### Exercício 9.7 - Tutorial Passo-a-Passo

Interface de Utilizador (UI) com Java, usando Vaadin Flow

Demonstração de Uso de MariaDB como Base de Dados Alternativa (à H2)

**Base de Dados de Livros com MariaDB (Demonstração de Alternativa à H2)**

### **Introdução**

Este manual apresenta um guia passo a passo para configurar um projeto Spring Boot com MariaDB usando XAMPP, gerir a base de dados com phpMyAdmin ou a extensão MySQL no VS Code, e criar uma interface gráfica com Vaadin Flow. A base de dados utilizada será chamada **"livros97"**.

### **1. Configuração do Projeto para Usar MariaDB**

#### **Passo 1: Configurar o Ficheiro application.properties**

No ficheiro application.properties, configure as propriedades para usar o MariaDB:

(Substituído as que estavam para a base de dados H2)

# Configuração do MariaDB  
spring.datasource.url=jdbc:mariadb://localhost:3306/livros97  
spring.datasource.driver-class-name=org.mariadb.jdbc.Driver  
spring.datasource.username=root  
spring.datasource.password=  
  
# Configuração do JPA  
spring.jpa.hibernate.ddl-auto=update  
spring.jpa.database-platform=org.hibernate.dialect.MariaDBDialect  
  
# Porta do servidor  
server.port=8080

#### **Passo 2: Dependências no pom.xml**

Certifique-se de incluir as dependências necessárias no seu projeto Spring Boot. Aqui está o trecho atualizado do pom.xml com comentários para facilitar o entendimento:

<dependencies>  
 <!-- Dependência do MariaDB: Driver JDBC para conectar a aplicações Java -->  
 <dependency>  
 <groupId>org.mariadb.jdbc</groupId>  
 <artifactId>mariadb-java-client</artifactId>  
 <version>3.1.2</version>  
 </dependency>  
  
 <!-- OSHI: Biblioteca para obter informações de hardware do sistema -->  
 <dependency>  
 <groupId>com.github.oshi</groupId>  
 <artifactId>oshi-core</artifactId>  
 <version>6.4.0</version> <!-- Versão estável -->  
 </dependency>  
  
 <!-- JNA: Acesso nativo a recursos do sistema operacional usado pelo OSHI -->  
 <dependency>  
 <groupId>net.java.dev.jna</groupId>  
 <artifactId>jna</artifactId>  
 <version>5.13.0</version> <!-- Última versão estável -->  
 </dependency>  
  
 <!-- JNA Platform: Complemento para acesso a APIs nativas da plataforma -->  
 <dependency>  
 <groupId>net.java.dev.jna</groupId>  
 <artifactId>jna-platform</artifactId>  
 <version>5.13.0</version>  
 </dependency>  
</dependencies>

Atualize as dependências do Maven com o comando:

./mvnw clean install

### **Alternativa A - Configuração do MariaDB no XAMPP**

#### **Passo 1: Instalar o XAMPP**

1. Baixe o XAMPP em <https://www.apachefriends.org>.
2. Instale o XAMPP e, durante o processo, certifique-se de incluir o **MariaDB** (vem por padrão).

#### 

#### **Passo 2: Inicializar o XAMPP**

1. Abra o **XAMPP Control Panel**.
2. **Garanta que o MariaDB está em execução:**
   * Localize o módulo **MySQL** no painel do XAMPP. (MariaDB é o servidor usado por padrão.)
   * Clique no botão **Start** ao lado de **MySQL**. O status do módulo deve mudar para "Running" e a barra correspondente deve ficar verde.

* **Nota Importante:** O MariaDB deve estar em execução **sempre que a aplicação Spring Boot for iniciada**. Caso contrário, a aplicação não conseguirá conectar-se à base de dados.

1. Se necessário, clique em **Admin** no módulo MySQL para abrir o **phpMyAdmin** no navegador.

#### 

#### **Passo 3: Criar a Base de Dados "livros97"**

1. No **phpMyAdmin**, vá à aba **Databases**.
2. No campo de nome, insira livros97 e clique em **Create**.
3. A base de dados estará pronta para ser usada no projeto.

### **Alternativa B - Configuração do MariaDB no VS Code**

#### 

#### **Passo 1: Instalar a Extensão MySQL**

1. No **VS Code**, instale a extensão oficial **MySQL (Oracle)**.
2. Após a instalação, clique no ícone da extensão na barra lateral.

#### 

#### **Passo 2: Criar a Base de Dados (Alternativa ao phpMyAdmin)**

1. Adicione uma nova conexão:
   * **Host:** localhost
   * **Port:** 3306
   * **Username:** root
   * **Password:** (vazio por padrão no XAMPP).
2. Após conectar, clique com o botão direito na conexão e escolha **New Query**.
3. Execute o comando SQL para criar a base de dados:

CREATE DATABASE livros97;

Agora a base de dados estará pronta para ser usada (faça “refresh” para ver a nova base de dados)

### **Muito Importante – Arrancar a Base de Dados**

**Certifique-se de que o MariaDB está ativo**

Antes de iniciar a aplicação, **confirme que o MariaDB está ativo no XAMPP**:

1. Abra o **XAMPP Control Panel**.
2. Verifique se o módulo **MySQL** está com o status **"Running"**.
   * Se não estiver, clique no botão **Start** ao lado do módulo **MySQL**.
   * Certifique-se de que a barra correspondente ao MySQL ficou verde, indicando que o MariaDB está em execução.
3. Este passo é **obrigatório**, pois, sem o MariaDB em execução, a aplicação Spring Boot não conseguirá conectar à base de dados e apresentará erros.

### **6. Testar a Aplicação**

1. Com o MariaDB em execução, inicie a aplicação Spring Boot no terminal com o comando:

* ./mvnw spring-boot:run

1. Após o servidor iniciar, abra o navegador e aceda ao endereço:

* <http://localhost:8080/livros>

1. Teste as funcionalidades:
   * Adicionar livros preenchendo o formulário.
   * Listar os livros adicionados na interface de utilizador.
   * Confirmar que os dados são persistidos na base de dados **livros97**.

### **7. Alteração Fácil para Outra Base de Dados**

Caso deseje alterar para outra base de dados no futuro:

1. Atualize o ficheiro application.properties para apontar para o novo servidor e as novas credenciais:
   * Substitua o spring.datasource.url, spring.datasource.driver-class-name e spring.jpa.database-platform de acordo com a base de dados desejada.
2. Adicione a dependência correspondente no pom.xml (por exemplo, o driver do PostgreSQL ou Oracle).
3. Reinicie a aplicação.

### **Resumo**

Este manual final destaca a necessidade de garantir que o **MariaDB esteja em execução** no XAMPP sempre que a aplicação Spring Boot for iniciada, além de fornecer todos os passos necessários para configurar o projeto, gerir a base de dados, e criar uma interface de utilizador funcional com o Vaadin Flow.

Se seguir estes passos corretamente, terá uma aplicação robusta, conectada ao MariaDB, e com uma interface gráfica moderna e integrada ao Spring Boot. 😊

### **Em Caso de Dificuldades: Tentar estes Comandos em Sequência:**

./mvnw dependency:purge-local-repository

./mvnw clean install

./mvnw spring-boot:run

Ver o Vídeo feito para Explicar este Exemplo, no YouTube