

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

-- query that returns rows where payment time is out of range of opening and closing times
-- (should return zero rows)

```
select *
from transactions T
      inner join bar B
        on B.Bar_Name = T.Bar_Name
where time(T.Payment_Time) < B.Open_Time or time(T.Payment_Time) > B.Close_Time;
```

PATTERN: 2) Drinkers cannot frequent bars in different state

-- query that returns rows where drinker frequents bar in a different state (should return zero rows)

```
select *
from frequents F
      inner join bar B
        on B.Bar_Name = F.Bar_Name
      inner join drinker D
        on D.Drinker_Name = F.Drinker_Name
where D.State != B.State;
```

PATTERN: 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. But Corona may be more expensive than Bud in one bar, and have the same price as Bud in another.

-- query that makes sure that beer prices are more expensive or less expensive in ALL bars, no exceptions
-- (should return zero rows)

```
select *
from sells S1
      inner join sells S2
        on S1.Beer_Name = S2.Beer_Name
      and S1.Bar_Name != S2.Bar_Name
      and S1.Price <= S2.Price
      inner join sells S3
        on S3.Beer_Name = S1.Beer_Name
      and S3.Bar_Name != S1.Bar_Name
      and S3.Bar_Name != S2.Bar_Name
```

and S3.Price < S1.Price

;