Create Springboot project using start.spring.io ->

Pom.xml

<?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>3.3.4</version>

<relativePath /> <!-- lookup parent from repository -->

</parent>

<groupId>com.var</groupId>

<artifactId>springrestjpademo</artifactId>

<version>0.0.1-SNAPSHOT</version>

<name>springrestjpademo</name>

<description>Demo project for Spring Boot</description>

<url />

<licenses>

<license />

</licenses>

<developers>

<developer />

</developers>

<scm>

<connection />

<developerConnection />

<tag />

<url />

</scm>

<properties>

<java.version>17</java.version>

</properties>

<dependencies>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-data-jpa</artifactId>

</dependency>

<!-- https://mvnrepository.com/artifact/com.oracle.database.jdbc/ojdbc6 -->

<dependency>

<groupId>com.oracle.database.jdbc</groupId>

<artifactId>ojdbc6</artifactId>

<version>11.2.0.4</version>

</dependency>

<!--

https://mvnrepository.com/artifact/org.springframework.boot/spring-boot-devtools -->

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-devtools</artifactId>

<version>3.3.0</version>

</dependency>

<!---->

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-web</artifactId>

</dependency>

<dependency>

<groupId>org.springdoc</groupId>

<artifactId>springdoc-openapi-starter-webmvc-ui</artifactId>

<version>2.5.0</version>

</dependency>

<dependency>

<groupId>org.springdoc</groupId>

<artifactId>springdoc-openapi-starter-webflux-ui</artifactId>

<version>2.1.0</version>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-test</artifactId>

<scope>test</scope>

</dependency>

</dependencies>

<build>

<plugins>

<plugin>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-maven-plugin</artifactId>

</plugin>

</plugins>

</build>

</project>

Application.properties :

spring.application.name=springrestjpademo

server.port=8082

spring.datasource.url = jdbc:oracle:thin:@//localhost:1521/xe

spring.datasource.username= system

spring.datasource.password= system

## Hibernate Properties

# SQL dialect makes Hibernate generate better SQL for chosen database

spring.jpa.properties.hibernate.dialect= org.hibernate.dialect.OracleDialect

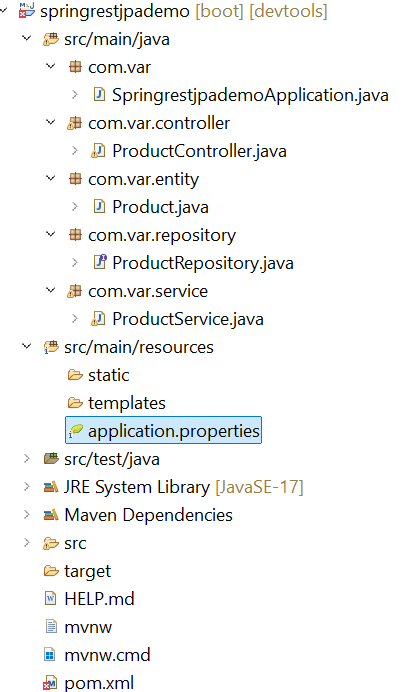
# Hibernate ddlauto (create, create-drop, validate, update)

spring.jpa.hibernate.ddl-auto = update

spring.jpa.show-sql=true

springdoc.api-docs.path=/api-docs

Create project structure as follows:



SpringrestjpademoApplication.java

**package** com.var;

**import** org.springframework.boot.SpringApplication;

**import** org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

**public** **class** SpringrestjpademoApplication {

**public** **static** **void** main(String[] args) {

SpringApplication.*run*(SpringrestjpademoApplication.**class**, args);

}

}

ProductController.java:

package com.var.controller;

import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.web.bind.annotation.DeleteMapping;

import org.springframework.web.bind.annotation.GetMapping;

import org.springframework.web.bind.annotation.PathVariable;

import org.springframework.web.bind.annotation.PostMapping;

import org.springframework.web.bind.annotation.PutMapping;

import org.springframework.web.bind.annotation.RequestBody;

import org.springframework.web.bind.annotation.RestController;

import com.var.entity.Product;

import com.var.service.ProductService;

import jakarta.websocket.server.PathParam;

@RestController

public class ProductController {

@Autowired

ProductService productService;

@PostMapping("/product")

public String addProduct(@RequestBody Product product)

{ productService.addProduct(product);

return "Added";

}

@GetMapping("/product/{id}")

public Product getProduct(@PathVariable("id") Integer id) {

return productService.getProduct(id);

}

@GetMapping("/product")

public List<Product> getAllProducts() {

return productService.getAllProducts();

}

@PutMapping("/product/{id}")

public String updateProduct(@PathVariable("id") Integer id,@RequestBody Product product) { productService.updateProduct(id,product);

return "updated";

}

@DeleteMapping("/product/{id}")

public String deleteProduct(@PathVariable("id") Integer id) {

productService.deleteProduct(id);

return "deleted";

}

}

Product.java

**package** com.var.entity;

**import** jakarta.persistence.Column;

**import** jakarta.persistence.Entity;

**import** jakarta.persistence.Id;

**import** jakarta.persistence.Table;

@Entity

@Table(name="product\_table")

**public** **class** Product {

@Id

@Column(name="product\_id")

Integer id;

@Column(name="product\_name")

String name;

**public** Product(){}

**public** Integer getId() {

**return** id;

}

**public** **void** setId(Integer id) {

**this**.id = id;

}

**public** String getName() {

**return** name;

}

**public** **void** setName(String name) {

**this**.name = name;

}

@Override

**public** String toString() {

**return** "Product [id=" + id + ", name=" + name + "]";

}

**public** Product(Integer id, String name) {

**super**();

**this**.id = id;

**this**.name = name;

}

}

ProductRepository.java

package com.var.repository;

import org.springframework.data.repository.CrudRepository;

import org.springframework.stereotype.Repository;

import com.var.entity.Product;

@Repository

public interface ProductRepository extends CrudRepository<Product,Integer>{

// save, findById(), findAll,deleteById,...

}

ProductService.java

package com.var.service;

import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.stereotype.Service;

import com.var.entity.Product;

import com.var.repository.ProductRepository;

import jakarta.transaction.Transactional;

@Service

@Transactional

public class ProductService {

@Autowired

ProductRepository productRepository;

public void addProduct(Product product) {

productRepository.save(product);

}

public List<Product> getAllProducts() {

return (List)productRepository.findAll();

}

public Product getProduct(Integer id) {

return productRepository.findById(id).get();

}

public void deleteProduct(Integer id) {

productRepository.deleteById(id);

}

public void updateProduct(Integer id,Product product) {

Product product1=productRepository.findById(id).get();

product1.setName(product.getName());

}

}