# Spring Boot参考指南

版本2.1.5

## |||.使用Spring Boot

这个模块会更为详细的讲述Spring Boot的使用方法。包括一些主题，例如开发生态（原文用的是build systems，本应译为构建Spring boot项目所需的各种软件）、自动配置和如何运行你的应用。还包括Spring Boot的实例练习。虽然没有十分特别的关于Spring Boot的内容，但还是有一些建议或提醒，可以使你的开发稍微简单一些。

### 1开发生态

强烈建议您在选择开发生态时遵循以下两条：

1. 开发生态支持依赖管理
2. 可以将项目发布到“Maven Central” repository.

我们建议使用maven或gradle。虽然其他开发生态可以用于开发Spring boot，但是他们并不完美支持Spring boot。

#### 1.1依赖管理

每个版本的Spring Boot都提供了它支持的依赖项的精选列表。 实际上，您不需要为构建配置中的任何这些依赖项提供版本，因为Spring Boot会为您管理。 当您升级Spring Boot时，这些依赖项也会以一致的方式升级。

您仍然可以指定版本和覆盖。如果需要可以这么做。

精选列表包含可以与Spring Boot一起使用的所有spring模块以及精确的第三方库列表。 该列表可用作标准的Bills of Materials（spring-boot-dependencies），可与Maven和Gradle一起使用。

每一个版本打Spring Boot 都有对应版本的Spring Framework。强烈建议您不要指定它的版本。

#### 1.2Maven

Maven用户可以继承spring-boot-starter-parent项目以获得合理的默认值。 父项目提供以下内容特征：

Java1.8作为默认编译版本

UTF-8源编码。

继承自spring-boot-dependencies pom的依赖关系管理部分，它管理公共依赖项的版本。这种依赖关系管理允许您在自己的pom中使用时忽略这些依赖关系的<version>标记。

使用重新打包执行ID执行重新打包目标。

明智的资源过滤。

明智的插件配置（exec插件，Git提交ID和阴影）。

application.properties和application.yml的合理资源过滤，包括特定于配置文件的文件（例如，application-dev.properties和application-dev.yml）

请注意，由于application.properties和application.yml文件接受Spring样式占位符（$ {...}），因此Maven过滤更改为使用@ .. @占位符。 （您可以通过设置名为resource.delimiter的Maven属性来覆盖它。）

##### 1.2.1继承初始配置

要将项目配置为从spring-boot-starter-parent继承，请按如下所示设置父项

*<!-- Inherit defaults from Spring Boot -->*

<parent>

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

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

<version>2.1.5.RELEASE</version>

</parent>

