Team -13 DBMS LAB EXAM

Team Members:

1. Sejal Rai 19BCS098
2. Aman Kumar 19BCS115
3. Akshat Mishra 19BCS009
4. Priyam 19BCS089
5. Uattam Kumar Jangid 19BCS108
6. Sai Madhav Reddy 19BCS074
7. Nishikant 19BCS079
8. Aakansh Agarwal 19BCS001
9. AniruddhMukesh 19BCS010

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

```
SELECT *
FROM EMPLOYEE AS E
WHERE E.ENO IN (SELECT D.ENO
FROM DEPENDENT D
WHERE E.LNAME = D.DEP_NAME AND E.SEX = D.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.**

```
SELECT PNO, PNAME, COUNT (*)
FROM PROJECT P, WORKS_NO W
```

```
WHERE P.PNO = W.PNO
GROUP BY P.PNO, P.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.**

```
SELECT DNO, COUNT (*)
FROM DEPARTMENT D, EMPLOYEE E
WHERE D.DNO = E.DNO AND SALARY > 50000 AND
E.DNO IN (SELECT E.DNO
FROM EMPLOYEE
GROUP BY E.DNO
HAVING COUNT (*) > 3)
GROUP BY D.DNO;
```

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.**

```
SELECT * FROM EMPLOYEE E
WHERE E.ENO IN (SELECT DP.ENO
FROM DEPENDENT DP
WHERE DP.RELATION = 'MANAGER'
GROUP BY DP.RELATION
HAVING COUNT(*) > 1)
GROUP BY E.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 D, PROJECT P
WHERE P.DNUM=D.DNO
GROUP BY P.PNO
HAVING COUNT(P.PNO) > 1 AND
COUNT(PLOCATION) > 1 ;
```

NORMALISATION:

1. **Convert the table into 1nf, 2nf, 3nf.**

   1NF: As in table we don't have any data we can assume the given table is in already in 1NF.

   2NF:
   PLOTS(Property-ID, CountryName, Plot, Area, Price, TaxRate)
   FD1: PropertyID → {CountryName, Lot#, Area, Price, TaxRate}
   FD2: {County-Name, Plot} → {Property-ID#, Area, Price, TaxRate}
   FD3: County-Name → TaxRate
   FD4: Area → Price

   2 candidate keys: {PropertyID},{CounrtyName, Plot}
   PLOTS is not in 2NF, because of CounrtyName → Tax-Rate
   TaxRate is partially dependent on the candidate key {CountryName, Plot}.

Resulting 2NF:
PLOTS1(PropertyID, CountryName, PLOT, Area, Price)
PLOTS2(CountryName, TaxRate)

3NF:

The relation PLOTS1 is not in 3NF, because of Area → Price
Area is not a superkey and Price is not prime attribute
PLOTS1A(PropertyID, CountryName, Plot, Area)
PLOTS1B(Area, Price)
PLOT2(CountryName, TaxRate)