



BASES DE DATOS.

GRUPO: 01.

TAREA 2: "GUIA DE INICIO PARA POSTGRESQL".

ALUMNO: AGUILAR BUSTILLOS ALAN RODRIGO.

PROFESOR: ING. FERNANDO ARREOLA FRANCO.

SEMESTRE 2024-1

PostgreSQL es un sistema gestor de bases de datos relacionales Orientadas a Objetos, es derivado del paquete Postgres escrito en Berkeley, y actualmente es el gesto de bases de datos más avanzado del mundo con casi una década de desarrollo. Este ofrece un control de concurrencia multi-version, que soporta casi toda la sintaxis de SQL, en conjunto con un gran número de lenguajes de programación.

Ventajas de PostgreSQL:

1. **Robustez y Confiabilidad:** PostgreSQL es conocido por su alta confiabilidad y estabilidad. Es capaz de manejar grandes volúmenes de datos y cargas de trabajo intensivas.
2. **Características Avanzadas:** PostgreSQL es altamente extensible y ofrece muchas características avanzadas, como soporte para tipos de datos personalizados, funciones y procedimientos almacenados, y una amplia variedad de índices.
3. **Escalabilidad:** PostgreSQL es escalable y puede manejar aplicaciones de pequeña a gran escala. Puedes implementar clústeres de réplica para distribuir la carga y mejorar la disponibilidad.
4. **Soporte de Lenguaje SQL:** PostgreSQL cumple en gran medida con los estándares SQL, lo que facilita la migración desde otras bases de datos relacionales.
5. **Soporte para JSON y NoSQL:** PostgreSQL ofrece soporte nativo para tipos de datos JSON y hstore, lo que lo hace adecuado para aplicaciones que requieren almacenamiento de datos semiestructurados o NoSQL.
6. **Comunidad Activa:** PostgreSQL cuenta con una comunidad de usuarios y desarrolladores muy activa, lo que significa que es probable que encuentres una amplia gama de recursos y soporte en línea.
7. **Licencia de Código Abierto:** PostgreSQL se distribuye bajo la Licencia de PostgreSQL, una licencia de código abierto que permite su uso y modificación gratuitos.

Desventajas de PostgreSQL:

1. Rendimiento: Aunque PostgreSQL es muy potente, puede no ser tan rápido en algunas operaciones comparado con bases de datos específicas para ciertas cargas de trabajo, como MySQL o MongoDB.
2. Curva de Aprendizaje: PostgreSQL puede tener una curva de aprendizaje pronunciada, especialmente para aquellos que no están familiarizados con las bases de datos relacionales o SQL.
3. Requisitos de Recursos: Para funcionar eficientemente, PostgreSQL puede requerir más recursos de hardware que algunas otras bases de datos, lo que puede aumentar los costos de infraestructura.
4. Herramientas de Administración: Aunque hay muchas herramientas de administración disponibles para PostgreSQL, algunas de ellas pueden no ser tan maduras o fáciles de usar como las disponibles para sistemas más populares como MySQL.
5. No es la Mejor Opción para Ciertas Cargas de Trabajo Específicas: Para cargas de trabajo muy específicas, como aplicaciones altamente transaccionales o de lectura intensiva, otras bases de datos pueden ser más adecuadas.

Guia de comandos básicos de PostgreSQL (Linux):

Ingresar a PostgreSQL por terminal:

```
psql -U nombreUsuario -W -h iphost nombreBaseDeDatos
```

Parámetros:

-U es el usuario de la base

-W mostrará el prompt de solicitud de password

-h IP del servidor de la base de datos en caso nos conectemos remotamente sino

bastaría con poner localhost

`\l` ---> Te muestra las bases de datos existentes.

`\d` ----> Te muestra las relaciones (tablas, secuencias, etc.) existentes en la base de datos.

`\d [nombre_tabla]` ---> Para ver la descripción (nombre de columnas, tipo de datos, etc.) de una tabla.

`\c [nombre_bd]` ---> Para conectarte a otra base de datos.

`SHOW search_path;` ---> Para ver la ruta de búsqueda actual.

`SET search_path TO [nombre_esquema];` ---> Para actualizar la ruta de búsqueda.

`\q` ----> Para salir de psql

Para crear una base de datos:

CREATE DATABASE nombreBD;

Para eliminar una base de datos:

DROP DATABASE nombreBD;

Para renombrar una base de datos:

ALTER DATABASE nombreBD;

Para conectarse a una base de datos:

\c nombreBD;

Para crear un nuevo usuario:

CREATE USER nombre_de_usuario WITH PASSWORD 'contraseña';

Para otorgar permisos:

GRANT CONNECT ON DATABASE nombre_de_base_de_datos TO nombre_de_usuario;

GRANT USAGE, SELECT, INSERT, UPDATE, DELETE ON ALL TABLES IN SCHEMA public TO nombre_de_usuario;

REFERENCIAS BIBLIOGRAFICAS:

- UNEWEB. Accedido el 11 de septiembre de 2023. [En línea]. Disponible: <https://uneweb.edu.ve/tuto-docs/guia-postgres-l.pdf>