CS4350 Database Systems HW#3

Name: Loc Nguyen
Given the following relational database schema:

Employee = (ID, name, position, officeN, phoneN,age)// **assume the name is unique**.
Committee = (title, meetingDate, startTime, endTime, location)// **You may use < , >, =, != between dates and times.**
Membership = (ID, title, task) // **task = 'member 'or 'chair'.**

Use a minimum number of operations and tables to express the following queries by SQL statements:

1. List the name of every employee who is a member (not chair) of at least three committees.

```
SELECT e.name
FROM Employee e
JOIN Membership m ON e.ID = m.ID
WHERE m.task = 'member'
GROUP BY e.name
HAVING COUNT(DISTINCT m.title) >= 3;
```

2. List the name of every employee who serving on every committee.

```
SELECT e.name
FROM Employee e
WHERE NOT EXISTS (
        SELECT c.title
        FROM Committee c
        WHERE NOT EXISTS (
                SELECT m.title
                FROM Membership m
                WHERE m.id = e.id AND m.title = c.title
        )
);
```

Employee=(ID, name, position, officeN, phoneN, age)// **assume the name is unique**.

Committee =(title, meetingDate, startTime, endTime, location)// You may use <,>,=,!= between dates and times.

Membership= (ID, title, task) // **task = ‘member ‘or ‘chair’**

3. For every employee, list the ID, name and number of committees he/she is serving on as member or chair.

```
SELECT e.ID, e.name, COUNT(DISTINCT m.title) AS num_committees
FROM Employee e
JOIN Membership m ON e.ID = m.ID
GROUP BY e.ID, e.name;
```

4. List the name of every employee who only serves (as member) on committees that meets in location H345.

```
SELECT e.name
FROM Employee E
JOIN Membership m on e.ID = m.ID
JOIN Committee c on m.title = c.title
WHERE m.task = ‘member’
AND c.location = ‘H345’
AND NOT EXISTS (
        SELECT 1
        FROM Committee c2
        WHERE c2.title = c.title
        AND c2.location <> ‘H345’
);
```

Employee=(ID, name, position, officeN, phoneN,age)// **assume the name is unique**.

Committee =(title, meetingDate, startTime, endTime, location)// You may use <,>,=,!= between dates and times.

Membership= (ID, title, task) // **task = ‘member ‘or ‘chair’**

5. List the name of every employee who does not have a phone number.

```
SELECT e.name
FROM Employee e
WHERE phoneN IS NULL OR phoneN = ‘’;
```

6. List the ID and name of every employee who is not serving on any committee.

```
SELECT e.ID, e.name
FROM Employee e
WHERE e.ID NOT IN(SELECT DISTINCT ID FROM Membership);
```

Employee=(ID, name, position, officeN, phoneN,age)// **assume the name is unique**.

Committee =(title, meetingDate, startTime, endTime, location)// You may use <,>,=,!= between dates and times.

Membership= (ID, title, task) // **task = ‘member ‘or ‘chair’**

7. List the title of every committee on which Sandy Liu or Barry Smith is serving.

```
SELECT DISTINCT m.title
FROM Membership m
JOIN Employee e ON m.ID = e.ID
WHERE e.name IN (‘Sandy Liu’, ‘Barry Smith’);
```

8. List the name and position of oldest employees.

```
SELECT name, position
FROM Employee
WHERE age = (SELECT MAX(age) FROM Employee);
```

Employee= (ID, name, position, officeN, phoneN,age)// **assume the name is unique**.

Committee = (title, meetingDate, startTime, endTime, location)// You may use <,>,=,!= between dates and times.

Membership= (ID, title, task) // **task = ‘member ‘or ‘chair’**

9. List the titles of every two committees which do not meet on the same date.

```
SELECT DISTINCT c1.title, c2.title
FROM Committee c1, Committee c2
WHERE c1.title < c2.title AND c1.meetingDate <> c2.meetingDate;
```

10. List the name and position of every employee who does have phone number.

```
SELECT name, position
FROM Employee e
WHERE phoneN IS NOT NULL and phoneN <> ‘’;
```