CPSC 368: Databases in Data Science
Practice Exercise on SQL Equivalencies

Based on the following relations, determine whether each of the following pairs of SQL statements are equivalent. If the pair is equivalent, just say yes and no explanation is needed. If you do not think the pair is equivalent, construct an instance of the relations to illustrate the difference between the pair of statements.

To approach these problems, you can test each of these queries the same way you would test a method that you have programed. Create relations that would represent normal and edge cases and test each of your queries using those relations.

Mover(e-num, addr, phone, driverlicence, indicator)
Van(plate-num, wt-category, capacity, driver-e-num, startdate, enddate)
Customer(c-num, oldaddr, newaddr, phone, capacity)
Job(job-num, date, hours-served, c-num, driver-e-num)
Job-assign(job-num, e-num)

1) SELECT enum FROM Job-assign
WHERE job-num = 1001 AND enum NOT IN (SELECT driver-e-num
FROM Job
WHERE job-num = 1001)

vs.

SELECT enum FROM Job-assign, Job
WHERE Job.job-num = 1001 AND Job-assign.job-num = 1001
AND enum <> driver-e-num

(<> means not equal)

2) SELECT v1.plate-num FROM Van v1, Van v2
WHERE v1.plate-num = v2.plate-num AND (
(v1.startdate < v2.startdate AND v2.startdate < v1.enddate) OR
(v2.startdate < v1.startdate and v1.startdate < v2.enddate))

vs

SELECT plate-num FROM Van v1
WHERE NOT EXISTS (
SELECT * FROM Van v2
WHERE v2.plate-num = v1.plate-num AND (
(v2.startdate < v1.startdate and v1.startdate < v2.enddate) OR
(v1.startdate < v2.startdate and v2.startdate < v1.enddate)
)

```
)
```

3)
```
SELECT phone FROM Mover
GROUP BY phone HAVING COUNT(enum) > 2
```

vs

```
SELECT phone FROM Mover m1
WHERE EXISTS
      (SELECT * FROM Mover  m2
       WHERE m1.phone = m2.phone AND m1.enum <> m2.enum)
```

4)
```
SELECT count(*) FROM Van, Job
WHERE Van.driver-e-num = Job.driver-e-num AND startdate ≤ 1/1/2004
GROUP BY plate-num
```

vs

```
SELECT count(*) FROM Van
WHERE startdate ≤ 1/1/2004
GROUP BY plate-num
```

5)
```
SELECT job-num FROM Job j1
WHERE NOT EXISTS (
        (SELECT driver-e-num FROM Van
        WHERE wt-category = light AND
                startdate ≤ 1/12/2004 AND enddate ≥ 1/12/2004)
        EXCECT
        (SELECT driver-e-num FROM Job j2
        WHERE date = 1/12/2004 AND j2.job-num = j1.job-num)
)
```

vs

```
SELECT job-num FROM Job j1
WHERE NOT EXISTS (
        SELECT * FROM Van v1
       WHERE wt-category = light AND startdate ≤ 1/12/2004 AND
                enddate ≥ 1/12/2004 AND EXISTS
                        (SELECT * FROM Job j2
                         WHERE j2.job-num = j1.job-num AND
                                 j2.driver-e-num = v1.driver-e-num AND
```

```
                                    j2.date = 1/12/2004)
    )
```

6)
```
SELECT v1.plate-num, v2.plate-num FROM Van v1, Van v2
WHERE v1.capacity + v2.capacity ≥ 2000
```

vs

```
SELECT v1.plate-num, v2.plate-num FROM Van v1, Van v2
WHERE v1.capacity ≥ 1000 AND v2.capacity ≥ 1000 AND
        v1.plate-num <> v2.plate-num
```

CPSC 368: Databases in Data Science
Practice Exercise on SQL Equivalencies

Answers

1) Equivalent.

   Both queries are trying to find all employees who worked job 1001 and was not the main driver.

2) Not equivalent.

   The first query finds plate numbers where the plate number was used on two different vehicles at the same time. The second query finds plate numbers which have not been used on two different vehicles in the same time period.

3) Not equivalent.

   The first query is trying to find all phone numbers associated with three or more movies (because count has to be greater than two, not greater than or equal to two). The second query is trying to find all phone numbers that are associated with at least two people. If you had a phone number that was only associated with two people, it would appear as a result for the second query but not the first.

4) Not equivalent.

   The first query finds the number of vans that were used in a job and were commissioned before 1/1/2004. The second query finds the number of vans commissioned before 1/1/2004. Idle vans that were not used for any job will not show up in the first query.

5) Equivalent.

6) Not equivalent.

   Imagine a table where there is only one tuple in Van and the tuple has a capacity of 1000. The first query would include that van in the result but the second query would not.