TEMA 4 - ORGANIZACIÓN, CONSULTA Y MODIFICACIÓN DE LA INFORMACIÓN

TEMA 4 - ORGANIZACIÓN, CONSULTA Y MODIFICACIÓN DE LA INFORMACIÓN

Índice de contenidos

1. Acceso a la base de datos con PGAdmin

La base de datos de un sistema ERP es muy grande. Los datos se almacenan en las tablas de la aplicación, las vistas de las tablas y cualquier otro elemento del cual necesitemos mostrar los datos o guardarlos.

Cualquier administrador del ERP puede acceder a la base de datos de Odoo directamente. Para ello solo necesitaremos conectar con ella a través de un cliente de la base de datos. Principalmente tenemos dos alternativas:

- **psql**: si sabemos manejar la línea de comandos, con este cliente podemos acceder y manipular cualquier base de datos Postgres.
- **PgAdmin** (en cualquiera de sus versiones): es un cliente gráfico de Postgres. Se trata de un software multiplataforma con el que podremos ver gráficamente las bases de datos de nuestro Postgres.

1.1. Instalación y configuración de pgAdmin

pgAdmin es un software gratuito distribuido bajo licencia PostgreSQL. Podemos encontrarlo en los repositorios oficiales, concretamente en la **página de descargas de pgAdmin**.

En nuestro caso instalaremos **la versión correspondiente a nuestro sistema operativo (Windows, macOS o Linux)**.

La opción "Container" que aparece en la web está pensada para instalaciones avanzadas donde también se desea ejecutar pgAdmin dentro de Docker, algo que **no necesitamos en este módulo**.

Una vez descargado el instalador, lo ejecutamos y abrimos pgAdmin.

¿Para qué sirve pgAdmin?

pgAdmin es una herramienta gráfica que nos permite **administrar y trabajar con bases de datos PostgreSQL** de manera visual.

Gracias a pgAdmin podemos:

- Ver todas las bases de datos creadas en un servidor PostgreSQL.
- Consultar tablas, columnas y tipos de datos.
- Ejecutar consultas SQL sin tener que usar la terminal.
- Crear nuevas bases de datos, tablas y usuarios.
- Importar y exportar datos.
- Realizar copias de seguridad (backups) y restauraciones.
- Analizar la estructura interna de las bases de datos, algo muy útil para comprender cómo Odoo organiza la información.

Esta herramienta facilita mucho el aprendizaje, ya que permite al alumnado observar directamente la relación entre lo que ocurre en Odoo y los datos almacenados en PostgreSQL.

Conexión de pgAdmin con PostgreSQL dentro del contenedor de Odoo

Odoo no incluye su propia base de datos interna. Siempre utiliza un contenedor adicional que ejecuta PostgreSQL.

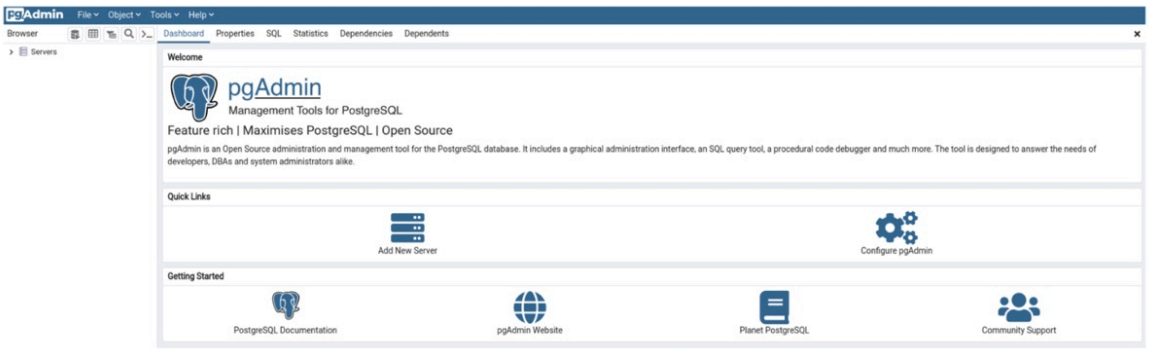
En el archivo `docker-compose.yml` suele aparecer algo parecido a:

```
db:
  image: postgres:15
  environment:
    POSTGRES_DB: postgres
    POSTGRES_USER: odoo
    POSTGRES_PASSWORD: odoo
  ports:
    - "5432:5432"
```

Ese mapeo de puertos permite que pgAdmin, instalado en nuestro equipo, pueda conectarse sin problemas.

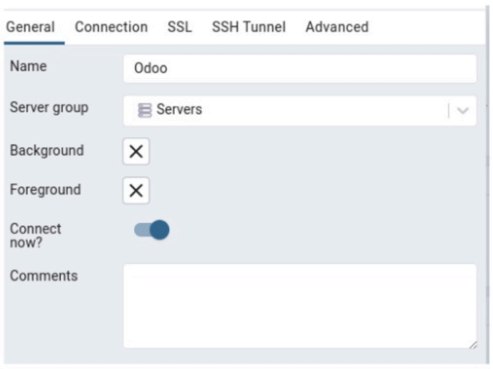
Añadir un nuevo servidor en pgAdmin

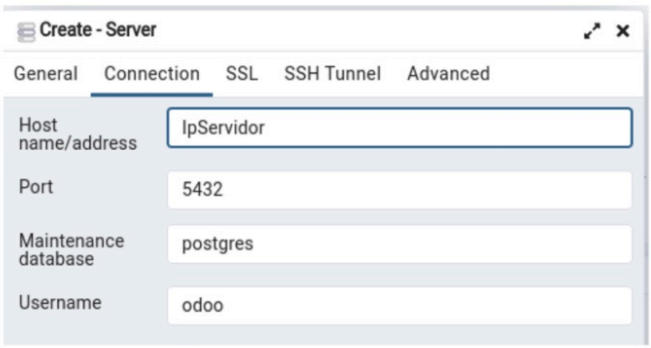
Cuando abrimos pgAdmin veremos la ventana principal con varias opciones. Desde el menú lateral añadimos un **nuevo servidor** y configuramos los datos necesarios.



En la pestaña **Connection** rellenamos:

- **Nombre**: nombre identificativo para la conexión, por ejemplo *Odoo-Docker*.
- **Host name/address**: `localhost`
  (pgAdmin se conecta al puerto publicado por Docker).
- **Port**: `5432`.
- **Maintenance database**: `postgres`.
- **Username**: `odoo`.
- **Password**: `odoo`.





Guardamos la conexión para reutilizarla siempre que queramos acceder a la base de datos.

Notas importantes cuando trabajamos con Docker

• El contenedor debe estar en ejecución

Podemos comprobarlo con:

```
docker ps
```

• Debemos tener publicado el puerto 5432

Si no aparece `"5432:5432"` en el `docker-compose.yml`, pgAdmin no podrá conectarse desde fuera del contenedor.

La solución es añadirlo, guardar y reiniciar:

```
docker compose down
docker compose up -d
```

• La conexión siempre se hace desde el equipo anfitrión

En contextos más avanzados se podría usar la IP del contenedor, pero en este módulo utilizaremos siempre `localhost`.

¿Sabías que...?

Cada vez que creamos una base de datos nueva en Odoo (por ejemplo, una empresa distinta), aparece automáticamente en la lista de pgAdmin.

Esto permite revisar estructuras, consultar tablas y realizar copias de seguridad.

1.2. Los datos de Odoo

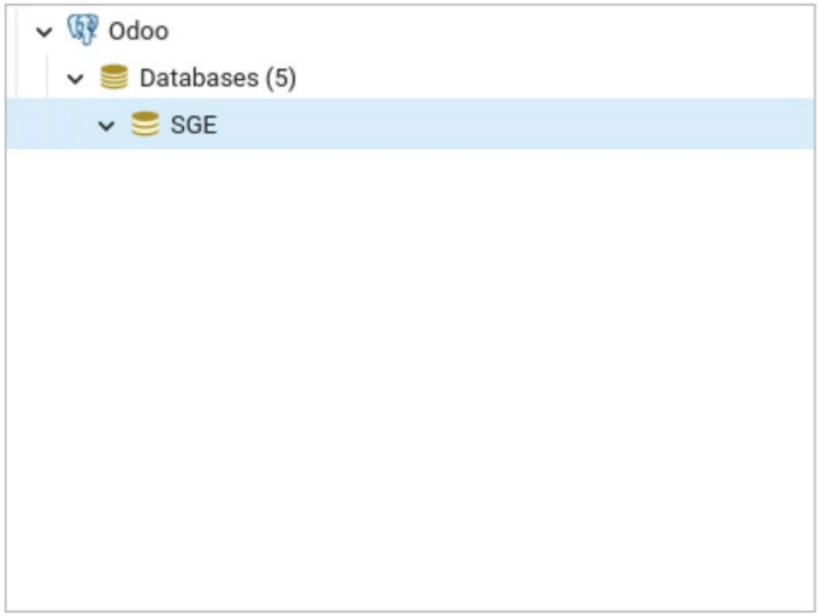
Para poder trabajar con Odoo es importante entender **qué datos guarda**, **dónde los guarda** y **cómo podemos consultarlos**.

Toda la información del ERP –clientes, productos, facturas, usuarios, etc.– se almacena en **PostgreSQL**, el sistema gestor de bases de datos que utiliza Odoo.

Desde pgAdmin podremos explorar esa información, consultarla, ver cómo se organiza y, si fuese necesario, modificarla.

1.2.1. Las tablas de la base de datos

Al abrir pgAdmin y conectarnos al servidor que hemos configurado, veremos las bases de datos que tenemos disponibles. Cada vez que creamos una base de datos en Odoo, esta aparece también en PostgreSQL.



Si desplegamos la base de datos y entramos en **Schemas** → **public** → **Tables**, veremos **todas las tablas** que utiliza Odoo.

Un ERP no es una aplicación pequeña: un simple módulo puede crear tres o cuatro tablas nuevas, y al instalar varios módulos el número de tablas puede llegar a ser enorme.

Desde pgAdmin podemos:

- ver las bases de datos existentes,
- crear nuevas,
- eliminar bases de datos,
- crear tablas,
- borrar tablas,
- consultar la información almacenada en ellas.

1.2.2. Identificar las tablas

Odoo organiza sus datos utilizando clases Python. Cada clase que existe en el código del ERP se transforma en una tabla dentro de PostgreSQL. A su vez, cada atributo de esa clase se convierte en una columna de la tabla.

Este detalle nos permite localizar con rapidez en qué tabla se guarda cierta información. Para ello debemos recordar dos ideas:

1. Cómo se nombran las clases Python de Odoo

- Siempre en minúscula.
- Las palabras compuestas usan guion bajo.
- La estructura es:

  nombre_del_modulo.nombre_de_la_clase

- En la base de datos, los puntos se convierten en guiones bajos.

Por ejemplo:

```
website.menu    →   website_menu
sale.order      →   sale_order
product.template → product_template
```

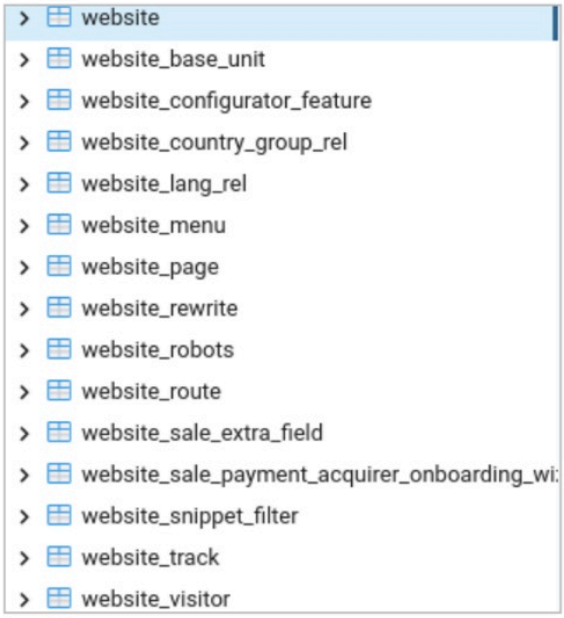
2. Cómo se nombran los atributos de las clases

- También van siempre en minúscula.
- Siguen el mismo estilo: palabras unidas con guion bajo.
- Si el atributo se guarda en la base de datos, aparece como una columna con el mismo nombre.

Ejemplo

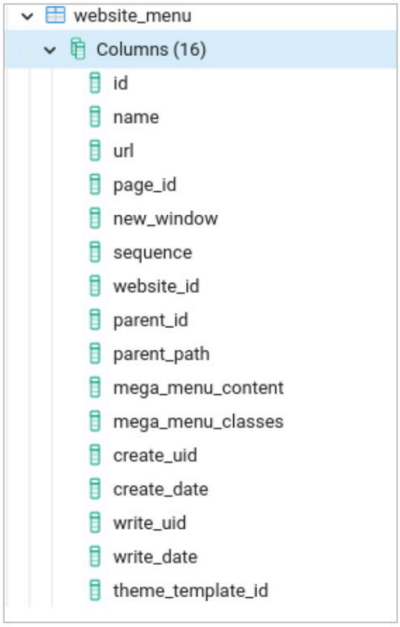
Todas las tablas relacionadas con el módulo *website* empiezan por:

```
website_
```



De esta manera resulta muy fácil localizar las tablas que pertenecen a un módulo concreto.

Si seleccionamos una tabla, pgAdmin nos muestra las columnas que contiene. Si además elegimos la opción **View/Edit Data**, podemos ver los datos que hay dentro. Esto nos ayuda a entender la estructura interna de Odoo.



Por ejemplo, si consultamos la tabla de productos veremos exactamente los productos que tenemos dados de alta en el ERP. Lo que estamos haciendo realmente es ejecutar una consulta SQL.

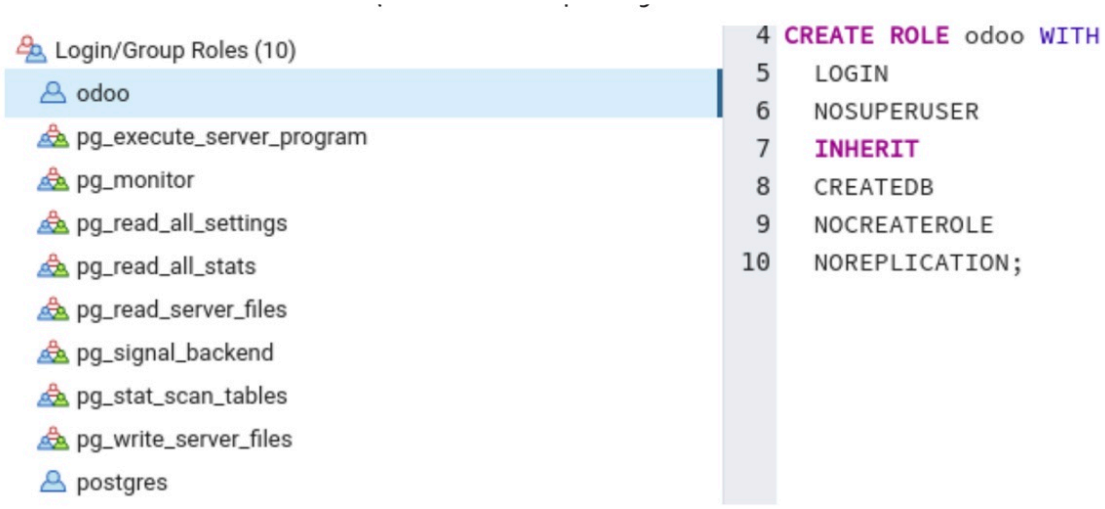
1.2.3. Los usuarios del SGBD

Además de las tablas, PostgreSQL también gestiona **usuarios**, que son quienes pueden acceder o no a las bases de datos.

En pgAdmin, dentro de la conexión que hemos creado, podemos abrir el apartado:

```
Login/Group Roles
```

Ahí veremos los usuarios existentes en la base de datos.

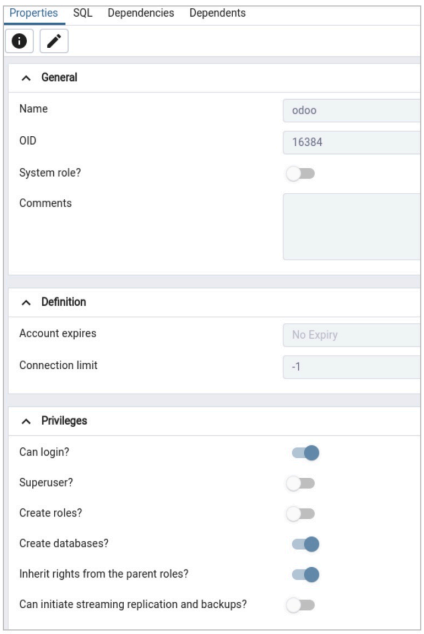


El usuario **odoo** es el que utiliza el ERP para conectarse a PostgreSQL y trabajar con las tablas.

Desde este mismo menú podemos:

- crear usuarios nuevos,
- modificarlos,
- asignar permisos,
- revisar qué operaciones puede realizar cada uno.

Si seleccionamos un usuario y abrimos **Properties**, aparecen todas sus características (si puede iniciar sesión, si puede crear bases de datos, si hereda permisos, etc.).



Si pulsamos en el icono del **lápiz**, podremos editarlas.