UNAM FACULTAD DE INGENIERIA

BASES DE DATOS

Tarea 02

Martínez García José Eduardo

- **¿Qué requiero para conectarme a una BD?**

1. **Controlador de Base de Datos:**

Es necesario un controlador o driver específico para el sistema de gestión de bases de datos (SGBD) que se esté utilizando, como MySQL, PostgreSQL, Oracle, etc. Este controlador permite la comunicación entre la aplicación y la base de datos [1].

2. **Cadena de Conexión:**

La cadena de conexión es una cadena de texto que contiene la información necesaria para establecer la conexión, como el nombre del servidor, el nombre de la base de datos, el puerto, y las credenciales de autenticación (usuario y contraseña) [2].

3. **Credenciales de Autenticación:**

Se deben proporcionar un nombre de usuario y una contraseña válidos para acceder a la base de datos. Estas credenciales deben tener los permisos necesarios para realizar las operaciones requeridas [3].

4. **Librerías o APIs:**

Dependiendo del lenguaje de programación que se esté utilizando, se necesitarán librerías o APIs específicas para manejar la conexión y las operaciones con la base de datos. Por ejemplo, en Java se utiliza JDBC, en Python se puede usar PyMySQL o SQLAlchemy, y en PHP se utiliza PDO o MySQLi [4].

5. **Configuración de Red:**

Si la base de datos no está alojada localmente, es necesario asegurarse de que la red esté configurada correctamente para permitir la conexión remota, incluyendo la configuración de firewalls y la apertura de puertos necesarios [5].

6. **Software de Gestión de Bases de Datos:**

Es recomendable tener instalado un software de gestión de bases de datos, como phpMyAdmin para MySQL o pgAdmin para PostgreSQL, para facilitar la administración y la ejecución de consultas [6].

- **Permisos a nivel sistema y objeto**

  - **Objeto**

  Estos permisos controlan el acceso a objetos específicos dentro de una base de datos, como tablas, vistas, procedimientos almacenados, etc. Permiten realizar operaciones como consultar, insertar, actualizar o eliminar datos [7].

1- **SELECT**: Permite consultar datos de una tabla o vista [7].

2- **INSERT**: Permite insertar registros en una tabla [7].

3- **UPDATE**: Permite modificar registros en una tabla [7].

4- **DELETE**: Permite eliminar registros de una tabla [7].

5- **EXECUTE**: Permite ejecutar procedimientos almacenados o funciones [7].

6- **ALTER**: Permite modificar la estructura de un objeto, como una tabla [7].

7- **INDEX**: Permite crear índices en una tabla [7].

  - **Sistema:**

  Estos permisos otorgan la capacidad de realizar operaciones administrativas o de gestión en toda la instancia del servidor de bases de datos. Ejemplos incluyen crear, modificar o eliminar bases de datos, gestionar usuarios y roles, y realizar tareas de mantenimiento [8]**.**

1- **CREATE**: Permite al usuario crear nuevos objetos, como bases de datos, tablas, índices o procedimientos almacenados [8].

2- **ALTER**: Permite al usuario modificar la estructura de objetos existentes, como agregar o eliminar columnas de una tabla, o cambiar propiedades de una base de datos [8].

3- **DROP**: Permite al usuario eliminar objetos, como bases de datos, tablas o vistas [8].

- **¿Cómo dar/quitar permisos?**

  - **Dar permisos (GRAND)**

  Es una sentencia SQL que se utiliza para conceder permisos a un usuario o grupo de usuarios para realizar ciertas acciones en una base de datos. Por ejemplo, puedes utilizar

GRANT para otorgar permisos para acceder a una tabla o vista, modificar datos, ejecutar procedimientos almacenados, entre otras acciones [9].

- **Quitar permisos (REVOKE)**

Es una sentencia SQL que se utiliza para revocar permisos concedidos previamente a un usuario o grupo de usuarios para realizar ciertas acciones en una base de datos. Por ejemplo, puedes utilizar REVOKE para revocar permisos para acceder a una tabla o vista, modificar datos, ejecutar procedimientos almacenados, entre otras acciones [9].

- **Diferencia entre role y usuario**

Un rol en SQL es un conjunto de permisos que se pueden conceder a varios usuarios de forma simultánea. Los roles se utilizan para simplificar la gestión de permisos en una base de datos y para evitar tener que conceder permisos individualmente a cada usuario [9].

Un usuario es una entidad individual que puede conectarse a la base de datos y realizar operaciones específicas. Cada usuario tiene un identificador único (username) y credenciales de autenticación (como una contraseña) [10].

Bibliografía (formato IEEE)

[1] A. Silberschatz, H. F. Korth, and S. Sudarshan, *Database System Concepts*, 7th ed. New York, NY, USA: McGraw-Hill, 2020.

[2] J. D. Ullman, *Principles of Database and Knowledge-Base Systems*, vol. 1. Rockville, MD, USA: Computer Science Press, 1988.

[3] C. J. Date, *An Introduction to Database Systems*, 8th ed. Boston, MA, USA: Addison-Wesley, 2003.

[4] M. Lutz, *Programming Python*, 4th ed. Sebastopol, CA, USA: O'Reilly Media, 2010.

[5] W. Stallings, *Data and Computer Communications*, 10th ed. Upper Saddle River, NJ, USA: Pearson, 2013.

[6] R. Elmasri and S. B. Navathe, *Fundamentals of Database Systems*, 7th ed. Boston, MA, USA: Pearson, 2016.

[7] Microsoft, “Database Engine Permissions and Security,” *Microsoft Docs*, 2023. [En línea]. Disponible en: https://docs.microsoft.com

[8] Oracle, “Oracle Database Security Guide,” *Oracle Documentation*, 2023. [En línea]. Disponible en: https://docs.oracle.com

[9] M. F. González, «GRANT y REVOKE: Permisos de base de datos en SQL | Programar SQL», *Programar en SQL*, 4 de julio de 2024. https://www.programarsql.com/grant-y-revoke-permisos-de-base-de-datos-en-sql/

[10] PostgreSQL Global Development Group, "PostgreSQL 13.0 Documentation," 2020. [Online]. Available: https://www.postgresql.org/docs/13/index.html. [Accessed: Oct. 10, 2023].