**SQL-QUERIES ASSIGNMENT**

* **CREATE DATABASE**

create database assignment;

* **CREATE TABLE: Employee**

create table Employee

(

EM\_ID int primary key AUTO\_INCREMENT,

FIRST\_NAME varchar(50),

LAST\_NAME varchar(50),

SALARY float,

JOINING\_DATE date,

DEPARTMENT varchar(50)

);

* **CREATE TABLE: Incentives**

create table Incentives

(

incentive\_id int primary key AUTO\_INCREMENT,

EMPLOYEE\_REF\_ID int references Employee(EM\_ID),

INCENTIVE\_DATE date,

INCEMTIVE\_AMT float

);

* **INSERT DATA INTO Employee TABLE**
* **EMPLOYEE 1**

insert into Employee(EM\_ID, FIRST\_NAME, LAST\_NAME, SALARY, JOINING\_DATE, DEPARTMENT) values(1, "JOHN", "ABRAHAM", 1000000, "2013-01-01", "BANKING");

* **EMPLOYEE 2**

insert into Employee(EM\_ID, FIRST\_NAME, LAST\_NAME, SALARY, JOINING\_DATE, DEPARTMENT) values(2, "MICHAEL", "CLERK", 800000, "2013-01-01", "INSURANCE");

* **EMPLOYEE 3**

insert into Employee(EM\_ID, FIRST\_NAME, LAST\_NAME, SALARY, JOINING\_DATE, DEPARTMENT) values(3, "ROY", "THOMAS", 700000, "2013-02-01", "BANKING");

* **EMPLOYEE 4**

insert into Employee(EM\_ID, FIRST\_NAME, LAST\_NAME, SALARY, JOINING\_DATE, DEPARTMENT) values(4, "TOM", "JOSE", 600000, "2013-02-01", "INSURANCE");

* **EMPLOYEE 5**

insert into Employee values("5", "JERRY", "PINTO", 650000, "2013-01-01", "INSURANCE");

* **EMPLOYEE 6**

insert into Employee values(6, "PHILIP", "MATHEW", 750000, "2013-01-01", "SERVICES");

* **EMPLOYEE 7**

insert into Employee values (7, "TESTNAME1", "123", 650000, "2013-01-01", "SERVICES");

* **EMPLOYEE 8**

insert into Employee values(8, "TESTNAME2", "LNAME%", 600000, "2013-02-01", "INSURANCE");

* **INSERT DATA INTO TABLE: Incentives**
* **INCENTIVE 1**

**AlTER THE NAME OF COLUMN INCEMTIVE\_AMT TO INCENTIVE\_AMT**

alter table incentives change INCEMTIVE\_AMT INCENTIVE\_AMT float;

insert into Incentives(incentive\_id, EMPLOYEE\_REF\_ID, INCENTIVE\_DATE, INCENTIVE\_AMT) values(1, 1, "2013-02-01", 5000);

* **INSENTIVE 2**

insert into Incentives(incentive\_id, EMPLOYEE\_REF\_ID, INCENTIVE\_DATE, INCENTIVE\_AMT) values(2, 2, "2013-02-01", 3000);

* **INCENTIVE 3**

insert into Incentives values(3, 3, "2013-02-01", 4000);

* **INCENTIVE 4**

insert into Incentives values(4, 1, "2013-01-01", 4500);

* **INCENTIVE 5**

insert into Incentives values(5, 2, "2013-01-01", 3500);

**TASK-1**

1. **Get FIRST\_NAME from Employee table using alias name “Employee Name”.**

select FIRST\_NAME as "First Name" from Employee;

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

select FIRST\_NAME, JOINING\_DATE from Employee;

1. **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;

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

select \* from Employee where FIRST\_NAME like "%o%";

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

select \* from Employee where MONTH(JOINING\_DATE)=1;

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

select DEPARTMENT as "Department", sum(SALARY) as "Salary" from Employee group by DEPARTMENT order by SALARY desc;

1. **Get department wise maximum salary from employee table order by salary ascending**

select DEPARTMENT as "Department", max(SALARY) as "Salary" from Employee group by DEPARTMENT order by SALARY asc;

1. **Select first\_name, incentive amount from employee and incentives table for those Employees who have incentives and incentive amount greater than 3000**
2. **Select 2nd Highest salary from employee table.**

select max(SALARY) from Employee where SALARY < (select max(SALARY) from Employee);

1. **Select first\_name, incentive amount from employee and incentives table for all Employees who got incentives using left join.**
2. **Create View OF Employee table in which store first name, last name and salary only.**

create view Employee\_salary as select FIRST\_NAME, LAST\_NAME, SALARY from Employee;

1. **Create Procedure to find out department wise highest salary.**
2. **Create after Insert trigger on Employee table which insert records in view table.**

**TASK-2**

**CREATE TABLE: SALES\_PERSON**

create table SALES\_PERSON

(

SNO int primary key AUTO\_INCREMENT,

SNAME varchar(50),

CITY varchar(50),

COMM float

);

**CREATE TABLE: CUSTOMER**

create table CUSTOMER

(

CNM int primary key AUTO\_INCREMENT,

CNAME varchar(50),

CITY varchar(50),

RATING int(5),

SNO int references SALES\_PERSON(SNO)

);

**CREATE TABLE: ORDER\_INFO**

create table ORDER\_INFO

(

ONM int primary key AUTO\_INCREMENT,

AMT float,

ODE date,

CNM int references CUSTOMER(CNM),

SNO int references SALES\_PERSON(SNO)

);

**INSERT DATA INTO TABLE: SALES\_PERSON**

**SALES\_PERSON: 1001**

insert into SALES\_PERSON(SNO, SNAME, CITY, COMM) values (1001, "PEEL", "LONDON", 0.12);

**SALES\_PERSON: 1002, 1003, 1004, 1007**

insert into SALES\_PERSON values(1002, "SERRES", "SAN JOSE", 0.13), (1003, "AXELROD", "NEW YORK", 0.1), (1004, "MOTIKA", "LONDON", 0.11), (1007, "RAFKIN", "BARCELONA", 0.15);

**INSERT DATA INTO TABLE: CUSTOMER**

**INSERT DATA OF CUSTOMER: 201, 202, 203, 204, 206, 207**

insert into CUSTOMER values(201, "HOFFMAN", "LONDON", 100, 1001), (202, "GIOVANNE", "ROME", 200, 1003), (203, "LIU", "SAN JOSE", 300, 1002), (204, "GRASS", "BARCELONA", 100, 1002), (206, "CLEMENS", "LONDON", 300, 1007), (207, "PEREIRA", "ROME", 100, 1004);

**INSERT DATA INTO TABLE: ORDER\_INFO**

**INSERT ORDER DATA: 3001, 3002, 3003, 3005, 3006**

insert into ORDER\_INFO values(3001, 18.69, "1994-10-03", 201, 1007), (3002, 1900.1, "1994-10-03", 207, 1004), (3003, 767.19, "1994-10-03", 201, 1001), (3005, 3005, "1994-10-03", 203, 1002), (3006, 3006, "1994-10-04", 201, 1007);

**INSERT ORDER DATA: 3007, 3008, 3009, 3010, 3011**

insert into ORDER\_INFO values(3007, 3007, "1994-10-05", 204, 1002), (3008, 3008, "1994-10-05", 206, 1001), (3009, 3009, "1994-10-04", 202, 1003), (3010, 3010, "1994-10-06", 204, 1002), (3011, 3011, "1994-10-06", 206, 1001);

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

select \* from ORDER\_INFO where AMT > 1000;

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

select SNAME, CITY FROM SALES\_PERSON where city="LONDON" && COMM > 0.10

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

select \* from SALES\_PERSON where CITY="BARCELONA" || CITY="LONDON";

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

**(first query shows the error)**

select \* from SALES\_PERSON where COMM limit 0.10, 0.12;

select \* from SALES\_PERSON where COMM between 0.10 and 0.12;

1. **All customers with NULL values in city column.**

select \* from customer where CITY=null;

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

select \* from ORDER\_INFO where ODE="1994-10-03" || ODE="1994-10-04";

1. **All customers serviced by peel or Motika.**

select \* from CUSTOMER where SNO=1001 || SNO=1004;

1. **All customers whose names begin with a letter from A to B**

select \* from CUSTOMER where CNAME like "A%" || CNAME like "B%";

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

select \* from CUSTOMER where RATING>100 || CITY="ROME";

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

select \* from ORDER\_INFO where AMT != 0 || AMT != NULL;

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

select count(SNO) as "Total Sales Persons" from (select ONM SNO from ORDER\_INFO group by SNO) SALES\_PERSON;