**

**Ans:-** SELECT 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?**

**Ans:-** SELECT department,max(salary)from employee GROUP by department ORDER by salary;

**h) Select first\_name, incentive amount from employee and incentives table for those**

**Employees who have incentives and incentive amount greater than 3000**

**Ans:-** SELECT E.FIRST\_NAME, I.INCENTIVE\_AMT FROM Employee E JOIN Incentives I ON E.EM\_ID =I.EMPLOYEE\_REF\_ID WHERE I.INCENTIVE\_AMT > 3000;

1. **Select 2nd Highest salary from employee table.**

**Ans:-** SELECT MAX(SALARY) AS Second\_Highest\_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.**

**Ans:-** SELECT E.FIRST\_NAME, I.INCENTIVE\_AMT FROM Employee E LEFT JOIN Incentives I ON E.EM\_ID = I.EMPLOYEE\_REF\_ID;

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

**Ans:-** CREATE VIEW EmployeeView AS SELECT FIRST\_NAME, LAST\_NAME, SALARY FROM Employee SELECT \* FROM EmployeeView;

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

**Ans:-** DELIMITER $$

CREATE PROCEDURE GetDepartmentWiseHighestSalary()

BEGIN

SELECT DEPARTMENT, MAX(SALARY) AS Highest\_Salary

FROM Employee

GROUP BY DEPARTMENT;

END $$

DELIMITER ;

Call GetDepartmentWiseHighestSalary();

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

**Ans:-** CREATE TABLE EmployeeViewTable (

FIRST\_NAME VARCHAR(50),

LAST\_NAME VARCHAR(50),

SALARY INT

);

DELIMITER $$

CREATE TRIGGER after\_employee\_insert

AFTER INSERT ON Employee

FOR EACH ROW

BEGIN

INSERT INTO EmployeeViewTable (FIRST\_NAME, LAST\_NAME, SALARY)

VALUES (NEW.FIRST\_NAME, NEW.LAST\_NAME, NEW.SALARY);

END $$

DELIMITER ;

**Task-2**

1. **All orders for more than $1000.**

**Ans:-** SELECT \* FROM orders WHERE AMT > 1000;

1. **Names and cities of all salespeople in London with commission above 0.10.**

**Ans:-** SELECT SNAME, CITY FROM SalesPerson WHERE CITY = 'LONDON' AND COMM > 0.10;

1. **All salespeople either in Barcelona or in London.**

**Ans:-** SELECT SNAME, CITY FROM SalesPerson WHERE CITY IN ('BARCELONA', 'LONDON');

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

**be excluded).**

**Ans:-** SELECT SNAME, COMM FROM SalesPerson WHERE COMM > 0.10 AND COMM < 0.12;

**e) All customers with NULL values in city column.**

**Ans:-** SELECT \* FROM Customer WHERE CITY IS NULL;

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

**Ans:-** SELECT \* FROM Orders WHERE ODE IN ('1994-10-03', '1994-10-04');

**g) All customers serviced by peel or Motika.**

**Ans:-** SELECT C.CNAME FROM Customer C JOIN SalesPerson S ON C.SNO = S.SNO WHERE S.SNAME IN ('PEEL', 'MOTIKA');

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

**Ans:-** 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.**

**Ans:-** SELECT \* FROM Customer WHERE (RATING > 100) OR (CITY = 'ROME');

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

**Ans:-** SELECT \* FROM Orders WHERE AMT > 0;

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

**Ans:-** SELECT COUNT(DISTINCT SNO) AS SalesPersonCount FROM Orders;