**Spring Boot 热部署**

**实际开发中，修改某个页面数据或逻辑功能都需要重启应用。这无形中降低了开发效率，所以使用热部署是十分必要的。**

**什么是热部署？**

**应用启动后会把编译好的Class文件加载的虚拟机中，正常情况下在项目修改了源文件是需要全部重新编译并重新加载（需要重启应用）。而热部署就是监听Class文件的变动，只把发生修改的Class重新加载，而不需要重启应用，使得开发变得简便。**

## Spring Boot 配置实现热部署

### 第一种方式：maven 插件-springloaded

#### 在spring-boot-maven-plugin插件配置中添加springloaded依赖。

<build> <plugins> <plugin> <groupId>org.springframework.boot</groupId> <artifactId>spring-boot-maven-plugin</artifactId> <dependencies> <dependency> <groupId>org.springframework</groupId> <artifactId>springloaded</artifactId> <version>1.2.5.RELEASE</version> </dependency> </dependencies> </plugin> </plugins> </build>

#### 通过命令：mvn spring-boot:run 启动应用，修改controller中代码 或 修改html页面 后再刷新页面，会发现显示结果跟着变化。

#### 注意：第一种方式 对 mvn spring-boot:run 这种方式启动应用有效，而对直接运行Application.java启动应用无效。

注意1：如果发现没有热部署效果，则需要检查idea配置中有没有打开自动编译：

#### 打开 Preferences > Compiler , 勾选 Build project automatically

注意2：如果使用Thymeleaf模板引擎，需要把模板默认缓存设置为false

**在application.properties中添加：**

#禁止thymeleaf缓存（建议：开发环境设置为false，生成环境设置为true）

spring.thymeleaf.cache=false

### 第二种方式：依赖 spring-boot-devtools 热部署模块

#### pom.xml 添加依赖：

<!-- 热部署模块 -->

<dependency>

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

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

<optional>true</optional> <!-- 这个需要为 true 热部署才有效 -->

</dependency>

#### 重启应用，修改controller中代码 或 修改html页面 后再刷新页面，会发现显示结果跟着变化。

第三种方式

springboot 在IDEA中实现热部署

快捷键Ctrl + F9 即 --> 启动设置旁边的 build project 按钮。

这种方式只对修改页面有效。