DBMS LAB EXAM

Date:15,May,2021

Name: G.Mohith Krishna
Reg no. :19BCS043

1.Retrieve the details of all the employee who have a dependent with the same last name and sex as the employee.

Sol)

```
SELECT          Emp.Lname
FROM            Employee AS Emp
WHERE           Emp.SSN IN   (SELECT   Emp.SSN
                              FROM      Dependent
 WHERE    Emp.LNAME =Dep_name  AND Emp.Sex=Sex);
```

2. For each project on which more than 2 employees word, retrieve the project number, the project name, and the number of employees who work on the project.

Sol)

```
SELECT           Pnumber, Pname,Count(*)
FROM             Project,Works_on
WHERE            Pnumber=Pno
GROUP BY         Pnumber,Pname
HAVING           COUNT(*)>2;
```

3. For each department that has more than 3 employees, retrieve the department number and the number of employees who are making more than Rs50,000

sol)

```
SELECT       Dnumber, Count(*)
FROM         Department,Employee
WHERE        Dnumber=Dno AND Salary >50000 AND
             Dno IN   (SELECT          DNO
                       FROM            Employee
                       GROUP BY        Dno
                       HAVING          Count(*)>3)
GROUP BY    Dnumber;
```

4. Retrieve details of all the employees and their dependents who has been managers with department more than once and has more than 10 employees.

Sol:

```
SELECT * from Employee E
WHERE E.Eno in (SELECT Eno FROM Dependent WHERE Relation =
                'Manager' GROUP BY relation HAVING
                count()>1);
GROUP BY Eno
HAVING count()>10;
```

5. Retrieve the department details with more than one project, where these projects are located in multiple locations.

SELECT *
FROM   Department ,Project
WHERE   Department.Dno=Project.Dnum
GROUP BY Pno
HAVING   Count(Pno)>1 **AND** Count(Plocation)>1

Normalization:
1. Convert the table into 1nf, 2nf, 3nf

Solution 1:

Based on the closure of given functional dependencies, we arrive at a result of two candidate keys possible. They are:

- ❖ {PropertyId}
- ❖ {CountryName, Plot}

Prime Attributes: PropertyId, CountryName, Plot

Non-prime attributes: Area, Price, TaxRate

Since, the minimal amongst candidate keys is the primary key, {PropertyId} is the primary key of the table PLOT.

1NF Normalization:

The table PLOT is already in 1NF. Thus, the 1NF Normalized form shall be:

PropertyId	CountryName	Plot	Area	Price	TaxRate

The table PLOT is not in 2NF because of the functional dependency CountryName → TaxRate. Thus, TaxRate is partially dependent on the candidate key {CountryName, Plot}.

The 2NF Normal form shall be:

PLOT_Table1				
PropertId (primary key)	CountryName	Plot	Area	Price

The other table shall be:

PLOT_Table2	
CountryName (primary key)	TaxRate

3NF Normalization:

The table PLOT_Table1 is not in 3NF because of the existence of transitive dependency due to Area → Price. Area is not a superkey and Price is not a prime attribute. Thus, PLOT_Table1 should be divided into PLOT_Table1A and PLOT_Table1B.

PLOT_Table1A			
PropertId (primary key)	CountryName	Plot	Area

The other table shall be:

PLOT_Table1B

Area (primary key)	Price