

SQLite base de dados

O SQLite é um sistema de gestão de base de dados relacional open source, utilizado em aplicações que requerem uma de dados embutida, como aplicativos móveis e pequenos projetos. Quando usado em memória, o SQLite é especialmente útil para testes e desenvolvimento, pois os dados são armazenados na RAM e não são persistidos no disco, o que torna as operações muito rápidas.

Principais Características do SQLite em Memória

Volatilidade: Os dados são armazenados na memória RAM, o que significa que eles são perdidos quando a aplicação é encerrada. **Velocidade:** Como os dados não são escritos no disco, as operações de leitura e escrita são muito rápidas. **Facilidade de Configuração:** É fácil de configurar e integrar em aplicações Java, especialmente para testes unitários e de integração. **Portabilidade:** O SQLite é uma base de dados sem servidor, o que significa que ele pode ser facilmente incluído em qualquer aplicação sem a necessidade de instalação de um servidor de banco de dados.

Passo 1 - Configurar o Maven

```
<dependency>
  <groupId>org.xerial</groupId>
  <artifactId>sqlite-jdbc</artifactId>
  <version>3.25.2</version>
</dependency>
```

Passo 2 - Java: Criar Connection

Sintaxe SQL O SQLite suporta a maioria dos comandos SQL padrão, incluindo:

- DDL (Data Definition Language): Comandos como CREATE, ALTER, DROP
- DML (Data Manipulation Language): Comandos como INSERT, UPDATE, DELETE
- DQL (Data Query Language): Comandos como SELECT para consulta de dados.

Tipos de dados

- TEXT
- NUMERIC
- INTEGER
- REAL
- BLOB

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;

public class DatabaseConnection {
    private static final String URL = "jdbc:sqlite::memory:";
```

```

private static final String USER = "";
private static final String PASSWORD = "";

public static Connection getConnection() throws SQLException {
    return DriverManager.getConnection(URL, USER, PASSWORD);
}
}

```

Passo 3 - Exemplo de um DAO

```

import java.sql.*;
import java.util.ArrayList;
import java.util.List;

public class UserDao {
    public void createTable() throws SQLException {
        String sql = "CREATE TABLE IF NOT EXISTS users (id INTEGER PRIMARY KEY AUTOINCREMENT, name TEXT)";
        try (Connection conn = DatabaseConnection.getConnection();
            Statement stmt = conn.createStatement()) {
            stmt.execute(sql);
        }
    }

    public void insertUser(String name) throws SQLException {
        String sql = "INSERT INTO users (name) VALUES (?)";
        try (Connection conn = DatabaseConnection.getConnection();
            PreparedStatement pstmt = conn.prepareStatement(sql)) {
            pstmt.setString(1, name);
            pstmt.executeUpdate();
        }
    }

    public List<String> getAllUsers() throws SQLException {
        List<String> users = new ArrayList<>();
        String sql = "SELECT * FROM users";
        try (Connection conn = DatabaseConnection.getConnection();
            Statement stmt = conn.createStatement();
            ResultSet rs = stmt.executeQuery(sql)) {
            while (rs.next()) {
                users.add(rs.getString("name"));
            }
        }
        return users;
    }
}

```