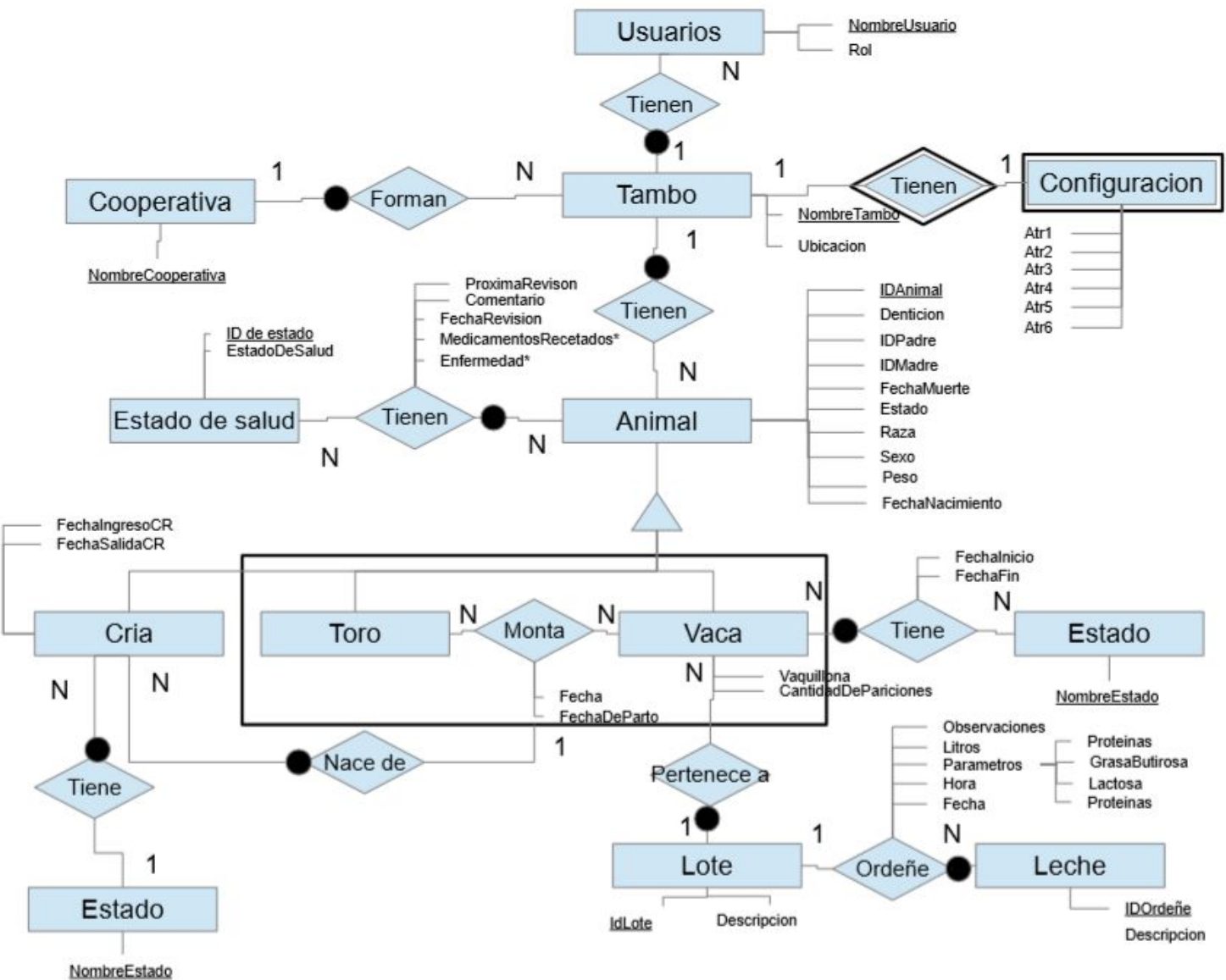




Índice

Índice	2
DER	3
Pasaje a tablas	3
RNE (Restricciones No Estructurales)	4
Diccionario de datos	4
Estudio de permisos para los distintos usuarios del sistema	10
Script de creación de la base de datos	11

DER



Pasaje a tablas

- Cooperativa (NombreCooperativa)
- Tambo(NombreTambo; Ubicacion; NombreCooperativa)
- TamboConfig(NombreTambo; atr1; atr2; atr3; atr4; atr5; atr6)



- Usuarios(NombreUsuario; Rol; NombreTambo)
- Animal(IdAnimal; Denticion; IdPadre; IdMadre; FechaMuerte; Estado; Raza; Sexo; Peso; FechaNacimiento; NombreTambo)
- EstadoSalud(EstadoSalud)
- TieneSalud(IdAnimal; EstadoSalud; FechaRevision; ProximaRevision; Comentario)
- Enfermedad(IdAnimal; EstadoSalud; FechaRevision; NombreEnfermedad)
- Medicamento(IdAnimal; EstadoSalud; FechaRevision; NombreMedicamento)
- Vaca(IdAnimal; CantidadPariciones; IdLote; Vaquillona)
- VacaEstado(EstadoVaca)
- TieneEstadoV(IdAnimal; EstadoVaca; FechaInicio; FechaFin)
- Toro(IdAnimal)
- Monta(ToroId; VacaId; FechaInicio; FechaParto)
- Cria(IdAnimal; EstadoCria; FechaIngresoCR; FechaSalidaCR)
- EstadoCria(EstadoCria)
- Lote(IdLote; Descripcion)
- Leche(IdLeche; Descripcion)
- Ordeño(IdLote; IdLeche; Hora; Fecha; Observaciones; Litros; GrasaButirosa; Proteinas; Lactosa; CelulasSomaticas)

RNE (Restricciones No Estructurales)

- Si Animal.Estado es "Vivo", no permitir el ingreso de fecha muerte
- Animal.FechaMuerte debe ser mayor o igual a Animal.FechaNacimiento
- Vaca.CantidadPariciones es "0" entonces Vaca.Vaquillona es "Vaquillona"
- Vaca.CantidadPariciones no puede tener valores negativos
- Cria.FechaIngresoCR debe ser mayor a Cria.FechaSalidaCR
- Cria.EstadoCria debe ser "En campo de recría" para que se pueda ingresar Cria.FechaIngresoCR y Cria.FechaSalidaCR
- TieneSalud.ProximaRevision debe ser mayor que TieneSalud.FechaRevision
- Si Animal.Sexo es "Macho" solo puede ser cria o toro
- Si Animal.Sexo es "Hembra" solo puede ser cria o vaca
- Si TieneSalud.EstadoSalud es "Sano", no debe de haber registro en Enfermedad que coincidan con ese animal en esa fecha

Diccionario de datos

-Cooperativa:

NombreCooperativa:Refiere al nombre de la cooperativa. Atributo tipo char(20);no puede ser null; clave primaria.

-Tambo:

NombreTambo:Refiere al nombre del tambo. Atributo tipo char(20); no puede ser null; clave primaria.

Ubicacion:Refiere a la ubicacion del tambo. Atributo tipo char(70); puede ser null.

NombreCooperativa: Es un atributo foráneo que proviene de Cooperativa; no puede ser null.

-TamboConfig:

NombreTambo: Es un atributo foráneo que proviene de Tambo; no puede ser null; clave primaria.

Atr1: Es un atributo que se encarga de guardar un valor de configuración dentro del programa. Atributo tipo integer, no puede ser null.

Atr2: Es un atributo que se encarga de guardar un valor de configuración dentro del programa. Atributo tipo integer, no puede ser null.

Atr3: Es un atributo que se encarga de guardar un valor de configuración dentro del programa. Atributo tipo integer, no puede ser null.

Atr4: Es un atributo que se encarga de guardar un valor de configuración dentro del programa. Atributo tipo integer, no puede ser null.

Atr5: Es un atributo que se encarga de guardar un valor de configuración dentro del programa. Atributo tipo integer, no puede ser null.

Atr6: Es un atributo que se encarga de guardar un valor de configuración dentro del programa. Atributo tipo integer, no puede ser null.

-Usuarios:

NombreUsuario: Debe de coincidir con el nombre de usuario que se utiliza para ingresar al sistema operativo de linux. Atributo tipo char(20), no puede ser null; es clave primaria.

Rol: Indica el rol del usuario en el sistema. Atributo tipo char(20), no puede ser null.

NombreTambo: Es un atributo foráneo que proviene de Tambo; no puede ser null.

-Animal:

IdAnimal: Refiere al código de la caravana del ganado. Atributo tipo char(20); no puede ser null; clave primaria.

NombreTambo: Es un atributo foráneo que proviene de Tambo; no puede ser null.

IdPadre: Refiere al código de la caravana del padre. Atributo tipo char(20); no puede ser null.



IdMadre: Refiere al código de la caravana de la madre. Atributo tipo char(20); no puede ser null.

Denticion: Es un código que se usa para datar la edad del animal en base a su dentadura. Atributo tipo char(20); puede ser null

Estado: Refiere a si el animal está vivo o muerto. Atributo tipo char(20); no puede ser null.

Peso: Es el peso en kilos del animal, en la base de datos se guarda el valor numérico únicamente. Atributo tipo smallfloat; puede ser null.

Raza: Es la raza del animal. Atributo tipo char(20); puede ser null.

Sexo: Refiere a si el animal es macho o hembra. Atributo tipo char(20); no puede ser null.

FechaNacimiento: Es la fecha de nacimiento del animal. Atributo tipo date; no puede ser null.

FechaMuerte: Es la fecha de muerte del animal. Atributo tipo date; puede ser null.

-EstadoSalud:

EstadoSalud: Se encarga de guardar solo dos valores "Sano" o "Enfermo", no puede ser null; clave primaria.

-TieneSalud:

IdAnimal: Es un atributo foráneo que proviene de Animal; no puede ser null; forma parte de la clave primaria.

EstadoSalud: Es un atributo foráneo que proviene de EstadoSalud; no puede ser null; forma parte de la clave primaria.

FechaRevision: Refiere a la fecha en la que se realizó la revisión al animal. Atributo tipo date; no puede ser null; forma parte de la clave primaria.

ProximaRevision: Refiere a la fecha cuando posiblemente se realice una próxima revisión. Atributo tipo date; puede ser null.

Comentario: Alguna observación que se quiera agregar o dato que sea útil para el usuario. Atributo tipo char(100), puede ser null.

-Enfermedad:

IdAnimal: Es un atributo foráneo que proviene de TieneSalud; no puede ser null; forma parte de la clave primaria.

EstadoSalud: Es un atributo foráneo que proviene de TieneSalud; no puede ser null; forma parte de la clave primaria.

FechaRevision: Es un atributo foráneo que proviene de TieneSalud; no puede ser null; forma parte de la clave primaria.

NombreEnfermedad: Refiere al nombre de la enfermedad que se le detectó al animal en la revisión. Atributo tipo char(20), no puede ser null, forma parte de la clave primaria.

-Medicamento:

IdAnimal: Es un atributo foráneo que proviene de TieneSalud; no puede ser null; forma parte de la clave primaria.

EstadoSalud: Es un atributo foráneo que proviene de TieneSalud; no puede ser null; forma parte de la clave primaria.

FechaRevision: Es un atributo foráneo que proviene de TieneSalud; no puede ser null; forma parte de la clave primaria.

NombreMedicamento: Refiere al nombre del medicamento recetado al animal luego de la revisión. Atributo tipo char(20), no puede ser null, forma parte de la clave primaria.

-Vaca:

IdAnimal: Es un atributo foráneo que proviene de Animal; no puede ser null; es la clave primaria.

CantidadPariciones: Refiere a la cantidad que la vaca tuvo crías. Atributo tipo integer; no puede ser null.

IdLote: Es un atributo foráneo que proviene de Lote; no puede ser null.

Vaquillona: Indica si la vaca es vaquillona (nunca a parido) o no. Atributo tipo char(20); no puede ser null.

-VacaEstado:

EstadoVaca: Refiere si la vaca está en pre-parto, en servicio, etc. Atributo tipo char(20); no puede ser null.



-TieneEstadoV:

IdAnimal: Es un atributo foráneo que proviene de Vaca; no puede ser null; forma parte de la clave primaria.

EstadoVaca: Es un atributo foráneo que proviene de VacaEstado. no puede ser null; forma parte de la clave primaria.

FechaInicio: Refiere a la fecha en la que la vaca entró en ese estado. Atributo tipo date; no puede ser null; forma parte de la clave primaria.

FechaFin: Refiere a la posible fecha en la que el animal cambie a otro estado.

Atributo tipo date; puede ser null.

-Toro:

IdAnimal: Es un atributo foráneo que proviene de Animal. no puede ser null; es la clave primaria.

-Monta:

Torold: Es un atributo foráneo que proviene de Toro, no puede ser null; forma parte de la clave primaria.

Vacald: Es un atributo foráneo que proviene de Vaca, no puede ser null; forma parte de la clave primaria.

FechaInicio: Refiere a la fecha en la que se produzco el acto. Atributo tipo date, no puede ser null, forma parte de la clave.

FechaParto: Refiere a la fecha en la que se espera el parto. Atributo tipo date, puede ser null.

-Cria:

IdAnimal: Es un atributo foráneo que proviene de Animal. No puede ser null; forma parte de la clave primaria.

EstadoCria: Es un atributo foráneo que proviene de EstadoCria. No puede ser null.

FechaIngresoCR: Indica la fecha en la que el animal ingresa al campo de recria. Atributo tipo date, puede ser null

FechaSalidaCR: Indica la fecha en la que el animal sale del campo de recria. Atributo tipo date, puede ser null.

-EstadoCria:

EstadoCria: Refiere al estado de la cria como "Ternero en pie" o "En campo de

recría". Atributo tipo
clave primaria.

char(20); no puede ser null; es la

-Lote:

IdLote: Es el código por el cual se identifican los lote. Atributo tipo integer; no puede ser null; forma parte de la clave primaria.

Descripcion: Es una descripcion del lote. Atributo tipo char(20); no puede ser null; forma parte de la clave primaria.

-Leche:

IdLeche: Es un código por el cual se identifican los tipos de leche. Atributo tipo integer; no puede ser null; es la clave primaria.

Descripcion: Es el nombre de el tipo de leche. Atributo tipo char(20); no puede ser null

-Ordenie:

IdLote: Es un atributo foráneo que proviene de Lote. No puede ser null; forma parte de la clave primaria.

IdLeche: Es un atributo foráneo que proviene de Leche. No puede ser null; forma parte de la clave primaria.

Hora: Refiere a la hora en la que fue hecho el ordeño. Atributo tipo datetime hour to minute; no puede ser null; forma parte de la clave primaria.

Fecha: Refiere a la fecha en la que fue hecho el ordeño. Atributo tipo date; no puede ser null; forma parte de la clave primaria.

Observaciones: Refiere a algún dato que el usuario quiera guardar. Atributo tipo char(70); puede ser null.

Litros: Refiere a la cantidad de litros que se obtuvieron del ordeño. Atributo tipo smallfloat; no puede ser null.

GrasaButirosa: Es un parámetro que conaprole mide de la leche. Atributo tipo smallfloat; puede ser null.

Proteinas: Es un parámetro que conaprole mide de la leche. Atributo tipo smallfloat, puede ser null.



Lactosa: Es un parámetro que
tipo smallfloat, puede ser null.

conaprole mide de la leche. Atributo

CelulasSomaticas: Es un parámetro que conaprole mide de la leche. Atributo tipo
smallfloat, puede ser null.

Estudio de permisos para los distintos usuarios del sistema

Usuario: Administrador del sistema

Permisos sobre la base: DBA

Permisos sobre las tabla: ALL en todas las tablas

Usuario:Gerente Administrador del sistema

Permisos sobre la base: Connect

Permisos sobre las tablas: Select en todas las tablas

Update en todas las tablas

Insert en todas las tablas

Index en todas las tablas

Usuario:Administrador de un establecimiento

Permisos sobre la base: Connect

Permisos sobre las tablas: Select en todas las tablas

Update en todas las tablas, menos en usuarios, cooperativa,
tambo

Insert en todas las tablas, menos en cooperativa, tambo

Index en todas las tablas

Usuario:Gerente Auxiliar

Permisos sobre la base: Connect

Permisos sobre las tablas: Select en todas las tablas

Update en todas las tablas, menos en tamboconfig, usuarios,
cooperativa, animal, vacaestado,



estadocria, cooperativa, tambo,
 estadosalud, leche, lote
 Insert en todas las tablas, menos en tamboconfig, usuarios,
 cooperativa, animal, vacaestado,
 estadocria, cooperativa, tambo,
 estadosalud, leche, lote
 Index en todas las tablas

Los permisos de alter y delete no se le asignan a ningún usuario además del administrador del sistema.

Script de creación de la base de datos

```
{ DATABASE tilfa delimiter | }
```

```
grant dba to "fdelbono";
```

```
{ TABLE "fdelbono".cooperativa row size = 20 number of columns = 1 index size = 25  
  }
```

```
{ unload file name = coope00100.unl number of rows = 0 }
```

```
create table "fdelbono".cooperativa  
(  
  nombrecooperativa char(20),  
  primary key (nombrecooperativa) constraint "fdelbono".pk1  
);
```

```
revoke all on "fdelbono".cooperativa from "public" as "fdelbono";
```

```
{ TABLE "fdelbono".tambo row size = 90 number of columns = 3 index size = 50 }
```

```
{ unload file name = tambo00101.unl number of rows = 0 }
```

```
create table "fdelbono".tambo  
(  
  nombretambo char(20) not null ,  
  ubicacion char(50),  
  nombrecooperativa char(20) not null ,  
  primary key (nombretambo) constraint "fdelbono".pk2  
);
```

```
revoke all on "fdelbono".tambo from "public" as "fdelbono";
```



```
{ TABLE "fdelbono".usuarios row size = 60 number of columns = 3 index
size = 50 }
{ unload file name = usuar00102.unl number of rows = 0 }
```

```
create table "fdelbono".usuarios
(
  nombreusuario char(20) not null ,
  rol char(20) not null ,
  nombretambo char(20) not null ,
  primary key (nombreusuario) constraint "fdelbono".pk3
);
```

```
revoke all on "fdelbono".usuarios from "public" as "fdelbono";
```

```
{ TABLE "fdelbono".animal row size = 228 number of columns = 12 index size = 50 }
{ unload file name = anima00103.unl number of rows = 0 }
```

```
create table "fdelbono".animal
(
  idanimal char(20) not null ,
  denticion char(20),
  idpadre char(20) not null ,
  idmadre char(20) not null ,
  fechamuerte date,
  estado char(20) not null ,
  raza char(20),
  sexo char(20) not null ,
  peso smallfloat,
  fechanacimiento date not null ,
  nombretambo char(20) not null ,
  primary key (idanimal) constraint "fdelbono".pk4
);
```

```
revoke all on "fdelbono".animal from "public" as "fdelbono";
```

```
{ TABLE "fdelbono".estadosalud row size = 20 number of columns = 1 index size = 25
}
{ unload file name = estad00104.unl number of rows = 0 }
```

```
create table "fdelbono".estadosalud
(
  estadosalud char(20) not null ,
  primary key (estadosalud) constraint "fdelbono".pk5
);
```

```
revoke all on "fdelbono".estadosalud from "public" as "fdelbono";
```

```
{ TABLE "fdelbono".tienesalud row size = 148 number of columns = 5 index size = 99
}
```

```
{ unload file name = tiene00105.unl number of rows = 0 }
```

```
create table "fdelbono".tienesalud
```

```
(
  idanimal char(20) not null ,
  estadosalud char(20) not null ,
  fecharevision date not null ,
  proximarevision date,
  comentario char(100),
  primary key (idanimal,estadosalud,fecharevision) constraint "fdelbono".pk6
);
```

```
revoke all on "fdelbono".tienesalud from "public" as "fdelbono";
```

```
{ TABLE "fdelbono".enfermedad row size = 64 number of columns = 4 index size = 118
}
```

```
{ unload file name = enfer00106.unl number of rows = 0 }
```

```
create table "fdelbono".enfermedad
```

```
(
  idanimal char(20) not null ,
  estadosalud char(20) not null ,
  fecharevision date not null ,
  nombreenfermedad char(20),
  primary key (idanimal,estadosalud,fecharevision,nombreenfermedad) constraint
    "fdelbono".pk7
);
```

```
revoke all on "fdelbono".enfermedad from "public" as "fdelbono";
```

```
{ TABLE "fdelbono".medicamento row size = 64 number of columns = 4 index size = 118
}
```

```
{ unload file name = medic00107.unl number of rows = 0 }
```

```
create table "fdelbono".medicamento
```

```
(
  idanimal char(20) not null ,
  estadosalud char(20) not null ,
  fecharevision date not null ,
  nombremedicamento char(20) not null ,
  primary key (idanimal,estadosalud,fecharevision,nombremedicamento) constraint
    "fdelbono".pk8
);
```

revoke all on "fdelbono".medicamento from "public" as "fdelbono";

```
{ TABLE "fdelbono".vaca row size = 48 number of columns = 4 index size = 34 }
{ unload file name = vaca_00108.unl number of rows = 0 }
```

```
create table "fdelbono".vaca
(
    idanimal char(20) not null ,
    cantidadpariciones integer not null ,
    idlote integer not null ,
    vaquillona char(20) not null ,
    primary key (idanimal) constraint "fdelbono".pk9
);
```

revoke all on "fdelbono".vaca from "public" as "fdelbono";

```
{ TABLE "fdelbono".vacaestado row size = 20 number of columns = 1 index size = 25
    }
{ unload file name = vacae00109.unl number of rows = 0 }
```

```
create table "fdelbono".vacaestado
(
    estadovaca char(20) not null ,
    primary key (estadovaca) constraint "fdelbono".pk10
);
```

revoke all on "fdelbono".vacaestado from "public" as "fdelbono";

```
{ TABLE "fdelbono".tienestadov row size = 48 number of columns = 4 index size = 99
    }
{ unload file name = tiene00110.unl number of rows = 0 }
```

```
create table "fdelbono".tienestadov
(
    idanimal char(20) not null ,
    estadovaca char(20) not null ,
    fechainicio date not null ,
    fechafin date,
    primary key (idanimal,estadovaca,fechainicio) constraint "fdelbono".pk11
);
```

revoke all on "fdelbono".tienestadov from "public" as "fdelbono";

```
{ TABLE "fdelbono".toro row size = 20 number of columns = 1 index size = 25 }
```



```
{ unload file name = toro_00111.unl          number of rows = 0 }
```

```
create table "fdelbono".toro
(
    idanimal char(20) not null ,
    primary key (idanimal) constraint "fdelbono".pk12
);
```

```
revoke all on "fdelbono".toro from "public" as "fdelbono";
```

```
{ TABLE "fdelbono".monta row size = 48 number of columns = 4 index size = 99 }
{ unload file name = monta00112.unl number of rows = 0 }
```

```
create table "fdelbono".monta
(
    toroid char(20) not null ,
    vacaid char(20) not null ,
    fechainicio date not null ,
    fechaparto date,
    primary key (toroid,vacaid,fechainicio) constraint "fdelbono".pk13
);
```

```
revoke all on "fdelbono".monta from "public" as "fdelbono";
```

```
{ TABLE "fdelbono".cria row size = 48 number of columns = 4 index size = 50 }
{ unload file name = cria_00113.unl number of rows = 0 }
```

```
create table "fdelbono".cria
(
    idanimal char(20) not null ,
    estadocria char(20) not null ,
    fechaingresocr date,
    fechasalidacr date,
    primary key (idanimal) constraint "fdelbono".pk14
);
```

```
revoke all on "fdelbono".cria from "public" as "fdelbono";
```

```
{ TABLE "fdelbono".lote row size = 24 number of columns = 2 index size = 9 }
{ unload file name = lote_00114.unl number of rows = 0 }
```

```
create table "fdelbono".lote
(
    idlote integer not null ,
    descripcion char(20) not null ,
    primary key (idlote) constraint "fdelbono".pk15
);
```



);

revoke all on "fdelbono".lote from "public" as "fdelbono";

```
{ TABLE "fdelbono".leche row size = 24 number of columns = 2 index size = 9 }
{ unload file name = leche00115.unl number of rows = 0 }
```

```
create table "fdelbono".leche
(
  idleche integer not null ,
  descripcion char(20) not null ,
  primary key (idleche) constraint "fdelbono".pk16
);
```

revoke all on "fdelbono".leche from "public" as "fdelbono";

```
{ TABLE "fdelbono".tamboconfig row size = 44 number of columns = 7 index size = 74
      }
{ unload file name = tambo00118.unl number of rows = 0 }
```

```
create table "fdelbono".tamboconfig
(
  nombretambo char(20) not null ,
  atr1 integer not null ,
  atr2 integer not null ,
  atr3 integer not null ,
  atr4 integer not null ,
  atr5 integer not null ,
  atr6 integer not null ,
  primary key (nombretambo,atr1,atr2,atr3,atr4,atr5,atr6) constraint "fdelbono".pk18
);
```

revoke all on "fdelbono".tamboconfig from "public" as "fdelbono";

```
{ TABLE "fdelbono".estadocria row size = 20 number of columns = 1 index size = 25
      }
{ unload file name = estad00119.unl number of rows = 0 }
```

```
create table "fdelbono".estadocria
(
  nombrestado char(20) not null ,
  primary key (nombrestado) constraint "fdelbono".pk19
);
```

revoke all on "fdelbono".estadocria from "public" as "fdelbono";



```
{ TABLE "fdelbono".ordenie row size = 105 number of columns = 10 index size = 38
}
{ unload file name = orden00120.unl number of rows = 0 }
```

```
create table "fdelbono".ordenie
(
  idlote integer not null ,
  idleche integer not null ,
  hora datetime hour to minute not null ,
  fecha date not null ,
  observaciones char(70),
  litros smallfloat not null ,
  grasabutirosa smallfloat,
  proteinas smallfloat,
  lactosa smallfloat,
  celulassomaticas smallfloat,
  primary key (idlote,idleche,hora,fecha) constraint "fdelbono".pk17
);
```

```
revoke all on "fdelbono".ordenie from "public" as "fdelbono";
```

```
grant select on "fdelbono".cooperativa to "public" as "fdelbono";
grant update on "fdelbono".cooperativa to "public" as "fdelbono";
grant insert on "fdelbono".cooperativa to "public" as "fdelbono";
grant delete on "fdelbono".cooperativa to "public" as "fdelbono";
grant index on "fdelbono".cooperativa to "public" as "fdelbono";
grant select on "fdelbono".tambo to "public" as "fdelbono";
grant update on "fdelbono".tambo to "public" as "fdelbono";
grant insert on "fdelbono".tambo to "public" as "fdelbono";
grant delete on "fdelbono".tambo to "public" as "fdelbono";
grant index on "fdelbono".tambo to "public" as "fdelbono";
grant select on "fdelbono".usuarios to "public" as "fdelbono";
grant update on "fdelbono".usuarios to "public" as "fdelbono";
grant insert on "fdelbono".usuarios to "public" as "fdelbono";
grant delete on "fdelbono".usuarios to "public" as "fdelbono";
grant index on "fdelbono".usuarios to "public" as "fdelbono";
grant select on "fdelbono".animal to "public" as "fdelbono";
grant update on "fdelbono".animal to "public" as "fdelbono";
grant insert on "fdelbono".animal to "public" as "fdelbono";
grant delete on "fdelbono".animal to "public" as "fdelbono";
grant index on "fdelbono".animal to "public" as "fdelbono";
grant select on "fdelbono".estadosalud to "public" as "fdelbono";
grant update on "fdelbono".estadosalud to "public" as "fdelbono";
grant insert on "fdelbono".estadosalud to "public" as "fdelbono";
grant delete on "fdelbono".estadosalud to "public" as "fdelbono";
```




```
grant index on "fdelbono".estadosalud to "public" as "fdelbono";
grant select on "fdelbono".tienesalud to "public" as "fdelbono";
grant update on "fdelbono".tienesalud to "public" as "fdelbono";
grant insert on "fdelbono".tienesalud to "public" as "fdelbono";
grant delete on "fdelbono".tienesalud to "public" as "fdelbono";
grant index on "fdelbono".tienesalud to "public" as "fdelbono";
grant select on "fdelbono".enfermedad to "public" as "fdelbono";
grant update on "fdelbono".enfermedad to "public" as "fdelbono";
grant insert on "fdelbono".enfermedad to "public" as "fdelbono";
grant delete on "fdelbono".enfermedad to "public" as "fdelbono";
grant index on "fdelbono".enfermedad to "public" as "fdelbono";
grant select on "fdelbono".medicamento to "public" as "fdelbono";
```

```
grant update on "fdelbono".medicamento to "public" as "fdelbono";
grant insert on "fdelbono".medicamento to "public" as "fdelbono";
grant delete on "fdelbono".medicamento to "public" as "fdelbono";
grant index on "fdelbono".medicamento to "public" as "fdelbono";
grant select on "fdelbono".vaca to "public" as "fdelbono";
grant update on "fdelbono".vaca to "public" as "fdelbono";
grant insert on "fdelbono".vaca to "public" as "fdelbono";
grant delete on "fdelbono".vaca to "public" as "fdelbono";
grant index on "fdelbono".vaca to "public" as "fdelbono";
grant select on "fdelbono".vacaestado to "public" as "fdelbono";
grant update on "fdelbono".vacaestado to "public" as "fdelbono";
grant insert on "fdelbono".vacaestado to "public" as "fdelbono";
grant delete on "fdelbono".vacaestado to "public" as "fdelbono";
grant index on "fdelbono".vacaestado to "public" as "fdelbono";
grant select on "fdelbono".tienestadov to "public" as "fdelbono";
grant update on "fdelbono".tienestadov to "public" as "fdelbono";
grant insert on "fdelbono".tienestadov to "public" as "fdelbono";
grant delete on "fdelbono".tienestadov to "public" as "fdelbono";
grant index on "fdelbono".tienestadov to "public" as "fdelbono";
grant select on "fdelbono".toro to "public" as "fdelbono";
grant update on "fdelbono".toro to "public" as "fdelbono";
grant insert on "fdelbono".toro to "public" as "fdelbono";
grant delete on "fdelbono".toro to "public" as "fdelbono";
grant index on "fdelbono".toro to "public" as "fdelbono";
grant select on "fdelbono".monta to "public" as "fdelbono";
grant update on "fdelbono".monta to "public" as "fdelbono";
grant insert on "fdelbono".monta to "public" as "fdelbono";
grant delete on "fdelbono".monta to "public" as "fdelbono";
grant index on "fdelbono".monta to "public" as "fdelbono";
grant select on "fdelbono".cria to "public" as "fdelbono";
grant update on "fdelbono".cria to "public" as "fdelbono";
grant insert on "fdelbono".cria to "public" as "fdelbono";
grant delete on "fdelbono".cria to "public" as "fdelbono";
```

```

grant index on "fdelbono".cria to "public" as "fdelbono";
grant select on "fdelbono".lote to "public" as "fdelbono";
grant update on "fdelbono".lote to "public" as "fdelbono";
grant insert on "fdelbono".lote to "public" as "fdelbono";
grant delete on "fdelbono".lote to "public" as "fdelbono";
grant index on "fdelbono".lote to "public" as "fdelbono";
grant select on "fdelbono".leche to "public" as "fdelbono";
grant update on "fdelbono".leche to "public" as "fdelbono";
grant insert on "fdelbono".leche to "public" as "fdelbono";
grant delete on "fdelbono".leche to "public" as "fdelbono";
grant index on "fdelbono".leche to "public" as "fdelbono";
grant select on "fdelbono".tamboconfig to "public" as "fdelbono";
grant update on "fdelbono".tamboconfig to "public" as "fdelbono";
grant insert on "fdelbono".tamboconfig to "public" as "fdelbono";
grant delete on "fdelbono".tamboconfig to "public" as "fdelbono";
grant index on "fdelbono".tamboconfig to "public" as "fdelbono";
grant select on "fdelbono".estadocria to "public" as "fdelbono";
grant update on "fdelbono".estadocria to "public" as "fdelbono";
grant insert on "fdelbono".estadocria to "public" as "fdelbono";
grant delete on "fdelbono".estadocria to "public" as "fdelbono";
grant index on "fdelbono".estadocria to "public" as "fdelbono";
grant select on "fdelbono".ordenie to "public" as "fdelbono";
grant update on "fdelbono".ordenie to "public" as "fdelbono";
grant insert on "fdelbono".ordenie to "public" as "fdelbono";
grant delete on "fdelbono".ordenie to "public" as "fdelbono";
grant index on "fdelbono".ordenie to "public" as "fdelbono";

```

```

revoke usage on language SPL from public ;

```

```

grant usage on language SPL to public ;

```

```

alter table "fdelbono".tambo add constraint (foreign key (nombrecooperativa)
references "fdelbono".cooperativa on delete cascade constraint
"fdelbono".fk2);
alter table "fdelbono".usuarios add constraint (foreign key (nombretambo)
references "fdelbono".tambo on delete cascade constraint
"fdelbono".fk3);
alter table "fdelbono".animal add constraint (foreign key (nombretambo)
references "fdelbono".tambo on delete cascade constraint
"fdelbono".fk4);
alter table "fdelbono".tienesalud add constraint (foreign key
(idanimal) references "fdelbono".animal on delete cascade
constraint "fdelbono".fk6_1);
alter table "fdelbono".tienesalud add constraint (foreign key
(estadosalud) references "fdelbono".estadosalud on delete
cascade constraint "fdelbono".fk6_2);

```



```
alter table "fdelbono".enfermedad add constraint (foreign key
(idanimal,estadosalud,fecharevision) references "fdelbono"
.tienesalud on delete cascade constraint "fdelbono".fk7);
```

```
alter table "fdelbono".medicamento add constraint (foreign key
(idanimal,estadosalud,fecharevision) references "fdelbono"
.tienesalud on delete cascade constraint "fdelbono".fk8);
```

```
alter table "fdelbono".vaca add constraint (foreign key (idanimal)
references "fdelbono".animal on delete cascade constraint
"fdelbono".fk9_1);
```

```
alter table "fdelbono".tienestadov add constraint (foreign key
(idanimal) references "fdelbono".vaca on delete cascade constraint
"fdelbono".fk11_1);
```

```
alter table "fdelbono".tienestadov add constraint (foreign key
(estadovaca) references "fdelbono".vacaestado on delete cascade
constraint "fdelbono".fk11_2);
```

```
alter table "fdelbono".toro add constraint (foreign key (idanimal)
references "fdelbono".animal on delete cascade constraint
"fdelbono".fk12);
```

```
alter table "fdelbono".monta add constraint (foreign key (toroid)
references "fdelbono".toro on delete cascade constraint "fdelbono"
.fk13_1);
```

```
alter table "fdelbono".monta add constraint (foreign key (vacaid)
references "fdelbono".vaca on delete cascade constraint "fdelbono"
.fk13_2);
```

```
alter table "fdelbono".cria add constraint (foreign key (idanimal)
references "fdelbono".animal on delete cascade constraint
"fdelbono".fk14);
```

```
alter table "fdelbono".vaca add constraint (foreign key (idlote)
references "fdelbono".lote on delete cascade constraint "fdelbono"
.fk9_2);
```

```
alter table "fdelbono".ordenie add constraint (foreign key (idlote)
references "fdelbono".lote on delete cascade constraint "fdelbono"
.fk17_1);
```

```
alter table "fdelbono".ordenie add constraint (foreign key (idleche)
references "fdelbono".leche on delete cascade constraint
"fdelbono".fk17_2);
```

```
alter table "fdelbono".tamboconfig add constraint (foreign key
(nombretambo) references "fdelbono".tambo on delete cascade
constraint "fdelbono".fk18);
```

```
alter table "fdelbono".cria add constraint (foreign key (estadocria)
references "fdelbono".estadocria on delete cascade constraint
"fdelbono".fk14_2);
```

