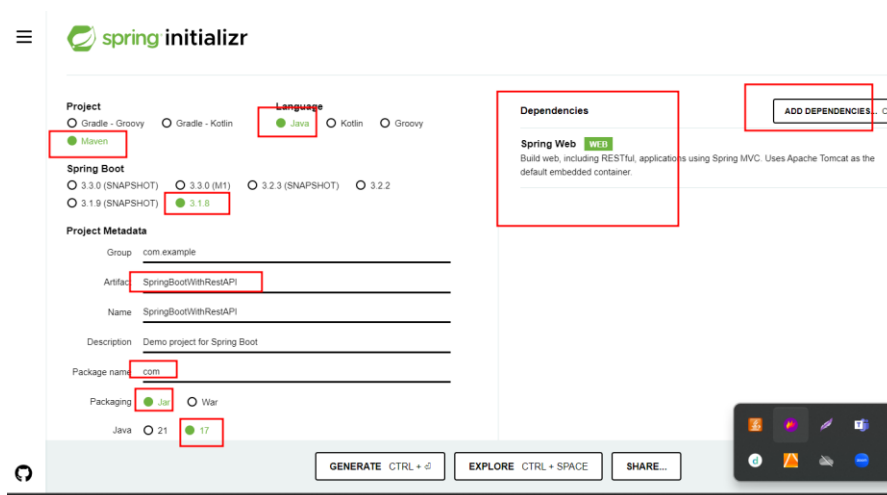


Spring boot Rest full Web Service

In Spring boot we need to add web starter and in place @Controller

We need to write @RestController annotation.

When we write @RestController annotation we can write return type of method string or user defined object with produce attribute.



@Controller annotation is use to return view as jsp or thymeleaf

@RestController annotation is use to return rest api.

Before @RestController we if want to use rest api concept

@RestController = @Controller + @ResponseBody;

Spring boot with Rest API interacting with database using Spring JPA Data.

Entity can be Employee, Customer, Manager, Product, order etc.

ProductManagementRestAPI -> project name

Web starter -> to create web service

Jpa starter --> to use core jpa or spring jpa data.

Mysql connector -> to connect mysql database

Dev starter -> it is use to refresh the project whenever we do any changes.

(only in development mode).

The screenshot displays the Spring Initializr web application interface. The 'Project' section shows 'Maven' selected under 'Project' and 'Java' under 'Language'. The 'Spring Boot' section shows '3.1.8' selected under 'Spring Boot'. The 'Project Metadata' section shows 'com.example' for 'Group', 'ProductManagementRestAPI' for 'Artifact', 'ProductManagementRestAPI' for 'Name', 'Demo project for Spring Boot' for 'Description', 'com' for 'Package name', 'Jar' for 'Packaging', and '17' for 'Java'. The 'Dependencies' section shows 'Spring Web', 'Spring Data JPA', 'MySQL Driver', and 'Spring Boot DevTools' selected. A red box highlights the 'ADD DEPENDENCIES...' button. The bottom of the page shows 'GENERATE', 'EXPLORE', and 'SHARE...' buttons.

spring initializr

Project
☐ Gradle - Groovy ☐ Gradle - Kotlin ☒ **Java** ☐ Kotlin ☐ Groovy

Spring Boot
☐ 3.3.0 (SNAPSHOT) ☒ 3.3.0 (M1) ☐ 3.2.3 (SNAPSHOT) ☐ 3.2.2
☐ 3.1.9 (SNAPSHOT) ☒ 3.1.8

Project Metadata
Group: com.example
Artifact: ProductManagementRestAPI
Name: ProductManagementRestAPI
Description: Demo project for Spring Boot
Package name: com
Packaging: ☒ Jar ☐ War
Java: ☐ 21 ☒ 17

Dependencies
Spring Web WEB
Build web, including RESTful, applications using Spring MVC. Uses Apache Tomcat as the default embedded container.
Spring Data JPA SQL
Persist data in SQL stores with Java Persistence API using Spring Data and Hibernate.
MySQL Driver SQL
MySQL JDBC driver.
Spring Boot DevTools DEVELOPER TOOLS
Provides fast application restarts, LiveReload, and configurations for enhanced development experience.

ADD DEPENDENCIES...

GENERATE CTRL + G **EXPLORE** CTRL + SPACE **SHARE...**

Spring JPA Data.

JPA Data provided two interface ie CrudRepository and JpaRepository.

These interface base upon java 8 features. Which provided implementation of dao layer.

Java 8 onward interface can contains method with body.

CrudRepository -> Basic Repository

JpaRepository -> Jpa specific repository

JpaRepository interface extends CrudRepository.

We need to create normal interface and that implements may be extends CrudRepository or JpaRepository.

Which takes generic type. 1 class name and 2nd data type of primary key.

@RequestBody annotation is use to extract the data from request body part.