

## 1. 什么是SpringBoot?

Spring Boot, 以 “约定大于配置” 为核心理念, 于 2014 年诞生。它旨在简化 Spring 应用的开发, 以“开箱即用”的方式, 快速搭建 Spring 应用。

## 2. SpringBoot的特征?

- 基于 Spring 4.x 版本
- 支持 Tomcat, Jetty, Undertow (默认使用 Tomcat, 可配置为 WAR 包)
- 提供 “starter” 依赖, 以简化配置
- 支持 Spring 3rd Party 集成
- 支持 于 产 生 可 执 行 的 可 执 行 文 件, 例 如 jar 包
- 代 理 器, 也 不 需 要 XML 配 置

## 3. 如何快速构建一个SpringBoot项目?

- 通过 Web 浏览器访问 <http://start.spring.io>
- 使用 Spring Tool Suite 使
- 使用 IntelliJ IDEA 使
- 使用 Spring Boot CLI 使

## 4. SpringBoot启动类注解?它是由哪些注解组成?

@SpringBootApplication

- @SpringBootConfiguration: 继承了 @Configuration , 标注为配置类
- @EnableAutoConfiguration: 启用自动配置, 也以 一个
- @SpringBootApplication(exclude = { DataSourceAutoConfiguration.class })
- @ComponentScan: Spring 组件

## 5. 什么是yaml?

YAML (/ˈjæməl/, 似camel ) 一个 , YAML 了  
他 , : C Python Perl

## 6. SpringBoot支持配置文件的格式?

1.properties

```
java.xiaokaxiu.name = xiaoka
```

2.yml

```
java:
  xiaokaxiu:
    name: xiaoka
```

## 7. SpringBoot启动方式?

1. main
2. 令 java -jar
3. mvn/gradle

## 8. SpringBoot需要独立的容器运行?

不 , 了 Tomcat/Jetty

## 9. SpringBoot配置途径?

1. 令
2. java:comp/env JNDI
3. JVM
4. 作

5. random.\* ( 他 , 以 们, \${random.  
long})
  6. 以 application.properties appliaction.yml 件
  7. application.properties appliaction.yml 件
  8. @PropertySource
  - 9.
- tips: 个 优 , 也 , 任何 优 会 低优

## 10. application.properties和application.yml文件可放位置?优先级?

1. , 于 /config
  2. ,
  3. , config
  4. , Classpath
- 个 优 ,优 会 优 低  
们 以 件 位 件

```
java -jar xiaoka.jar --spring.config.location=/home/application.yml
```

## 11. SpringBoot自动配置原理?

@EnableAutoConfiguration ( ) 了AutoConfigurationImportSelector, 中  
会 META-INF/spring.factories jar

## 12. SpringBoot热部署方式?

- spring-boot-devtools
- Spring Loaded
- Jrebel
- 

## 13. 「bootstrap.yml」 和 「application.yml」 ?

bootstrap.yml 优 于application.yml

## 14. SpringBoot如何修改端口号?

yaml中:

```
server :  
  port : 8888
```

properties:

```
server.port = 8888
```

令1:

```
java -jar xiaoka.jar --- server.port=8888
```

令2:

```
java -Dserver.port=8888 -jar xiaoka.jar
```

## 15. 开启SpringBoot特性的几种方式?

1. spring-boot-starter-parent
2. spring-boot-dependencies

## 16. SpringBoot如何兼容Spring项目?

:

```
@ImportResource(locations = {"classpath:spring.xml"})
```

## 17. SpringBoot配置监控?

```
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-actuator</artifactId>
</dependency>
```

## 18. 获得Bean装配报告信息访问哪个端点?

/beans

## 19. 关闭应用程序访问哪个端点?

/shutdown

, , 下

```
endpoints:
  shutdown:
    enabled: true
```

properties 也 以

## 20. 查看发布应用信息访问哪个端点?

/info

## 21. 针对请求访问的几个组合注解?

@PatchMapping

@PostMapping

@GetMapping

@PutMapping

@DeleteMapping

## 22. SpringBoot 中的starter?

以 , starter会 个 , 了  
事, 了  
且 会 个 , 们 , 会

## 23. SpringBoot集成Mybatis?

mybatis-spring-boot-starter

## 24. 什么是SpringProfiles?

我们从生产, (dev) (test) 上 (prod) 不会不  
Spring Profiles 件 (dev, test, prod ) bean 们以们  
什么 什么

## 25. 不同的环境的配置文件?

以 application-{profile}.properties/yml , 但 主 件application.properties,  
们不

- `application.properties`: 主 件
- `application-dev.properties`: 件
- `application-test.properties`: 件
- `application.prod.properties`: 产 件

## 26. 如何激活某个环境的配置?

们

yml:

```
spring:
  profiles:
    active: dev
```

properties:

```
spring.profiles.active=dev
```

令 :

```
java -jar xiaoka-v1.0.jar ---spring.profiles.active=dev
```

## 27. 编写测试用例的注解?

@SpringBootTest

## 28. SpringBoot异常处理相关注解?

@ControllerAdvice

@ExceptionHandler

## 29. SpringBoot 1.x 和 2.x区别?.....

1. SpringBoot 2 于Spring5 JDK8, Spring 1x 低
2. ,
3. SpringBoot2 件 低
4. 2.x 中 中 以 , 不
5. Actuator
6. CacheManager

## 30. SpringBoot读取配置相关注解有?

- @PropertySource
- @Value
- @Environment
- @ConfigurationProperties