

13 abr 2023

Usuarios, permisos, esquemas

```
postgres=> select user;
      user
      -----
\l
```

\du -> describe los usuarios

	Role name	Attributes	Member
of			
\x1B\x1B[3~dúkjvkjvévb			{}
julian	Superuser		{}
no			{}
oracle			{}
postgres	Superuser, Create role, Create DB, Replication, Bypass RLS	{}	
readonly	Cannot login		{}
ua			{}
ub			{readonly}
nly}			
user3			{}
user4			{}
user5			{}
(END)			

se crea cuando se instala ese usu

readonly - no es un usuario. es un rol(usuarios que no tienen login)

\l

lista las bases de datos creadas

List of databases						
Name	Owner	Encoding	Collate	Ctype	Access privileges	Tablespace
asirl	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	=Tc/postgres + postgres=CTc/postgres+ oracle=c/postgres	
postgres	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	postgres=CTc/postgres+ oracle=CTc/postgres + =c/postgres + user4=c/postgres	
proba	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	=T/postgres + postgres=CTc/postgres	
proba3	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	=c/postgres + postgres=CTc/postgres	
template0	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	postgres=CTc/postgres	
template1	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	=c/postgres + postgres=CTc/postgres	
(6 rows)						
postgres	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	oracle=c/postgres postgres=CTc/postgres+ oracle=CTc/postgres + =c/postgres + user4=c/postgres	

que significa esto

```
oracle=c/postgres
postgres=CTc/postgres+
oracle=CTc/postgres +
=c/postgres +
user4=c/postgres
```

que postgres le dio al usuario oracle la posibilidad de (por la c minus) conectarse

lo que hay a la derecha es quien concede el privilegio
 izq usu que recibe el privilegio
 en el medio son los privilegios
 a que basen van los privilegios??

postgres	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	
----------	----------	------	-------------	-------------	--

a esa

cd no hay nada y este la 'c' es que cualquier usuario se puede conectar a las siguientes bases:

- | =Tc/postgres +
 • | postgres=CTc/postgres+
- | =c/postgres +

en ese caso no se puede : (no hay 'c')

=T/postgres	+
-------------	---

Crear base de datos

tengo que ser admin
 (de momento no soy super)

julian	Superuser
no	
oracle	
postgres	Superuser, Create role
readonly	Cannot login

```
postgres=> \du
postgres=> create database dam1;
ERROR: permission denied to create database
postgres=>
```

oracle no puede crear base de datos pero el usu postgres si. entonces me conecto a ese usu:

```
postgres=> \c postgres postgres
Password for user postgres:
psql (14.7 (Ubuntu 14.7-0ubuntu0.22.04.1), server 12.12 (Ubuntu 12.12-0ubuntu0.20.04.1))
You are now connected to database "postgres" as user "postgres".
postgres=#
```

c= conectar

primer postgres=base de datos

segundo postgres=postgres

```
postgres=# create database dam1;
CREATE DATABASE
postgres=#
```

|| para ver que esta creada

List of databases						
Name	Owner	Encoding	Collate	Ctype	Access privileges	
asir1	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	=Tc/postgres + postgres=CTc/postgres+ oracle=c/postgres	
dam1	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	postgres=CTc/postgres+ oracle=CTc/postgres + =c/postgres + user4=c/postgres	
postgres	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	postgres=CTc/postgres oracle=CTc/postgres + =c/postgres + user4=c/postgres	
proba	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	=T/postgres + postgres=CTc/postgres	
proba3	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8		
template0	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	=c/postgres + postgres=CTc/postgres	
template1	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	=c/postgres + postgres=CTc/postgres	
(7 rows)						

no pone nada pq no esta actualizada de momento(es un error de la app)

dam1	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	oracle=c/postgres
postgres	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	postgres=CTc/postgres+

me conecto a la bd 'dam1' creada con el usu 'oracle'

```
postgres=# \c dam1 oracle;
Password for user oracle:
psql (14.7 (Ubuntu 14.7-0ubuntu0.22.04.1), server 12.12 (Ubuntu 12.12-0ubuntu0.20.04.1))
You are now connected to database "dam1" as user "oracle".
dam1=>
```

esto no deberia dejar de hacerlo ya que no se podria conectar ya que no tiene los permisos correspondientes. Entonces, voy a quitarle a oracle la posibilidad de conectarse a cualquier base de datos ya que por defecto 'public' tiene el permiso de conectarse a cualquier base de datos y por tanto incluye a 'oracle'. oracle realmente no tiene ningun permiso pero public si.

public es un usuario generico que no se puede modificar. cualquier permiso que se le de a public se la da a todos los usuario, por ello, es normal que al empezar conviene quitarle todos los permisos a public

para quitar el permiso hay que quitarselo a 'public'

```
dam1=>
dam1=> \c postgres postgres;
Password for user postgres:
psql (14.7 (Ubuntu 14.7-0ubuntu0.22.04.1), server 12.12 (Ubuntu 12.12-0ubuntu0.20.04.1))
You are now connected to database "postgres" as user "postgres".
postgres=#
```

```

Password for user postgres:
psql (14.7 (Ubuntu 14.7-0ubuntu0.22.04.1), server 12.12 (Ubuntu 12.12-0ubuntu0.20.04.1))
You are now connected to database "postgres" as user "postgres".
REVOKE
:: postgres=#

```

Actividades Terminal Kav 13 de Abr. 13:55 oracle@oracle-VirtualBox:~

Name	Owner	Encoding	Collate	Ctype	Access privileges
asirl	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	=Tc/postgres + postgres=CTc/postgres+ oracle=c/postgres
dam1	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	=T/postgres + postgres=CTc/postgres
postgres	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	postgres=CTc/postgres+ oracle=CTc/postgres + =c/postgres + user4=c/postgres
proba	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	=T/postgres + postgres=CTc/postgres
proba3	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	=c/postgres + postgres=CTc/postgres
template0	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	=c/postgres + postgres=CTc/postgres
template1	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	=c/postgres + postgres=CTc/postgres
(7 rows)					

```

:: postgres=#

```

Ahora quiero conceder permiso SOLO a oracle

```

postgres=# grant connect on database dam1 to oracle;
: GRANT
: postgres=#

```

grant-garantiza

lo comprobamos:

Name	Owner	Encoding	Collate	Ctype	Access privileges
asirl	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	=Tc/postgres + postgres=CTc/postgres+ oracle=c/postgres
dam1	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	=T/postgres + postgres=CTc/postgres+ oracle=c/postgres
postgres	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	postgres=CTc/postgres+ oracle=CTc/postgres + =c/postgres + user4=c/postgres
proba	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	=T/postgres + postgres=CTc/postgres
proba3	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	=c/postgres + postgres=CTc/postgres
template0	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	=c/postgres + postgres=CTc/postgres
template1	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	=c/postgres + postgres=CTc/postgres
(7 rows)					

(esto antes no aparecía)

esta linea siempre se crea:

```

| postgres@pc:postgres]
| oracle=c/postgres
-8 | postgres=CTc/postgres+
| oracle=CTc/postgres+
| =c/postgres+
| user4=c/postgres

```

(ya que es el usuario el que crea ese objeto??????)

un schema es un espacio dentro de una base d datos que tiene un nombre (es como un apartado). el propietario es el usu postgres. por tanto todo lo que cree postgres va a estar aqui.

```

dam1=> \dn
List of schemas
 Name   | Owner
-----+-----
 public | postgres
(1 row)

dam1=>

```

solo hay un esquema llamado public

para ver todos las tablas:

```

dam1=> \d
Did not find any relations.

dam1=>

```

cambiamos de usuario para ver tablas creadas por oracle

```

Did not find any relations.

dam1=> \c postgres oracle
psql (14.7 (Ubuntu 14.7-0ubuntu0.22.04.1), server 12.12 (Ubuntu 12.12-0ubuntu0.20.04.1))
: You are now connected to database "postgres" as user "oracle".
: postgres=>

```

aqui hay varios esquemas pq los habia creado ricardo anteriormente. nosotros durante todo el curso hemos trabajado en el esquema 'public':

(el esquemas public siempre viene por defecto)

```

postgres=> \dn
List of schemas
 Name   | Owner
-----+-----
 esquema3 | user3
 esquema4 | user4
 public   | postgres
(3 rows)

postgres=>

```

(postgrtess es el propietario de public pero oracle pudo crear tablas dentro de public)

\d se me muestran todas las tablas

List of relations				
Schema	Name	Type	Owner	
public	a	table	oracle	
public	alc	table	oracle	
public	a2c	table	oracle	
public	acciones	table	oracle	
public	achados	table	oracle	
public	actores	table	oracle	
public	adestra	table	oracle	
public	adscrito	table	oracle	
public	agudos	table	oracle	
public	alozan	table	oracle	
public	apoian	table	oracle	
public	area	table	oracle	
public	asegurado	table	oracle	
public	asignan	table	oracle	
public	augas	table	oracle	
public	autor	table	oracle	
public	balnearios	table	oracle	
:				

Crear esquema dentro de dam1

listar esquemas

```
postgres=> \c dam1 postgres;
Password for user postgres:
psql (14.7 (Ubuntu 14.7-0ubuntu0.22.04.1), server 12.12 (Ubuntu 12.12-0ubuntu0.20.04.1))
You are now connected to database "dam1" as user "postgres".
dam1=# \dn
List of schemas
 Name | Owner
-----+-----
 public | postgres
(1 row)

dam1=#
```

crear un esquema

```
dam1=# create schema esquema3;
CREATE SCHEMA
dam1=# \dn
List of schemas
 Name | Owner
-----+-----
 esquema3 | postgres
 public | postgres
(2 rows)

dam1=#
```

```
dam1=# create user user33 with password 'user33';
CREATE ROLE
dam1=#
```

(creo usu nuevo)

```
CREATE ROLE
dam1=# create schema esquema33;
CREATE SCHEMA
dam1=# \dn
List of schemas
 Name   | Owner
-----+-
 esquema3 | postgres
 esquema33 | postgres
 public   | postgres
(3 rows)
dam1=#
```

quiero poner como owner al usu creado ‘usu33’

```
dam1=# ALTER SCHEMA esquema33 OWNER TO user33;
ALTER SCHEMA
dam1=# \dn
List of schemas
 Name   | Owner
-----+-
 esquema3 | postgres
 esquema33 | user33
 public   | postgres
(3 rows)
```

(los permisos para conceder permisos de esquemas solo los puede hacer postgres)

entonces es independiente el usuario en el que estas para crear un esquema ya que se crea x defecto el propietario es postgres????

- no, no cualquier usuario, tienes que tener un permiso para crear esquemas → ‘C’

asirl	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	=Tc/postgres + postgres=CTc/postgres+ oracle=c/postgres
dam1	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	=T/postgres + postgres=CTc/postgres+ oracle=c/postgres
postgres	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	postgres=CTc/postgres+ oracle=CTc/postgres + =c/postgres + user4=c/postgres
proba	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	=T/postgres + postgres=CTc/postgres
proba3	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	=c/postgres + postgres=CTc/postgres
template0	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	=c/postgres + postgres=CTc/postgres
template1	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	=c/postgres + postgres=CTc/postgres

oracle puede crear esquemas en la base de datos de postgres

**'T' es una tabla temporal. se borran automaticamente cuando cierra sesion.
normalmente se usan para hacer calculos intermedios**

Dar permisos de una tabla a un usuario

```
Password for user user3:  
connection to server on socket "/var/run/postgresql/.s.PGSQL.5432" failed: FATAL: password authentication failed for user "user3"  
Previous connection kept  
dam1=#  
dam1=#  
dam1=#  
dam1=#  
dam1=# \c dam1 user33;  
Password for user user33:  
connection to server on socket "/var/run/postgresql/.s.PGSQL.5432" failed: FATAL: permission denied for database "dam1"  
DETAIL: User does not have CONNECT privilege.  
Previous connection kept  
dam1=# select user;  
 user  
-----  
 postgres  
(1 row)
```

```
dam1=# grant connect on database dam1 to user33;  
GRANT  
dam1=# \c dam1 user33;  
Password for user user33:  
psql (14.7 (Ubuntu 14.7-0ubuntu0.22.04.1), server 12.12 (Ubuntu 12.12-0ubuntu0.20.04.1))  
You are now connected to database "dam1" as user "user33".  
dam1=> ■ I
```

Crear tablas en distintos esquemas

Se crean las tablas x defecto (indep del usuario) en el esquema 'public':

```

dam1=> create table probal (codigo integer);
CREATE TABLE
dam1=> \d
      List of relations
 Schema |  Name   | Type  | Owner
-----+-----+-----+
 public | probal | table | user33
(1 row)

:: dam1=>

```

hay que especificar en que esquema:

```

dam1=> create table esquema33.probal (codigo integer);
CREATE TABLE
dam1=> \d
      List of relations
 Schema |  Name   | Type  | Owner
-----+-----+-----+
 public | probal | table | user33
(1 row)

dam1=> \d esquema3.*
Did not find any relation named "esquema3.*".
dam1=> \d esquema33.*
      Table "esquema33.probal"
 Column |  Type   | Collation | Nullable | Default
-----+-----+-----+-----+
 codigo | integer |           |         |
:: dam1=>

```

tengo que decirle que me muestre las tablas de ese esquema al que le hemos asignado la tabla:

```

dam1=> \d esquema33.*
      Table "esquema33.probal"

```

al poner \d me muestra las tablas de 'public' por defecto

si quiero eliminar tablas

```

dam1=> drop table probal;
DROP TABLE
dam1=> \d esquema33.*
      Table "esquema33.probal"
 Column |  Type   | Collation | Nullable | Default
-----+-----+-----+-----+
 codigo | integer |           |         |
dam1=> \d
Did not find any relations.
:: dam1=>

```

(al no especificar el esquema borra la de 'public' que es la de por defecto

Si quiero que de forma predeterminada me haga las que tablas creadas sean de un esquema en concreto

tengo que cambiararme al usu postgres

```
did not find any relations.
dam1=> \c dam1 postgres;
Password for user postgres:
psql (14.7 (Ubuntu 14.7-0ubuntu0.22.04.1), server 12.12 (Ubuntu 12.12-0ubuntu0.20.04.1))
You are now connected to database "dam1" as user "postgres".
dam1=# alter user user33 in database dam1 set search_path to esquema33;
::: ALTER ROLE
dam1=#

```

compruebo:

\dn muestra los esquemas de la base de datos en la que me encuentro

```
dam1=# \dn
List of schemas
 Name   | Owner
-----+-
 esquema3 | postgres
 esquema33 | user33
 public   | postgres
(3 rows)
```

```
dam1=# show search_path
dam1-# ;
      search_path
-----
 "$user", public
(1 row)

dam1=#

```

```
dam1=# \c dam1 user33;
Password for user user33:
psql (14.7 (Ubuntu 14.7-0ubuntu0.22.04.1), server 12.12 (Ubuntu 12.12-0ubuntu0.20.04.1))
You are now connected to database "dam1" as user "user33".
dam1=> show search_path;
      search_path
-----
 esquema33
(1 row)

dam1=>
```

ahora, cuando cree una bd va a crearlo por defecto en el esquema33

compruebo:

```

dam1=> create table proba2 (codigo integer);
CREATE TABLE
dam1=> \d
      List of relations
 Schema | Name   | Type  | Owner
-----+-----+-----+
 esquema33 | proba1 | table | user33
 esquema33 | proba2 | table | user33
(2 rows)

dam1=>

```

14 abr 2023

quitar el permiso de conectarse a public:

```

dam1=# revoke connect on database dam1 from public ;
: REVOKE
: dam1=#

```

24 abr 2023

borramos una base de datos

- \drop database dam1;

comprobamos con \l:

	Name	Owner	Encoding	Collate	Ctype	Access privileges
	asir1	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	=T/postgres + postgres=CTc/postgres+ oracle=c/postgres + user3=c/postgres + user4=c/postgres
	postgres	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	postgres=CTc/postgres+ oracle=CTc/postgres + =c/postgres + user4=c/postgres
	proba	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	=T/postgres + postgres=CTc/postgres
	proba3	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	
	template0	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	=c/postgres + postgres=CTc/postgres
	template1	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	=c/postgres + postgres=CTc/postgres
	(6 rows)					

crear usuario: create user angel with password '1234';

postgres:

- “C” crear tablas en este caso pq en “List of schemas” el significado cambia con respecto a cuando usamos \l para ver “List of databases” cuyo significado es permisos para crear schemas

```
dam1=> \dn+
                                         List of schemas
   Name | Owner | Access privileges |      Description
-----+-----+-----+-----+
 e1    | udam1 | postgres=UC/postgres+| standard public schema
 public | postgres | =UC/postgres           |
(2 rows)

dam1=>
```

para que me cree por la tabla en el esquema que yo quiero, en este caso: e1:

```
dam1=> create table e1.tudamlel(codigo integer);
CREATE TABLE
dam1=> \d
                                         List of relations
 Schema |   Name   | Type  | Owner
-----+-----+-----+-----+
 public | proba   | table | user3
 public | tudamlel | table | udam1
(2 rows)
```

si no pongo "e1." delante de "tudamlel" se crear en el esquema por defecto que sería public.

muestro las tablas de un esquema

Si no especificamos el esquema, me muestra las tablas que tiene el esquema por defecto(el esquema: public). esto pasa lo mismo con \d y \z:

\d te desplegaría las tablas todas con sus datos

```
dam1=> \d e1.*
                                         Table "e1.tudamlel"
 Column |  Type   | Collation | Nullable | Default
-----+-----+-----+-----+-----+
 codigo | integer |           |          |
```



```
dam1=> \z e1.*
                                         Access privileges
 Schema |   Name   | Type  | Access privileges | Column privileges | Policies
-----+-----+-----+-----+-----+-----+
 e1    | tudamlel | table |               |                 |
```

\z muestra las tablas con los permisos en un formato mas reducido

27 abr 2023

crear esquema y poner como propietario en un solo comando (en vez de crearlo y luego hacer el alter)

```
dam1=# create schema e2 authorization udam2;
CREATE SCHEMA
dam1=# \dn
List of schemas
 Name | Owner
-----+-----
e1   | udam1
e2   | udam2
public | postgres
(3 rows)
```

si quiero crear tabla en e2 asi

- incorrecta: esta estaria mal ya que se crearia por defecto en public"
- correcta:

```
dam1=> create table e2.tudam2e2(codigo integer, nome varchar(20));
CREATE TABLE
dam1=> \d e2./*
              Table "e2.tudam2e2"
 Column |          Type          | Collation | Nullable | Default
-----+-----+-----+-----+-----+
 codigo | integer          |           |         |
 nome   | character varying(20) |           |         |
```

ahora borro la tabla que hice en public porque no es la que queria hacer:

```
dam1=> drop table tudam2e2;
DROP TABLE
dam1=> \dt public./*
Did not find any relation named "public.*".
dam1=> \d e2./*
              Table "e2.tudam2e2"
 Column |          Type          | Collation | Nullable | Default
-----+-----+-----+-----+-----+
 codigo | integer          |           |         |
 nome   | character varying(20) |           |         |
```

aqui no hace falta especificar la "ruta" ya que por defecto va a ir a la de public, que es la que queremos borrar.

****preguntarle a Nico una cosa nueva que va a hacer que es darle a permiso a no se que, lo hizo despues de las 14.30, que es cd me pire, y lo explico creo que sdespues de hacer el insert en la tabla**

28 abr 2023

para añadir una nueva columna a una tabla que ya existía

```
You are now connected to database "dam1" as user "udam1".
dam1=> alter table e1.tudamlel add column nome varchar(20);
ALTER TABLE
dam1=> select * from e1.tudamlel;
 codigo | nome
-----+-----
 2 | 
 3 |
(2 rows)
```

cambiar un dato o campos de filas que ya existen:

```
dam1=> update e1.tudamlel set nome='pedro' where codigo=2;
UPDATE 1
dam1=> select * from e1.tudamlel;
 codigo | nome
-----+-----
 3 |
 2 | pedro
(2 rows)

dam1=>
```

dar permisos a un usuario SOLO para ver datos de una columna especifica

```
dam1=> grant select(nome) on e1.tudamlel to udam2;
GRANT
```

(entre parentesis le digo cuales son las columnas que quiero que udam2 pueda visualizar, el resto no las veria)

Schema	Name	Type	Access privileges	Column privileges	Policies
e1	tudamlel	table	udam1=arwdDxt/udam1+	nome: udam2=r/udam1	+ udam2=r/udam1

- comprobamos:

me muevo a udam2 y hago el select a la tabla

```

dam1=> \c dam1 udam2
;
Password for user udam2:
psql (14.7 (Ubuntu 14.7-0ubuntu0.22.04.1), server 12.12 (Ubuntu 12.12-0ubuntu0.20.04.1))
You are now connected to database "dam1" as user "udam2".
dam1=> select * from e1.tudamle1;
ERROR: permission denied for table tudamle1
dam1=> select nome from e1.tudamle1;
 nome
-----
 pedro
 ana
(2 rows)

```

con el "*" no me deja porque no tiene permiso para ver toda la tabla pero con solo "nome" sí que me deja ya que sí que tengo permisos para ver esa columna.

Se puede dar permisos para insertar valores pero SOLO dentro de una columna? SI

grant insert(nome) on e1.tudamle1 to udam2

```

You are now connected to database "dam1" as user "udam2".
dam1=> \z e1.*
                                         Access privileges
Schema |   Name    | Type | Access privileges | Column privileges | Policies
-----+-----+-----+-----+-----+
e1     | tudamle1 | table | udam1=arwdDxt/udam1 | nome:             +|
                   |           |       |                   | udam2=ar/udam1  |
(1 row)

Column privileges |
-----+
 nome:             +
 udam2=ar/udam1  |

```

(en este caso "nome" es la columna en la que tendré permiso para insertar)

- Comprobamos

```

dam1=> insert into e1.tudamle1 values( 1,'juan');
ERROR: permission denied for table tudamle1
dam1=> insert into e1.tudamle1 values('juan');
ERROR: invalid input syntax for type integer: "juan"
LINE 1: insert into e1.tudamle1 values('juan');

dam1=> insert into e1.tudamle1 (nome) values('juan');
INSERT 0 1
dam1=>

```

En el primer ERROR: porque estoy intentando insertar en dos columnas distintos valores y solo tengo permiso para insertar valores en una.

En el segundo ERROR: le tengo que especificar la columna en la que tengo permiso para insertar, tengo que ser explícito.

miramos la tabla con el nuevo dato incorporado

```
dam1=> select * from el.tudamlel;
 codigo | nome
-----+-----
 2 | pedro
 3 | ana
 1 | juan
(3 rows)
```

al no poder modificar el otro dato “codigo”, me lo deja en nulo

Quiero añadir / modificar una nueva clave primaria pero uno de los campos de la columna es nulo

```
dam1=> select * from el.tudamlel;
 codigo | nome
-----+-----
 2 | pedro
 3 | ana
 1 | juan
(3 rows)
```

me da error porque ninguno de los campos puede ser nulo para ser clave primaria:

```
dam1=> alter table el.tudamlel add primary key (codigo);
ERROR: column "codigo" contains null values
dam1=>
```

actualizo el campo que es nulo:

```
ERROR: column "codigo" contains null values
dam1=> update el.tudamlel set codigo=1 where nome='juan';
UPDATE 1
dam1=> select * from el.tudamlel;
 codigo | nome
-----+-----
 2 | pedro
 3 | ana
 1 | juan
(3 rows)
```

- comprobamos:

Intento volver a modificar la clave primaria: ya me dejaría!

```
dam1=> alter table el.tudamlel add primary key (codigo);
ALTER TABLE
```

comprobación 2. intentamos insertar nuevos valores en la tabla poniendo el campo “codigo”, que ahora es primario, como nulo.

```

dam1=> insert into e1.tudamle1 values (null,'windar');
ERROR: null value in column "codigo" violates not-null constraint
DETAIL: Failing row contains (null, windar).
dam1=> \d e1.tudamle1
              Table "e1.tudamle1"
 Column |          Type          | Collation | Nullable | Default
-----+-----+-----+-----+-----+
 codigo | integer           |           | not null |
 nome   | character varying(20) |           |           |
Indexes:
    "tudamle1_pkey" PRIMARY KEY, btree (codigo)

```

efectivamente, me da error ya que no puedo introducir ningun nulo como clave primaria.

4 may 2023

preguntar para que sirve???

```

baseX=> select * from information_schema.table_privileges where grantee='u2';
baseX=>

```

```

baseX=> select grantor , grantee , table_schema ,table_name , privilege_type,
is_grantable from information_schema.table_privileges where grantee='u2';

grantor | grantee | table_schema | table_name | privilege_type | is_grantable
-----+-----+-----+-----+-----+-----+
u1     | u2      | e1        | basex_e1_u1_t1 | INSERT       | YES
u1     | u2      | e1        | basex_e1_u1_t1 | SELECT       | NO
u1     | u2      | e1        | basex_e1_u1_t1 | UPDATE       | YES

```

serias capaz sabiendo que estoy conectado por postgres y la tabla es de u1 que grant hay que dar para que aparezca esto asi?

```

GRANT
baseX=> \z e1.*
               Access privileges
 Schema |      Name      | Type | Access privileges | Column privileges | Policies
-----+-----+-----+-----+-----+-----+
 e1    | basex_e1_u1_t1 | table | u1=arwdDxt/u1    +| nome:             +|
      |                  |       | u2=a*rw*/u1      +| u3=r/u1          |
      |                  |       | u3=a/u2          +|
      |                  |       | u3=r/u1          |
(1 row)

```

```

| Column privileges
+-----+
+| nome:
+| u3=r/u1
+|

```

```

baseX=# grant select(nome) on e1.baseX_e1_u1_t1 to u3;
GRANT
baseX=# \z e1.*
          Access privileges
Schema |      Name      | Type | Access privileges | Column privileges | Policies
-----+-----+-----+-----+-----+-----+
e1    | baseX_e1_u1_t1 | table | u1=arwdDxt/u1    +| nome:           +|
      |                  |       | u2=a*rw*/u1     +| u3=r/u1        |
      |                  |       | u3=a/u2         +|
      |                  |       | u3=r/u1         |
(1 row)

```

si quito permisos a la tabla, te quitaria tambien los permisos a los campos pero si solo quitas permisos a los campos no se quitan los permisos de las tablas

privilegios de columnas

```

baseX=# select grantor , grantee , table_schema ,table_name ,column_name, privilege_type, is_grantable
  from information_schema.column_privileges where grantee ='u3';
baseX=# select grantor , grantee , table_schema ,table_name ,column_name, privilege_type
  from information_schema.column_privileges where grantee ='u3';
          grantor | grantee | table_schema | table_name | column_name | privilege_type
-----+-----+-----+-----+-----+
u2   | u3    | e1      | baseX_e1_u1_t1 | codigo      | INSERT
u1   | u3    | e1      | baseX_e1_u1_t1 | nome        | SELECT
u2   | u3    | e1      | baseX_e1_u1_t1 | nome        | INSERT
(3 rows)

baseX=# \z e1.*
          Access privileges
Schema |      Name      | Type | Access privileges | Column privileges | Policies
-----+-----+-----+-----+-----+
e1    | baseX_e1_u1_t1 | table | u1=arwdDxt/u1    +| nome:           +|
      |                  |       | u2=a*rw*/u1     +| u3=r/u1        |
      |                  |       | u3=a/u2         +|
      |                  |       | u3=r/u1         |
(1 row)
baseX=#

```

Explicacion roles

List of databases						
Name	Owner	Encoding	Collate	Ctype	Access privileges	
asir1	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	=T/postgres	+
					postgres=CTc/postgres	+
					oracle=c/postgres	+
					user3=c/postgres	+
					user4=c/postgres	
base_nova	oracle	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	=T/postgres	+
baseX	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	postgres=CTc/postgres	
basey	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	postgres=CTc/postgres	+
					readonly=c/postgres	+
dam1	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	ua=c/postgres	
					=T/postgres	+
					postgres=CTc/postgres	+
					user3=c/postgres	+
					udam1=c/postgres	+
					udam2=c/postgres	
postgres	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	postgres=CTc/postgres	+

impedimos crear esquemas en el esquema public a todos. nadie podria crear ningun solo esquema (excepto que tuvieran concedidos privilegios individuales para crear esquemas). solo los podria crear el administrador.

```
: basex=# REVOKE CREATE ON SCHEMA public FROM PUBLIC; | I
```

nos movemos a postgres pq es el unico que puede hacerlos

Crear un rol “redonli”

```
base=# create role readonli;
CREATE ROLE
```

no se puede logear, es la unica diferencia entre rol y usuario:

Role name	Attributes	Member
of		{}
\x1B\x1B[3~dükjvkljvéeb		{}
julian	Superuser	{}
no		{}
oracle		{}
pedro		{}
postgres	Superuser, Create role, Create DB, Replication, Bypass RLS	{}
readonli	Cannot login	{}
readonly	Cannot login	{}
u1		{}
u2		{}
u3		{}
ua		{}
ub		{}
ily}		{}
udam1		{}

```
base=# \du+
base=# create schema sa authorization ua;
```

List of schemas	
Name	Owner
e1	u1
e2	u2
e3	u3
public	postgres
sa	ua
(5 rows)	

```
base#
```

```
base=# create schema sb authorization ub;
CREATE SCHEMA
base=# \dn
```

```
baseX=# CREATE SCHEMA sb
CREATE SCHEMA
baseX=# \dn
List of schemas
 Name | Owner
-----+
 e1   | u1
 e2   | u2
 e3   | u3
 public | postgres
 sa    | ua
 sb    | ub
(6 rows)

baseX=#

```

```
baseX=# grant connect on database baseX to ua;
GRANT
baseX=#

```

List of databases						
	Name	Owner	Encoding	Collate	Ctype	Access privileges
	asirl	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	=T/postgres + postgres=CTc/postgres+ oracle=c/postgres + user3=c/postgres + user4=c/postgres
	base_nova	oracle	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	=T/postgres + postgres=CTc/postgres+ user3=c/postgres +
	baseX	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	ua=c/postgres I postgres=CTc/postgres+ readonly=c/postgres +
	basey	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	postgres=CTc/postgres+ readonly=c/postgres + ua=c/postgres
	dam1	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	=T/postgres + postgres=CTc/postgres+ user3=c/postgres + udam1=c/postgres +

8 may 2023

dar uso / permiso de esquema a un rol

(lo puede dar el propietario o el creador)

nos conectamos al usu "ua", creador del esquema, y asignamos permiso:

```
You are now connected to database "baseX" as user "
baseX=> grant usage on schema sa to readonly;
GRANT
baseX=>
```

porque no me deja en este caso? pq esto lo intenté hacer con el usu "ua" y este no es el creador del esquema "readonly"

```
baseX=> grant readonly to ub;
```

- cambio a postgres y si que me deja

```
baseX=> \c baseX postgres;
Password for user postgres:
psql (14.7 (Ubuntu 14.7-0ubuntu0.22.04.1), server 12.12 (Ubuntu 12.12-0ubuntu0.20.04.1))
You are now connected to database "baseX" as user "postgres".
baseX=# grant readonly to ub;
NOTICE:  role "ub" is already a member of role "readonly"
GRANT ROLE
```

Todos los usuarios que tengan el rol tienen los permisos asignados a este rol.

?????????

no hay una vista específica para ver propiedades de un rol (por lo menos no directamente): privilegios sobre tablas que tiene un rol (en este caso "readonly")

```
baseX=> SELECT * FROM information_schema.table_privileges where grantee='readonly';

+-----+-----+-----+-----+-----+
| grantor | grantee | table_catalog | table_schema | table_name | privilege_type | is_grantable |
+-----+-----+-----+-----+-----+
| ua     | readonly | baseX        | sa          | probauat1   | SELECT        | NO           |
| YES   |           |               |             |             |               |              |
(1 row)
```

(solo te estaría mostrando una parte - NO LO ENTENDO)

???????????

Para ver tablas en varios esquemas a la vez y que no me salga por defecto el esquema public

```
select user; --usuario ub
create table sb.probaubT1(codigo intefer, nome varchar(20));
baseX=> \dt sb.*
               List of relations
 Schema |    Name    | Type  | Owner
-----+-----+-----+-----+
 sb    | probaubt1 | table | ub
(1 row)
```

```

basex=> insert into sb.probaubT1 values(1,'azucar');
INSERT 0 1
basex=> insert into sb.probaubT1 values(2,'leite');
INSERT 0 1
basex=> \q

```

nos conectamos a basex como postgres.

alter user postgres in database basex set search_path to public,sa,sb; /*para cambiar que por defecto no me muestre las tablas del esquema public (con el comando \dt), si no que muestre las tablas de todos los esquemas:

```

basex=# alter user postgres in database basex set search_path to public,sa,sb;
ALTER ROLE
basex=# \dt
      List of relations
 Schema | Name | Type  | Owner
-----+-----+-----+
 public | pra  | table | u1
(1 row)

```

comprobamos y me muestra por defecto las tablas de el esquema public a pesar de que hemos introducido el anterior comando. Esto es porque no se actualiza hasta que cerramos y volvemos a abrir sesion

salgo y me vuelvo a entrar:

```

basex=# \q
oracle@oracle-VirtualBox:~$ psql basex postgres;
Password for user postgres:
psql (14.7 (Ubuntu 14.7-0ubuntu0.22.04.1), server 12.12 (Ubuntu 12.12-0ubuntu0.20.04.1))
Type "help" for help.

basex=# \dt
      List of relations
 Schema |   Name   | Type  | Owner
-----+-----+-----+
 public | pra    | table | u1
 sa    | probauat1 | table | ua
 sb    | probaubt1 | table | ub
(3 rows)

basex=#

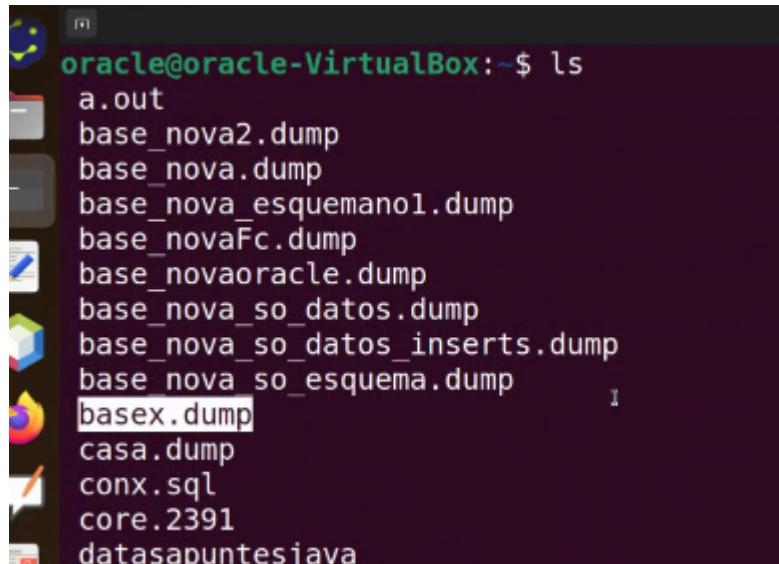
```

copia de seguridad mediante comandos

```
basex=# \q
oracle@oracle-VirtualBox:~$ pg_dump -U postgres -Fp basex > basex.dump
```

pg_dump
-U usu que lanza la copia
-Fp formato que lanza la copia

parece que no hicimos nada pero buscamos eso y lo abrimos:



```
oracle@oracle-VirtualBox:~$ ls
a.out
base_nova2.dump
base_nova.dump
base_nova_esquemanol.dump
base_novaFc.dump
base_novaoracle.dump
base_nova_so_datos.dump
base_nova_so_datos_inserts.dump
base_nova_so_esquema.dump
basex.dump
casa.dump
conx.sql
core.2391
datasapuntesjava
```

(en password: postgres)

-U // usuario

```
pg_dump -U postgres -Fp basex > basex.dump
```

//...texto plano

```
pg_dump -U postgres -Fp basex > basex.dump
```

hace copia de seguridad (para regenerar la base de datos) con todo incluso con los inserts que habiamos hecho

```
-- Data for Name: probaubt1; Type: TABLE DATA; Schema: sb; Owner: ub
-- 

COPY sb.probaubt1 (codigo, nome) FROM stdin;
1      azucar
2      leite
\.

-- 
-- Name: SCHEMA e1; Type: ACL; Schema: -; Owner: u1
-- 

GRANT USAGE ON SCHEMA e1 TO u2;
GRANT USAGE ON SCHEMA e1 TO u2;


```

Texto plano ▾ Largura de tabulación: 8 ▾ Lí 166, Col 1 ▾ INS

pone "COPY" pero son los "INSERT"

Borro la base e intento regenerarla a partir de la copia
borro:

```
oracle@oracle-VirtualBox:~$ dropdb -U postgres basex;
Password:
Password:
oracle@oracle-VirtualBox:~$ dropdb -U postgres basex;
Password:
dropdb: error: database removal failed: ERROR:  database "basex" does not exist
oracle@oracle-VirtualBox:~$ psql basex postgres;
Password for user postgres:
psql: error: connection to server on socket "/var/run/postgresql/.s.PGSQL.5432" failed: FATAL:
  database "basex" does not exist
```

creamos la basex: create data..

```
postgres=# \c basex postgres
psql (14.7 (Ubuntu 14.7-0ubuntu0.22.04.1), server 12.12 (Ubuntu 12.12-0ubuntu0.20.04.1))
You are now connected to database "basex" as user "postgres".
basex=# \dn
   List of schemas
   Name | Owner
-----+-----
 public | postgres
(1 row)

basex=# \dt
Did not find any relations.
```

y vemos que no tenemos nada

Si 'nombreScript_deLaCopiaSeguridad'
asi se regenera la base.

este tipo de copias no copia los usuarios y asi. Hay otros tipos de copias que si que hacen esto.

Ejemplo: soy usuario A y quiero exportar mis tablas

1. salgo \q
2. exporto la base

```
oracle@oracle-VirtualBox:~$ pg_dump -U ua -Fp basex > basexua.dump
Password:
pg_dump: error: query failed: ERROR: permission denied for schema el
pg_dump: error: query was: LOCK TABLE el.basex_el_u1_t1 IN ACCESS SHARE MODE
oracle@oracle-VirtualBox:~$
```

me da error porque no puede exportar cosas que no son suyas.

3. intento exportar solo su tabla

```
oracle@oracle-VirtualBox:~$ pg_dump -U ua -Fp -t probauat1 basex > basexua.dump
Password:
oracle@oracle-VirtualBox:~$
```

4. comprobamos

```
oracle@oracle-VirtualBox:~$ gedit basexua.dump
```

```
COPY sa.probauat1 (coda, nom) FROM stdin;
1      ola
1      ola
\.

-- 
-- Name: TABLE probauat1; Type: ACL; Schema: sa; Owner: ua
--

GRANT SELECT ON TABLE sa.probauat1 TO readonly;

-- 
-- PostgreSQL database dump complete
--
```

para ver mas info sobre esto: postgres pg_dump

```
oracle@oracle-VirtualBox:~$ gedit basexua.dump
oracle@oracle-VirtualBox:~$ pg_dump -U ua -Fp --column-inserts -t probauat1 basex > basexua.dump
Password:
```

- antes

```
INSERT INTO sa.probauat1 VALUES (1, 'ola');
INSERT INTO sa.probauat1 VALUES (1, 'ola');
```

- ahora

```
INSERT INTO sa.probauat1 (coda, nom) VALUES (1, 'ola');
INSERT INTO sa.probauat1 (coda, nom) VALUES (1, 'ola');
```

mas especifico pero mas lento

Exportacion normal

- Operacion 1

```
ox:~$ pg_dump -U postgres --inserts -Fp basex > basexcompleta.dump
```

ahora edito fichero:

```
baseX:~$ gedit basexua.dump
```

Para exportar tablas

- Operacion 2

```
oracle@oracle-VirtualBox:~$ pg_dump -U ua -t 'sa.probauat1' --inserts -Fp basex > uat1.dump
Password:
oracle@oracle-VirtualBox:~$
```

(estamos indicando el esquema “sa”)

“-t” incluye lo que quieres que exporte

“-T” excluye lo que no quieres que exporte

si no incluye “-t” o “-T” incluye todo lo que pueda exportar ese usuario

Por ejemplo aqui nos da ERROR porque ese usuario “ua” no tiene los permisos sobre las tablas que quiere exportar (que sería todo menos lo que está después de la “-T”)

```
oracle@oracle-VirtualBox:~$ pg_dump -U ua -T 'sa.probauat1' --inserts -Fp basex > uat1.dump
Password:
pg_dump: error: query failed: ERROR: permission denied for schema el
pg_dump: error: query was: LOCK TABLE el.basex el.ul t1 IN ACCESS SHARE MODE
```

no se puede ni tratar el error y ni hay ninguna opcion para que exporte todo menos lo que no puede, es así como está hecho

Para exportar esquemas en concreto

```
oracle-VirtualBox:~$ gedit uat1.dump
oracle-VirtualBox:~$ pg_dump -U postgres -n 'sa' --inserts -Fp basex > uat1.dump
```

con -N excluye y con -n incluye (igual que con la T)

se importarán todas las tablas que estén dentro de los esquemas que incluyo

te incluye todos los esquemas que empiezan por “s..”

```
ox:~$ pg_dump -U postgres -n 's*' --inserts -Fp basex > uat1.dump
```

Combinación de las 2 cosas

```
alBox:~$ gedit uat1.dump
alBox:~$ pg_dump -U postgres -t 's*.p*' --inserts -Fp basex > uat1.dump
```

pero realmente, lo que estoy exportando solo es la tabla “-t” y no exporto el esquema

Exportacion comprimida

se hace con “-Fc”

```
root@box:~$ pg_dump -U postgres -Fc baseX > baseXcomprimida.dump
```

el fichero estara con todos los datos encriptados → entonces, como recupero lo datos??:
(pg_restore para recuperar)

```
x:~$ pg_restore -U postgres -C -d postgres baseXcomprimida.dump
```

este es el usuario que la va a recuperar:

```
not exist
x:~$ pg_restore -U postgres -C -d postgres baseXcomprimida.dump
```

esto es la base de datos: “postgres”

```
tore - U postgres -C -d postgres baseXcomprimida.dump
```

pero no tendria que poner: baseXcomprimida.dump??? NO porque esta es la base de datos que coge como referencia (necesita una base de referencia para recuperar los datos)

generalmente usamos “postgres” como la base de referencia para esta recuperacion

para borrar una base:

```
x:~$ dropdb -U postgres baseX
```

crear una base cun nome distinto a partir de la base comprimida

imaginar que la nueva base debe llamarse baseZ

```
createdb -U postgres -T template0 baseZ
pg_restore -U postgres -d baseZ baseXcomprimida.dump
```

- hay otra forma para hacerlo pero con este comando (haces lo mismo que los 2 de arriba)

```
 Password:
oracle@oracle-VirtualBox:~$ pg_restore -U postgres -C -d postgres -L listado.txt basexcomprimida
a.dump
Password:
```

“-T” = template; indica que utiliza como referencia la plantilla “template()”

template0??

- es una plantilla con el catálogo que contiene cosas necesarias para la tabla. Io necesito hacer asi porque el formato es “-Fc”.

Cargaria lo que esta en 2 “basexcomprimida” en “basez”

comprobacion:

List of databases						
Name	Owner	Encoding	Collate	Ctype	Access privileges	
asirl	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	=T/postgres + postgres=CTc/postgres+ oracle=c/postgres + user3=c/postgres + user4=c/postgres	
base_nova	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8		
baseex	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8		
basey	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	postgres=CTc/postgres+ readonly=c/postgres + ua=c/postgres	
basez	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8		
dam1	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8	=T/postgres + postgres=CTc/postgres+ user3=c/postgres + udam1=c/postgres + udam2=c/postgres	
futbol2	postgres	UTF8	gl_ES.UTF-8	gl_ES.UTF-8		

(basez == basex)

Crear fichero resumen con formato especial

ESTE RESUMEN NO VALE DE NADA SIN TENER EL FICHERO COMPRIMIDO → NO LO ENTIENDO MUY BIEN???

Va a hacer un listado en formato especial (legible) basado en lo que hay en basexcomprimida y lo va a meter en listado.txt

REALMENTE ES UN RESUMEN de lo que tenemos:

```
alBox:~$ pg_restore -l basexcomprimida.dump > listado.txt
```

abro el documento:

```
*Documento sin título 1
8; 2615 1028385 SCHEMA - e3 u3
2990; 0 0 ACL - SCHEMA public postgres
10; 2615 1028386 SCHEMA - sa ua
2991; 0 0 ACL - SCHEMA sa ua
9; 2615 1028387 SCHEMA - sb ub
207; 1259 1028388 TABLE e1 basex_e1_u1_t1_u1
2992; 0 0 ACL e1 TABLE basex_e1_u1_t1_u1
2993; 0 0 ACL e1 COLUMN basex_e1_u1_t1.nome u1
208; 1259 1028391 TABLE e2 basex_e2_u2_t1_u2
2994; 0 0 ACL e2 TABLE basex_e2_u2_t1_u2
209; 1259 1028394 TABLE public pra u1
210; 1259 1028397 TABLE sa probauat1 ua
2995; 0 0 ACL sa TABLE probauat1 ua
211; 1259 1028400 TABLE sb probaubt1 ub
2977; 0 1028388 TABLE DATA e1 basex_e1_u1_t1_u1
2978; 0 1028391 TABLE DATA e2 basex_e2_u2_t1_u2
2979; 0 1028394 TABLE DATA public pra u1
```

CARACTERISTICAS

- hace referencia esto es lo que crea la tabla

```
*Documento sin título 1
10; 2615 1028386 SCHEMA - sa ua
2991; 0 0 ACL - SCHEMA sa ua
9; 2615 1028387 SCHEMA - sb ub
207; 1259 1028388 TABLE e1 basex_e1_u1_t1_u1
2992; 0 0 ACL e1 TABLE basex_e1_u1_t1_u1
2993; 0 0 ACL e1 COLUMN basex_e1_u1_t1.nome u1
208; 1259 1028391 TABLE e2 basex_e2_u2_t1_u2
2994; 0 0 ACL e2 TABLE basex_e2_u2_t1_u2
209; 1259 1028394 TABLE public pra u1
210; 1259 1028397 TABLE sa probauat1 ua
2995; 0 0 ACL sa TABLE probauat1 ua
211; 1259 1028400 TABLE sb probaubt1 ub
2977; 0 1028388 TABLE DATA e1 basex_e1_u1_t1_u1
2978; 0 1028391 TABLE DATA e2 basex_e2_u2_t1_u2
2979; 0 1028394 TABLE DATA public pra u1
2980; 0 1028397 TABLE DATA sa probauat1 ua
2981; 0 1028400 TABLE DATA sb probaubt1 ub
```

- hace referencia a los datos de esa tabla

```
*Documento sin título 1
2990; 0 0 ACL SCHEMA public postgres
10; 2615 1028386 SCHEMA - sa ua
2991; 0 0 ACL - SCHEMA sa ua
9; 2615 1028387 SCHEMA - sb ub
207; 1259 1028388 TABLE e1 basex_e1_u1_t1 u1
2992; 0 0 ACL e1 TABLE basex_e1_u1_t1 u1
2993; 0 0 ACL e1 COLUMN basex_e1_u1_t1.nome u1
208; 1259 1028391 TABLE e2 basex_e2_u2_t1 u2
2994; 0 0 ACL e2 TABLE basex_e2_u2_t1 u2
209; 1259 1028394 TABLE public pra u1
210; 1259 1028397 TABLE sa probauat1 ua
2995; 0 0 ACL sa TABLE probauat1 ua
211; 1259 1028400 TABLE sb probaubt1 ub
2977; 0 1028388 TABLE DATA e1 basex_e1_u1_t1 u1
2978; 0 1028391 TABLE DATA e2 basex_e2_u2_t1 u2
2979; 0 1028394 TABLE DATA public pra u1
2980; 0 1028397 TABLE DATA sa probauat1 ua
2981; 0 1028400 TABLE DATA sb probaubt1 ub
```

- hace referencia a dd estan estos datos que estan dentro del fichero comprimido

```
*Documento sin título 1
2990; 0 0 ACL SCHEMA public postgres
10; 2615 1028386 SCHEMA - sa ua
2991; 0 0 ACL - SCHEMA sa ua
9; 2615 1028387 SCHEMA - sb ub
207; 1259 1028388 TABLE e1 basex_e1_u1_t1 u1
2992; 0 0 ACL e1 TABLE basex_e1_u1_t1 u1
2993; 0 0 ACL e1 COLUMN basex_e1_u1_t1.nome u1
208; 1259 1028391 TABLE e2 basex_e2_u2_t1 u2
2994; 0 0 ACL e2 TABLE basex_e2_u2_t1 u2
209; 1259 1028394 TABLE public pra u1
210; 1259 1028397 TABLE sa probauat1 ua
2995; 0 0 ACL sa TABLE probauat1 ua
211; 1259 1028400 TABLE sb probaubt1 ub
2977; 0 1028388 TABLE DATA e1 basex_e1_u1_t1 u1
2978; 0 1028391 TABLE DATA e2 basex_e2_u2_t1 u2
2979; 0 1028394 TABLE DATA public pra u1
2980; 0 1028397 TABLE DATA sa probauat1 ua
2981; 0 1028400 TABLE DATA sb probaubt1 ub
```

- ACL hace referencia a los comandos con los permisos de los usuarios.

```
0; 2615 1028386 SCHEMA - sa ua
991; 0 0 ACL - SCHEMA sa ua
10; 1259 1028397 TABLE sa probauat1 ua
995; 0 0 ACL sa TABLE probauat1 ua
980; 0 1028397 TABLE DATA sa probauat1 ua
```

si no hubiera ningun privilegio de acceso,esta linea no existiria:

```
110; 1259 1028397 TABLE sa probauat1 ua
2995; 0 0 ACL sa TABLE probauat1 ua
```

estos serian los privilegios sobre el esquema:

```
991; 0 0 ACL - SCHEMA sa ua
```

comprobacion:

List of schemas			
Name	Owner	Access privileges	Description
public	postgres	postgres=UC/postgres+ =UC/postgres	standard public schema
sa	ua	ua=UC/ua readonly=U/ua	+
(2 rows)			

por ejemplo el “readonly” estará dentro de esas coordenadas:

```
991; 0 0 ACL
```

Editando el resumen

Desde este resumen puedo eliminar cosas como por ejemplo borro todo menos todo lo que hay en “sa”. tendría que borrar todas aquellas líneas que no tengan que ver con esta.

si borro esto no la va a cargar: **importante**

```
0; 2615 1028386 SCHEMA - sa ua
991; 0 0 ACL - SCHEMA sa ua
10; 1259 1028397 TABLE sa probauat1 ua
995; 0 0 ACL sa TABLE probauat1 ua
980; 0 1028397 TABLE DATA sa probauat1 ua
```

Restaurar parte de la informacion del fichero comprimido (partiendo de que la base original no fue eliminada)

- antes

```
pg_restore -l basexcomprimida.dump > listado.txt
edito el fichero 'listado.txt' y elimino todo lo que no tiene que ver con lo que
no quiero restaurar
pg_restore -U postgres -C -d postg|es -L listado.txt basexcomprimida.dump
```

- despues

```
pg_restore -U postgres -d basex| -L listado.txt basexcomprimida.dump
```

12 may 2023

18 may 2023