Tarea 2

1. ¿Qué requiero para conectarme a una base de datos?

Para conectarte a una base de datos necesitas:

- **Controlador/Driver**: Software que nos permite la comunicación entre la aplicación y la BD (ej: JDBC para Java, ODBC)
- **Cadena de conexión**: Incluyendo:
    - Host/Dirección del servidor
    - Puerto
    - Nombre de la base de datos
    - Credenciales (usuario y contraseña)
    - Parámetros adicionales (SSL, timeout, etc.)
- **Cliente/Herramienta**: Puede ser una aplicación (MySQL Workbench, SQL Server Management Studio), etc
- **Protocolo de red**: TCP/IP generalmente
- **Acceso a red**: Permisos de firewall y conectividad al servidor

2. Permisos a nivel sistema y objeto

Permisos a nivel sistema:

- CREATE DATABASE
- CREATE USER
- CREATE TABLE
- BACKUP DATABASE
- SHUTDOWN
- CREATE PROCEDURE
- ALTER ANY LOGIN
- CONTROL SERVER

Permisos a nivel objeto:

- SELECT (consultar datos)
- INSERT (insertar datos)
- UPDATE (actualizar datos)
- DELETE (eliminar datos)
- EXECUTE (ejecutar procedimientos)
- REFERENCES (crear claves foráneas)
- ALTER (modificar estructura)
- CONTROL (todos los permisos)

3. ¿Cómo dar/quitar permisos?

La administración de permisos se realiza mediante dos comandos principales: **GRANT** (para otorgar permisos) y **REVOKE** (para quitarlos). Estos comandos permiten controlar qué acciones puede realizar cada usuario sobre los objetos de la base de datos.

Otorgar permisos (GRANT)

El comando GRANT asigna privilegios específicos a usuarios o roles. La sintaxis básica sigue esta estructura: GRANT [permisos] ON [objeto] TO [usuario o rol];

Quitar permisos (REVOKE)

El comando REVOKE elimina permisos previamente otorgados. La sintaxis es similar a GRANT: REVOKE [permisos] ON [objeto] FROM [usuario o rol];

4. Diferencia entre role y usuario

Para entender la diferencia, primero debemos comprender qué representa cada concepto en el entorno de una base de datos.

El Usuario

Un **usuario** es una **identidad** que puede conectarse a la base de datos y realizar acciones. Representa a una persona, aplicación o servicio que necesita acceder a los datos.

Características principales del usuario:

- **Es una entidad única**: Cada usuario tiene credenciales propias (nombre de usuario y contraseña) para autenticarse
- **Posee identidad propia**: Se utiliza para auditoría (saber quién hizo qué)
- **Puede tener permisos directos**: Se le pueden asignar privilegios específicos
- **Es el que ejecuta las acciones**: Cuando alguien se conecta, lo hace como un usuario específico

El Role

Un **role** es un **contenedor de permisos** que agrupa privilegios relacionados. No puede conectarse a la base de datos, solo sirve para organizar y facilitar la administración de permisos.

Características principales del role:

- **No tiene identidad propia**: No puede conectarse, no tiene contraseña
- **Es una plantilla de permisos**: Agrupa privilegios que suelen ir juntos
- **Facilita la administración**: En lugar de asignar 10 permisos a 20 usuarios, creas un role y lo asignas
- **Permite cambios centralizados**: Modificas el role y todos los usuarios con ese role se actualizan automáticamente

La Relación entre Usuario y Role

La clave está en cómo se relacionan: **los usuarios pueden tener roles asignados**, heredando todos los permisos que contienen.

Ventajas de usar Roles

1. **Administración eficiente**: Un cambio en el role afecta a todos los usuarios que lo tienen
2. **Consistencia**: Todos los usuarios con el mismo rol tienen exactamente los mismos permisos
3. **Seguridad**: Facilita aplicar el principio de mínimo privilegio
4. **Organización**: Los permisos se agrupan lógicamente por funciones de trabajo
5. **Auditoría simplificada**: Es más fácil revisar qué permisos tiene cada rol que revisar usuario por usuario