# Problem solving in SQL

**Bars:**

| Name | Addr | License |
|---|---|---|
| Australia Hotel | The Rocks | 123456 |
| Coogee Bay Hotel | Coogee | 966500 |
| Lord Nelson | The Rocks | 123888 |
| Marble Bar | Sydney | 122123 |
| Regent Hotel | Kingsford | 987654 |
| Royal Hotel | Randwick | 938500 |

**Drinkers:**

| Name | Addr | Phone |
|---|---|---|
| Adam | Randwick | 9385-4444 |
| Gernot | Newtown | 9415-3378 |
| John | Clovelly | 9665-1234 |
| Justin | Mosman | 9845-4321 |

# Problem solving in SQL

**Beers:**

| Name | Manf |
|---|---|
| 80/- | Caledonian |
| Bigfoot Barley Wine | Sierra Nevada |
| Burragorang Bock | George IV Inn |
| Crown Lager | Carlton |
| Fosters Lager | Carlton |
| Invalid Stout | Carlton |
| Melbourne Bitter | Carlton |
| New | Toohey's |
| Old | Toohey's |
| Old Admiral | Lord Nelson |
| Pale Ale | Sierra Nevada |
| Premium Lager | Cascade |
| Red | Toohey's |
| Sheaf Stout | Toohey's |
| Sparkling Ale | Cooper's |
| Stout | Cooper's |
| Three Sheets | Lord Nelson |
| Victoria Bitter | Carlton |

# Problem solving in SQL

**Frequents:**

| Drinker | Bar |
|---|---|
| Adam | Coogee Bay Hotel |
| Gernot | Lord Nelson |
| John | Coogee Bay Hotel |
| John | Lord Nelson |
| John | Australia Hotel |
| Justin | Regent Hotel |
| Justin | Marble Bar |

**Likes:**

| Drinker | Beer |
|---|---|
| Adam | Crown Lager |
| Adam | Fosters Lager |
| Adam | New |
| Gernot | Premium Lager |
| Gernot | Sparkling Ale |
| John | 80/- |
| John | Bigfoot Barley Wine |
| John | Pale Ale |
| John | Three Sheets |
| Justin | Sparkling Ale |
| Justin | Victoria Bitter |

# Problem solving in SQL

**Sells:**

| Bar | Beer | Price |
|---|---|---|
| Australia Hotel | Burragorang Bock | 3.5 |
| Coogee Bay Hotel | New | 2.25 |
| Coogee Bay Hotel | Old | 2.5 |
| Coogee Bay Hotel | Sparkling Ale | 2.8 |
| Coogee Bay Hotel | Victoria Bitter | 2.3 |
| Lord Nelson | Three Sheets | 3.75 |
| Lord Nelson | Old Admiral | 3.75 |
| Marble Bar | New | 2.8 |
| Marble Bar | Old | 2.8 |
| Marble Bar | Victoria Bitter | 2.8 |
| Regent Hotel | New | 2.2 |
| Regent Hotel | Victoria Bitter | 2.2 |
| Royal Hotel | New | 2.3 |
| Royal Hotel | Old | 2.3 |
| Royal Hotel | Victoria Bitter | 2.3 |

# SQL/PLpgSQL

- Createdb
- Create table: attributes, constraints, primary key, foreign key
- Select … from … where
- Aggregation, Group by, Having, order by
- Set operation: union, intersect, except
- Views
- inset, delete, update, atler
- SQL function
- PLpgSQL function
- Triggers

# Problem solving in SQL

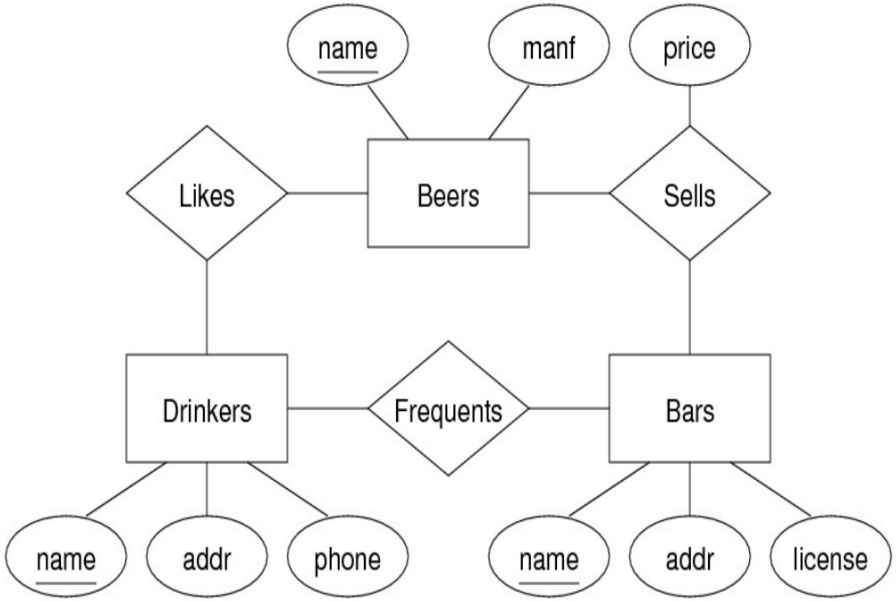
Request: description of required information from database.

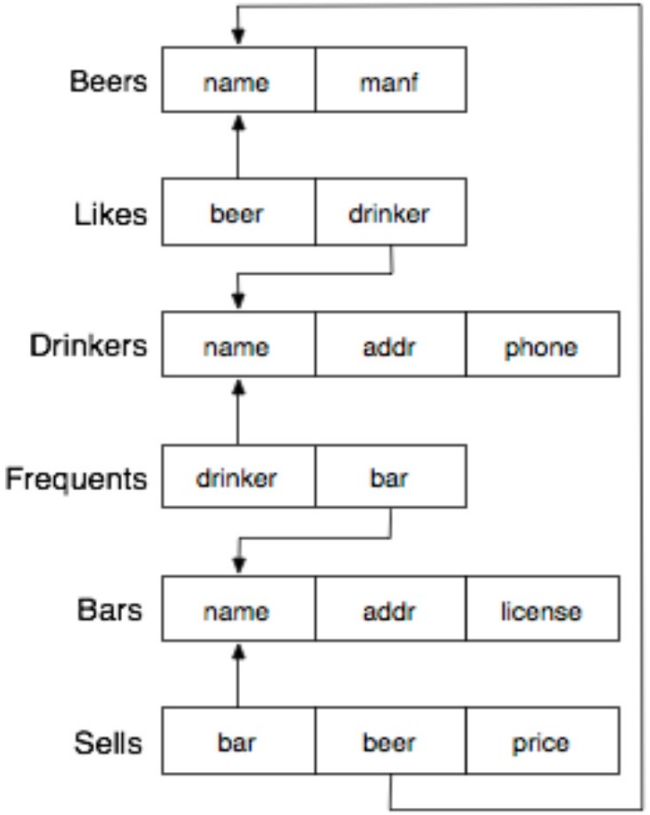
Pre-req: know your schema

Look for keywords in request to identify required data :

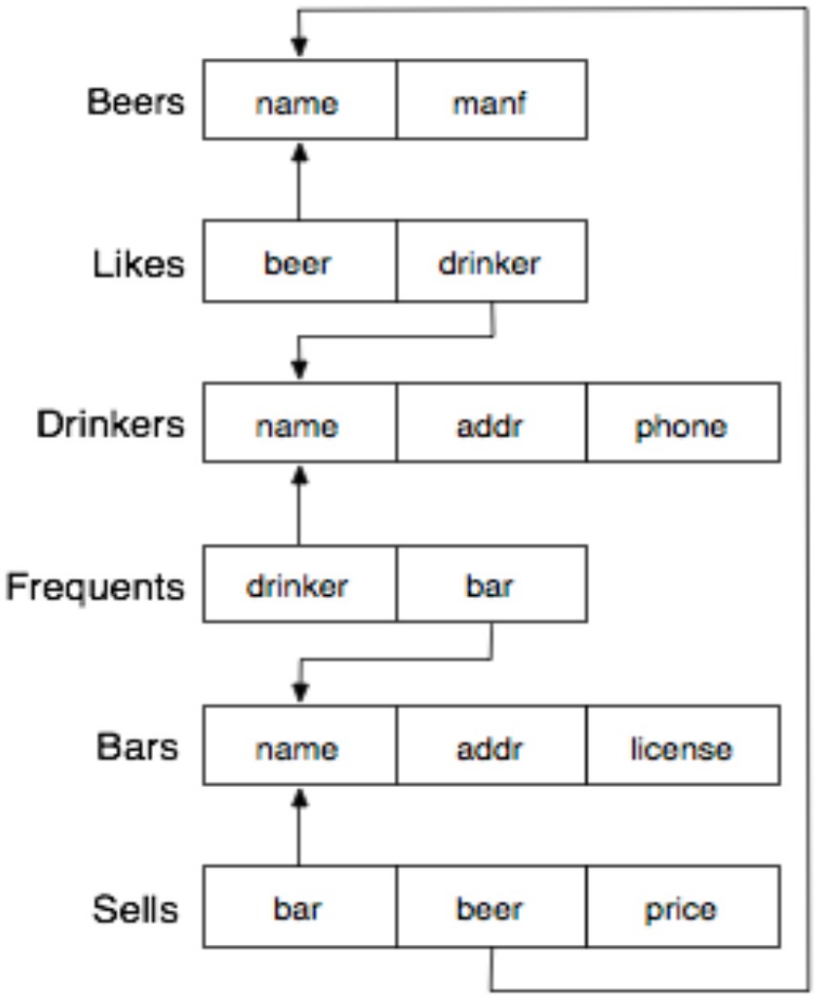
- tell me the names of all students...
- how many students failed ...
- what is the highest mark in ...
- which courses are ... (course codes?)

# Problem solving on Beer Database



# Problem solving on Beer Database

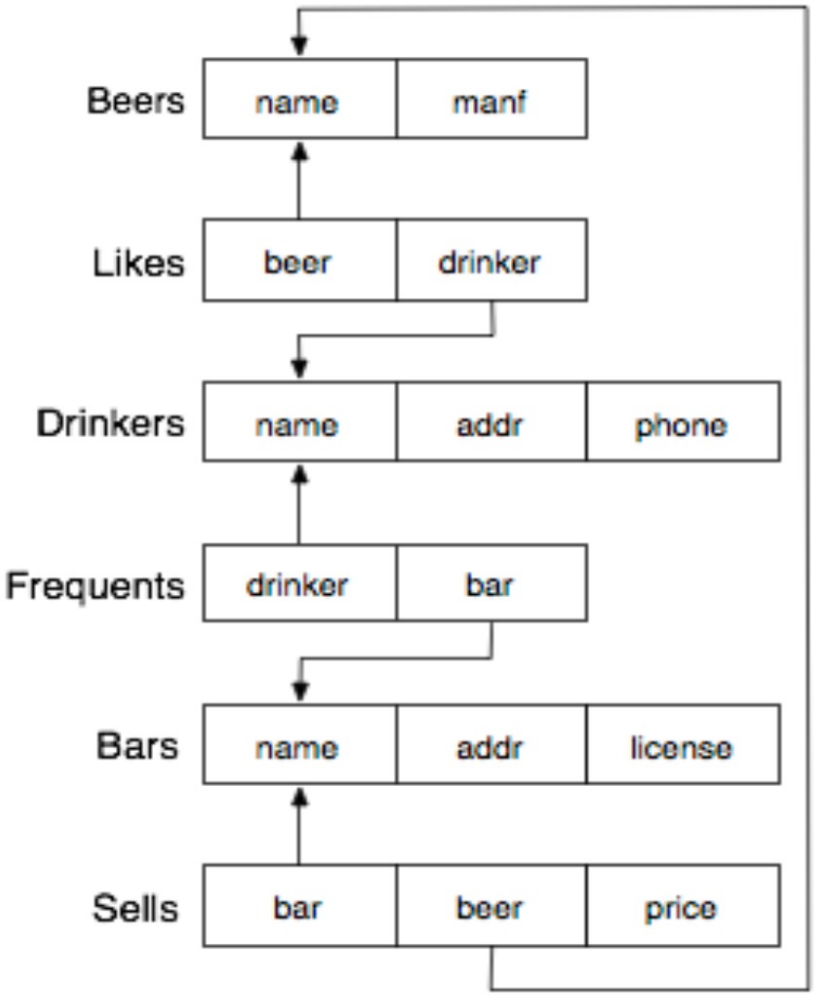



What beers are made by Toohey's?

Show beers with headings "Beer", "Brewer".

Find the brewers whose beers John likes

# Problem solving on Beer Database

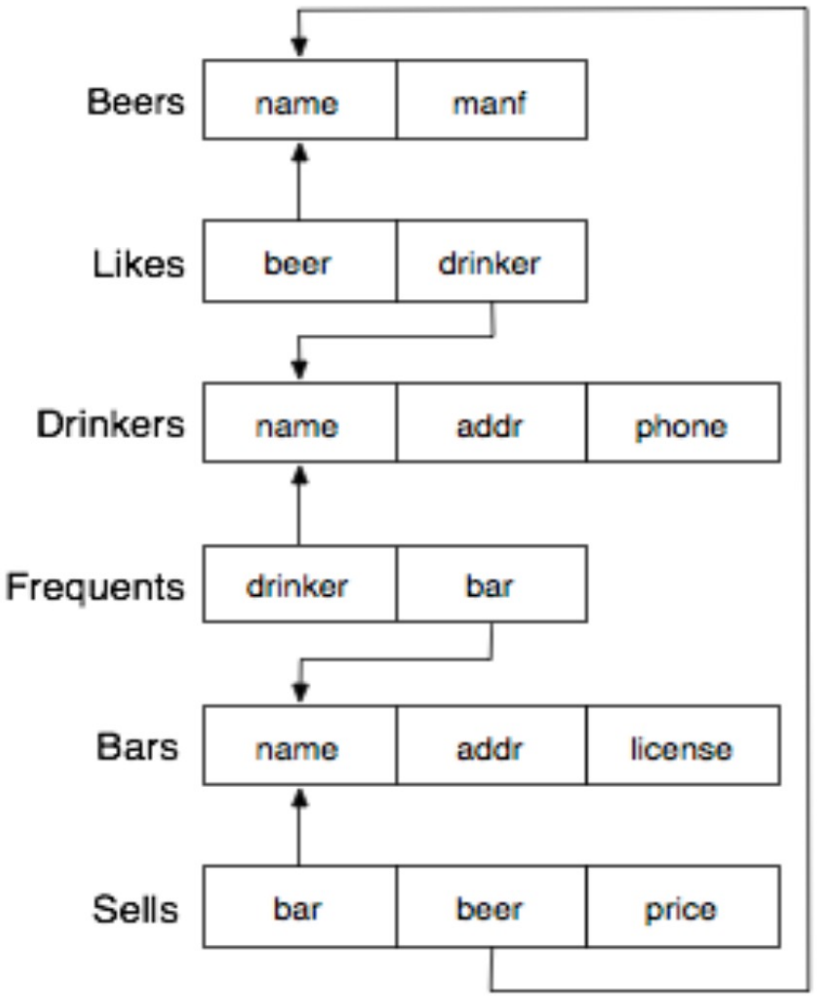



```
select name
from beers
where manf = 'Toohey"s';
```

```
select name as Beer, manf as
Brewer  from beers;
```

```
select distinct Beers.manf as
brewer  from   Likes
join Beers on (Likes.beer =
Beers.name)
where  Likes.drinker = 'John';
```

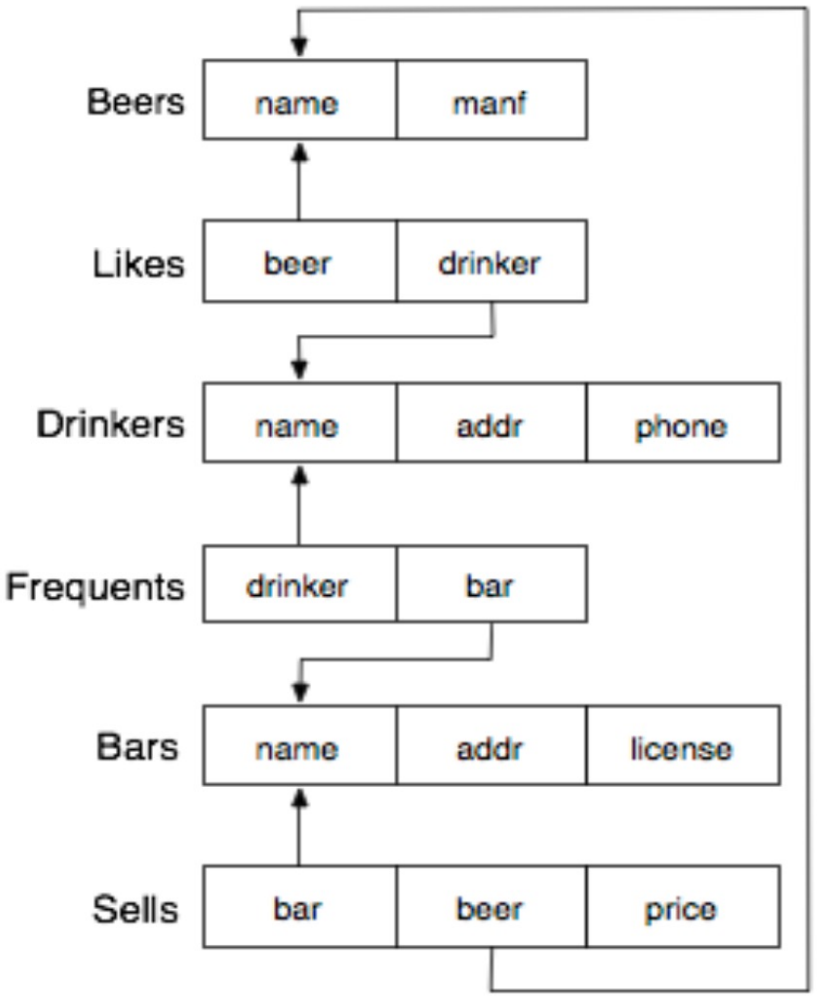
# Problem solving on Beer Database




How many different beers are there?

How many different brewers are there?

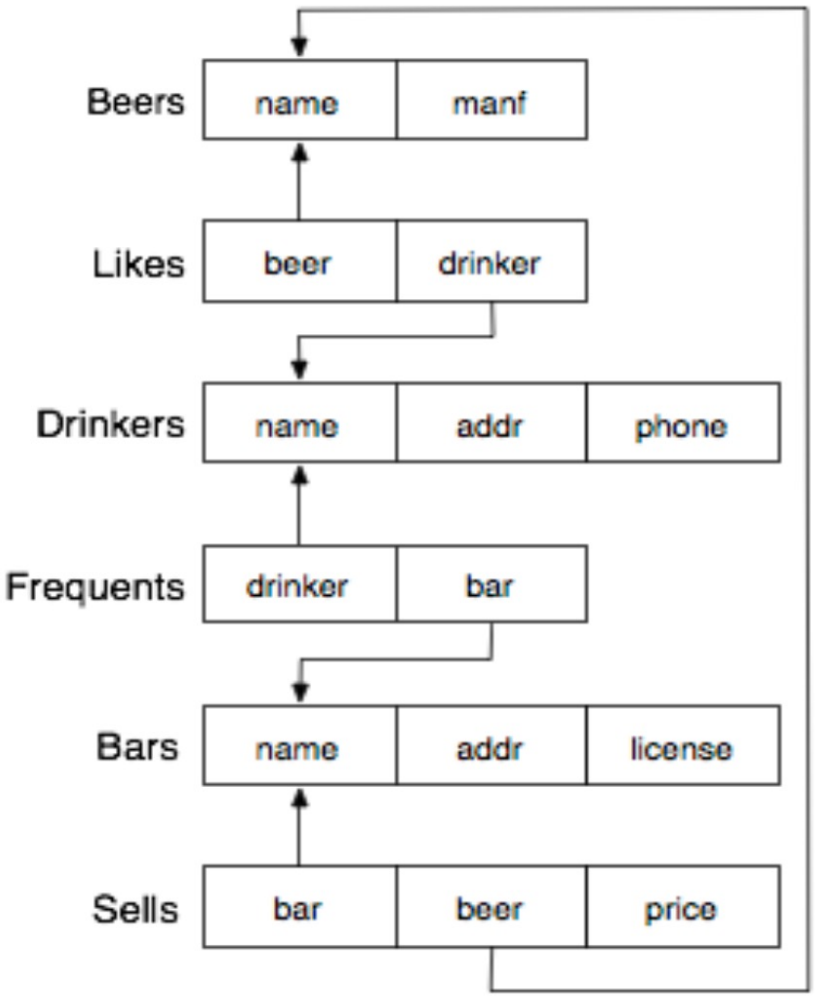
# Problem solving on Beer Database




select count(name) from beers;

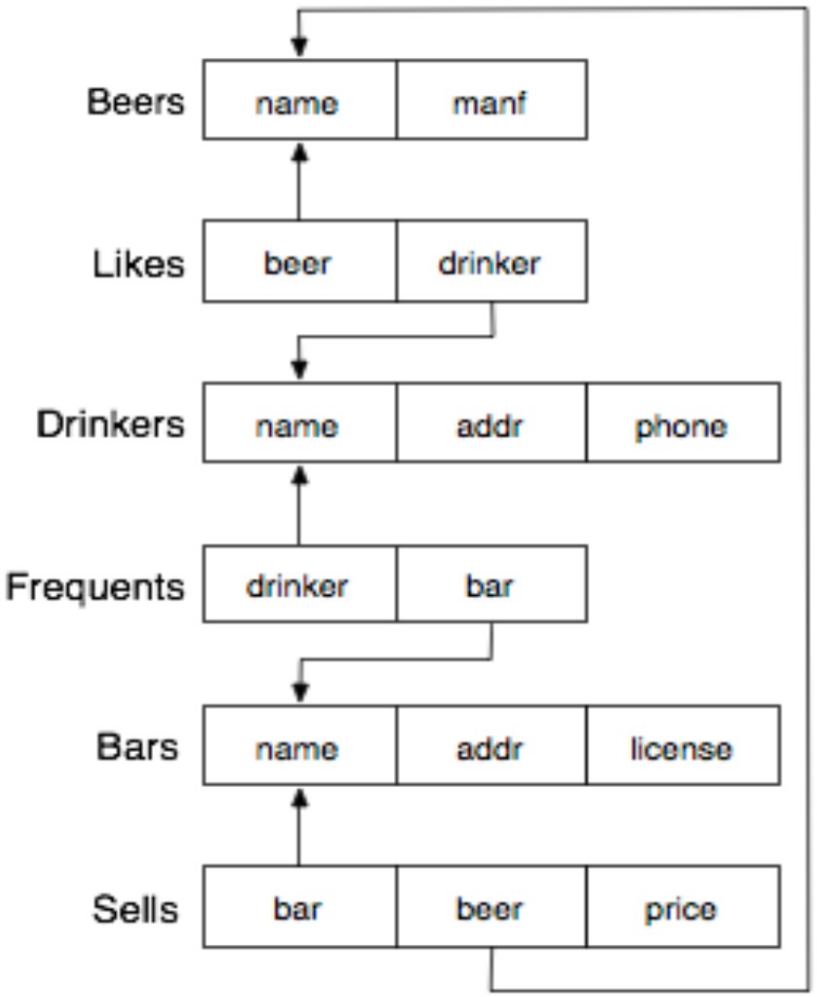
select count(distinct manf) from Beers;

# Problem solving on Beer Database



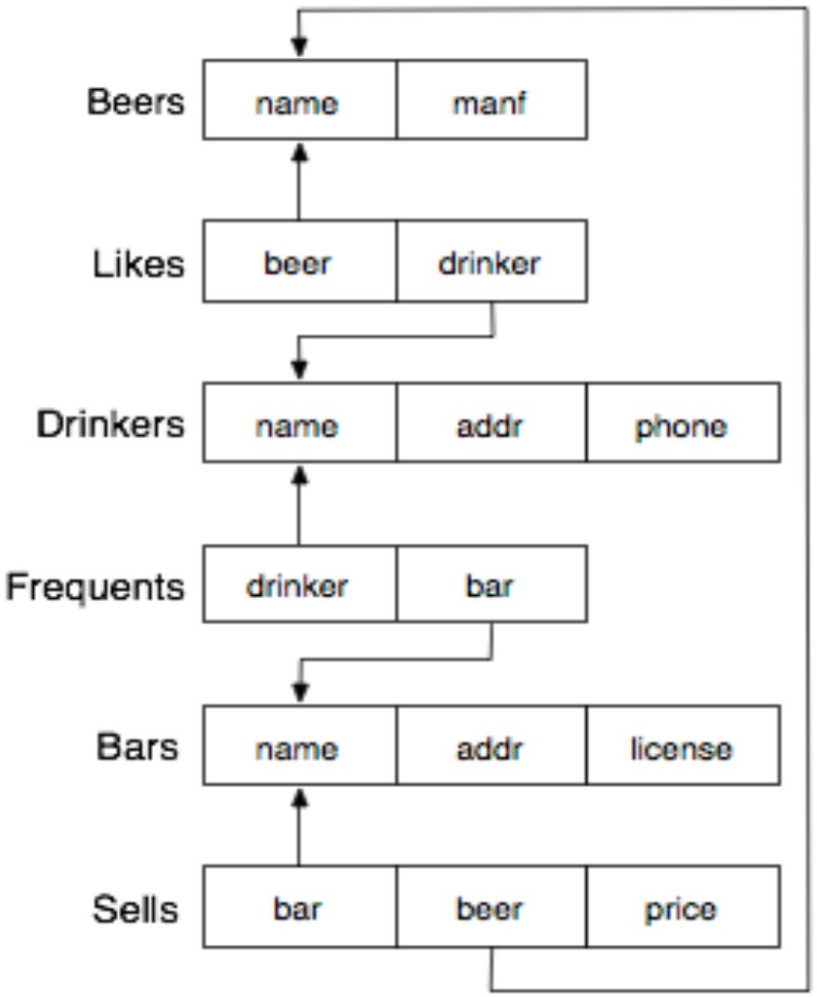

Find the beers uniquely made by their manufacturer.

# Problem solving on Beer Database



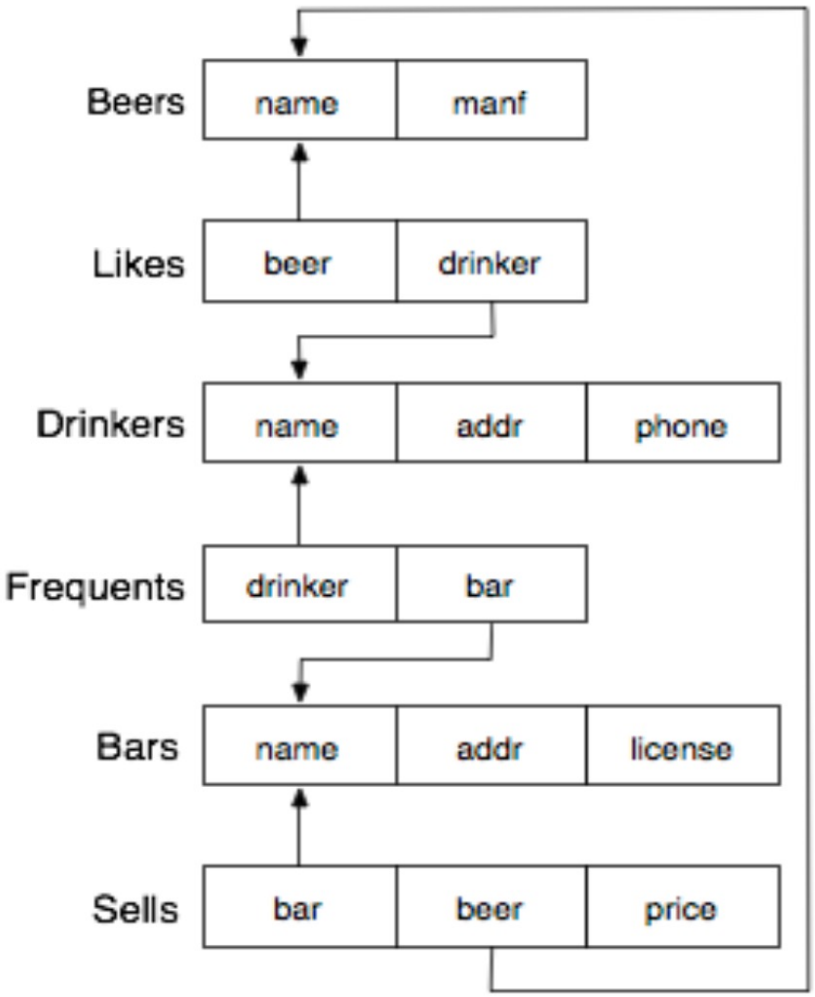

```
select b.name
from   Beers b
where  not exists
(select * from   Beers b1
where  b1.manf = b.manf and
b1.name != b.name);
```

# Problem solving on Beer Database



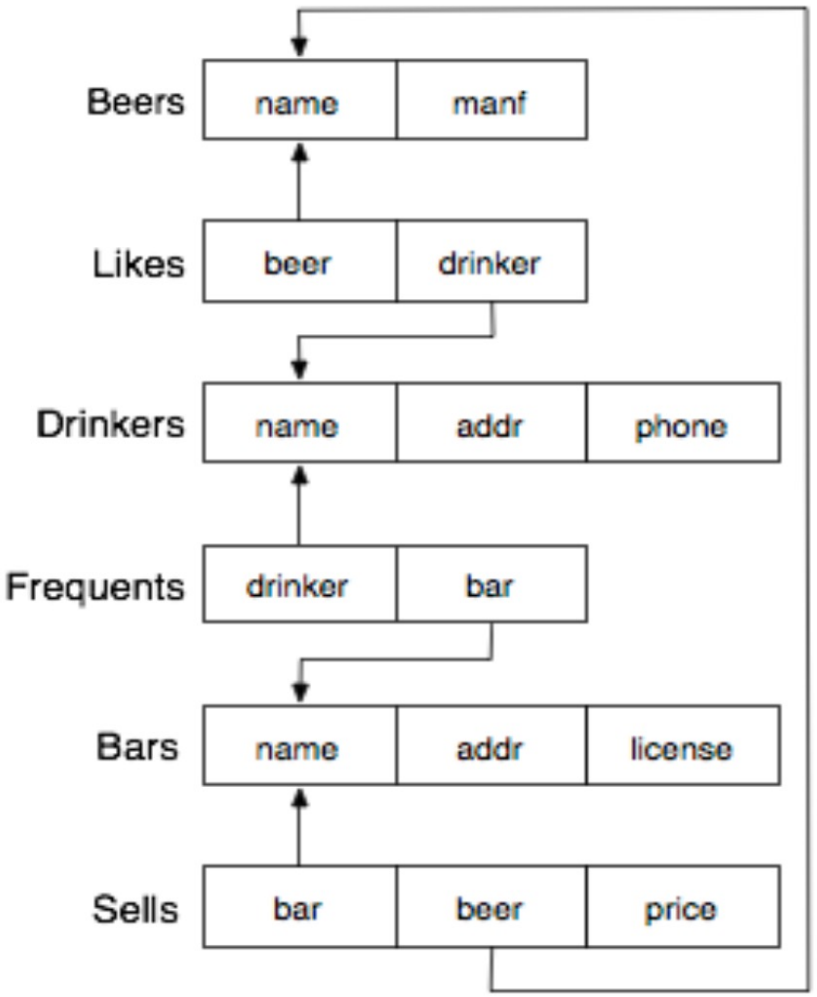

```
select name
from   Beers
where  manf in
(select manf  from beers group
by manf having count(name) = 1)
```

# Problem solving on Beer Database




Which brewer makes the most beers?

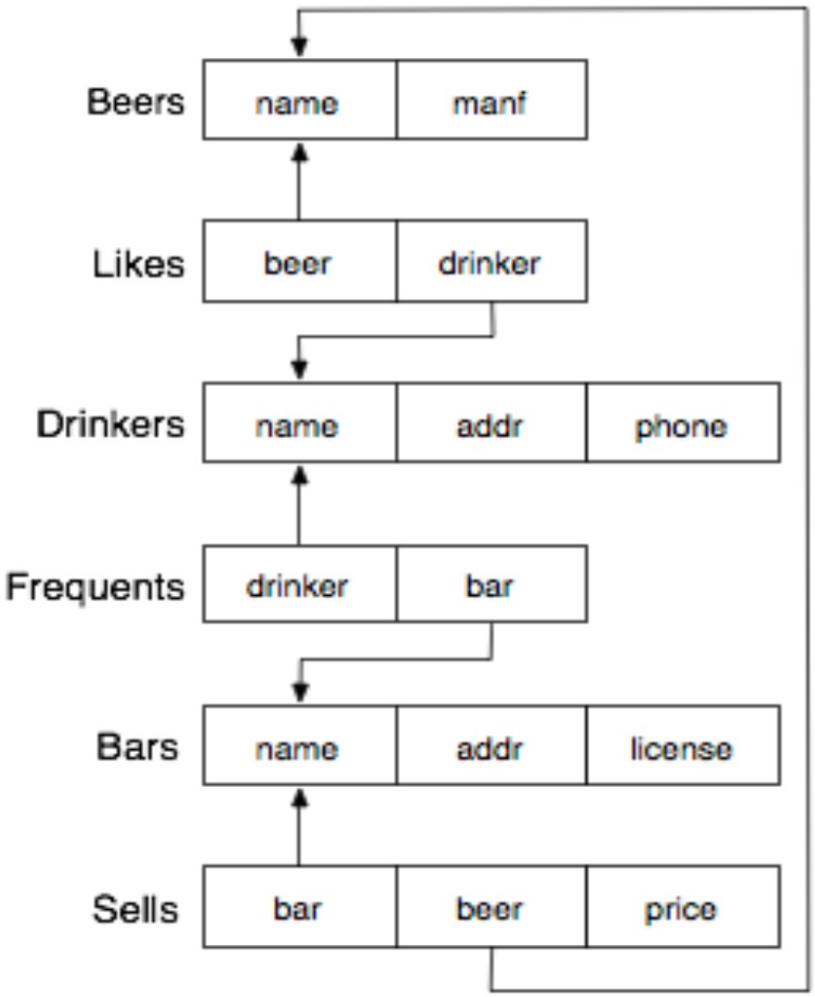
# Problem solving on Beer Database




```
create view
BrewersBeers(brewer,nbeers) as
select manf,count(name)
from   Beers
group by manf;

select brewer
from   BrewersBeers
where  nbeers =
(select max(nbeers) from
BrewersBeers);
```

# Problem solving on Beer Database




Which bar is most popular?

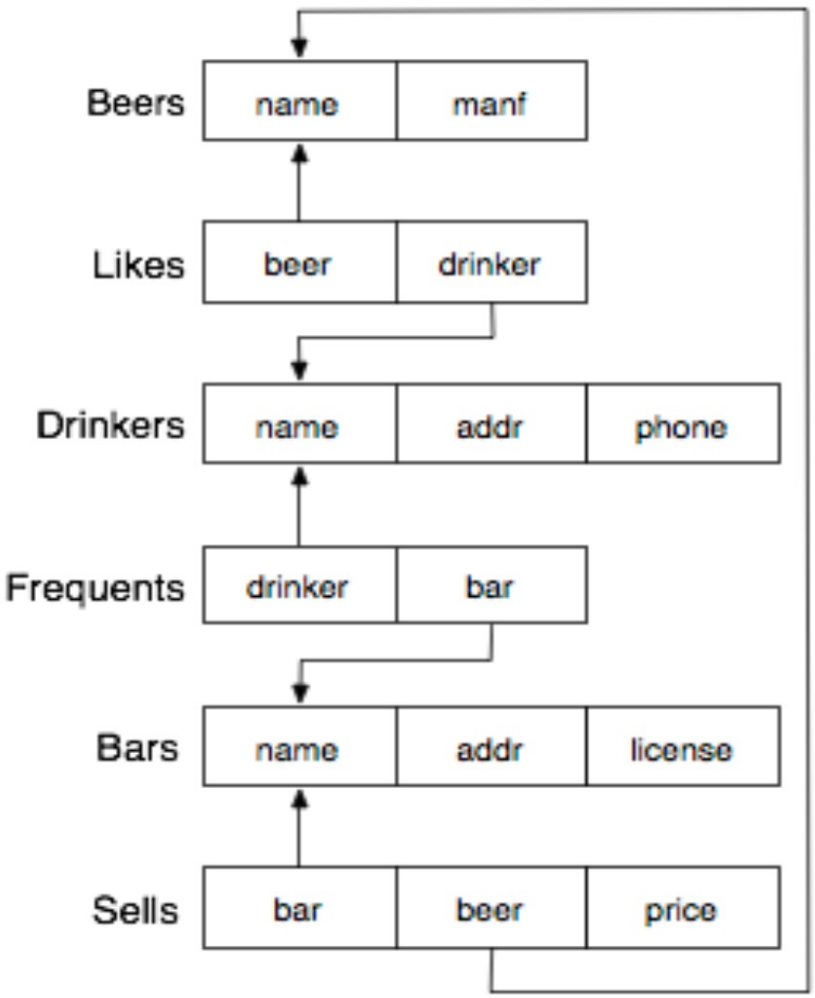
(Most drinkers)

Which bar is most expensive?

(Highest average price)

Name of cheapest beer at each bar?

# Problem solving on Beer Database

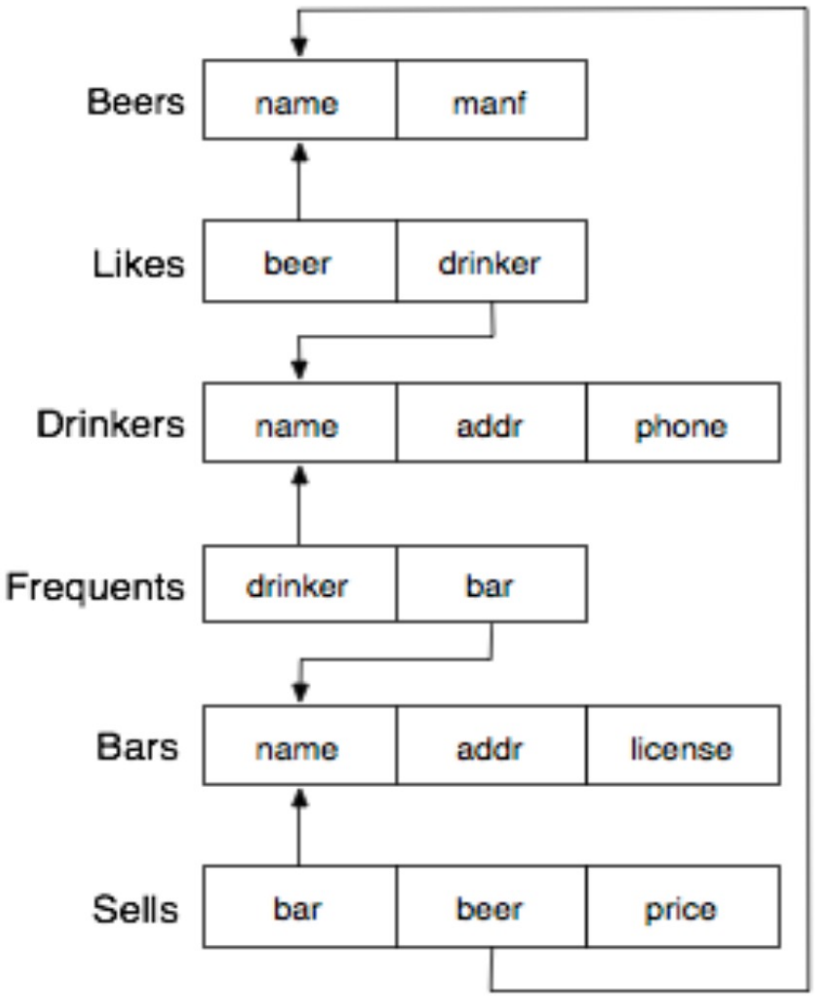



create or replace view v(bar, ndrink) as

select bar, count(drinker) from frequents group by bar;

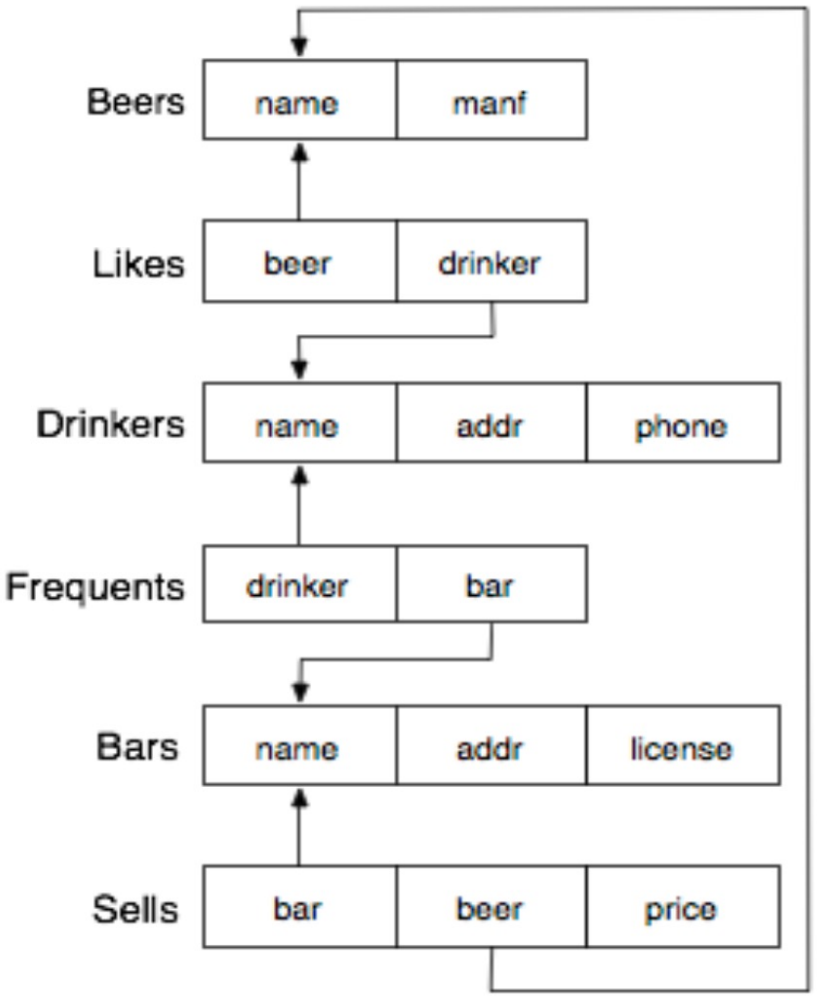
select bar from v where ndrink = (select max(ndrink) from v);

# Problem solving on Beer Database



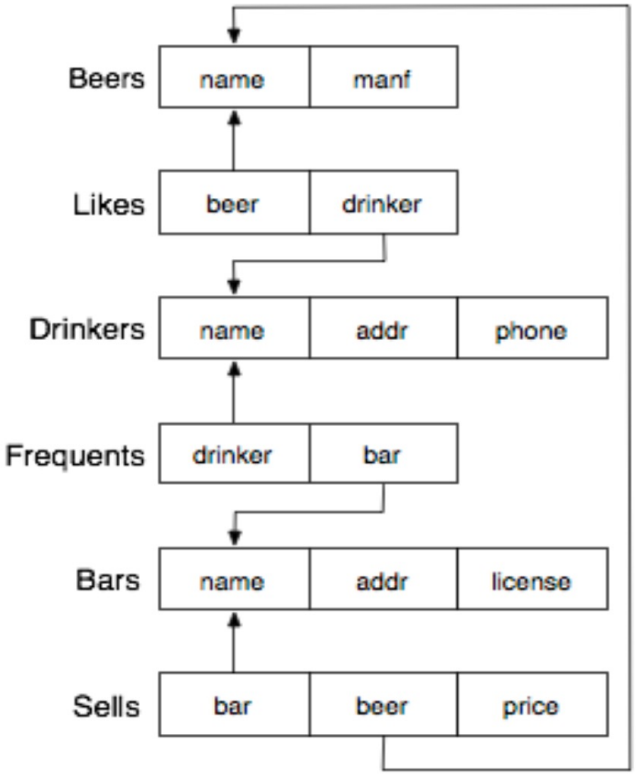

Find the average price of common beers (i.e. served in more than two bars).

# Problem solving on Beer Database




```
select beer, count(bar),
avg(price)
from sells
group by beer
having count(bar) > 2;
```

# Problem solving on Beer Database




```
cp /home/cs9311/web/24T1/lab/beer_data/data.sql ./
cp /home/cs9311/web/24T1/lab/beer_data/schema.sql ./
```