Dbschema
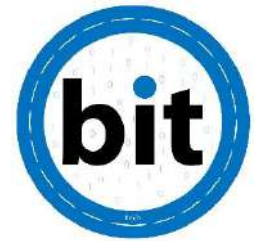
Tercera entrega 1/11/2019

Ruta en GitLab: /Actividades/BD01006/

```sql
grant dba to "root";


create role "operario" ;
create role "transportista" ;
create role "administrativo" ;

grant "operario" to "ope" ;
grant "transportista" to "trans" ;
grant "administrativo" to "admin" ;

grant default role "operario" to "ope" ;
grant default role "transportista" to "trans" ;
grant default role "administrativo" to "admin" ;
```

```sql
{ TABLE "root".usuario row size = 633 number of columns = 16 index size = 44 }

create table "root".usuario
  (
    idusuario serial not null ,
    nombredeusuario varchar(20) not null ,
    hash_contra char(60) not null ,
    email varchar(255) not null ,
    fechanac date,
    telefono varchar(15) not null ,
    primernombre varchar(50) not null ,
    primerapellido varchar(50) not null ,
    preguntasecreta varchar(50),
    respuestasecreta varchar(50),
    creador integer,
    fechacreacion date not null ,
    sexo char(1) not null ,
    rol char(1) not null ,
    imagen byte
        default null,
    invalido "informix".boolean
        default 'f' not null ,
    unique (nombredeusuario) ,

    check (rol IN ('A' ,'O' ,'T' )),
```

```sql
      check (sexo IN ('M' ,'F' ,'O' )),
      primary key (idusuario)
   );

revoke all on "root".usuario from "public" as "root";

{ TABLE "root".cliente row size = 129 number of columns = 6 index size = 36 }

create table "root".cliente
   (
      rut varchar(12) not null ,
      nombre varchar(100) not null ,
      idcliente serial not null ,
      fecharegistro datetime year to day not null ,
      invalido "informix".boolean not null ,
      usuarioregistro integer not null ,
      unique (rut) ,
      primary key (idcliente)
   );

revoke all on "root".cliente from "public" as "root";

{ TABLE "root".lugar row size = 155 number of columns = 9 index size = 18 }

create table "root".lugar
   (
      idlugar serial not null ,
      nombre varchar(100) not null ,
      capacidad integer,
      geox float not null ,
      geoy float not null ,
      usuariocreador integer not null ,
      fecharegistro datetime year to second not null ,
      tipo varchar(15) not null ,
      inhabilitado "informix".boolean
          default 'f' not null ,

      check (capacidad > 0 ),

      check (tipo IN ('Patio' ,'Puerto' ,'Establecimiento' ,'Zona' ,'Subzona' )),
      primary key (idlugar)
   );

revoke all on "root".lugar from "public" as "root";

{ TABLE "root".incluye row size = 8 number of columns = 2 index size = 18 }

create table "root".incluye
   (
      menor integer,
      mayor integer,
      primary key (menor)
   );
```

```sql
revoke all on "root".incluye from "public" as "root";

{ TABLE "root".pertenecea row size = 8 number of columns = 2 index size = 18 }

create table "root".pertenecea
  (
    idlugar integer,
    clienteid integer,
    primary key (idlugar)
  );

revoke all on "root".pertenecea from "public" as "root";

{ TABLE "root".trabajaen row size = 20 number of columns = 5 index size = 27 }

create table "root".trabajaen
  (
    id serial not null ,
    idlugar integer,
    idusuario integer not null ,
    fechainicio date not null ,
    fechafin date,
    primary key (id)
  );

revoke all on "root".trabajaen from "public" as "root";

{ TABLE "root".conexion row size = 24 number of columns = 4 index size = 35 }

create table "root".conexion
  (
    idtrabajaen integer,
    horaingreso datetime year to second not null ,
    horasalida datetime year to second,
    usuario integer,
    primary key (usuario,horaingreso)
  );

revoke all on "root".conexion from "public" as "root";

{ TABLE "root".vehiculo row size = 145 number of columns = 8 index size = 40 }

create table "root".vehiculo
  (
    idvehiculo serial not null ,
    vin char(17) not null ,
    marca varchar(50),
    modelo varchar(50),
    color char(6),
    tipo varchar(7),
    anio integer,
    cliente integer not null ,
    unique (vin) ,
```

```
      check (tipo IN ('Auto' ,'MiniVan' ,'SUV' ,'Camion' ,'Van' )),

      check ((anio >= 1900 ) AND (anio <= 10000 ) ),
      primary key (idvehiculo)
   );

revoke all on "root".vehiculo from "public" as "root";

{ TABLE "root".vehiculoingresa row size = 4128 number of columns = 5 index size = 39 }

create table "root".vehiculoingresa
   (
      idvehiculo integer,
      fecha datetime year to second,
      tipoingreso varchar(10) not null ,
      usuario integer,
      detalle "informix".bson,

      check (tipoingreso IN ('Precarga' ,'Alta' ,'Baja' )),
      primary key (idvehiculo,usuario,fecha)
   );

revoke all on "root".vehiculoingresa from "public" as "root";

{ TABLE "root".informedanios row size = 285 number of columns = 7 index size = 40 }

create table "root".informedanios
   (
      id serial not null ,
      descripcion varchar(255) not null ,
      fecha datetime year to day not null ,
      tipo varchar(7) not null ,
      idvehiculo integer not null ,
      idlugar integer not null ,
      idusuario integer not null ,

      check (tipo IN ('Total' ,'Parcial' )),
      primary key (id,idvehiculo)
   );

revoke all on "root".informedanios from "public" as "root";

{ TABLE "root".registrodanios row size = 268 number of columns = 4 index size = 30 }

create table "root".registrodanios
   (
      idvehiculo integer not null ,
      informedanios integer not null ,
      idregistro serial not null ,
      descripcion varchar(255) not null ,
      primary key (idvehiculo,informedanios,idregistro)
   );
```

```sql
revoke all on "root".registrodanios from "public" as "root";

{ TABLE "root".imagenregistro row size = 72 number of columns = 5 index size = 38 }

create table "root".imagenregistro
  (
    vehiculo integer,
    informe integer,
    nrolista integer,
    nroimagen serial not null ,
    imagen byte not null ,
    primary key (vehiculo,informe,nrolista,nroimagen)
  );

revoke all on "root".imagenregistro from "public" as "root";

{ TABLE "root".actualiza row size = 40 number of columns = 7 index size = 34 }

create table "root".actualiza
  (
    vehiculo1 integer,
    informe1 integer,
    registro1 integer,
    vehiculo2 integer,
    informe2 integer,
    registro2 integer,
    tipo varchar(15) not null ,

    check (tipo IN ('Anulacion' ,'Correccion' )),

    check (vehiculo1 = vehiculo2 ),
    primary key (vehiculo1,informe1,registro1)
  );

revoke all on "root".actualiza from "public" as "root";

{ TABLE "root".posicionado row size = 32 number of columns = 6 index size = 48 }

create table "root".posicionado
  (
    idlugar integer,
    idvehiculo integer,
    desde datetime year to second,
    hasta datetime year to second,
    posicion integer not null ,
    idusuario integer,

    check (posicion > 0 ),
    primary key (idlugar,idvehiculo,desde)
  );

revoke all on "root".posicionado from "public" as "root";

{ TABLE "root".tipotransporte row size = 55 number of columns = 2 index size = 65 }
```

```sql
create table "root".tipotransporte
  (
    idtipo serial not null ,
    nombre varchar(50) not null ,
    unique (nombre) ,
    primary key (idtipo)
  );

revoke all on "root".tipotransporte from "public" as "root";

{ TABLE "root".habilitado row size = 8 number of columns = 2 index size = 31 }

create table "root".habilitado
  (
    idlugar integer,
    idtipo integer,
    primary key (idtipo,idlugar)
  );

revoke all on "root".habilitado from "public" as "root";

{ TABLE "root".mediotransporte row size = 185 number of columns = 11 index size = 134 }

create table "root".mediotransporte
  (
    idtipo integer,
    idlegal varchar(50),
    nombre varchar(50) not null ,
    tipo varchar(50) not null ,
    creador integer not null ,
    fechacreacion date not null ,
    cantcamiones integer not null ,
    cantautos integer not null ,
    cantsuv integer not null ,
    cantvan integer not null ,
    cantminivan integer not null ,

    check (cantcamiones > -1 ),

    check (cantautos > -1 ),

    check (cantsuv > -1 ),

    check (cantvan > -1 ),

    check (cantminivan > -1 ),
    primary key (idtipo,idlegal)
  );

revoke all on "root".mediotransporte from "public" as "root";

{ TABLE "root".permite row size = 61 number of columns = 4 index size = 133 }
```

```sql
create table "root".permite
  (
    idtipo integer,
    idlegal varchar(50),
    usuario integer,
    invalido "informix".boolean,
    primary key (idtipo,idlegal,usuario)
  );

revoke all on "root".permite from "public" as "root";

{ TABLE "root".lote row size = 66 number of columns = 9 index size = 36 }

create table "root".lote
  (
    idlote serial not null ,
    nombre varchar(20),
    origen integer not null ,
    destino integer not null ,
    creadorid integer not null ,
    fechacreacion datetime year to day not null ,
    prioridad varchar(10) not null ,
    estado varchar(10) not null ,
    invalido "informix".boolean
        default 'f' not null ,

    check (prioridad IN ('Normal' ,'Alta' )),

    check (estado IN ('Abierto' ,'Cerrado' )),
    primary key (idlote)
  );

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

{ TABLE "root".integra row size = 21 number of columns = 5 index size = 47 }

create table "root".integra
  (
    idvehiculo integer,
    lote integer,
    fecha datetime year to minute,
    invalidado "informix".boolean not null ,
    idusuario integer not null ,
    primary key (idvehiculo,lote,fecha)
  );

revoke all on "root".integra from "public" as "root";

{ TABLE "root".transporte row size = 77 number of columns = 6 index size = 73 }

create table "root".transporte
  (
    transporteid serial not null ,
    usuario integer not null ,
```

```
      idtipo integer not null ,
      idlegal varchar(50) not null ,
      fechahoracreacion datetime year to minute not null ,
      fechahorasalida datetime year to minute,
      primary key (transporteid)
   );

revoke all on "root".transporte from "public" as "root";

{ TABLE "root".transporta row size = 33 number of columns = 5 index size = 31 }

create table "root".transporta
   (
      transporteid integer,
      idlote integer,
      estado varchar(10) not null ,
      fechahorallegadaestm datetime year to minute,
      fechahorallegadareal datetime year to minute,

      check (estado IN ('Proceso' ,'Fallo' ,'Exitoso' ,'Cancelado' )),
      primary key (transporteid,idlote)
   );

revoke all on "root".transporta from "public" as "root";

{ TABLE "root".link row size = 260 number of columns = 2 index size = 9 }

create table "root".link
   (
      link varchar(255) not null ,
      transportista integer,
      primary key (transportista)
   );

revoke all on "root".link from "public" as "root";

{ TABLE "root".evento row size = 4112 number of columns = 3 index size = 9 }

create table "root".evento
   (
      id serial not null ,
      datos "informix".bson not null ,
      fechaagregado datetime year to second,
      primary key (id)
   );

revoke all on "root".evento from "public" as "root";


grant select on "root".usuario to "administrativo" as "root";
grant update on "root".usuario to "administrativo" as "root";
grant insert on "root".usuario to "administrativo" as "root";
grant delete on "root".usuario to "administrativo" as "root";
grant index on "root".usuario to "administrativo" as "root";
```

```sql
grant alter on "root".usuario to "administrativo" as "root";
grant references on "root".usuario to "administrativo" as "root";
grant select on "root".usuario to "operario" as "root";
grant update on "root".usuario to "operario" as "root";
grant select on "root".usuario to "transportista" as "root";
grant update on "root".usuario to "transportista" as "root";
grant select on "root".cliente to "administrativo" as "root";
grant update on "root".cliente to "administrativo" as "root";
grant insert on "root".cliente to "administrativo" as "root";
grant delete on "root".cliente to "administrativo" as "root";
grant index on "root".cliente to "administrativo" as "root";
grant alter on "root".cliente to "administrativo" as "root";
grant references on "root".cliente to "administrativo" as "root";
grant select on "root".cliente to "operario" as "root";
grant select on "root".cliente to "transportista" as "root";
grant select on "root".lugar to "administrativo" as "root";
grant update on "root".lugar to "administrativo" as "root";
grant insert on "root".lugar to "administrativo" as "root";
grant delete on "root".lugar to "administrativo" as "root";
grant index on "root".lugar to "administrativo" as "root";
grant alter on "root".lugar to "administrativo" as "root";
grant references on "root".lugar to "administrativo" as "root";
grant select on "root".lugar to "operario" as "root";
grant select on "root".lugar to "transportista" as "root";
grant select on "root".incluye to "administrativo" as "root";
grant update on "root".incluye to "administrativo" as "root";
grant insert on "root".incluye to "administrativo" as "root";
grant delete on "root".incluye to "administrativo" as "root";
grant index on "root".incluye to "administrativo" as "root";
grant alter on "root".incluye to "administrativo" as "root";
grant references on "root".incluye to "administrativo" as "root";
grant select on "root".incluye to "operario" as "root";
grant select on "root".incluye to "transportista" as "root";
grant select on "root".pertenecea to "administrativo" as "root";
grant update on "root".pertenecea to "administrativo" as "root";
grant insert on "root".pertenecea to "administrativo" as "root";
grant delete on "root".pertenecea to "administrativo" as "root";
grant index on "root".pertenecea to "administrativo" as "root";
grant alter on "root".pertenecea to "administrativo" as "root";
grant references on "root".pertenecea to "administrativo" as "root";
grant select on "root".pertenecea to "operario" as "root";
grant select on "root".trabajaen to "administrativo" as "root";
grant update on "root".trabajaen to "administrativo" as "root";
grant insert on "root".trabajaen to "administrativo" as "root";
grant delete on "root".trabajaen to "administrativo" as "root";
grant index on "root".trabajaen to "administrativo" as "root";
grant alter on "root".trabajaen to "administrativo" as "root";
grant references on "root".trabajaen to "administrativo" as "root";
grant select on "root".trabajaen to "operario" as "root";
grant select on "root".conexion to "administrativo" as "root";
grant update on "root".conexion to "administrativo" as "root";
grant insert on "root".conexion to "administrativo" as "root";
grant delete on "root".conexion to "administrativo" as "root";
grant index on "root".conexion to "administrativo" as "root";
```

```sql
grant alter on "root".conexion to "administrativo" as "root";
grant references on "root".conexion to "administrativo" as "root";
grant select on "root".conexion to "operario" as "root";
grant update on "root".conexion to "operario" as "root";
grant insert on "root".conexion to "operario" as "root";
grant select on "root".conexion to "transportista" as "root";
grant update on "root".conexion to "transportista" as "root";
grant insert on "root".conexion to "transportista" as "root";
grant select on "root".vehiculo to "administrativo" as "root";
grant update on "root".vehiculo to "administrativo" as "root";
grant insert on "root".vehiculo to "administrativo" as "root";
grant delete on "root".vehiculo to "administrativo" as "root";
grant index on "root".vehiculo to "administrativo" as "root";
grant alter on "root".vehiculo to "administrativo" as "root";
grant references on "root".vehiculo to "administrativo" as "root";
grant select on "root".vehiculo to "operario" as "root";
grant update on "root".vehiculo to "operario" as "root";
grant select on "root".vehiculo to "transportista" as "root";
grant select on "root".vehiculoingresa to "administrativo" as "root";
grant update on "root".vehiculoingresa to "administrativo" as "root";
grant insert on "root".vehiculoingresa to "administrativo" as "root";
grant delete on "root".vehiculoingresa to "administrativo" as "root";
grant index on "root".vehiculoingresa to "administrativo" as "root";
grant alter on "root".vehiculoingresa to "administrativo" as "root";
grant references on "root".vehiculoingresa to "administrativo" as "root";
grant select on "root".vehiculoingresa to "operario" as "root";
grant update on "root".vehiculoingresa to "operario" as "root";
grant insert on "root".vehiculoingresa to "operario" as "root";
grant select on "root".informedanios to "administrativo" as "root";
grant update on "root".informedanios to "administrativo" as "root";
grant insert on "root".informedanios to "administrativo" as "root";
grant delete on "root".informedanios to "administrativo" as "root";
grant index on "root".informedanios to "administrativo" as "root";
grant alter on "root".informedanios to "administrativo" as "root";
grant references on "root".informedanios to "administrativo" as "root";
grant select on "root".informedanios to "operario" as "root";
grant update on "root".informedanios to "operario" as "root";
grant insert on "root".informedanios to "operario" as "root";
grant select on "root".registrodanios to "administrativo" as "root";
grant update on "root".registrodanios to "administrativo" as "root";
grant insert on "root".registrodanios to "administrativo" as "root";
grant delete on "root".registrodanios to "administrativo" as "root";
grant index on "root".registrodanios to "administrativo" as "root";
grant alter on "root".registrodanios to "administrativo" as "root";
grant references on "root".registrodanios to "administrativo" as "root";
grant select on "root".registrodanios to "operario" as "root";
grant update on "root".registrodanios to "operario" as "root";
grant insert on "root".registrodanios to "operario" as "root";
grant delete on "root".registrodanios to "operario" as "root";
grant select on "root".imagenregistro to "administrativo" as "root";
grant update on "root".imagenregistro to "administrativo" as "root";
grant insert on "root".imagenregistro to "administrativo" as "root";
grant delete on "root".imagenregistro to "administrativo" as "root";
grant index on "root".imagenregistro to "administrativo" as "root";
```

```sql
grant alter on "root".imagenregistro to "administrativo" as "root";
grant references on "root".imagenregistro to "administrativo" as "root";
grant select on "root".imagenregistro to "operario" as "root";
grant update on "root".imagenregistro to "operario" as "root";
grant insert on "root".imagenregistro to "operario" as "root";
grant delete on "root".imagenregistro to "operario" as "root";
grant select on "root".actualiza to "administrativo" as "root";
grant update on "root".actualiza to "administrativo" as "root";
grant insert on "root".actualiza to "administrativo" as "root";
grant delete on "root".actualiza to "administrativo" as "root";
grant index on "root".actualiza to "administrativo" as "root";
grant alter on "root".actualiza to "administrativo" as "root";
grant references on "root".actualiza to "administrativo" as "root";
grant select on "root".actualiza to "operario" as "root";
grant update on "root".actualiza to "operario" as "root";
grant insert on "root".actualiza to "operario" as "root";
grant delete on "root".actualiza to "operario" as "root";
grant select on "root".posicionado to "administrativo" as "root";
grant update on "root".posicionado to "administrativo" as "root";
grant insert on "root".posicionado to "administrativo" as "root";
grant delete on "root".posicionado to "administrativo" as "root";
grant index on "root".posicionado to "administrativo" as "root";
grant alter on "root".posicionado to "administrativo" as "root";
grant references on "root".posicionado to "administrativo" as "root";
grant select on "root".posicionado to "operario" as "root";
grant update on "root".posicionado to "operario" as "root";
grant insert on "root".posicionado to "operario" as "root";
grant select on "root".posicionado to "transportista" as "root";
grant update on "root".posicionado to "transportista" as "root";
grant insert on "root".posicionado to "transportista" as "root";
grant select on "root".tipotransporte to "administrativo" as "root";
grant update on "root".tipotransporte to "administrativo" as "root";
grant insert on "root".tipotransporte to "administrativo" as "root";
grant delete on "root".tipotransporte to "administrativo" as "root";
grant index on "root".tipotransporte to "administrativo" as "root";
grant alter on "root".tipotransporte to "administrativo" as "root";
grant references on "root".tipotransporte to "administrativo" as "root";
grant select on "root".tipotransporte to "operario" as "root";
grant select on "root".tipotransporte to "transportista" as "root";
grant select on "root".habilitado to "administrativo" as "root";
grant update on "root".habilitado to "administrativo" as "root";
grant insert on "root".habilitado to "administrativo" as "root";
grant delete on "root".habilitado to "administrativo" as "root";
grant index on "root".habilitado to "administrativo" as "root";
grant alter on "root".habilitado to "administrativo" as "root";
grant references on "root".habilitado to "administrativo" as "root";
grant select on "root".habilitado to "operario" as "root";
grant select on "root".habilitado to "transportista" as "root";
grant select on "root".mediotransporte to "administrativo" as "root";
grant update on "root".mediotransporte to "administrativo" as "root";
grant insert on "root".mediotransporte to "administrativo" as "root";
grant delete on "root".mediotransporte to "administrativo" as "root";
grant index on "root".mediotransporte to "administrativo" as "root";
grant alter on "root".mediotransporte to "administrativo" as "root";
```

```sql
grant references on "root".mediotransporte to "administrativo" as "root";
grant select on "root".mediotransporte to "operario" as "root";
grant select on "root".mediotransporte to "transportista" as "root";
grant select on "root".permite to "administrativo" as "root";
grant update on "root".permite to "administrativo" as "root";
grant insert on "root".permite to "administrativo" as "root";
grant delete on "root".permite to "administrativo" as "root";
grant index on "root".permite to "administrativo" as "root";
grant alter on "root".permite to "administrativo" as "root";
grant references on "root".permite to "administrativo" as "root";
grant select on "root".permite to "operario" as "root";
grant select on "root".permite to "transportista" as "root";
grant select on "root".lote to "administrativo" as "root";
grant update on "root".lote to "administrativo" as "root";
grant insert on "root".lote to "administrativo" as "root";
grant delete on "root".lote to "administrativo" as "root";
grant index on "root".lote to "administrativo" as "root";
grant alter on "root".lote to "administrativo" as "root";
grant references on "root".lote to "administrativo" as "root";
grant select on "root".lote to "operario" as "root";
grant update on "root".lote to "operario" as "root";
grant insert on "root".lote to "operario" as "root";
grant select on "root".lote to "transportista" as "root";
grant update on "root".lote to "transportista" as "root";
grant select on "root".integra to "administrativo" as "root";
grant update on "root".integra to "administrativo" as "root";
grant insert on "root".integra to "administrativo" as "root";
grant delete on "root".integra to "administrativo" as "root";
grant index on "root".integra to "administrativo" as "root";
grant alter on "root".integra to "administrativo" as "root";
grant references on "root".integra to "administrativo" as "root";
grant select on "root".integra to "operario" as "root";
grant update on "root".integra to "operario" as "root";
grant insert on "root".integra to "operario" as "root";
grant select on "root".integra to "transportista" as "root";
grant select on "root".transporte to "administrativo" as "root";
grant update on "root".transporte to "administrativo" as "root";
grant insert on "root".transporte to "administrativo" as "root";
grant delete on "root".transporte to "administrativo" as "root";
grant index on "root".transporte to "administrativo" as "root";
grant alter on "root".transporte to "administrativo" as "root";
grant references on "root".transporte to "administrativo" as "root";
grant select on "root".transporte to "operario" as "root";
grant select on "root".transporte to "transportista" as "root";
grant update on "root".transporte to "transportista" as "root";
grant insert on "root".transporte to "transportista" as "root";
grant select on "root".transporta to "administrativo" as "root";
grant update on "root".transporta to "administrativo" as "root";
grant insert on "root".transporta to "administrativo" as "root";
grant delete on "root".transporta to "administrativo" as "root";
grant index on "root".transporta to "administrativo" as "root";
grant alter on "root".transporta to "administrativo" as "root";
grant references on "root".transporta to "administrativo" as "root";
grant select on "root".transporta to "operario" as "root";
```

```sql
grant select on "root".transporta to "transportista" as "root";
grant update on "root".transporta to "transportista" as "root";
grant insert on "root".transporta to "transportista" as "root";
grant select on "root".link to "administrativo" as "root";
grant update on "root".link to "administrativo" as "root";
grant insert on "root".link to "administrativo" as "root";
grant delete on "root".link to "administrativo" as "root";
grant index on "root".link to "administrativo" as "root";
grant alter on "root".link to "administrativo" as "root";
grant references on "root".link to "administrativo" as "root";
grant select on "root".link to "operario" as "root";
grant select on "root".link to "transportista" as "root";
grant update on "root".link to "transportista" as "root";
grant insert on "root".link to "transportista" as "root";
grant select on "root".evento to "administrativo" as "root";
grant update on "root".evento to "administrativo" as "root";
grant insert on "root".evento to "administrativo" as "root";
grant delete on "root".evento to "administrativo" as "root";
grant index on "root".evento to "administrativo" as "root";
grant alter on "root".evento to "administrativo" as "root";
grant references on "root".evento to "administrativo" as "root";
grant select on "root".evento to "operario" as "root";
grant update on "root".evento to "operario" as "root";
grant insert on "root".evento to "operario" as "root";
grant select on "root".evento to "public" as "root";
grant update on "root".evento to "public" as "root";
grant insert on "root".evento to "public" as "root";
grant delete on "root".evento to "public" as "root";
grant index on "root".evento to "public" as "root";
grant select on "root".evento to "transportista" as "root";
grant update on "root".evento to "transportista" as "root";
grant insert on "root".evento to "transportista" as "root";


create function "root".crear_lugar(nombrel like lugar.nombre, pos_x like lugar.geox, pos_y
 like lugar.geoy, tipo like lugar.tipo, capacidad like lugar.capacidad, creador like lugar
.usuariocreador)
    returning integer
     DEFINE lugarid int;
     IF capacidad < 1 THEN
         return -1;
     END IF
     IF tipo not in ('Puerto', 'Patio', 'Establecimiento') THEN
         return -2;
     END IF
     insert into lugar
     (idlugar, nombre, geox, geoy, capacidad, usuariocreador, tipo, fecharegistro)
     values
     (0,       nombrel,pos_x,pos_y,capacidad, creador,        tipo, current);
     select dbinfo('sqlca.sqlerrd1') into lugarid from systables where tabid=1;
     return lugarid;
end function;

CREATE FUNCTION "root".maximo_ancestro(lugarid LIKE lugar.idlugar)
```

```
        returning integer
        define ttx int;
        IF lugarid NOT IN (SELECT menor FROM incluye) THEN
            RETURN lugarid;
        END if;
        select min(mayor) into ttx from incluye start with menor=lugarid connect by menor = pr
ior mayor;
        return ttx;
END FUNCTION;

create function "root".crear_subzona(nombrez like lugar.Nombre, enLugar like lugar.IDLugar
, capacidadz int)
    returning integer
    DEFINE existelugar, capacidadlugar, creador, lugarid int;
    DEFINE gx, gy float;
    SELECT count(*), capacidad, geox, geoy, usuariocreador
    into existelugar, capacidadlugar, gx, gy, creador
    from lugar where idlugar=enLugar and tipo='Zona'
    group by idlugar, capacidad, geox, geoy, usuariocreador;
    IF existelugar < 1 THEN
        return -1;
    END IF;
    IF capacidadlugar < capacidadz THEN
        return -1;
    END IF;
    IF gx is null or gy is null THEN
    END IF;
    insert into lugar(idlugar, nombre, capacidad, geox, geoy,
            usuariocreador, tipo,fechaRegistro)
        values(0, nombrez, capacidadz, gx, gy, creador, 'Subzona',current);
    select dbinfo('sqlca.sqlerrd1') into lugarid from systables where tabid=1;
    insert into incluye values(lugarid, enLugar);
    return lugarid;
end function;

create function "root".crear_zona(nombrez like lugar.Nombre, enLugar like lugar.IDLugar, c
apacidadz int)
    returning integer
    DEFINE existelugar, capacidadlugar, creador, lugarid int;
    DEFINE gx, gy float;
    SELECT count(*), capacidad, geox, geoy, usuariocreador
    into existelugar, capacidadlugar, gx, gy, creador
    from lugar where idlugar=enLugar and tipo in ('Puerto', 'Patio')
    group by idlugar, capacidad, geox, geoy, usuariocreador;
    IF existelugar < 1 THEN
        return -1;
    END IF;
    IF capacidadlugar < capacidadz THEN
        return -1;
    END IF;
    insert into lugar(idlugar, nombre, capacidad, geox, geoy,
            usuariocreador, tipo,fechaRegistro) values(0, nombrez, capacidadz, gx, gy, cre
ador, 'Zona', current);
    select dbinfo('sqlca.sqlerrd1') into lugarid from systables where tabid=1;
```

```
      insert into incluye values(lugarid, enLugar);
      return lugarid;
end function;

create function "root".zonas_en_lugar(lugarid like lugar.idlugar)
    returning integer, varchar(100), integer
     DEFINE idz integer;
     DEFINE nmz varchar(100);
     DEFINE cpz integer;
     FOREACH cursor1 FOR
       select lugar.idlugar, nombre, capacidad
       into idz, nmz, cpz
       from lugar inner join
       (select menor as idlugar from incluye
        start with mayor=lugarid
        connect by prior menor=mayor) as children on lugar.idlugar=children.idlugar
       where lugar.tipo="Zona" and inhabilitado='f'
       return idz, nmz, cpz WITH RESUME;
     END FOREACH;
end function;

create function "root".subzonas_en_zona(lugarid like lugar.idlugar, zonaid like lugar.idlu
gar)
    returning integer, varchar(100), integer
     DEFINE idz integer;
     DEFINE nmz varchar(100);
     DEFINE cpz integer;
     FOREACH cursor1 FOR
       select lugar.idlugar, nombre, capacidad
       into idz, nmz, cpz
       from lugar inner join
       (select menor as idlugar from incluye
        start with mayor=lugarid and menor=zonaid
        connect by prior menor=mayor) as children on lugar.idlugar=children.idlugar
       where lugar.tipo="Subzona" and inhabilitado='f'
       return idz, nmz, cpz WITH RESUME;
     END FOREACH;
end function;

create function "root".subzonas_en_lugar(lugarid like lugar.idlugar)
    returning integer, varchar(100), integer
     DEFINE idz integer;
     DEFINE nmz varchar(100);
     DEFINE cpz integer;
     FOREACH cursor1 FOR
       select lugar.idlugar, nombre, capacidad
       into idz, nmz, cpz
       from lugar inner join
       (select menor as idlugar from incluye
        start with mayor=lugarid
        connect by prior menor=mayor) as children on lugar.idlugar=children.idlugar
       where lugar.tipo="Subzona" and inhabilitado='f'
       return idz, nmz, cpz WITH RESUME;
     END FOREACH;
```

```
end function;

create function "root".ocupacion_en_lugar(lugarid like lugar.idlugar)
    returning integer
     DEFINE tipo varchar(15);
     DEFINE ocup integer;
     select lugar.tipo into tipo from lugar where idlugar=lugarid;
     IF tipo <> "Subzona" THEN
        select count(*)  into ocup
        from posicionado where hasta is null
        and idlugar in (select unnamed_col_1 from table(subzonas_en_lugar(lugarid)));
     ELSE
        select count(*) into ocup
        from posicionado where hasta is null and idlugar = lugarid;
     END IF;
     return ocup;
end function;

create function "root".subzonas_en_lugar_por_nombre(lugarnombre like lugar.nombre)
    returning integer, varchar(100), integer
     DEFINE idz integer;
     DEFINE nmz varchar(100);
     DEFINE cpz integer;
     DEFINE lugarid integer;
     select lugar.idlugar into lugarid from lugar where nombre=lugarnombre;
     FOREACH cursor1 FOR
       select lugar.idlugar, nombre, capacidad
       into idz, nmz, cpz
       from lugar inner join
       (select menor as idlugar from incluye
        start with mayor=lugarid
        connect by prior menor=mayor) as children on lugar.idlugar=children.idlugar
       where lugar.tipo="Subzona"
       return idz, nmz, cpz WITH RESUME;
     END FOREACH;
end function;

create function "root".cerrar_lote(loteid like lote.idlote)
    returning boolean
        DEFINE unverif boolean;
     select count(*) > 0
     into unverif
     from integra
     inner join vehiculo on integra.idvehiculo=vehiculo.idvehiculo and integra.lote=loteid
and integra.invalidado='f'
     inner join lote on integra.lote=lote.idlote
     left join informedanios on informedanios.idvehiculo=vehiculo.idvehiculo and informedan
ios.idlugar=lote.origen
     where informedanios.id is null;
     IF unverif THEN
     return 'f';
     ELSE
     update lote set estado="Cerrado" where idlote=loteid;
     return 't';
```

```
    END IF;
end function;


grant  execute on function "root".crear_lugar (char,char,char,char,char,char) to "public"
as "root";
grant  execute on function "root".maximo_ancestro (char) to "public" as "root";
grant  execute on function "root".crear_subzona (char,char,integer) to "public" as "root";
grant  execute on function "root".crear_zona (char,char,integer) to "public" as "root";
grant  execute on function "root".zonas_en_lugar (char) to "public" as "root";
grant  execute on function "root".subzonas_en_zona (char,char) to "public" as "root";
grant  execute on function "root".subzonas_en_lugar (char) to "public" as "root";
grant  execute on function "root".ocupacion_en_lugar (char) to "public" as "root";
grant  execute on function "root".subzonas_en_lugar_por_nombre (char) to "public" as "root
";
grant  execute on function "root".cerrar_lote (char) to "public" as "root";

revoke usage on language SPL from public ;

grant usage on language SPL to public ;




alter table "root".usuario add constraint (foreign key (creador)
    references "root".usuario );
alter table "root".cliente add constraint (foreign key (usuarioregistro)
    references "root".usuario );
alter table "root".lugar add constraint (foreign key (usuariocreador)
    references "root".usuario );
alter table "root".incluye add constraint (foreign key (menor)
    references "root".lugar );
alter table "root".incluye add constraint (foreign key (mayor)
    references "root".lugar );
alter table "root".pertenecea add constraint (foreign key (idlugar)
    references "root".lugar );
alter table "root".pertenecea add constraint (foreign key (clienteid)
    references "root".cliente );
alter table "root".trabajaen add constraint (foreign key (idlugar)
    references "root".lugar  on delete cascade);
alter table "root".trabajaen add constraint (foreign key (idusuario)
    references "root".usuario  on delete cascade);
alter table "root".conexion add constraint (foreign key (usuario)
    references "root".usuario  on delete cascade);
alter table "root".conexion add constraint (foreign key (idtrabajaen)
    references "root".trabajaen  on delete cascade);
alter table "root".vehiculo add constraint (foreign key (cliente)
    references "root".cliente  on delete cascade);
alter table "root".vehiculoingresa add constraint (foreign key
    (idvehiculo) references "root".vehiculo  on delete cascade);

alter table "root".vehiculoingresa add constraint (foreign key
    (usuario) references "root".usuario  on delete cascade);
alter table "root".informedanios add constraint (foreign key
```

```sql
    (idusuario) references "root".usuario  on delete cascade);

alter table "root".informedanios add constraint (foreign key
    (idvehiculo) references "root".vehiculo  on delete cascade);

alter table "root".informedanios add constraint (foreign key
    (idlugar) references "root".lugar  on delete cascade);
alter table "root".registrodanios add constraint (foreign key
    (informedanios,idvehiculo) references "root".informedanios
     on delete cascade);
alter table "root".imagenregistro add constraint (foreign key
    (vehiculo,informe,nrolista) references "root".registrodanios
     on delete cascade);
alter table "root".actualiza add constraint (foreign key (vehiculo1,
    informe1,registro1) references "root".registrodanios  on delete
    cascade);
alter table "root".actualiza add constraint (foreign key (vehiculo2,
    informe2,registro2) references "root".registrodanios  on delete
    cascade);
alter table "root".posicionado add constraint (foreign key (idvehiculo)
    references "root".vehiculo  on delete cascade);
alter table "root".posicionado add constraint (foreign key (idlugar)
    references "root".lugar  on delete cascade);
alter table "root".posicionado add constraint (foreign key (idusuario)
    references "root".usuario  on delete cascade);
alter table "root".habilitado add constraint (foreign key (idlugar)
    references "root".lugar );
alter table "root".habilitado add constraint (foreign key (idtipo)
    references "root".tipotransporte );
alter table "root".mediotransporte add constraint (foreign key
    (tipo) references "root".tipotransporte (nombre) on delete
    cascade);
alter table "root".mediotransporte add constraint (foreign key
    (idtipo) references "root".tipotransporte  on delete cascade);

alter table "root".mediotransporte add constraint (foreign key
    (creador) references "root".usuario  on delete cascade);
alter table "root".permite add constraint (foreign key (idtipo,
    idlegal) references "root".mediotransporte  on delete cascade);

alter table "root".permite add constraint (foreign key (usuario)
    references "root".usuario  on delete cascade);
alter table "root".lote add constraint (foreign key (origen)
    references "root".lugar  on delete cascade);
alter table "root".lote add constraint (foreign key (destino)
    references "root".lugar  on delete cascade);
alter table "root".lote add constraint (foreign key (creadorid)
    references "root".usuario  on delete cascade);
alter table "root".integra add constraint (foreign key (idvehiculo)
    references "root".vehiculo  on delete cascade);
alter table "root".integra add constraint (foreign key (lote)
    references "root".lote  on delete cascade);
alter table "root".integra add constraint (foreign key (idusuario)
    references "root".usuario  on delete cascade);
```

```sql
alter table "root".transporte add constraint (foreign key (idtipo,
    idlegal,usuario) references "root".permite  on delete cascade);

alter table "root".transporta add constraint (foreign key (transporteid)
    references "root".transporte  on delete cascade);
alter table "root".transporta add constraint (foreign key (idlote)
    references "root".lote  on delete cascade);
alter table "root".link add constraint (foreign key (transportista)
    references "root".usuario  on delete cascade);
```