package com.cars.controllers;  
  
import com.cars.services.CarService;  
import com.cars.entities.CarEntity;  
import lombok.RequiredArgsConstructor;  
import org.springframework.data.domain.Page;  
import org.springframework.web.bind.annotation.\*;  
  
@RestController  
@RequestMapping("api/v1/cars")  
@RequiredArgsConstructor  
public class CarController {  
  
 private final CarService carService;  
  
 @GetMapping  
 public Page<CarEntity> getAllCars(  
 @RequestParam(defaultValue = "0") int page,  
 @RequestParam(defaultValue = "10") int size) {  
 return carService.getAllCars(page, size);  
 }  
}

package com.cars.entities;  
  
import jakarta.persistence.\*;  
import lombok.\*;  
  
@Entity  
@Table(name = "cars")  
@Getter  
@Setter  
@NoArgsConstructor  
@AllArgsConstructor  
@ToString  
@EqualsAndHashCode  
public class CarEntity {  
  
 @Id  
 @GeneratedValue(strategy = GenerationType.*IDENTITY*)  
 private Long id;  
  
 private String brand;  
 private String model;  
 private int year;  
}

package com.cars.services;  
  
import com.cars.entities.CarEntity;  
import com.cars.repositories.CarRepository;  
import lombok.RequiredArgsConstructor;  
import org.springframework.data.domain.Page;  
import org.springframework.data.domain.PageRequest;  
import org.springframework.stereotype.Service;  
  
@Service  
@RequiredArgsConstructor  
public class CarService {  
  
 private final CarRepository carRepository;  
  
 public Page<CarEntity> getAllCars(int page, int size) {  
 return carRepository.findAll(PageRequest.*of*(page, size));  
 }  
}

package com.cars.repositories;  
  
import com.cars.entities.CarEntity;  
import org.springframework.data.jpa.repository.JpaRepository;  
import org.springframework.stereotype.Repository;  
  
@Repository  
public interface CarRepository extends JpaRepository<CarEntity, Long> {  
}

spring:  
 profiles:  
 active: prod *# Define 'dev' como perfil padrão*---  
*# 🌱 Configuração para DESENVOLVIMENTO (banco de dados local no MySQL)*spring:  
 config:  
 activate:  
 on-profile: dev  
  
 datasource:  
 url: jdbc:mysql://localhost:3306/carsdb  
 username: root  
 password: 55643988  
 driver-class-name: com.mysql.cj.jdbc.Driver  
  
 jpa:  
 hibernate:  
 ddl-auto: update  
 database-platform: org.hibernate.dialect.MySQLDialect  
 open-in-view: false  
 show-sql: false *# true para Mostrar as queries no console,*---  
*# 🚀 Configuração para PRODUÇÃO (usa variáveis de ambiente)*spring:  
 config:  
 activate:  
 on-profile: prod  
  
 datasource:  
 url: ${DATABASE\_URL:jdbc:mysql://localhost:3306/prod\_database?serverTimezone=UTC&useSSL=false}  
 username: ${DATABASE\_USER:root}  
 password: ${DATABASE\_PASSWORD:root}  
 driver-class-name: com.mysql.cj.jdbc.Driver  
  
 jpa:  
 hibernate:  
 ddl-auto: update  
 database-platform: org.hibernate.dialect.MySQLDialect  
 open-in-view: false  
 show-sql: false *# Desativa logs SQL em produção* server:  
 port: 8080  
  
---  
*# 🛠 Configuração para TESTES (banco separado para testes)*spring:  
 config:  
 activate:  
 on-profile: test  
  
 datasource:  
 url: jdbc:mysql://localhost:3306/test\_database?serverTimezone=UTC&useSSL=false  
 username: root  
 password: root  
 driver-class-name: com.mysql.cj.jdbc.Driver  
  
 jpa:  
 hibernate:  
 ddl-auto: update  
 database-platform: org.hibernate.dialect.MySQLDialect  
 open-in-view: false  
 show-sql: true  
  
 server:  
 port: 8081 *# Usa uma porta diferente para testes*

<?xml version="1.0" encoding="UTF-8"?>  
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
 xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://maven.apache.org/xsd/maven-4.0.0.xsd">  
  
 <modelVersion>4.0.0</modelVersion>  
  
 <parent>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-parent</artifactId>  
 <version>3.4.1</version>  
 <relativePath/> <!-- lookup parent from repository -->  
 </parent>  
  
 <groupId>com.cars</groupId>  
 <artifactId>Cars</artifactId>  
 <version>0.0.1-SNAPSHOT</version>  
 <name>Cars</name>  
 <description>Demo project for Spring Boot</description>  
  
 <properties>  
 <java.version>21</java.version>  
 </properties>  
  
 <dependencies>  
 <!-- Spring Boot Core -->  
 <dependency>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-data-jpa</artifactId>  
 </dependency>  
<!-- <dependency>-->  
<!-- <groupId>org.springframework.boot</groupId>-->  
<!-- <artifactId>spring-boot-starter-security</artifactId>-->  
<!-- </dependency>-->  
 <dependency>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-web</artifactId>  
 </dependency>  
  
 <!-- Banco de Dados -->  
 <dependency>  
 <groupId>com.mysql</groupId>  
 <artifactId>mysql-connector-j</artifactId>  
 <scope>runtime</scope>  
 </dependency>  
  
 <!-- Lombok -->  
 <dependency>  
 <groupId>org.projectlombok</groupId>  
 <artifactId>lombok</artifactId>  
 <scope>provided</scope>  
 </dependency>  
<!-- <dependency>-->  
<!-- <groupId>org.projectlombok</groupId>-->  
<!-- <artifactId>lombok-mapstruct-binding</artifactId>-->  
<!-- <version>0.2.0</version>-->  
<!-- <scope>provided</scope>-->  
<!-- </dependency>-->  
  
 <!-- Testes -->  
 <dependency>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-test</artifactId>  
 <scope>test</scope>  
 </dependency>  
<!-- <dependency>-->  
<!-- <groupId>org.springframework.security</groupId>-->  
<!-- <artifactId>spring-security-test</artifactId>-->  
<!-- <scope>test</scope>-->  
<!-- </dependency>-->  
 </dependencies>  
  
 <build>  
 <plugins>  
 <!-- Compilador Java -->  
 <plugin>  
 <groupId>org.apache.maven.plugins</groupId>  
 <artifactId>maven-compiler-plugin</artifactId>  
 <configuration>  
 <source>${java.version}</source>  
 <target>${java.version}</target>  
 <annotationProcessorPaths>  
 <path>  
 <groupId>org.projectlombok</groupId>  
 <artifactId>lombok</artifactId>  
 </path>  
 </annotationProcessorPaths>  
 </configuration>  
 </plugin>  
  
 <!-- Plugin do Spring Boot -->  
 <plugin>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-maven-plugin</artifactId>  
 </plugin>  
 </plugins>  
 </build>  
</project>