PART 2

1) Transactions/bills cannot be issued at times when the given bar is closed

```
select distinct 'False' as 'output' from bills where EXISTS(SELECT bills.bill_id FROM bills
LEFT JOIN operates ON operates.bar = bills.bar WHERE bills.bar = operates.bar AND
bills.day = operates.day AND (bills.time < operates.start OR bills.time > operates.end))
UNION
select distinct 'True' as 'output' from bills where NOT EXISTS(SELECT bills.bill_id FROM
bills LEFT JOIN operates ON operates.bar = bills.bar WHERE bills.bar = operates.bar AND
bills.day = operates.day AND (bills.time < operates.start OR bills.time > operates.end))
```

2) Drinkers cannot frequent bars in different state

```
SELECT distinct 'False' as 'output' from drinker
where EXISTS
(SELECT drinker.name FROM drinker LEFT JOIN frequent ON drinker.name = frequent.drinker
LEFT JOIN bar ON bar.state = drinker.state WHERE bar.state != drinker.state AND
drinker.name = frequent.drinker)
UNION
select distinct 'True' as 'output' from drinker where EXISTS
(SELECT drinker.name FROM drinker LEFT JOIN frequent ON drinker.name = frequent.drinker
LEFT JOIN bar ON bar.state = drinker.state WHERE bar.state = drinker.state AND drinker.name
= frequent.drinker)
```

3) For every two beers, b1 and b2, different bars may charge differently for b1 and b2 but b1 should either be less expensive than b2 in ALL bars or more expensive than b2 in ALL bars. Cannot be the case that in one bar Corona is more expensive than Bud and in another Bud is more expensive than Corona.

```
SELECT distinct 'true' AS 'output' FROM sellsbeer
WHERE EXISTS(SELECT * FROM sellsbeer B1, sellsbeer B2
WHERE b1.beername = b2.beername AND b1.barname <> b2.barname
AND b1.price = b2.price)
UNION
SELECT distinct 'false' AS 'output' FROM sellsbeer
WHERE EXISTS(SELECT * FROM sellsbeer B1, sellsbeer B2
WHERE b1.beername = b2.beername AND b1.barname <> b2.barname
AND b1.price <> b2.price)
```