ACTIVIDAD 4 – GESTIÓN DE PRIVILEGIOS

Cristóbal Suárez Abad

ADMINISTRACIÓN DE SISTEMAS GESTORES DE BASES DE DATOS - 2º ASIR

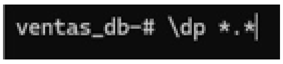
Contenido

Actividad 4 – Gestión de privilegios (criterios d–g)

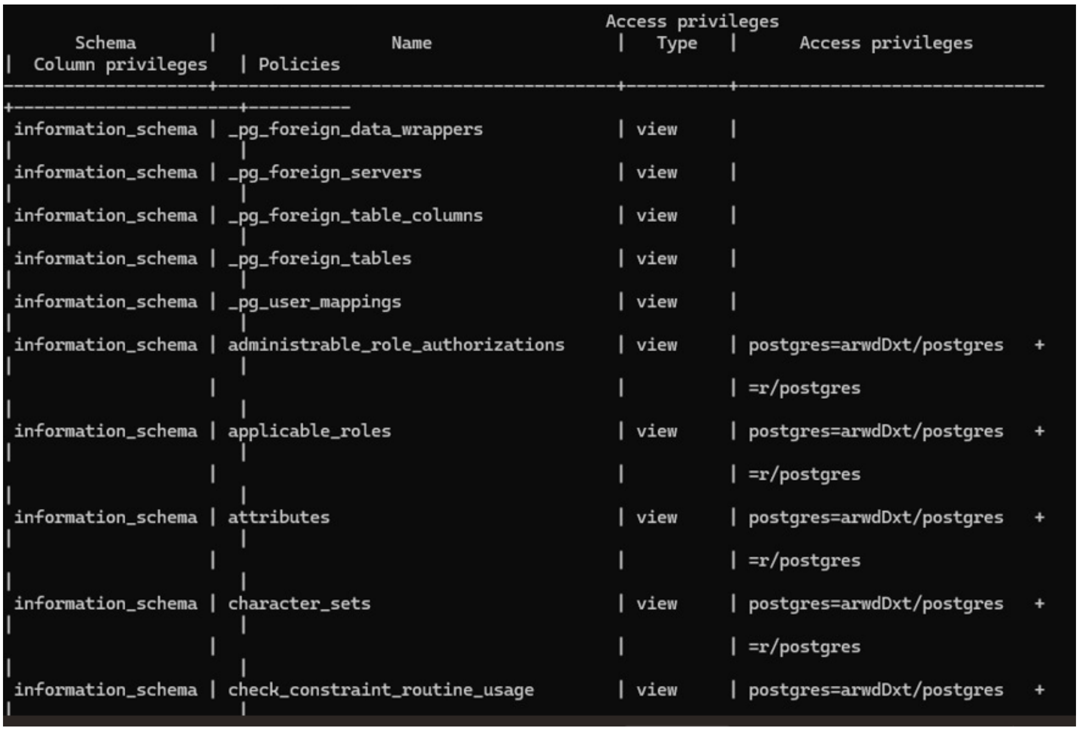
Comprobación previa

- Lista los privilegios actuales sobre todas las tablas y vistas[1]:

\dp *.*



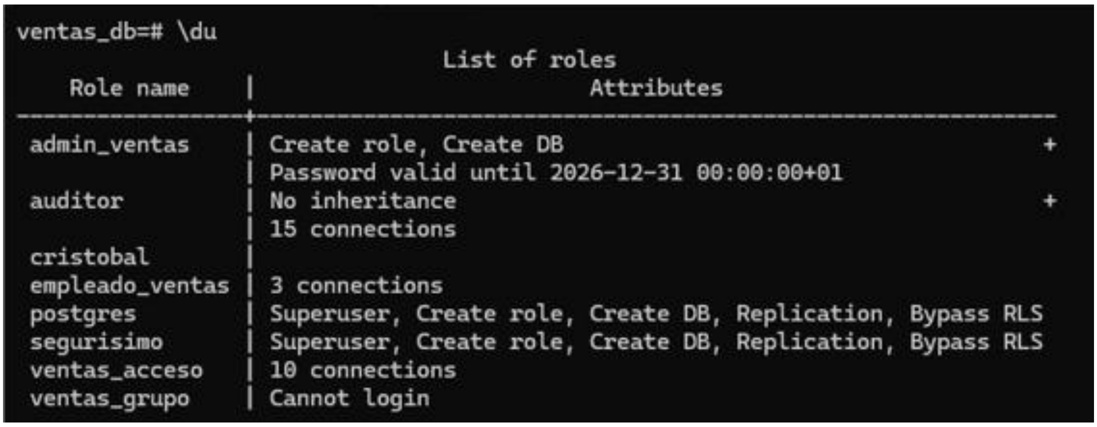
```
ventas_db-# \dp *.*
```



```
                                                              Access privileges
       Schema       |                    Name                    |   Type   |        Access privileges
|  Column privileges   | Policies
--------------------+--------------------------------------------+----------+------------------------------
+----------------------+----------
 information_schema | _pg_foreign_data_wrappers                  | view     |
|                      |
 information_schema | _pg_foreign_servers                        | view     |
|                      |
 information_schema | _pg_foreign_table_columns                  | view     |
|                      |
 information_schema | _pg_foreign_tables                         | view     |
|                      |
 information_schema | _pg_user_mappings                          | view     |
|                      |
 information_schema | administrable_role_authorizations          | view     | postgres=arwdDxt/postgres    +
|                      |
                    |                                            |          | =r/postgres
|                      |
 information_schema | applicable_roles                           | view     | postgres=arwdDxt/postgres    +
|                      |
                    |                                            |          | =r/postgres
|                      |
 information_schema | attributes                                 | view     | postgres=arwdDxt/postgres    +
|                      |
                    |                                            |          | =r/postgres
|                      |
 information_schema | character_sets                             | view     | postgres=arwdDxt/postgres    +
|                      |
                    |                                            |          | =r/postgres
|                      |
 information_schema | check_constraint_routine_usage             | view     | postgres=arwdDxt/postgres    +
|                      |
```

- Comprueba también los roles y sus pertenencias:

\du



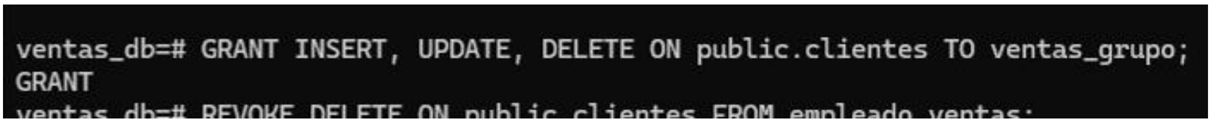
```
ventas_db=# \du
                              List of roles
   Role name    |                         Attributes
----------------+------------------------------------------------------------
 admin_ventas   | Create role, Create DB                                     +
                | Password valid until 2026-12-31 00:00:00+01
 auditor        | No inheritance                                             +
                | 15 connections
 cristobal      |
 empleado_ventas | 3 connections
 postgres       | Superuser, Create role, Create DB, Replication, Bypass RLS
 segurisimo     | Superuser, Create role, Create DB, Replication, Bypass RLS
 ventas_acceso  | 10 connections
 ventas_grupo   | Cannot login
```

[1] https://supabase.com/blog/postgres-roles-and-privileges

Asignación y prueba de privilegios

- Sobre la tabla clientes
  - Concede privilegios de manipulación a `ventas_grupo` sobre `clientes`

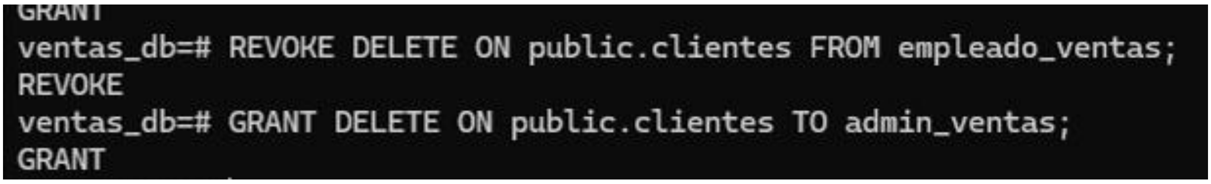
GRANT SELECT, INSERT, UPDATE, DELETE ON public.clientes TO ventas_grupo



```
ventas_db=# GRANT INSERT, UPDATE, DELETE ON public.clientes TO ventas_grupo;
GRANT
ventas_db=# REVOKE DELETE ON public.clientes FROM empleado_ventas;
```

  - Da permisos de eliminación **solo** al usuario `admin_ventas`:

GRANT DELETE ON public.clientes TO admin_ventas;



```
GRANT
ventas_db=# REVOKE DELETE ON public.clientes FROM empleado_ventas;
REVOKE
ventas_db=# GRANT DELETE ON public.clientes TO admin_ventas;
GRANT
```

  - Revoca explícitamente el permiso de eliminación a `empleado_ventas`:

REVOKE DELETE ON public.clientes TO empleado_ventas;

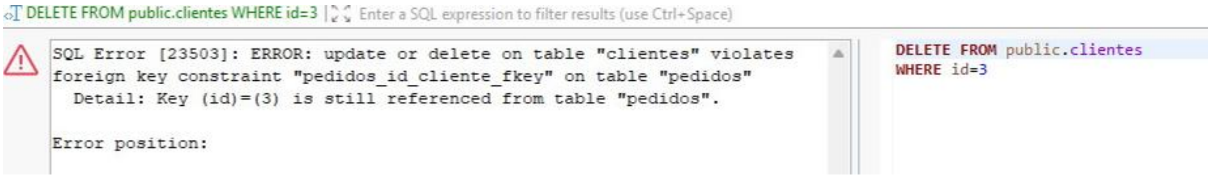
  - Verifica el efecto práctico:

Desde el usuario admin_ventas:

DELETE FROM public.clientes

WHERE id=3;

En teoría podemos borrarlo, pero no nos deja porque está siendo usada en otra tabla.

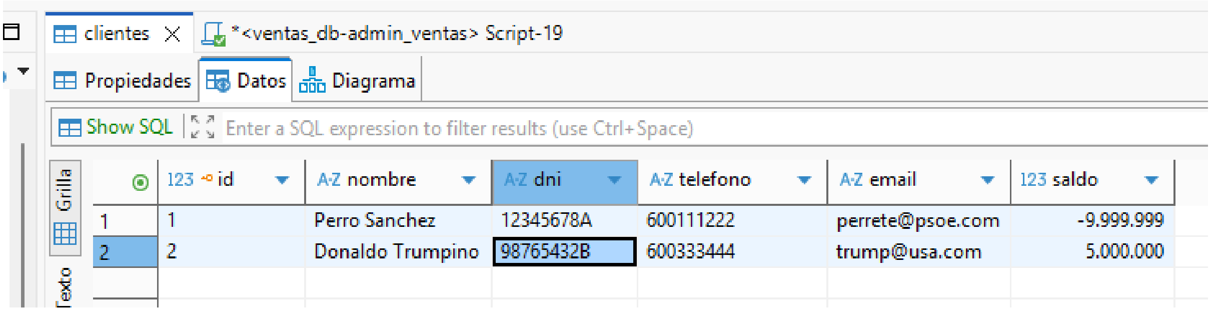


DELETE FROM public.clientes WHERE id=3
Enter a SQL expression to filter results (use Ctrl+Space)

```
SQL Error [23503]: ERROR: update or delete on table "clientes" violates
foreign key constraint "pedidos_id_cliente_fkey" on table "pedidos"
  Detail: Key (id)=(3) is still referenced from table "pedidos".

Error position:
```

```
DELETE FROM public.clientes
WHERE id=3
```

Si desde un usuario con permisos borramos las entradas del cliente con “id” igual a tres en la tabla “pedidos” (**DELETE FROM pedidos WHERE id_cliente = 3;**), podremos hacer que el comando tenga efecto:


clientes
*<ventas_db-admin_ventas> Script-19
Propiedades
Datos
Diagrama
Show SQL
Enter a SQL expression to filter results (use Ctrl+Space)
Grilla
Texto

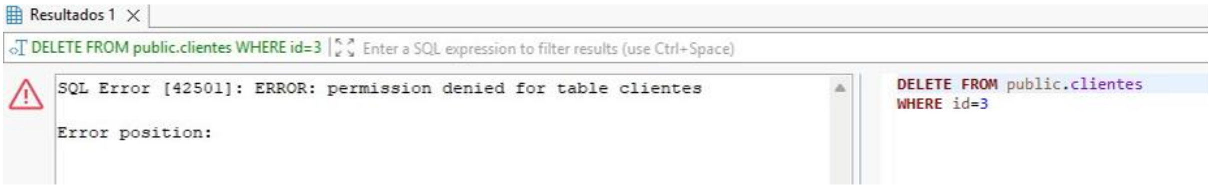
	id	nombre	dni	telefono	email	saldo
1	1	Perro Sanchez	12345678A	600111222	perrete@psoe.com	-9.999.999
2	2	Donaldo Trumpino	98765432B	600333444	trump@usa.com	5.000.000

Desde el usuario **empleado_ventas**:

DELETE FROM public.clientes

WHERE id=3;

A este no le deja porque no tiene permiso.


Resultados 1
DELETE FROM public.clientes WHERE id=3
Enter a SQL expression to filter results (use Ctrl+Space)

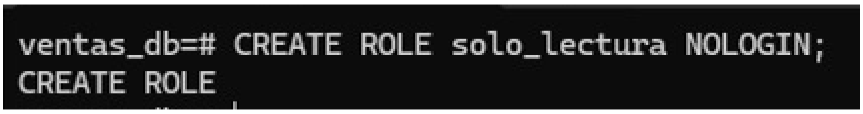
```
SQL Error [42501]: ERROR: permission denied for table clientes

Error position:
```

```
DELETE FROM public.clientes
WHERE id=3
```

Agrupación de privilegios y rol de solo lectura

- Crea un rol de solo lectura:

CREATE ROLE solo_lectura NOLOGIN;



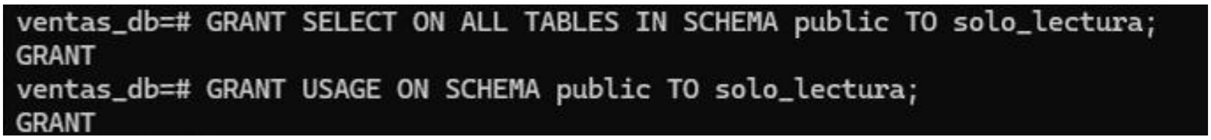
```
ventas_db=# CREATE ROLE solo_lectura NOLOGIN;
CREATE ROLE
```

- Concede privilegios de lectura sobre todas las tablas y vistas actuales:

GRANT SELECT ON ALL TABLES IN SCHEMA public TO solo_lectura;

Para poder acceder a los objetos también hay que darle este permiso:

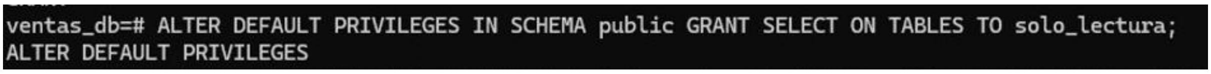
GRANT USAGE ON SCHEMA public TO solo_lectura;



```
ventas_db=# GRANT SELECT ON ALL TABLES IN SCHEMA public TO solo_lectura;
GRANT
ventas_db=# GRANT USAGE ON SCHEMA public TO solo_lectura;
GRANT
```

- Haz que los **nuevos objetos creados** también sean legibles por este rol:

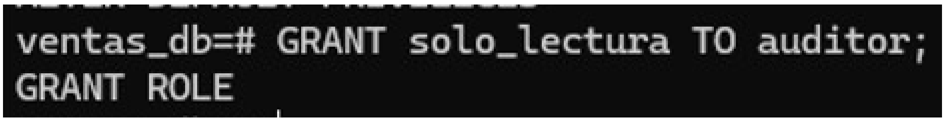
ALTER DEFAULT PRIVILEGES IN SCHEMA public GRANT SELECT ON TABLES TO solo_lectura;



```
ventas_db=# ALTER DEFAULT PRIVILEGES IN SCHEMA public GRANT SELECT ON TABLES TO solo_lectura;
ALTER DEFAULT PRIVILEGES
```

- Asigna el rol al usuario `auditor`:

GRANT solo_lectura TO auditor;

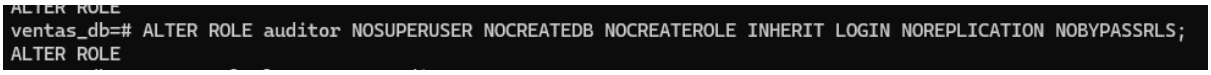


```
ventas_db=# GRANT solo_lectura TO auditor;
GRANT ROLE
```

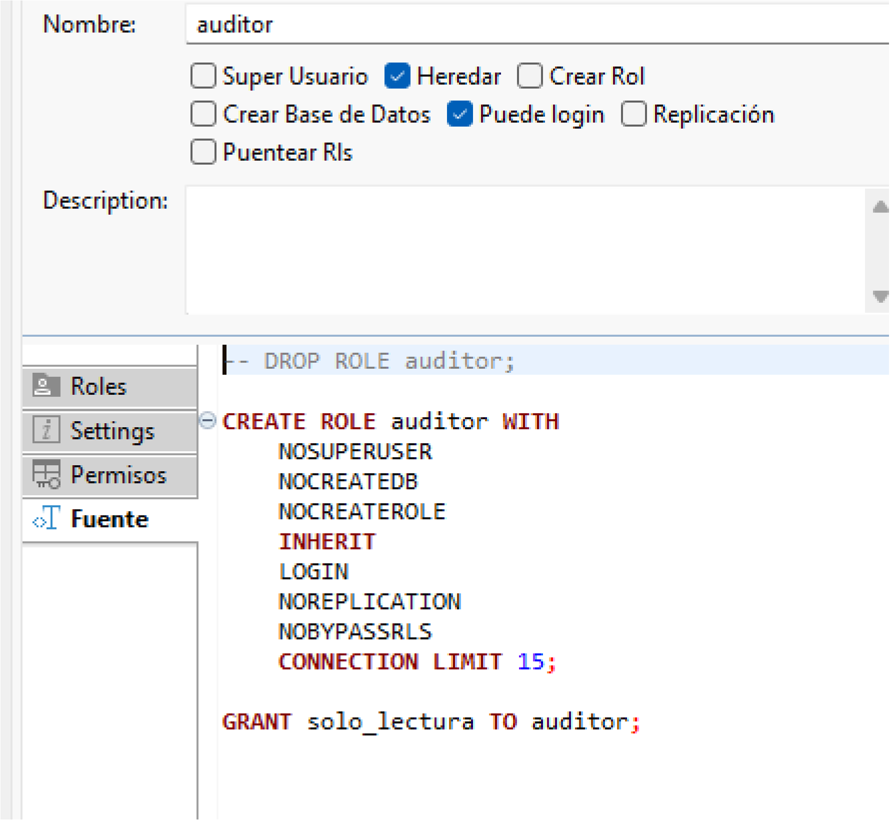
- Comprueba que el usuario `auditor` hereda los permisos:

Su configuración modificada para que pueda heredar:

ALTER ROLE auditor NOSUPERUSER NOCREATEDB NOCREATEROLE INHERIT LOGIN NOREPLICATION NOBYPASSRLS;



```
ALTER ROLE
ventas_db=# ALTER ROLE auditor NOSUPERUSER NOCREATEDB NOCREATEROLE INHERIT LOGIN NOREPLICATION NOBYPASSRLS;
ALTER ROLE
```



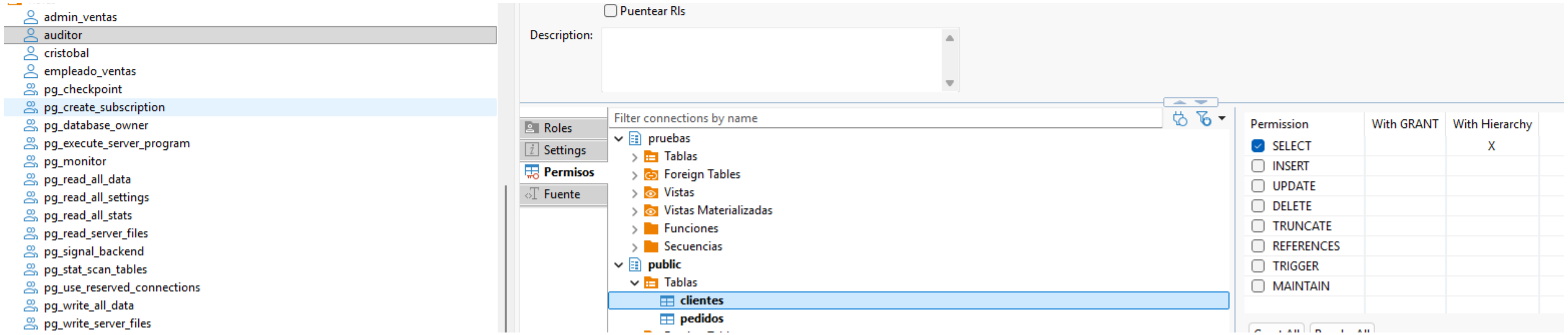
Nombre: auditor

□ Super Usuario ☑ Heredar □ Crear Rol
□ Crear Base de Datos ☑ Puede login □ Replicación
□ Puentear Rls

Description:

```
-- DROP ROLE auditor;

CREATE ROLE auditor WITH
    NOSUPERUSER
    NOCREATEDB
    NOCREATEROLE
    INHERIT
    LOGIN
    NOREPLICATION
    NOBYPASSRLS
    CONNECTION LIMIT 15;

GRANT solo_lectura TO auditor;
```

Los hereda en ambas tablas.

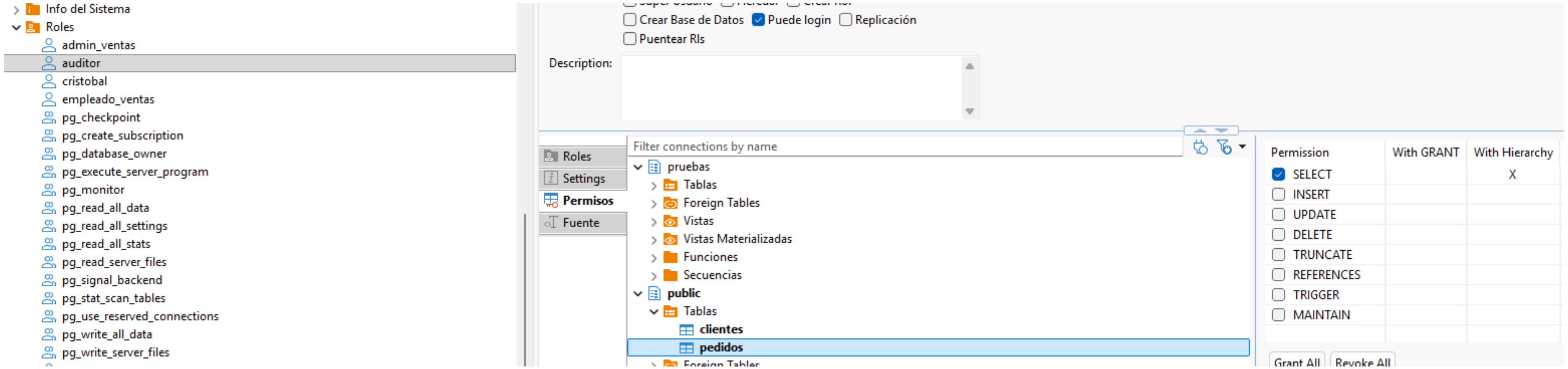


admin_ventas
auditor
cristobal
empleado_ventas
pg_checkpoint
pg_create_subscription
pg_database_owner
pg_execute_server_program
pg_monitor
pg_read_all_data
pg_read_all_settings
pg_read_all_stats
pg_read_server_files
pg_signal_backend
pg_stat_scan_tables
pg_use_reserved_connections
pg_write_all_data
pg_write_server_files

□ Puentear Rls

Description:

Permission	With GRANT	With Hierarchy
☑ SELECT		X
□ INSERT		
□ UPDATE		
□ DELETE		
□ TRUNCATE		
□ REFERENCES		
□ TRIGGER		
□ MAINTAIN		



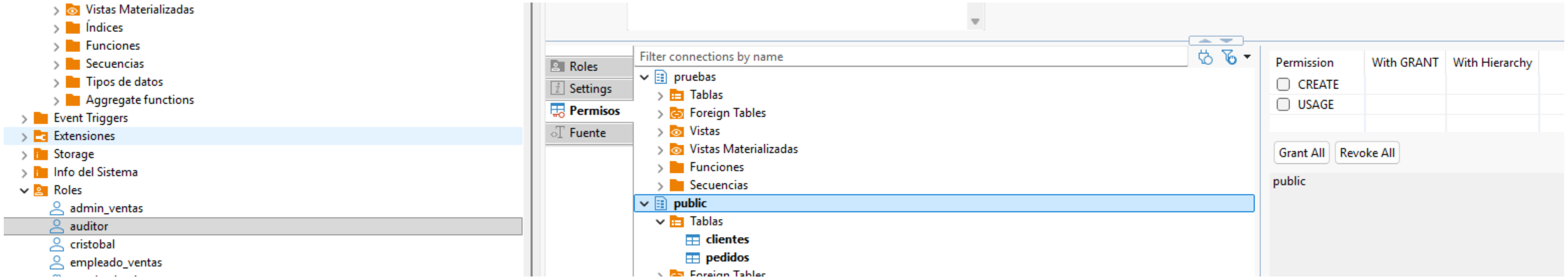
admin_ventas
auditor
cristobal
empleado_ventas
pg_checkpoint
pg_create_subscription
pg_database_owner
pg_execute_server_program
pg_monitor
pg_read_all_data
pg_read_all_settings
pg_read_all_stats
pg_read_server_files
pg_signal_backend
pg_stat_scan_tables
pg_use_reserved_connections
pg_write_all_data
pg_write_server_files

□ Crear Base de Datos ☑ Puede login □ Replicación
□ Puentear Rls

Description:

Permission	With GRANT	With Hierarchy
☑ SELECT		X
□ INSERT		
□ UPDATE		
□ DELETE		
□ TRUNCATE		
□ REFERENCES		
□ TRIGGER		
□ MAINTAIN		

Por motivos desconocidos no nos aparece en el esquema la herencia de “**Usage**”:

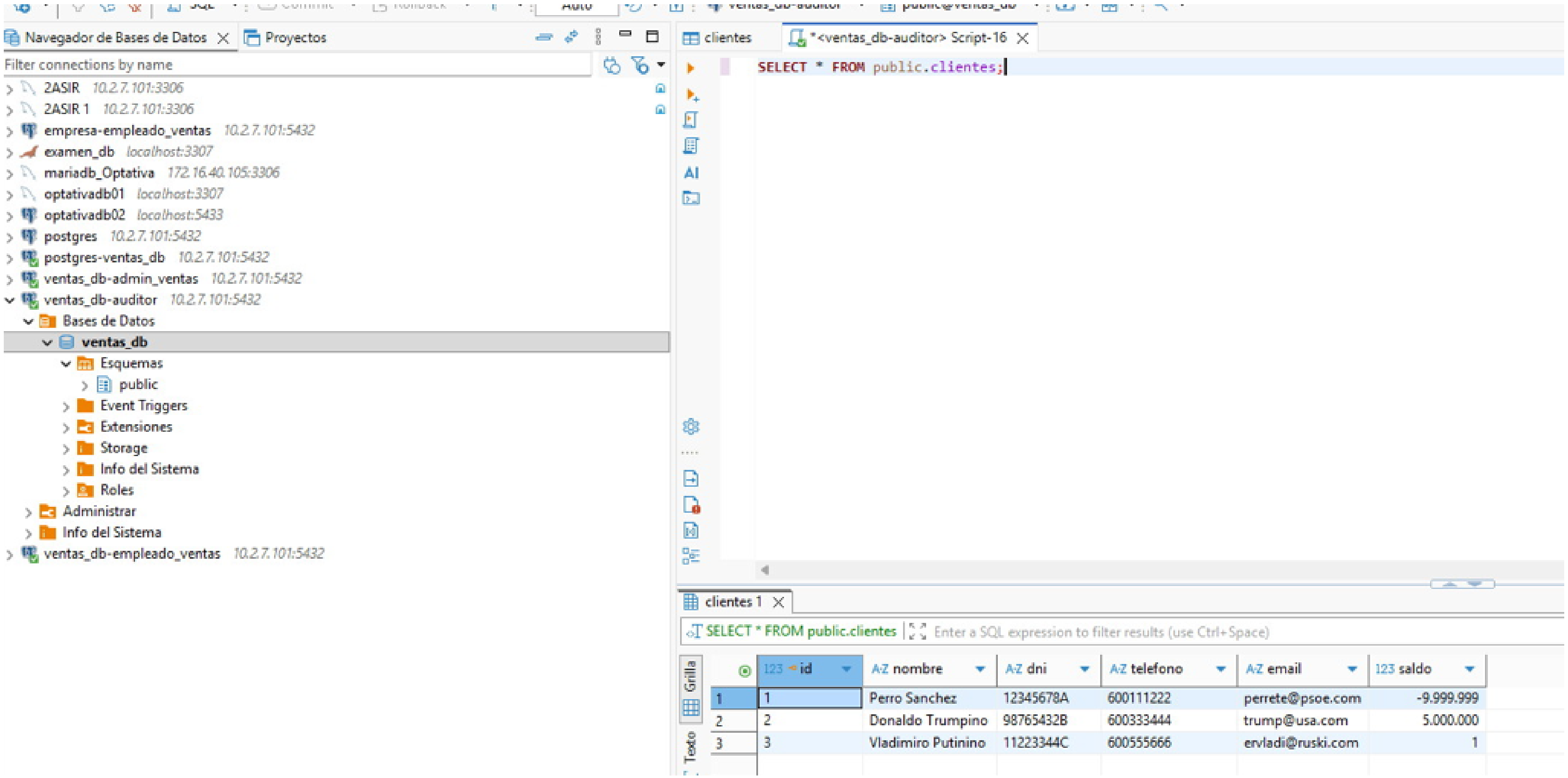


Permission	With GRANT	With Hierarchy
☐ CREATE		
☐ USAGE		

- Inicia sesión como `auditor` y prueba:

Desde la sesión del auditor, ejecutamos:

SELECT * FROM public.clientes;

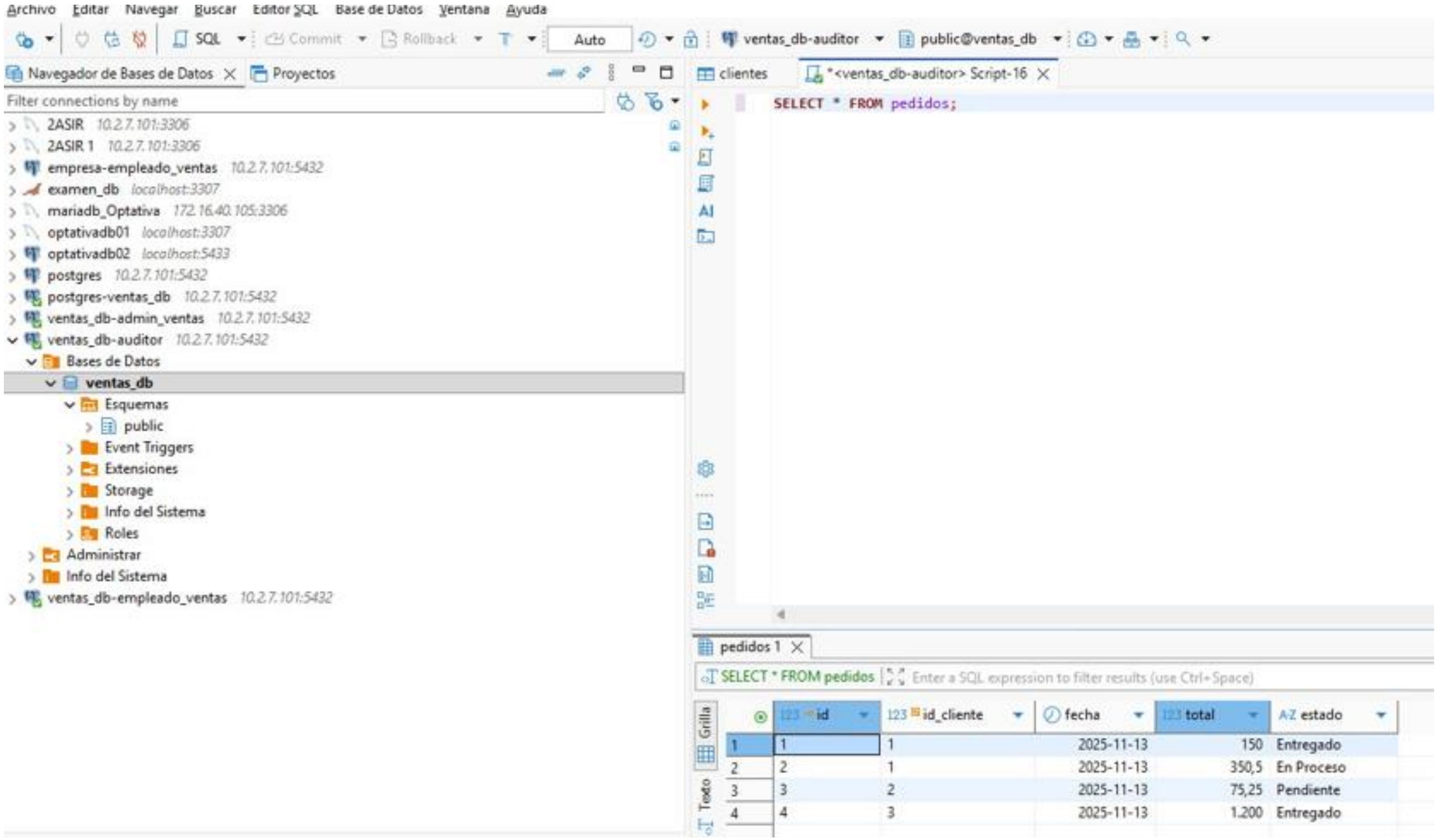


Navegador de Bases de Datos
Proyectos
Filter connections by name
2ASIR 10.2.7.101:3306
2ASIR 1 10.2.7.101:3306
empresa-empleado_ventas 10.2.7.101:5432
examen_db localhost:3307
mariadb_Optativa 172.16.40.105:3306
optativadb01 localhost:3307
optativadb02 localhost:5433
postgres 10.2.7.101:5432
postgres-ventas_db 10.2.7.101:5432
ventas_db-admin_ventas 10.2.7.101:5432
ventas_db-auditor 10.2.7.101:5432
Bases de Datos
ventas_db
Esquemas
public
Event Triggers
Extensiones
Storage
Info del Sistema
Roles
Administrar
Info del Sistema
ventas_db-empleado_ventas 10.2.7.101:5432
clientes
*<ventas_db-auditor> Script-16
clientes 1
Enter a SQL expression to filter results (use Ctrl+Space)
Grilla
Texto

```
SELECT * FROM public.clientes;
```

	id	nombre	dni	telefono	email	saldo
1	1	Perro Sanchez	12345678A	600111222	perrete@psoe.com	-9.999.999
2	2	Donaldo Trumpino	98765432B	600333444	trump@usa.com	5.000.000
3	3	Vladimiro Putinino	11223344C	600555666	ervladi@ruski.com	1

SELECT * FROM pedidos;



Archivo
Editar
Navegar
Buscar
Editor SQL
Base de Datos
Ventana
Ayuda
SQL
Commit
Rollback
Auto
ventas_db-auditor
public@ventas_db
Navegador de Bases de Datos
Proyectos
Filter connections by name
2ASIR 10.2.7.101:3306
2ASIR 1 10.2.7.101:3306
empresa-empleado_ventas 10.2.7.101:5432
examen_db localhost:3307
mariadb_Optativa 172.16.40.105:3306
optativadb01 localhost:3307
optativadb02 localhost:5433
postgres 10.2.7.101:5432
postgres-ventas_db 10.2.7.101:5432
ventas_db-admin_ventas 10.2.7.101:5432
ventas_db-auditor 10.2.7.101:5432
Bases de Datos
ventas_db
Esquemas
public
Event Triggers
Extensiones
Storage
Info del Sistema
Roles
Administrar
Info del Sistema
ventas_db-empleado_ventas 10.2.7.101:5432
clientes
*<ventas_db-auditor> Script-16
pedidos 1
Enter a SQL expression to filter results (use Ctrl+Space)
Grilla
Texto

```sql
SELECT * FROM pedidos;
```

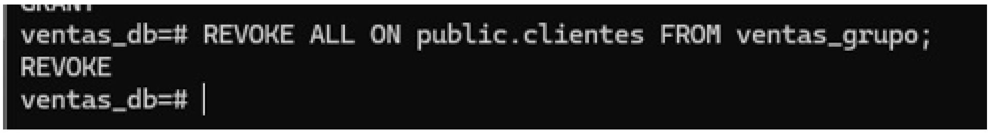
	id	id_cliente	fecha	total	estado
1	1	1	2025-11-13	150	Entregado
2	2	1	2025-11-13	350,5	En Proceso
3	3	2	2025-11-13	75,25	Pendiente
4	4	3	2025-11-13	1.200	Entregado

Gestión dinámica de privilegios

Sobre la tabla clientes:

- Elimina los permisos del grupo de ventas:

REVOKE ALL ON public.clientes FROM ventas_grupo;

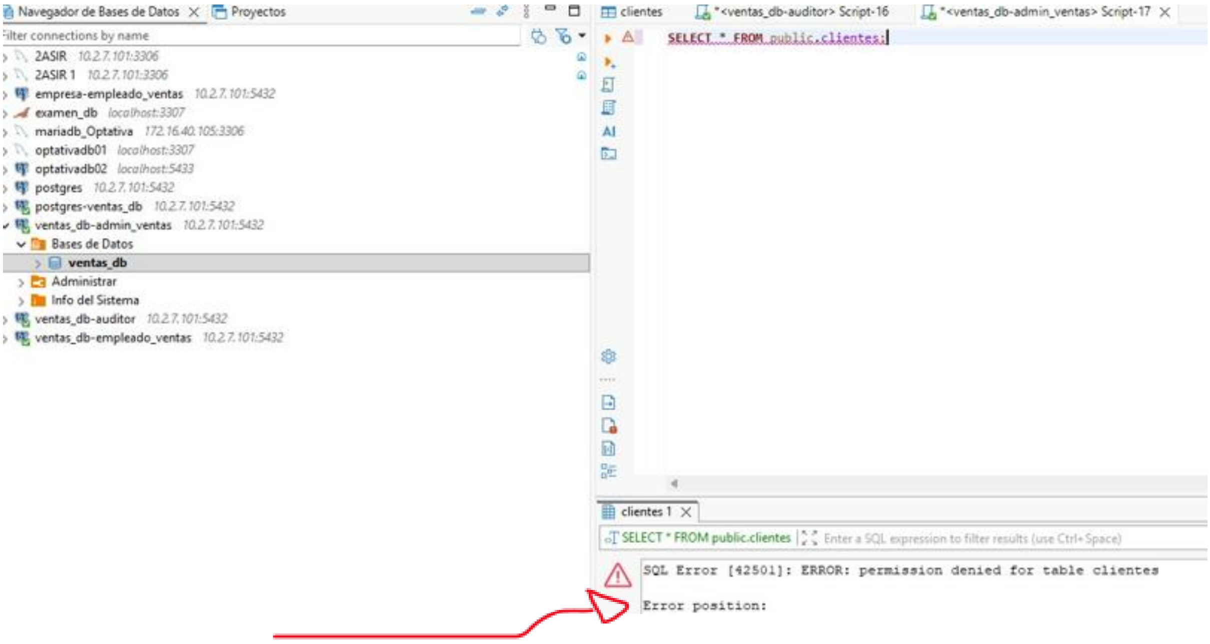


```
GRANT
ventas_db=# REVOKE ALL ON public.clientes FROM ventas_grupo;
REVOKE
ventas_db=# |
```

- Observa el efecto

Desde el usuario “admin_ventas”:

SELECT * FROM public.clientes;


Navegador de Bases de Datos
Proyectos
Filter connections by name
2ASIR 10.2.7.101:3306
2ASIR 1 10.2.7.101:3306
empresa-empleado_ventas 10.2.7.101:5432
examen_db localhost:3307
mariadb_Optativa 172.16.40.105:3306
optativadb01 localhost:3307
optativadb02 localhost:5433
postgres 10.2.7.101:5432
postgres-ventas_db 10.2.7.101:5432
ventas_db-admin_ventas 10.2.7.101:5432
Bases de Datos
ventas_db
Administrar
Info del Sistema
ventas_db-auditor 10.2.7.101:5432
ventas_db-empleado_ventas 10.2.7.101:5432
clientes
*<ventas_db-auditor> Script-16
*<ventas_db-admin_ventas> Script-17
AI
clientes 1
Enter a SQL expression to filter results (use Ctrl+Space)

```
SELECT * FROM public.clientes;
```

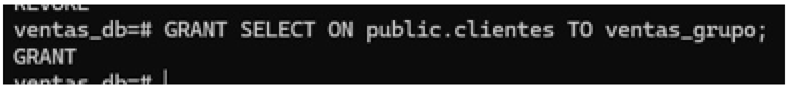
SELECT * FROM public.clientes

```
SQL Error [42501]: ERROR: permission denied for table clientes

Error position:
```

No tiene permiso.

- Añade privilegios de consulta

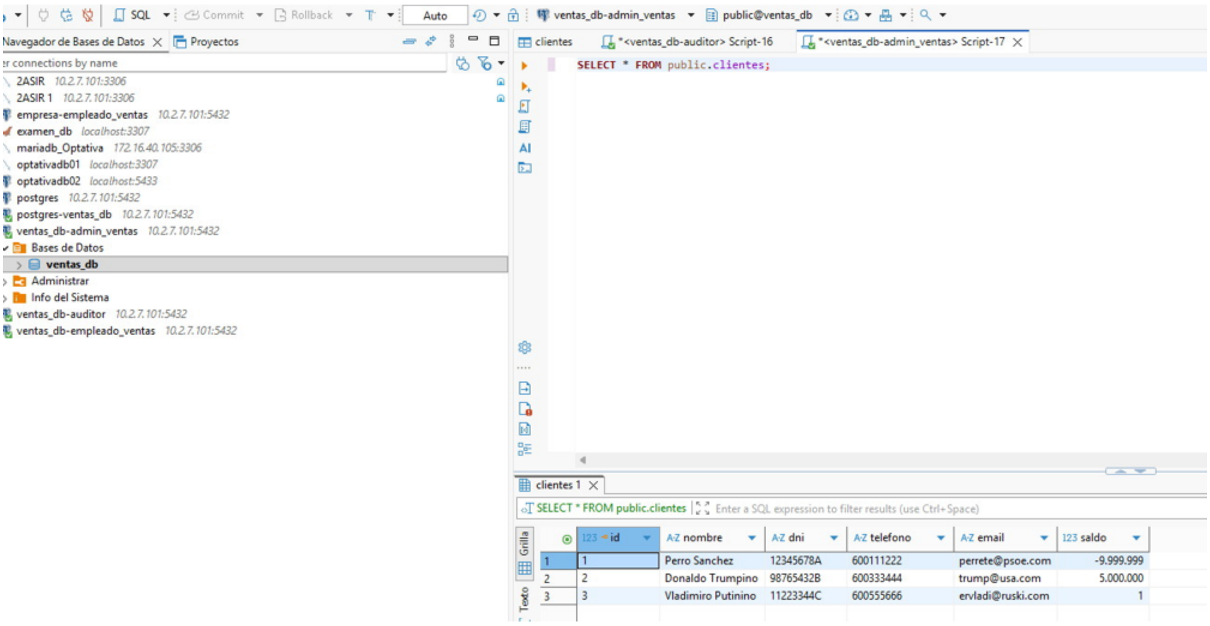
GRANT SELECT ON public.clientes TO ventas_grupo;



```
REVOKE
ventas_db=# GRANT SELECT ON public.clientes TO ventas_grupo;
GRANT
ventas_db=# |
```

- Comprueba si el permiso vuelve a estar disponible.

Funciona


SQL
Commit
Rollback
Auto
ventas_db-admin_ventas
public@ventas_db
Navegador de Bases de Datos
Proyectos
2ASIR 10.2.7.101:3306
2ASIR 1 10.2.7.101:3306
empresa-empleado_ventas 10.2.7.101:5432
examen_db localhost:3307
mariadb_Optativa 172.16.40.105:3306
optativadb01 localhost:3307
optativadb02 localhost:5433
postgres 10.2.7.101:5432
postgres-ventas_db 10.2.7.101:5432
ventas_db-admin_ventas 10.2.7.101:5432
Bases de Datos
ventas_db
Administrar
Info del Sistema
ventas_db-auditor 10.2.7.101:5432
ventas_db-empleado_ventas 10.2.7.101:5432
clientes
*<ventas_db-auditor> Script-16
*<ventas_db-admin_ventas> Script-17
clientes 1
Enter a SQL expression to filter results (use Ctrl+Space)
Grilla
Texto

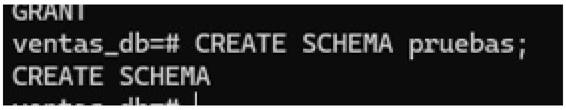
```
SELECT * FROM public.clientes;
```

	id	nombre	dni	telefono	email	saldo
1	1	Perro Sanchez	12345678A	600111222	perrete@psoe.com	-9.999.999
2	2	Donaldo Trumpino	98765432B	600333444	trump@usa.com	5.000.000
3	3	Vladimiro Putinino	11223344C	600555666	ervladi@ruski.com	1

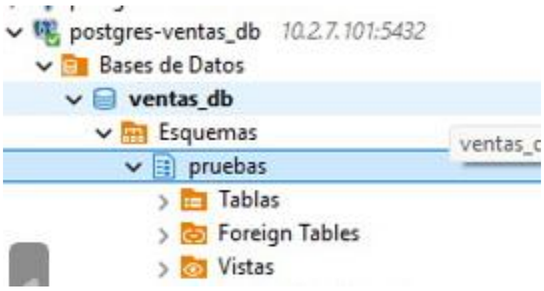
Privilegios sobre esquemas

- Crea un nuevo esquema llamado pruebas:

CREATE SCHEMA pruebas;



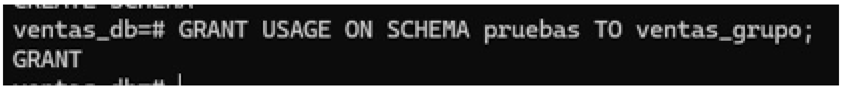
```
GRANT
ventas_db=# CREATE SCHEMA pruebas;
CREATE SCHEMA
```



```
postgres-ventas_db   10.2.7.101:5432
  Bases de Datos
    ventas_db
      Esquemas
        pruebas
          Tablas
          Foreign Tables
          Vistas
```

- Concede a `ventas_grupo` permiso para usar el esquema, pero no para crear objetos

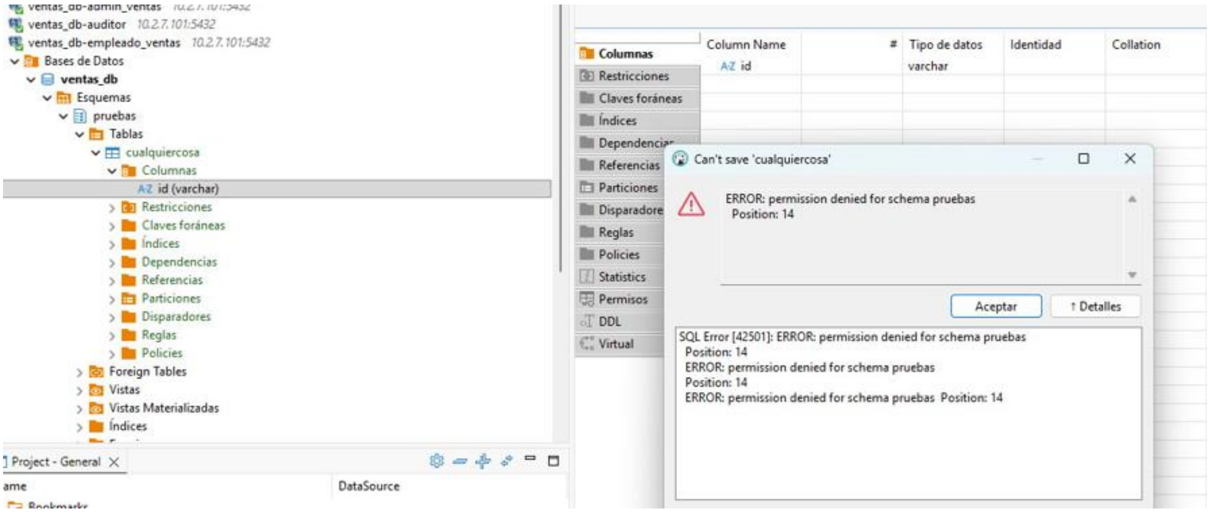
GRANT USAGE ON SCHEMA pruebas TO ventas_grupo;



```
ventas_db=# GRANT USAGE ON SCHEMA pruebas TO ventas_grupo;
GRANT
```

- Intenta crear una tabla dentro del esquema con `empleado_ventas` y verifica el resultado.

No se puede: No tiene permisos.



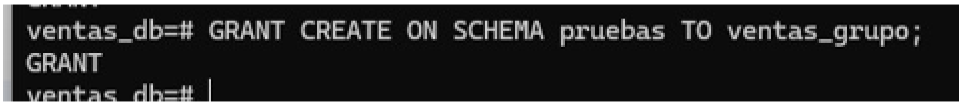
Can't save 'cualquiercosa'

ERROR: permission denied for schema pruebas
Position: 14

```
SQL Error [42501]: ERROR: permission denied for schema pruebas
 Position: 14
 ERROR: permission denied for schema pruebas
 Position: 14
 ERROR: permission denied for schema pruebas  Position: 14
```

- Da permisos para crear

GRANT CREATE ON SCHEMA pruebas TO ventas_grupo;



```
ventas_db=# GRANT CREATE ON SCHEMA pruebas TO ventas_grupo;
GRANT
ventas_db=#
```

- Vuelve a probar y observa la diferencia.

Ahora ya se puede.

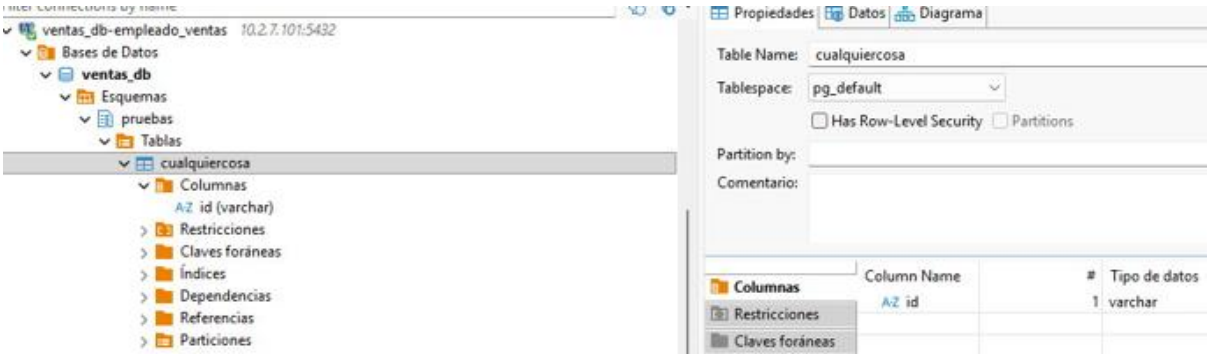


Table Name: cualquiercosa
Tablespace: pg_default
□ Has Row-Level Security □ Partitions
Partition by:
Comentario:

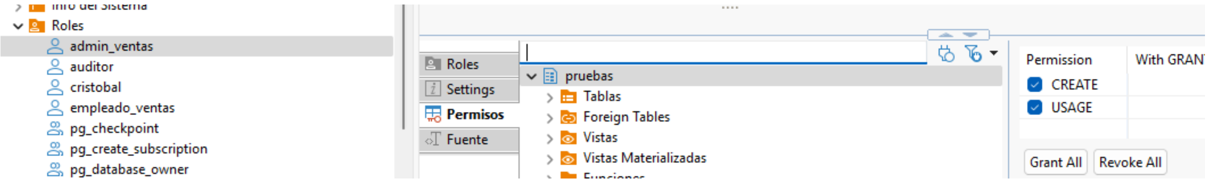
Column Name	#	Tipo de datos
id	1	varchar

Auditoría final de roles y privilegios

- Consulta todos los privilegios otorgados a cada usuario:

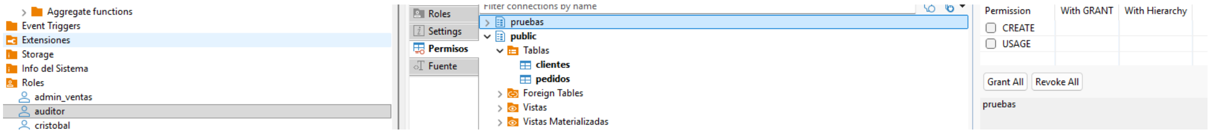
Para esquema “**pruebas**”:

“admin_ventas”



Permission	With GRAN
☑ CREATE	
☑ USAGE	

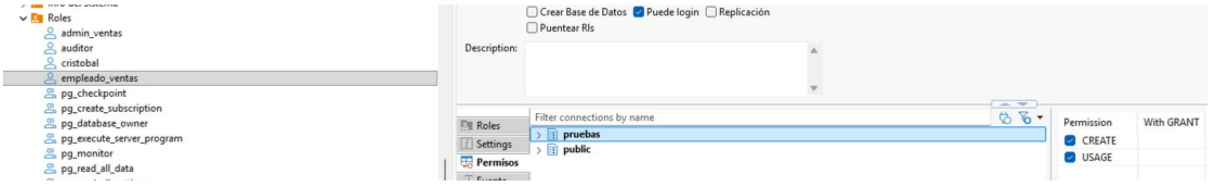
“auditor”



Permission	With GRANT	With Hierarchy
☐ CREATE		
☐ USAGE		

pruebas

“empleado_ventas”



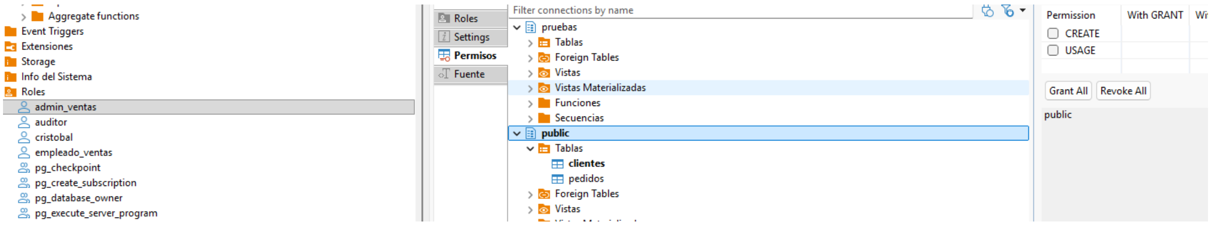
☐ Crear Base de Datos ☑ Puede login ☐ Replicación
☐ Puentear Rls

Description:

Permission	With GRANT
☑ CREATE	
☑ USAGE	

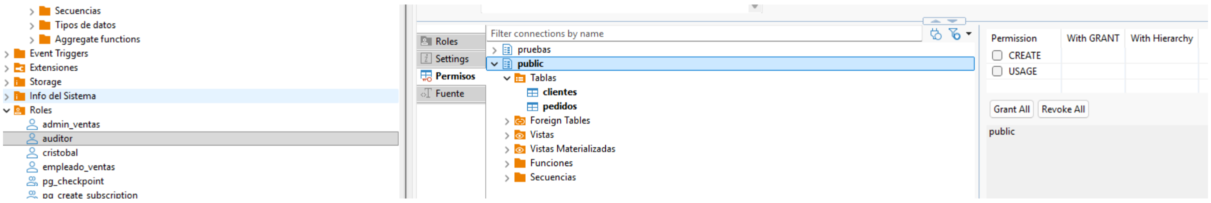
Para esquema “**public**”:

“admin_ventas”:



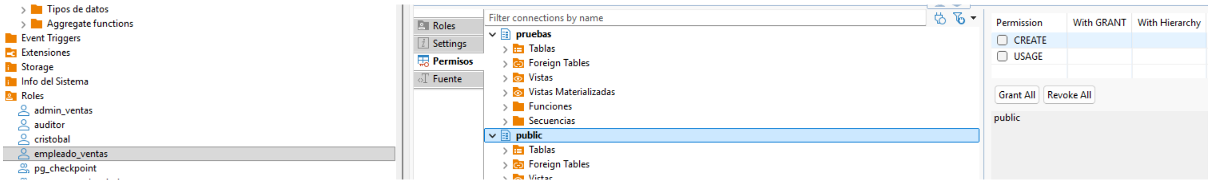
Permission	With GRANT	Wi
□ CREATE		
□ USAGE		

“auditor”:



Permission	With GRANT	With Hierarchy
□ CREATE		
□ USAGE		

“empleado_ventas”:



Permission	With GRANT	With Hierarchy
□ CREATE		
□ USAGE		