# 第一章：SpringBoot入门

## 本章大纲

* SpringBoot介绍
* 构建SpringBoot项目及启动器
* SpringBoot入门案例

## SpringBoot介绍

Spring 诞生时是 Java 企业版（Java Enterprise Edition，JEE，也称 J2EE）的轻量级框架。

Spring 为企业级Java 开发提供了一种相对简单的方法，通过依赖注入和面向切面编程，用简单的Java 对象（Plain Old Java Object，POJO）实现了 EJB 的功能。虽然 Spring 的组件代码是轻量级的，但它的配置却是重量级的。

第一阶段：xml配置

在Spring 1.x时代，使用Spring开发满眼都是xml配置的Bean，随着项目的扩大，我们需要把xml配置文件放到不同的配置文件里，那时需要频繁的在开发的类和配置文件之间进行切换

第二阶段：注解配置

在Spring 2.x 时代，随着JDK1.5带来的注解支持，Spring提供了声明Bean的注解（例如@Component、@Service），大大减少了配置量。主要使用的方式是应用的基本配置（如数据库配置）用xml，业务配置用注解

第三阶段：java配置

Spring 3.0 引入了基于 Java 的配置能力，这是一种类型安全的可重构配置方式，可以代替 XML。我们目前刚好处于这个时代，Spring4.x和Spring Boot都推荐使用Java配置。

Spring Boot 简化了基于Spring的应用开发，只需要“run”就能创建一个独立的、生产级别的Spring应用。Spring Boot为Spring平台及第三方库提供开箱即用的设置（提供默认设置），这样我们就可以简单的开始。多数Spring Boot应用只需要很少的Spring配置。

Spring Boot 主要目标是：

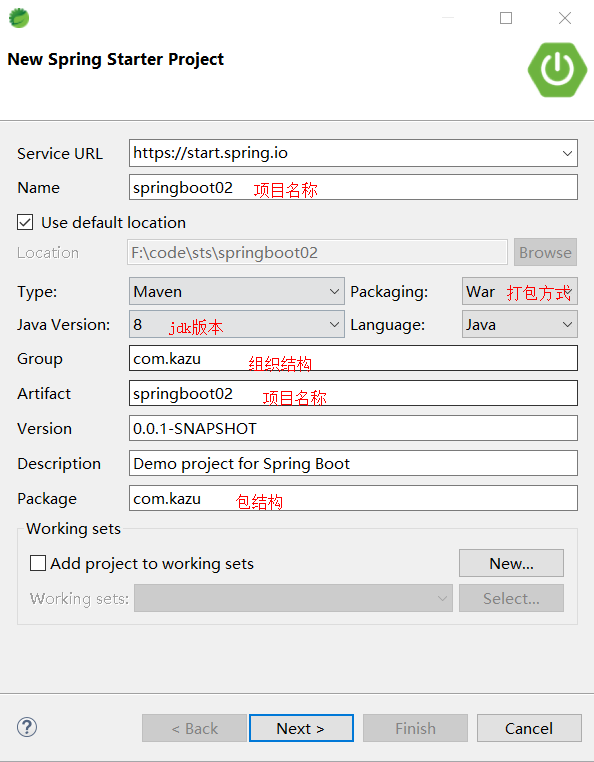
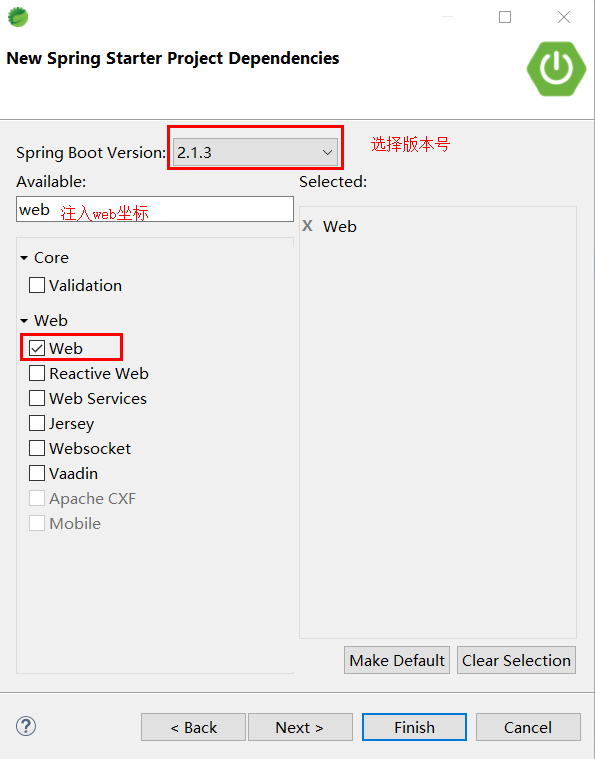
* 为所有 Spring 的开发提供一个从根本上更快的入门体验
* 开箱即用，但通过自己设置参数，即可快速摆脱这种方式。
* 提供了一些大型项目中常见的非功能性特性，如内嵌服务器、安全、指标，健康检测、外部化配置等
* 绝对没有代码生成，也无需 XML 配置。

## 构建SpringBoot项目及启动器

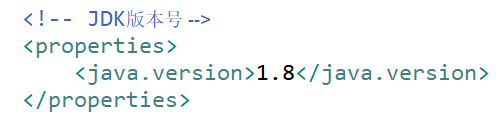
### 环境准备

* Spring Tool Suite4
* Maven 3.5+
* Jdk 8+

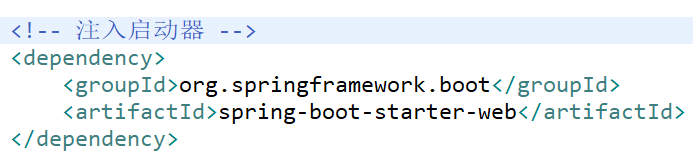
### 使用STS构建SpringBoot项目

### 修改pom文件，更新JDK版本



### 注入SpringBoot启动坐标



### Spring启动器

所谓的 springBoot 启动器其实就是一些 jar 包的集合。SprigBoot 一共提供 44 启动器。

* spring-boot-starter-web

支持全栈式的 web 开发，包括了 tomcat 和 springMVC 等 jar

* spring-boot-starter-jdbc

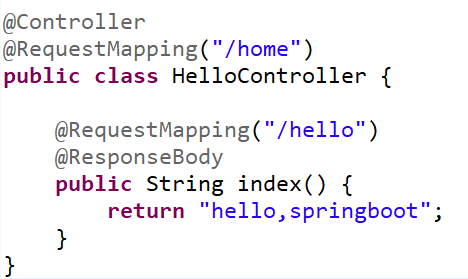
支持 spring 以 jdbc 方式操作数据库的 jar 包的集合

* spring-boot-starter-redis

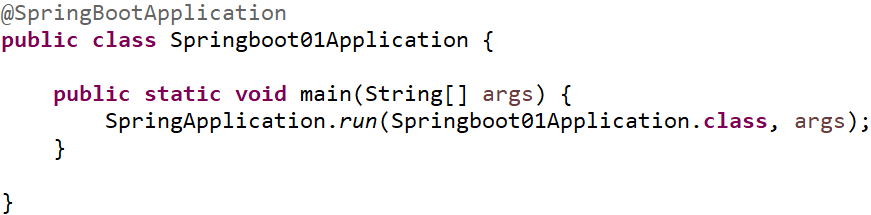
支持 redis 键值存储的数据库操作

## SpringBoot入门案例

### 编写HelloWorld的Controller



### 启动SpringBoot



### 启动器需要注意的问题

启动器存放的位置。启动器可以和 controller 位于同一个包下，或者位于 controller 的上一级

包中，但是不能放到 controller 的平级以及子包下。

### 修改属性文件

在resources资源文件夹下找到属性文件：application.properties

