**NAME : MANGESH A. GHADWAJE**

**ROLL NO:24**

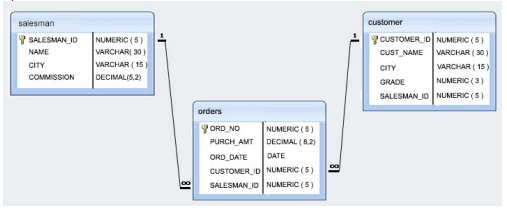
**BATCH : B2**

**COURSE: DBMS PRACTICAL**

**Assignment No:-5**

**Problem Statement:**

**A)**



**Create tables with primary key , foreign key constraint as shown in above schema, salesman\_id is**

**foreign key in customer table**

**Salesman**

**salesman\_id | name | city | commission**

**-------------+------------+----------+------------**

**5001 | James Hoog | New York | 0.15**

**5002 | Nail Knite | Paris | 0.13**

**5005 | Pit Alex | London | 0.11**

**5006 | Mc Lyon | Paris | 0.14**

**5007 | Paul Adam | Rome | 0.13**

**5003 | Lauson Hen | San Jose | 0.12**

**Customer**

**customer\_id | cust\_name | city | grade | salesman\_id**

**-------------+----------------+------------+-------+------------**

**-**

**3002 | Nick Rimando | New York | 100 | 5001**

**3007 | Brad Davis | New York | 200 | 5001**

**3005 | Graham Zusi | California | 200 | 5002**

**3008 | Julian Green | London | 300 | 5002**

**3004 | Fabian Johnson | Paris | 300 | 5006**

**3009 | Geoff Cameron | Berlin | 100 | 5003**

**3003 | Jozy Altidor | Moscow | 200 | 5007**

**3001 | Brad Guzan | London | | 5003**

**Orders**

**ord\_nopurch\_amtord\_datecustomer\_idsalesman\_id**

**---------- ---------- ---------- ----------- -----------**

**70001 150.5 2012-10-05 3005 5002**

**70009 270.65 2012-09-10 3001 5003**

**70002 65.26 2012-10-05 3002 5001**

**70004 110.5 2012-08-17 3009 5003**

**70007 948.5 2012-09-10 3005 5002**

**70005 2400.6 2012-07-27 3007 5001**

**70008 5760 2012-09-10 3002 5001**

**70010 1983.43 2012-10-10 3004 5006**

**70003 2480.4 2012-10-10 3009 5003**

**70012 250.45 2012-06-27 3008 5002**

**70011 75.29 2012-08-17 3003 5007**

**70013 3045.6 2012-04-25 3002 5001**

**Queries: (use appropriate join types and join condition like, cross join, natural join, left , right , full**

**outer join, Join... on, join...using)**

**1) Find the salesmen and customers with their name and cities, who belongs to the same city.**

**2) Prepare a list for which salesman are working for which customer along with city and**

**commissions earned by the salesman.**

**3) Write a query to make a join on the tables salesman and orders in such a form that the same**

**column of each table will appear once and only the relational rows will come.**

**4) Prepare a list for the salesmen who works either for one or more customer or not yet join under**

**any of the customers**

**5) Prepare a list in ascending order for the salesmen who works either for one or more customer**

**or not yet join under any of the customers**

**B) employee (employee-name, street, city)**

**works (employee-name, company-name, salary)**

**company (company-name, city)**

**manages (employee-name, manager-name)**

**Insert appropriate data in the tables.**

**Queries:**

**Solve Query 1 to 4 using appropriate operator in, not in , all, some , exists etc**

**Q1)Find the names, street address, and cities of residence of all employees who work for Tata**

**and earn more than $10,000**

**Q2)Find all employees in the database who do not work for Tata**

**Q3) Find all employees in the database who earn more than every employee of Reliance**

**Q4) Find the company that has the smallest payroll**

**Q5) Create a view for names and cities of residence of all employees who work for First Bank**

**Corporation**

**Solution :**

**CREATE TABLE SALESMAN(SALESMAN\_ID NUMBER(5) PRIMARY KEY,NAME VARCHAR(30), CITY VARCHAR(15),**

**COMMISION DECIMAL(5,2));**

**DROP TABLE SALESMAN;**

**CREATE TABLE CUSTOMER(CUSTOMER\_ID NUMBER(5) PRIMARY KEY,CUST\_NAME VARCHAR(30),**

**CITY VARCHAR(15),GRADE NUMBER(3),SALESMAN\_ID NUMBER(5),**

**FOREIGN KEY(SALESMAN\_ID) REFERENCES SALESMAN(SALESMAN\_ID));**

**DROP TABLE CUSTOMER;**

**CREATE TABLE ORDERS (**

**ORDER\_NO NUMBER(5,0) ,**

**PURCH\_AMT DECIMAL(8,2),**

**ORD\_DATE DATE,**

**CUSTOMER\_ID NUMBER(5),**

**SALESMAN\_ID NUMBER(5),**

**PRIMARY KEY (ORDER\_NO, CUSTOMER\_ID, SALESMAN\_ID),**

**FOREIGN KEY (CUSTOMER\_ID) REFERENCES CUSTOMER(CUSTOMER\_ID),**

**FOREIGN KEY (SALESMAN\_ID) REFERENCES SALESMAN(SALESMAN\_ID)**

**);**

**DELETE FROM customer**

**WHERE customer\_id= 3002;**

**DROP TABLE ORDERS;**

**INSERT INTO SALESMAN VALUES(5001,'James Hoog','New York',0.15);**

**INSERT INTO SALESMAN VALUES(5002,'Nail Knite','Paris',0.13);**

**INSERT INTO SALESMAN VALUES(5005,'Pit Alex','London',0.11);**

**INSERT INTO SALESMAN VALUES(5006,'Mc Lyon','Paris',0.14);**

**INSERT INTO SALESMAN VALUES(5007,'Paul Adam','Rome',0.13);**

**INSERT INTO SALESMAN VALUES(5003,'Lauson Hen','San Jose',0.12);**

**select \*from salesman;**

**INSERT INTO CUSTOMER VALUES(3002,'Nick Rimando','New York',100,5001);**

**INSERT INTO CUSTOMER VALUES(3007,' Brad Davis' ,'New York ',200 ,5001);**

**INSERT INTO CUSTOMER VALUES(3005, 'Graham Zusi ',' California ',200 , 5002);**

**INSERT INTO CUSTOMER VALUES(3008, 'Julian Green', 'London ',300 , 5002);**

**INSERT INTO CUSTOMER VALUES(3004,'Fabian Johnson ',' Paris' , 300 , 5006);**

**INSERT INTO CUSTOMER VALUES(3009,'Geoff Cameron' , 'Berlin' , 100 , 5003);**

**INSERT INTO CUSTOMER VALUES(3003,'Jozy Altidor','Moscow',200,5007);**

**INSERT INTO CUSTOMER VALUES(3001,'Brad Guzan', 'London','' ,5003);**

**select \*from customer;**

**INSERT INTO ORDERS VALUES (70001, 150.5, Date'2012-10-05', 3005, 5002);**

**INSERT INTO ORDERS VALUES (70009, 270.65, Date'2012-09-10', 3001, 5003);**

**INSERT INTO ORDERS VALUES (70002, 65.26, Date'2012-10-05', 3002, 5001);**

**INSERT INTO ORDERS VALUES (70004, 110.5, Date'2012-08-17', 3009, 5003);**

**INSERT INTO ORDERS VALUES (70007, 948.5, Date'2012-09-10', 3005, 5002);**

**INSERT INTO ORDERS VALUES (70005, 2400.6, Date'2012-07-27', 3007, 5001);**

**INSERT INTO ORDERS VALUES (70008, 5760, Date'2012-09-10', 3002, 5001);**

**INSERT INTO ORDERS VALUES (70010, 1983.43, Date'2012-10-10', 3004, 5006);**

**INSERT INTO ORDERS VALUES (70003, 2480.4, Date'2012-10-10', 3009, 5003);**

**INSERT INTO ORDERS VALUES (70012, 250.45, Date'2012-06-27', 3008, 5002);**

**INSERT INTO ORDERS VALUES (70011, 75.29, Date'2012-08-17', 3003, 5007);**

**INSERT INTO ORDERS VALUES (70013, 3045.6, DATE'2012-04-25', 3002, 5001);**

**select \*from ORDERS;**

**SELECT S.NAME AS salesman\_name, s.city AS salesman\_city, c.cust\_name AS customer\_name, c.city AS customer\_city**

**FROM salesman s**

**JOIN customer c ON s.city = c.city;**

**SELECT s.name AS salesman\_name, c.cust\_name AS customer\_name, c.city AS customer\_city, s.commision AS salesman\_commision**

**FROM salesman s**

**JOIN orders o ON s.salesman\_id = o.salesman\_id**

**JOIN customer c ON o.customer\_id = c.customer\_id;**

**SELECT \***

**FROM salesman**

**NATURAL JOIN orders;**

**SELECT s.salesman\_id, s.name**

**FROM salesman s**

**RIGHT JOIN customer c ON s.salesman\_id = c.salesman\_id**

**WHERE s.salesman\_id IS NOT NULL OR s.salesman\_id IS NULL;**

**SELECT s.salesman\_id, s.name**

**FROM salesman s**

**LEFT JOIN customer c ON s.salesman\_id = c.salesman\_id**

**order by salesman\_id asc;**

**-----------------------------Other Table ---------------**

**create table employee(employee\_name varchar(50) primary key,**

**street varchar(50), city varchar(20));**

**drop table employee;**

**create table company(company\_name varchar(50) primary key,city varchar(20));**

**drop table company;**

**create table works(employee\_name varchar(50),**

**company\_name varchar(50),**

**salary integer,**

**primary key(employee\_name, company\_name),**

**foreign key(employee\_name) references employee(employee\_name),**

**foreign key(company\_name) references company(company\_name)**

**);**

**drop table works;**

**create table manages(employee\_name varchar(50), manager\_name varchar(50),**

**primary key(employee\_name),**

**foreign key(employee\_name) references employee(employee\_name));**

**drop table manages;**

**insert into employee values('kajal patil','bhagvati nagar','chalisgaon');**

**insert into employee values('aishwarya','ozar','chalisgaon');**

**insert into employee values('damini jadhav','pawarwadi','mumbai');**

**insert into employee values('aastha sali','jalgoan road','erandol');**

**insert into company values('Tata','mumbai');**

**insert into company values('Reliances','mumbai');**

**insert into company values('first bank corporation','pune');**

**insert into works values('kajal patil','Tata',18000);**

**insert into works values('aishwarya','Reliances',15000);**

**insert into works values('damini jadhav','first bank corporation',19000);**

**insert into works values('aastha sali','Tata',11000);**

**insert into manages values('kajal patil','aditya patil');**

**insert into manages values('aishwarya','mohit patil');**

**insert into manages values('damini jadhav','varun bhadane');**

**SELECT e.employee\_name, e.street, e.city**

**FROM employee e**

**JOIN works w ON e.employee\_name = w.employee\_name**

**WHERE w.company\_name = 'Tata' AND w.salary > 10000;**

**SELECT employee\_name**

**FROM employee**

**WHERE employee\_name NOT IN (SELECT employee\_name FROM works WHERE company\_name = 'Tata');**

**SELECT company\_name**

**FROM works**

**GROUP BY company\_name**

**HAVING SUM(salary) = (**

**SELECT MIN(total\_salary)**

**FROM (**

**SELECT SUM(salary) AS total\_salary**

**FROM works**

**GROUP BY company\_name**

**) AS min\_payroll**

**);**

**CREATE VIEW First\_Bank\_Employees AS**

**SELECT e.employee\_name, e.city**

**FROM employee e**

**JOIN works w ON e.employee\_name = w.employee\_name**

**WHERE w.company\_name = 'First Bank Corporation';**

**show VIEW First\_Bank\_Employees;**