

8 DE ENERO DE 2026



## TAREA 4.1 - EXPLORACIÓN DE LA BASE DE DATOS DE ODOO CON PGADMIN (V2)

[SUBTÍTULO DEL DOCUMENTO]

**Nombre:** Daniel García Méndez

**Módulo:** Sistemas de gestión empresarial

**Curso:** 2º DAM

## CONTENIDO

TAREA 4.1 - Exploración de la base de datos de Odoo con pgAdmin.....	2
1. Puesta en marcha y conexión con la BD .....	2
1.1. Comprueba que los contenedores están activos .....	2
1.2. Abre pgAdmin y crea una conexión al servidor PostgreSQL de Odoo .....	2
2. Exploración de la base de datos de Odoo.....	4
2.1. Localiza la base de datos que usas en clase .....	4
2.2. Explora las tablas .....	6
3. Interpretación de nombres y estructuras.....	8
4. Usuarios de la base de datos .....	8
5. Reflexión final .....	10
Webgrafía .....	10

## TAREA 4.1 - EXPLORACIÓN DE LA BASE DE DATOS DE ODOO CON PGADMIN

En esta tarea vas a trabajar directamente con la base de datos de Odoo utilizando **pgAdmin**, un cliente gráfico para PostgreSQL. El objetivo es entender qué datos guarda el ERP, cómo se organizan sus tablas y quién tiene acceso a la información.

La tarea combina pequeñas pruebas prácticas y preguntas breves que te ayudarán a comprender cómo Odoo almacena y gestiona la información internamente.

### 1. PUESTA EN MARCHA Y CONEXIÓN CON LA BD

#### 1.1. COMPRUEBA QUE LOS CONTENEDORES ESTÁN ACTIVOS

Antes de usar pgAdmin, asegúrate de que los contenedores de Odoo y PostgreSQL están arrancados:

```
docker ps
```

Haz una captura donde se vean los contenedores funcionando.

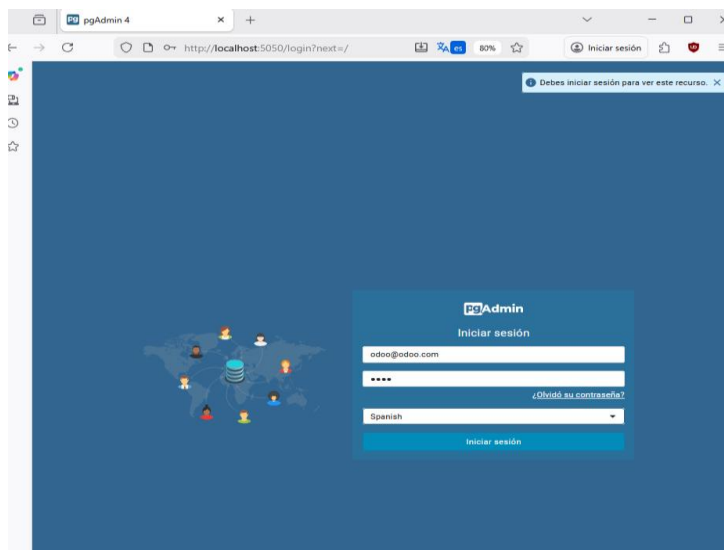
CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
0e18dccc55acd	odoo:18	"/entrypoint.sh odoo"	45 hours ago	Up 4 seconds	0.0.0.0:8069->8069/tcp, [::]:8069->8069/tcp	odoo_app
2ba4976a2973	dpape/pgadmin4	"/entrypoint.sh"	45 hours ago	Up 4 seconds	0.0.0.0:5050->80/tcp, [::]:5050->80/tcp	pgadmin_container
46e04b9301d8	postgres:15	"/docker-entrypoint.s..."	45 hours ago	Up 4 seconds	0.0.0.0:5432->5432/tcp, [::]:5432->5432/tcp	odoo_db
1277a510fd9c	portainer/portainer-ce:latest	"/portainer"	4 weeks ago	Up 4 seconds	0.0.0.0:9000->9000/tcp, [::]:9000->9000/tcp, 0.0.0.0:9443->9443/tcp, [::]:9443->9443/tcp	portainer

#### 1.2. ABRE PGADMIN Y CREA UNA CONEXIÓN AL SERVIDOR POSTGRESQL DE ODOO

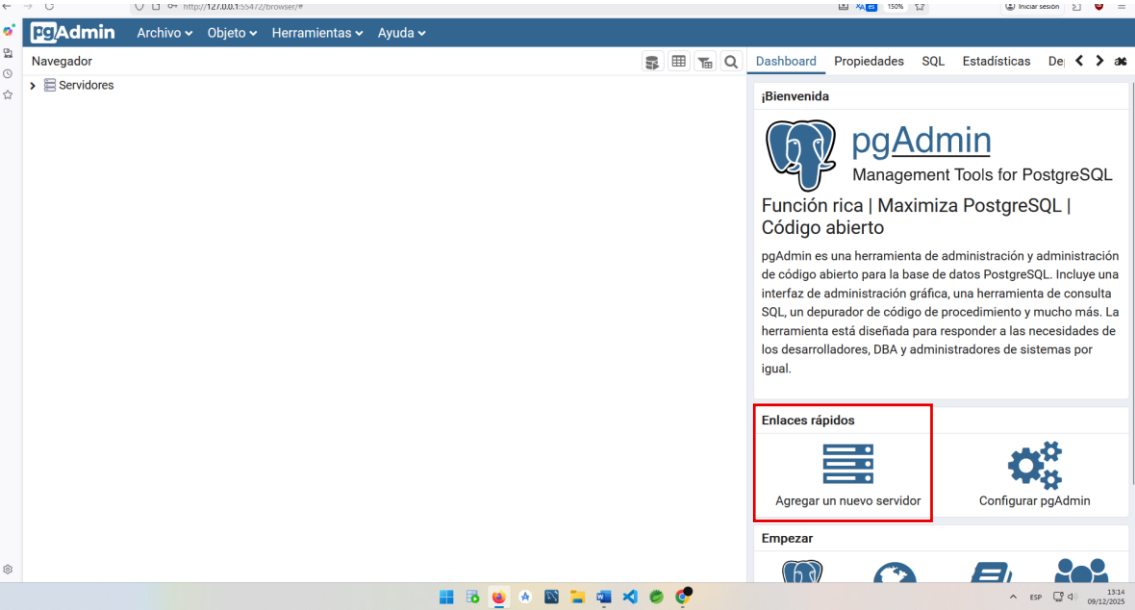
Añade un nuevo servidor con estos datos:

- **Nombre del servidor:** Odoo-Docker
- **Host:** localhost
- **Puerto:** 5432
- **Usuario:** odoo
- **Contraseña:** odoo

**Captura obligatoria:** pantalla de configuración o pantalla donde se vea el servidor conectado.



Iremos a la opción señalada para agregar un servidor y pondremos los datos anteriormente vistos en el apartado anterior:



**Crear - Servidor**

General Conexión SSL Túnel SSH Avanzado

Nombre:

Grupo de servidores:

Antecedentes: ☐

Primer plano: ☐

¿Conectar ahora?: ☒

Comentarios:

**Crear - Servidor**

General Conexión SSL Túnel SSH Avanzado

Nombre/dirección del host:

Puerto:

Base de datos de mantenimiento:

Nombre de usuario:

Contraseña:

¿Guardar contraseña?: ☒

Papel:

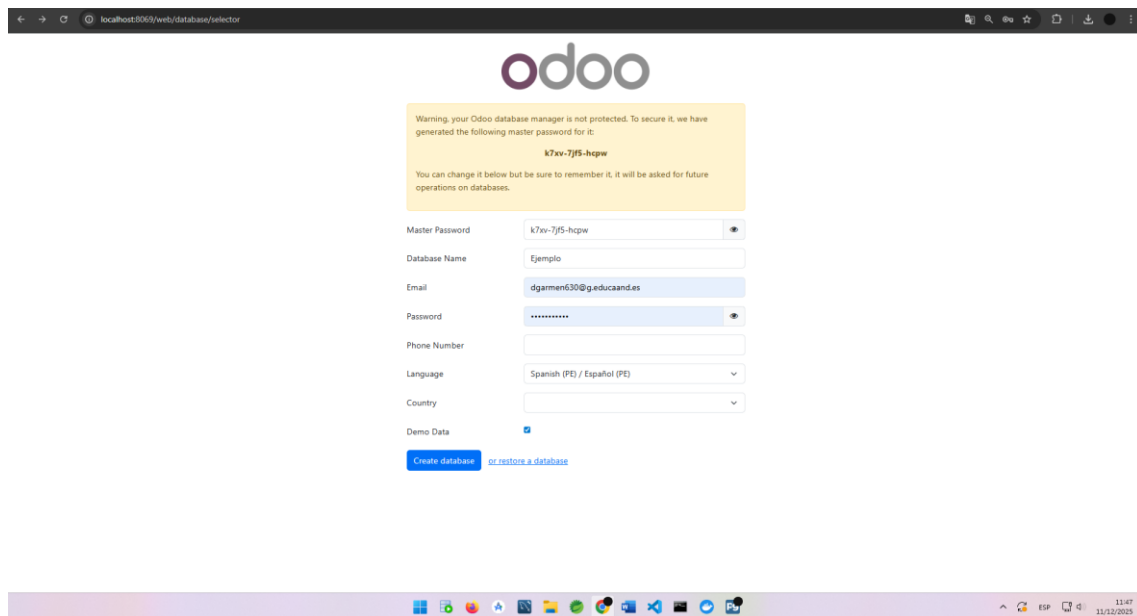
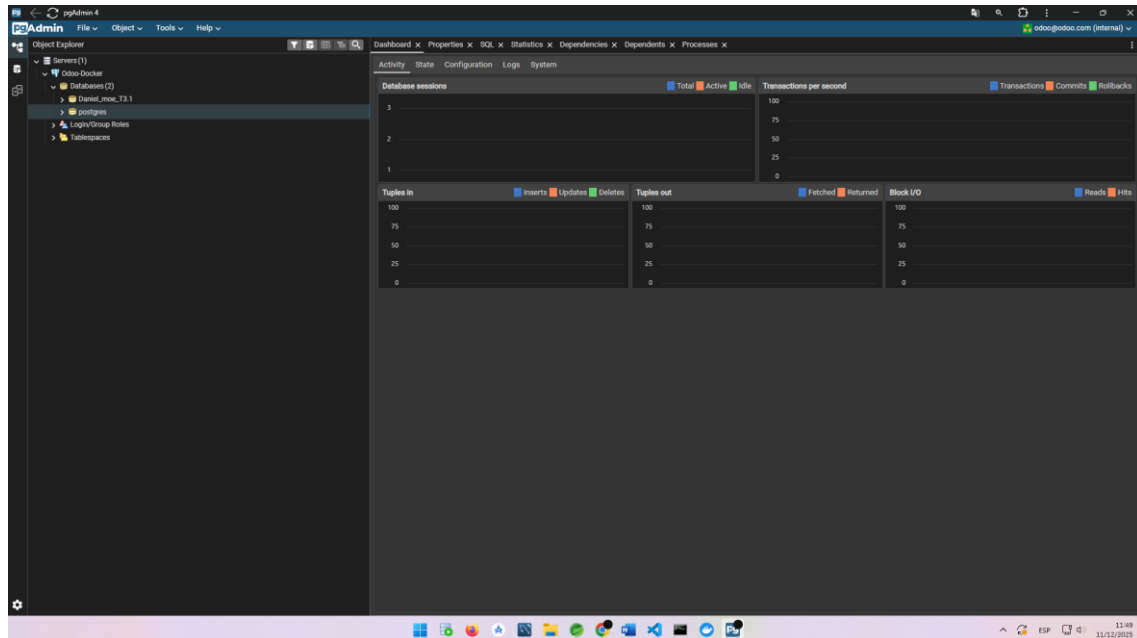
Servicio:

## 2. EXPLORACIÓN DE LA BASE DE DATOS DE ODOO

### 2.1. LOCALIZA LA BASE DE DATOS QUE USAS EN CLASE

Despliega:

Servers → Odoo-Docker → Databases



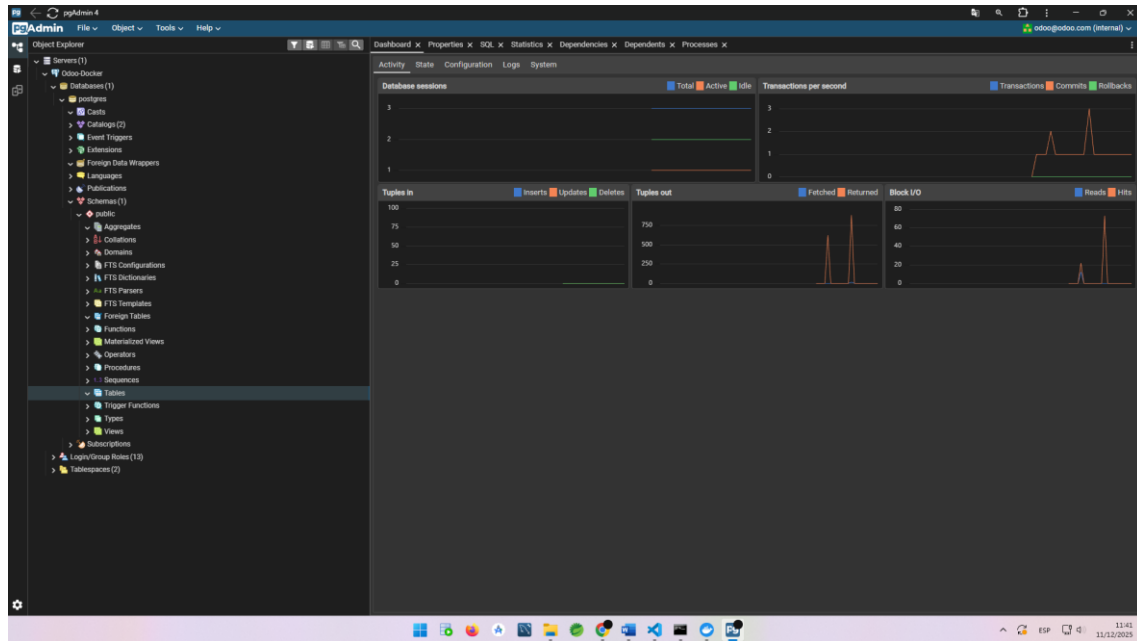
1. ¿Qué bases de datos aparecen?

Aparece la base de datos de postgres, esta es la base de datos en mantenimiento por defecto y la base de datos de Daniel\_moe\_T3.1 de la tarea anterior.

2. ¿Cuál corresponde a tu instalación de Odoov?

La base de Datos Daniel\_moe\_T3.1 y la de prueba.

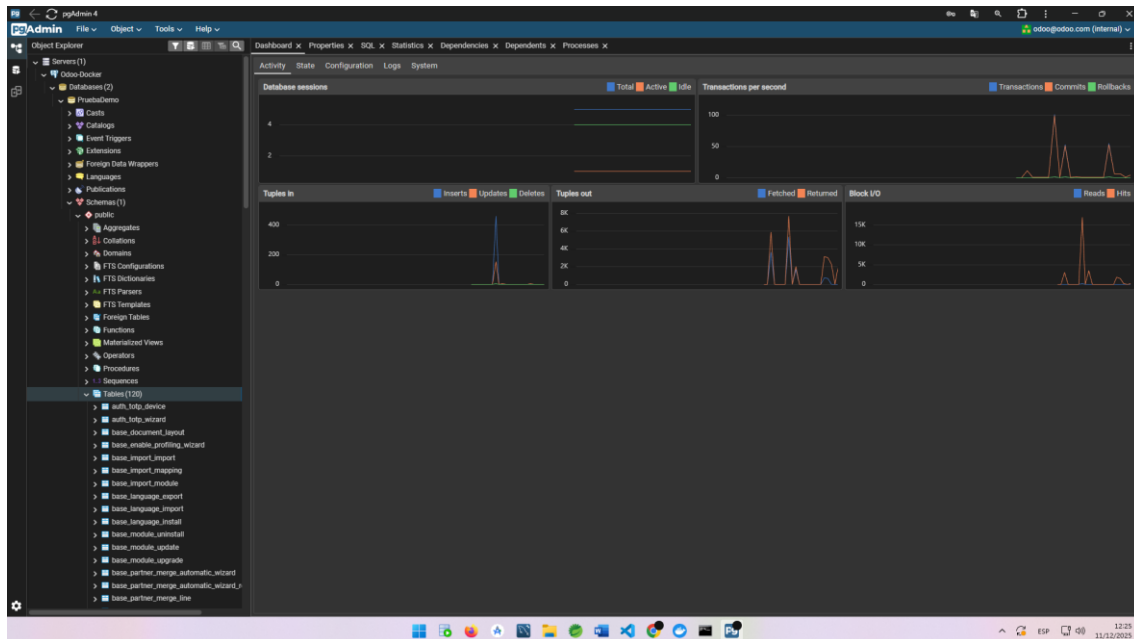
Incluye una captura.



## 2.2. EXPLORA LAS TABLAS

Ve a:

Schemas → public → Tables



Responde:

1. ¿Cuántas tablas aparecen en la base de datos?  
Aparecen 120 tablas.
2. Elige **dos módulos de Odoo** (por ejemplo, Ventas y Contactos) y localiza **una tabla de cada uno**.

Explica por qué sabes que esas tablas pertenecen al módulo elegido (usa pistas como el pre-fijo: *sale\_*, *res\_*...).

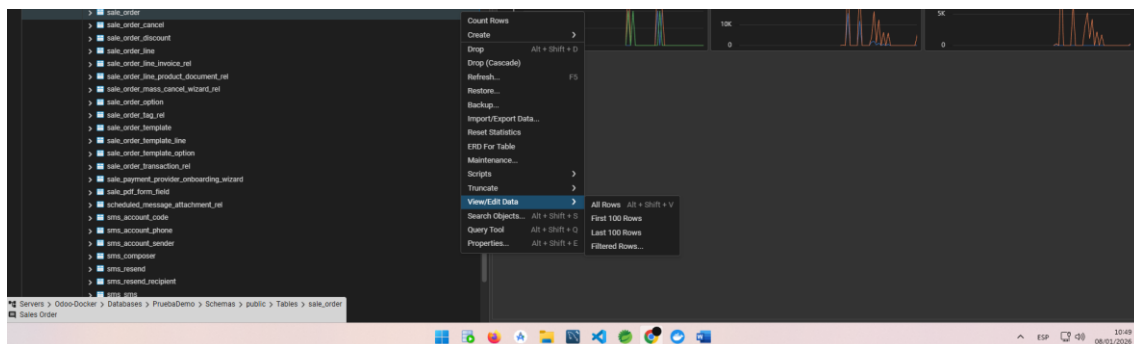
Incluye una captura de la lista de tablas.

## 2.3. Consulta datos desde pgAdmin

Elige **una tabla que contenga datos** (por ejemplo *res\_partner* o *product\_template*).

•

Abre **View/Edit Data** → *All Rows*.



## Captura de registros de la tabla SALE\_ORDER

id	campaign_id	source_id	medium_id	company_id	partner_id	journal_id	partner_invoice_id	partner_shipping_id	fiscal_position_id	payment_term_id	price_list_id	currency_id	user_id	team_id	create_uid	write_uid	access_token
9	9	4	10	4	1	11	[null]	32	[null]	[null]	[null]	[null]	1	6	4	1	[null]
10	4	10	4	1	11	[null]	32	32	[null]	[null]	[null]	[null]	1	6	4	1	46c5d6c5-447b-4d0f-ba58-045268c
11	4	10	4	1	11	[null]	32	32	[null]	[null]	[null]	[null]	1	6	4	1	[null]
12	4	10	4	1	11	[null]	32	32	[null]	[null]	[null]	[null]	1	6	4	1	3792114e-ce91-403b-ae53-696222c
13	4	10	4	1	11	[null]	32	32	[null]	[null]	[null]	[null]	1	6	4	1	[null]
14	4	10	4	1	11	[null]	32	32	[null]	[null]	[null]	[null]	1	6	1	1	[null]
15	4	10	4	1	11	[null]	32	32	[null]	[null]	[null]	[null]	1	6	1	1	[null]
16	4	10	4	1	11	[null]	32	32	[null]	[null]	[null]	[null]	1	6	1	1	50df90e1-816a-41a6-a849-057456a
17	4	10	4	1	11	[null]	32	32	[null]	[null]	[null]	[null]	1	6	1	1	[null]
18	4	10	4	1	11	[null]	32	32	[null]	[null]	[null]	[null]	1	6	1	1	[null]
19	[null]	[null]	[null]	1	8	[null]	8	8	[null]	[null]	[null]	[null]	1	2	1	1	19852b4c-d5a1-468b-a3e4-425a93
20	[null]	[null]	[null]	1	8	[null]	8	8	[null]	[null]	[null]	[null]	1	2	1	1	[null]

- ¿Cuántos registros aparecen?  
En total salen 20 registros.
- Describe tres columnas que te llamen la atención:  
Id, campaign\_id y company\_id son identificadores que se usan para relacionar tablas.

### 1. id

- **Tipo de dato:** integer (clave primaria)
- **Qué representa:** Es el identificador único de cada registro en la tabla resource\_calendar\_leaves. En el contexto de Odoo, esta tabla gestiona las ausencias o días no laborables en los calendarios de recursos (como empleados o máquinas). El id permite distinguir cada ausencia registrada.

### 2. campaign\_id

- **Tipo de dato:** integer
- **Qué representa:** Referencia a una campaña de marketing (utm.campaign). Se usa para rastrear el origen de una acción comercial o venta. Aunque esta tabla está relacionada con calendarios, Odoo permite asociar ausencias a campañas si están vinculadas a procesos como vacaciones por incentivos o planificación comercial.

### 3. company\_id

- **Tipo de dato:** integer
- **Qué representa:** Identifica a qué empresa pertenece el recurso o la ausencia registrada. En entornos multiempresa, esta columna es clave para separar la información entre compañías dentro del mismo sistema Odoo. Apunta a la tabla res.company.
- nombre de la columna
- tipo de dato
- qué información representa dentro del ERP



Incluye una captura de la tabla con algunos registros.

### 3. INTERPRETACIÓN DE NOMBRES Y ESTRUCTURAS

Los nombres de las tablas en PostgreSQL provienen de las clases Python de Odoo.

Responde:

#### 1. ¿Cómo se transforma el nombre de una clase como `sale.order` para generar la tabla correspondiente?

Se quita el punto y se pone guion bajo: `sale.order` → `sale_order`

#### 2. Pon dos ejemplos más a partir de nombres de módulos que ya hayas explorado.

- `res.partner` → `res_partner` → tabla de contactos (clientes, proveedores, etc.)
- `sale.order.line` → `sale_order_line` → líneas de cada pedido (productos, precios, cantidades)

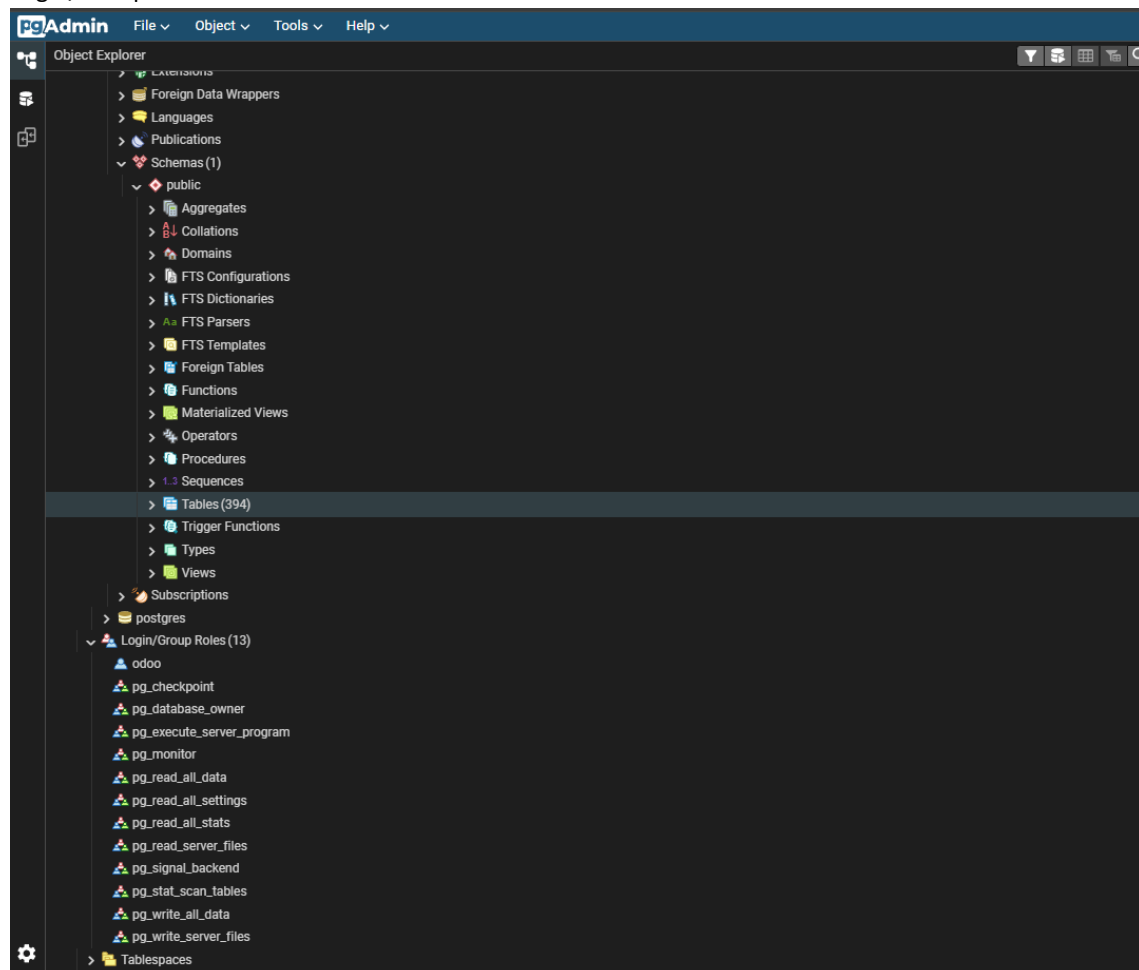
#### 3. Explica por qué este sistema de nombres facilita identificar los datos del ERP.

### 4. USUARIOS DE LA BASE DE DATOS

Ve a:

Login/Group

Roles



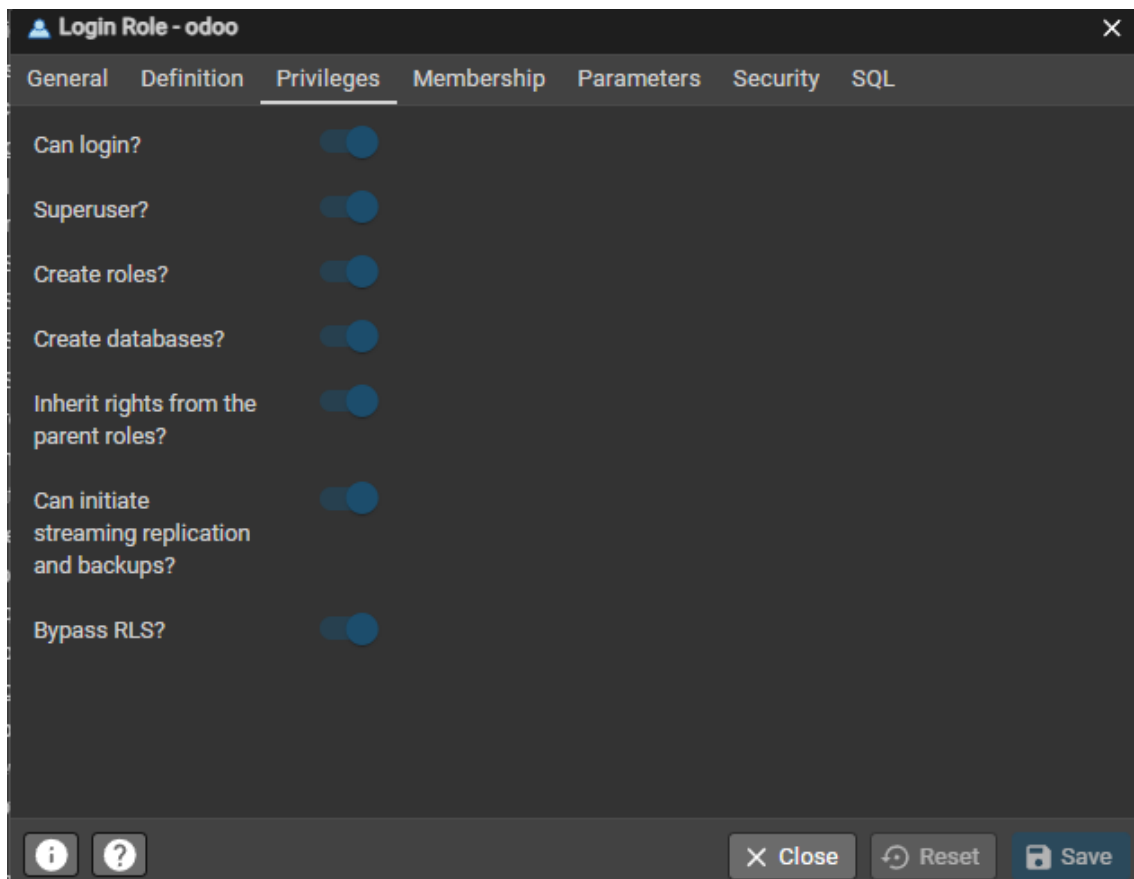
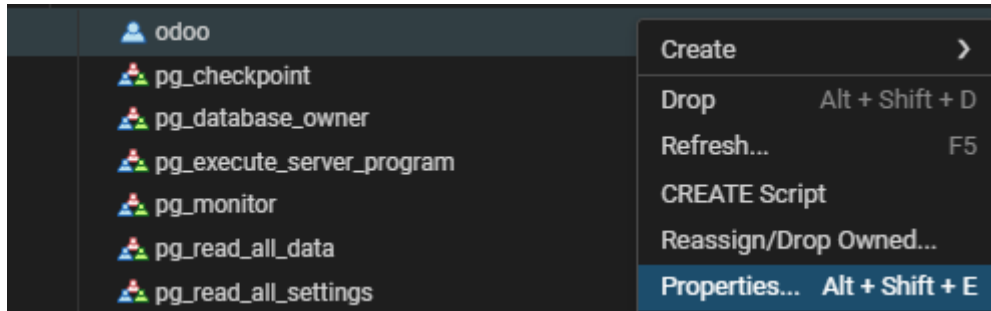
Responde:

1. ¿Qué usuarios existen en el servidor PostgreSQL?

Solo existe uno llamado odoo, lo demás son roles.

2. ¿Cuál es el usuario que utiliza Odoo para conectarse?

3. Abre las propiedades del usuario **odoo**:



- ¿Puede crear bases de datos? ¿Tiene permisos de login?

Como podemos ver en la pantalla, este usuario puede crear las bases de datos y también permite hacer login.

#### **4. Explica por qué un ERP necesita un usuario propio para la base de datos.**

Porque así el ERP trabaja con su propio usuario y no toca nada que no debe. Ese usuario solo tiene los permisos justos para crear tablas, leer datos y guardar cambios. De esta forma:

- **se evita romper la base por accidente**
- **se controla mejor la seguridad**
- **y todo lo que hace el ERP queda separado del resto de cosas del servidor.**

Incluye una captura donde se vea la lista de roles.

#### **5. REFLEXIÓN FINAL**

Responde con tus palabras:

##### **1. ¿Qué parte te ha servido más para entender cómo funciona un ERP por dentro?**

Ver las tablas. Ahí es donde ves realmente qué guarda el sistema y cómo todo lo que haces acaba ahí.

##### **2. ¿Te ha sorprendido la cantidad de tablas que utiliza Odoo? ¿Por qué?**

Sí, porque desde fuera Odoo parece más sencillo, pero por dentro tiene un montón de cosas y cada módulo mete las suyas.

##### **3. ¿Qué relación ves entre lo que ocurre en la interfaz de Odoo y los datos que has visto en las tablas?**

Muy directa, ya que lo que haces en la pantalla se convierte en datos en esas tablas. La interfaz es la parte bonita, lo importante está guardado ahí.

#### **WEBGRAFÍA**

<https://experts-deny-b9a.craft.me/xOJQn4khXm30Ob>

<https://openwebinars.net/blog/pgadmin-para-novatos-como-gestionar-bases-de-datos-sin-complicaciones/>