Takeaway Sheet 3 (Lectures 7-9) - Jonathan Jacobs

Lecture 8

L8.1

Using a subquery, find the number of cars each person drives

```sql
SELECT
    p.userid,
    p.name,
    (SELECT COUNT(*)
     FROM Regist r
     WHERE r.userid = p.userid) AS NumberOfCars
FROM Payroll p;
```

L8.2

Logical Proposition

Let (P(x)) be the proposition "Person (x) drives at least one car". We are interested in people for whom (P(x)) is false. Using the given tables, a person who does not drive any car will not have an entry in the `Regist` table. Therefore, the logical proposition for people who do not drive cars is the negation of (P(x)), represented as (\neg P(x)).

SQL Query

```sql
SELECT
    p.name,
    p.Salary
FROM
    Payroll p
LEFT JOIN
    Regist r ON p.userid = r.userid
WHERE
    r.userid IS NULL;
```

L8.3

Is the Query Monotone? A query is considered monotone if, whenever its input is extended, the output is extended as well. In other words, adding more data to the input of a monotone query cannot result in a smaller output set; it can only make the output set the same size or larger.

Demonstrating Monotonicity

- If we add another person with an existing job, say another "TA" with a different userid, the count for the "TA" group will increase.
- Alternatively, if we add a person with a new job title not currently in the Payroll table, such as "Admin", this will introduce a new job group to the results.

Let's say the original Payroll table looked like this:

userid	name	job	Salary
123	Leslie	TA	50k
345	Frances	TA	60k
567	Magda	Prof	120k

And we add a record:

userid	name	job	Salary
890	Alex	TA	70k

After adding Alex as another TA, the query's result set extends by increasing the count of TAs, demonstrating the query's monotonicity.

Lecture 9

L9.1

Find the number of each car each person drives (Including Frances Quinn!)

```
SELECT
    p.name,
    IFNULL(r.Car, 'No Car') as Car,
    COUNT(DISTINCT r.Car) as Count
FROM
    Payroll p
LEFT JOIN
    Regist r ON p.userid = r.userid
GROUP BY
    p.name
ORDER BY
    p.name;
```

L9.2

Select each Driver in Person who drives all the vehicles in Car:

```
SELECT d.driver
FROM Driver d
JOIN Car c ON d.Car = c.car
```

```
GROUP BY d.driver
HAVING COUNT(DISTINCT d.Car) = (SELECT COUNT(*) FROM Car);
```

L9.3

How many records are returned? The SQL query would return 0 records, as no entries in the provided table meet the specified criteria of having a price less than $1000 and either being size 2 or colored red.