Views and tables:

```
CREATE temporary TABLE Animal (
animal_id integer primary key,
lrid integer NOT NULL default 0,
tag varchar(16) NOT NULL default '',
rfid varchar(15) NOT NULL default '',
nlis varchar(16) NOT NULL default '',
is_new integer NOT NULL default 1,
draft varchar(20) NOT NULL default '',
sex varchar(20) NOT NULL default '',
dob timestamp,
sire varchar(16) NOT NULL default '',
dam varchar(16) NOT NULL default '',
breed varchar(20) NOT NULL default '',
colour varchar(20) NOT NULL default '',
weaned integer NOT NULL default 0 ,
prev_tag varchar(10) NOT NULL default '',
prev_pic varchar(20) NOT NULL default '',
note varchar(30) NOT NULL default '',
note_date timestamp,
is_exported integer NOT NULL default 0,
is_history integer NOT NULL default 0,
is_deleted integer NOT NULL default 0,
tag_sorter varchar(48) NOT NULL default '',
donordam varchar(16) NOT NULL default '',
whp timestamp,
esi timestamp,
status varchar(20) NOT NULL default '',
status_date timestamp,
overall_adg varchar(20) NOT NULL default '',
current_adg varchar(20) NOT NULL default '',
last_weight varchar(20) NOT NULL default '',
last_weight_date timestamp,
selected integer default 0,
animal_group varchar(20) NOT NULL default '',
current_farm varchar(20) NOT NULL default '',
current_property varchar(20) NOT NULL default '',
current_area varchar(20) NOT NULL default '',
current_farm_date timestamp,
current_property_date timestamp,
current_area_date timestamp,
animal_group_date timestamp,
sex_date timestamp,
```

```
breed_date timestamp,
dob_date timestamp,
colour_date timestamp,
prev_pic_date timestamp,
sire_date timestamp,
dam_date timestamp,
donordam_date timestamp,
prev_tag_date timestamp,
tag_date timestamp,
rfid_date timestamp,
nlis_date timestamp,
modified timestamp,
full_rfid varchar(16) default '',
full_rfid_date timestamp);
CREATE temporary  TABLE SessionAnimalActivity (
session_id integer NOT NULL,
animal_id integer NOT NULL,
activity_code integer NOT NULL,
when_measured timestamp NOT NULL,
latestForSessionAnimal integer default 1,
latestForAnimal integer default 1,
is_history integer NOT NULL default 0,
is_exported integer NOT NULL default 0,
is_deleted integer default 0,
primary key( session_id, animal_id, activity_code, when_measured ));
CREATE temporary  TABLE SessionAnimalTrait (
session_id integer NOT NULL,
animal_id integer NOT NULL,
trait_code integer NOT NULL,
alpha_value varchar(20) NOT NULL default '',
alpha_units varchar(10) NOT NULL default '',
when_measured timestamp NOT NULL,
latestForSessionAnimal integer default 1,
latestForAnimal integer default 1,
is_history integer NOT NULL default 0,
is_exported integer NOT NULL default 0,
is_deleted integer default 0,
primary key(session_id, animal_id, trait_code, when_measured));
CREATE temporary  TABLE PicklistValue (
picklistvalue_id integer primary key,
picklist_id integer,
value varchar(30));
-- read the CSV file into the table
\copy Animal from 'Animal.csv' with DELIMITER ',' CSV HEADER;
```

```
-- read the CSV file into the table
\copy SessionAnimalActivity from 'SessionAnimalActivity.csv' WITH DELIMITER ',' CSV
HEADER;
-- read the CSV file into the table
\copy SessionAnimalTrait from 'SessionAnimalTrait.csv' WITH DELIMITER ',' CSV HEADER;
-- read the CSV file into the table
\copy PicklistValue from 'PicklistValue.csv' WITH DELIMITER ',' CSV HEADER;
-- drop table goat cascade;
Create table goat as select Animal_id, rfid, Tag, Dob, Dam, prev_tag sire from animal;
-- drop table birth_weight cascade;
Create table birth_weight2 as select animal_id, alpha_value from sessionanimaltrait where
trait_code = 357 and alpha_value<>'';
create table birth_weight as select animal_id, min(alpha_value) as alpha_value from
birth_weight2 group by animal_id;
-- drop table weight cascade;
Create table weight as select animal_id, alpha_value, when_measured from sessionanimaltrait
where trait_code = 53;
update weight set alpha_value = '0' where alpha_value = '';
-- drop table dam cascade;
create or replace view dams as select distinct(dam) from animal;
Create table dam as select goat.* from goat join dams on dams.dam = goat.tag;
-- drop table child;
-- drop view dams cascade;
Create table child as select animal_id, tag, dam from goat order by dam;

-- drop table Season CASCADE;
Create table Season (season integer primary key, seasonName varchar(6) not null);
-- drop table Season_Month;
Create table Season_Month (
        Month integer primary key,
        Season integer references season
);
create table saledate as select Animal_id,status_date as saledate from animal where status
='Sold';
create or replace view lastweight as select animal_id, max(when_measured) as salesweight
from weight group by animal_id;
create or replace view salesweights as select l.animal_id, l.salesweight from lastweight as l
inner join saledate as s on l.animal_id = s.animal_id;
create or replace view sws as select s.*, w.alpha_value from salesweights as s inner join weight
as w on s.animal_id = w.animal_id and s.salesweight=w.when_measured ;

--drop table Animal;
drop table PicklistValue;
drop table SessionAnimalActivity;
```

```
drop table SessionAnimalTrait;

Insert into Season (season, seasonname) values(1,'Winter');
Insert into Season (season, seasonName) values(2,'Spring');
Insert into Season (season, seasonName) values(3,'Summer');
Insert into Season (season, seasonName) values(4,'Fall');
INSERT into Season_Month (month, season) values(1,1);
INSERT into Season_Month (month, season) values(2,2);
INSERT into Season_Month (Month, season) values(3,2);
INSERT into Season_Month (Month, season) values(4,2);
INSERT into Season_Month (Month, season) values(5,3);
INSERT into Season_Month (Month, season) values(6,3);
INSERT into Season_Month (Month, season) values(7,3);
INSERT into Season_Month (Month, season) values(8,4);
INSERT into Season_Month (Month, season) values(9,4);
INSERT into Season_Month (Month, season) values(10,4);
INSERT into Season_Month (Month, season) values(11,1);
INSERT into Season_Month (Month, season) values(12,1);

drop view dams;
Create or replace view damtags as select distinct dam as tag from goat;
Create or replace view dams as select a.* from goat as a natural join damtags as d;
Create or replace view kids as select a.* from goat as a inner join dams as d on a.dam = d.tag
order by a.dam;
Create or replace view nk as select dam, count(animal_id) as num_kids from kids group by
dam;
Create or replace view damwnk as select dams.*, nk.num_kids from dams inner join nk on
dams.tag = nk.dam;
Create or replace view kidwns as select kids.*, nk.num_kids as num_sibs from kids join nk on
kids.dam = nk.dam;
Create or replace view damwbw as select d.*, b.alpha_value as birth_weight from damwnk as d
join birth_weight as b on d.animal_id = b.animal_id order by d.dob;
Create or replace view kidwbw as select k.*, b.alpha_value as birth_weight from kidwns as k join
birth_weight as b on k.animal_id = b.animal_id order by k.dob;
create view bbm as select extract(month from dob) as month ,count(*) from goat group by
month order by month;

create or replace view winterweights as select * from weight where extract(month from
when_measured) = 11 or extract(month from when_measured) = 12 or extract(month from
when_measured) = 1;

create or replace view yob as select animal_id, extract(year from dob) as year, extract(month
from dob) as month  from goat;
```

```
create or replace view ww2 as select y.animal_id, w.alpha_value from yob as y join weight as w
on (y.animal_id=w.animal_id) and (extract(year from w.when_measured)=y.year) and
((extract(month from w.when_measured) > y.month+2) and (extract(month from
w.when_measured)<y.month+5)) and y.month<8;
create or replace view ww as select animal_id, min(alpha_value) as alpha_value from ww2
group by animal_id;
```