

Cómo agregar una Base de Datos al proyecto de Tabla Hash en Java

Actualmente:

El proyecto utiliza una tabla hash en memoria y guarda los datos en un archivo de texto (datos.txt).

Esto funciona como una "mini base de datos" pero no es tan flexible ni robusto como un motor de base de datos real.

Pasos para usar una Base de Datos (ejemplo con SQLite o MySQL):

1. Agregar librería JDBC

- Para SQLite: se usa `sqlite-jdbc.jar`.
- Para MySQL: se usa `mysql-connector-j.jar`.

Estas librerías permiten que Java se conecte a la base de datos.

2. Crear una clase para manejar la base de datos (BDManager)

Esta clase tendrá métodos para conectar, insertar, buscar y eliminar datos.

Ejemplo con SQLite:

- Conectar a la base de datos con `jdbc:sqlite:datos.db`
- Crear la tabla si no existe: `CREATE TABLE IF NOT EXISTS tabla_hash (clave INTEGER PRIMARY KEY, valor INTEGER)`
- Insertar/Actualizar: `INSERT OR REPLACE INTO tabla_hash (clave, valor)`
- Buscar: `SELECT valor FROM tabla_hash WHERE clave = ?`
- Eliminar: `DELETE FROM tabla_hash WHERE clave = ?`

3. Integrar con la GUI

En la interfaz gráfica, cada botón llamará también a la base de datos:

- Insertar: además de guardar en la tabla hash, se ejecuta `BDManager.insertar`.
- Buscar: puede consultarse en memoria o en la base de datos.
- Eliminar: además de borrar en memoria, se ejecuta `BDManager.eliminar`.

Al iniciar el programa, se debe llamar a `BDManager.crearTabla()` para asegurarse de que la tabla exista.

Diferencias frente al archivo de texto:

- Archivo: guarda y carga todos los datos de golpe.
- Base de datos: cada operación (insertar, buscar, eliminar) se hace directamente en la BD.
- SQLite: funciona con un solo archivo local (datos.db), fácil de usar y no requiere servidor.
- MySQL: requiere un servidor, pero es más potente para aplicaciones grandes o en red.

Resumen:

El cambio principal es reemplazar la persistencia en archivo de texto por una base de datos real mediante JDBC.

SQLite es ideal para pruebas o proyectos pequeños (archivo local).

MySQL es mejor para sistemas más grandes y multiusuario.