



链滴

Spring Boot 揭秘与实战（一）快速上手

作者: [idxiu](#)

原文链接: <https://ld246.com/article/1482291806499>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

简介

什么是Spring Boot

Spring Boot 设计目的是用来简化 Spring 应用的初始搭建以及开发过程。Spring Boot 的目标不在于为已解决的题域提供新的解决方案，而是为平台带来另一种开发体验，从而简化对这些已有技术的使用。Spring Boot 充分利用了 JavaConfig 的配置模式以及“约定优于配置”的理念，能够极大的简化基于 Spring MVC 的 Web 应用和 REST 服务开发。对于已经熟悉 Spring 生态系统的开发人员来说，Spring Boot 是一个很理想的选择。

为什么选择Spring Boot

- Spring Boot 使编码变简单
- Spring Boot 使配置变简单
- Spring Boot 使部署变简单
- Spring Boot 使监控变简单

相关知识

Spring Boot的spring-boot-starter

Spring Boot 供了很多“开箱即用”的依赖模块，都是以spring-boot-starter-xx作为命名的。下面列举一些常用模块。

- spring-boot-starter-logging：使用 Spring Boot 默认的日志框架 Logback。
- spring-boot-starter-log4j：添加 Log4j 的支持。
- spring-boot-starter-web：支持 Web 应用开发，包含 Tomcat 和 spring-mvc。
- spring-boot-starter-tomcat：使用 Spring Boot 默认的 Tomcat 作为应用服务器。
- spring-boot-starter-jetty：使用 Jetty 而不是默认的 Tomcat 作为应用服务器。
- spring-boot-starter-test：包含常用的测试所需的依赖，如 JUnit、Hamcrest、Mockito 和 spring-est 等。
- spring-boot-starter-aop：包含 spring-aop 和 AspectJ 来支持面向编程 (AOP)。
- spring-boot-starter-security：包含 spring-security。
- spring-boot-starter-jdbc：支持使用 JDBC 访问数据库。
- spring-boot-starter-redis：支持使用 Redis。
- spring-boot-starter-data-mongodb 包含 spring-data-mongodb 来支持 MongoDB。
- spring-boot-starter-data-jpa：包含 spring-data-jpa、spring-orm 和 Hibernate 来支持 JPA。
- spring-boot-starter-amqp：通过 spring-rabbit 支持 AMQP。
- spring-boot-starter-actuator：添加适用于生产环境的功能，如性能指标和监测等功能。

Java Config 自动配置

Spring Boot 推荐采用基于 Java Config 的配置方式，而不是传统的 XML。例如，@Configuration、@Bean、@EnableAutoConfiguration、@ComponentScan、@PropertySource、@Repository、@Service、@RestController等。

快速上手

Spring Boot初体验

通过 Spring Boot，创建新的 Spring 应用变得非常容易，只需要简单的几个步骤就可以创建一个应用。

首先，我们先创建一个 POM 文件。

```
<?xml version="1.0" encoding="UTF-8" ?>  
<project xmlns="http://maven.apache.org/POM/4.0.0"
```

```
<groupId>org.springframework.boot</groupId>  
<artifactId>spring-boot-starter-parent</artifactId>  
<version>1.3.3-RELEASE</version>  
<parent>  
<groupId>com.lianggzone.demo</groupId>  
<artifactId>springboot-action</artifactId>  
<version>0.1</version>  
<packaging>war</packaging>  
<name>springboot-action</name>  
<dependencies>  
<dependency>  
<groupId>org.springframework.boot</groupId>  
<artifactId>spring-boot-starter</artifactId>
```

```

<dependency>
</dependency>
</dependencies>
</build>
</plugins>
</plugin>
</groupId>org.springframework.boot</groupId>
</artifactId>spring-boot-maven-plugin</artifactId>
</plugin>
</plugins>
</build>
</project></pre>
<p><span>然后，第二步，创建 Java 代码。</span></p>
<pre class="brush:java;toolbar:false">
@SpringBootApplication
public class ApplicationDemo {
    public static void main(String[] args) throws Exception {
        System.out.println("springApplication run!");
        SpringApplication.run(ApplicationDemo.class, args);
    }
}</pre>
<h3 id="toc_h3_8">快速搭建 web 工程</h3>
<p>通过 Spring Boot，创建 web 工程也非常简单。修改 POM 文件，添加依赖。</p>
<pre class="brush:xml;toolbar:false">
<groupId>org.springframework.boot</groupId>
<artifactId>spring-boot-starter-web</artifactId>
</dependency></pre>
<p><span>创建 Java 代码</span></p>
<pre class="brush:java;toolbar:false">
@RestController
@EnableAutoConfiguration
public class RestfulApiWebDemo {
    @RequestMapping("/")
    String home() {
        return "Hello World!";
    }
    public static void main(String[] args) throws Exception {
        SpringApplication.run(RestfulApiWebDemo.class, args);
    }
}</pre>
<p><span>直接运行 Java 类，或者可以通过“mvn spring-boot:run”在命令行启动该应用。会启动一个内嵌的 Tomcat 服务器运行在 8080 端口。访问</span><span><a href="http://ld246.com/forward?goto=http%3A%2F%2Flocalhost%3A8080%2F" target="blank" rel="nofollow ugc">http://localhost:8080</a></span><span>可以看到页面上显示“Hello World!”。</span><span>此外，在 POM 文件添加插件。</span></p>
<pre class="brush:xml;toolbar:false">
<groupId>org.springframework.boot</groupId>
<artifactId>spring-boot-maven-plugin</artifactId>
</plugin></pre>
<p><span>在添加了插件后，当运行“mvn package”进行打包时，会打

```

成一个可以直接运行的 JAR 文件，使用 “java -jar” 命令就可以直接运行。

>转自： [http://blog.720ui.com/2016/springboot_01_quickstart/](https://ld246.com/forward?goto=http%3A%2F%2Fblog.720ui.com%2F2016%2Fspringboot_01_quickstart%2F)
