

为什么要用SpringBoot ?

传统项目:整合SSH或者SSM,考虑配置文件、Jar

冲突问题、整合起来繁琐,

打包方式:打包成一个war放入到 tomcatwebapps目录下进行执行。

什么是 SpringBoot ?

Springboot是一个快速开发框架,能够帮助我们快速整合第三方框架(Maven依赖关系## Maven继承),内置嵌入Http服务器(Tomcat、 Jetty),默认嵌入 Tomcat服务器。最终以Java应用程序进行执

1、启动注解

SpringBoot启动原理：**SpringMVC注解方式启动，内置http服务器（默认是tomcat）**

@EnableAutoConfiguration 作用：扫包范围，默认是当前类所在的包

@ComponentScan 启动加扫包范围，可以加多个包[(basePackages = {"com.....","com....."})]

@SpringBootApplication 作用：等同于@EnableAutoConfiguration +

@ComponentScan 同级包加当前包

@RestController 表示：该类中所有的方法返回json格式 等同于 @Controller + @ResponseBody

@ResponseBody 作用于Controller类的方法上

@RestController与@Controller的区别：

@RestController 表示：该类中所有的方法返回json格式

@Controller 表示：该类中的所有方法返回为视图

2、SpringCloud与SpringBoot关系

SpringBoot是一个快速开发框架，能够帮助我们快速整合第三方常用框架，完全采用注解化、简化XML配置、最终以Java应用程序进行执行。

SpringCloud是一套目前完整微服务解决框架，功能非常强大。有注册中心、客户端调用工具、服务治理（负载均衡、断路器、分布式配置中心、网关、服务链路、消息总线等）

关系：

SpringBoot Web组件默认集成SpringMVC

SpringCloud依赖于SpringBoot实现微服务，使用SpringMVC编写微服务接口。

区别：

微服务开发 - - - >RPC远程通讯技术，服务治理。 微服务通讯技术：http + json (restful)轻量级

SpringCloud + SpringBoot 是微服务开发

SpringBoot 实现快速开发

3、SpringBoot与SpringMVC关系

SpringBoot Web组件默认集成SpringMVC，但是SpringBoot启动SpringMVC的时候没有传统的配置文件

SpringMVC3.0之后产生注解，支持注解方式启动SpringMVC（使用Java代码启动SpringMVC）

4、SpringBoot静态资源访问

图片、js、css等静态资源，在SpringBoot中默认提供静态资源目录位置需置于classpath下，目录名需符合

/static

/public

/resources

/META-INF/resources

例如：[在src/main/resources/目录下创建static目录放入静态资源，启动项目后通](#)

过:localhost:8080/图片名.jpg可以访问