Java 最常见的 208 道面试题：第十一模块答案

zy java经验总结 3月28日

十一、Spring Boot / Spring Cloud

104. 什么是 spring boot?

在Spring框架这个大家族中，产生了很多衍生框架，比如 Spring、SpringMvc框架等，Spring的核心内容在于控制反转(IOC)和依赖注入(DI),所谓控制反转并非是一种技术，而是一种思想，在操作方面是指在spring配置文件中创建<bean>，依赖注入即为由spring容器为应用程序的某个对象提供资源，比如 引用对象、常量数据等。

SpringBoot是一个框架，一种全新的编程规范，他的产生简化了框架的使用，所谓简化是指简化了Spring众多框架中所需的大量且繁琐的配置文件，所以 SpringBoot是一个服务于框架的框架，服务范围是简化配置文件。

105. 为什么要用 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
- Spring的不足

106. spring boot 核心配置文件是什么?

Spring Boot提供了两种常用的配置文件：

- properties文件
- yml文件

107. spring boot 配置文件有哪几种类型？它们有什么区别?

Spring Boot提供了两种常用的配置文件，分别是properties文件和yml文件。相对于properties文件而言，yml文件更年轻，也有很多的坑。可谓成也萧何败萧何，yml通过空

格来确定层级关系，使配置文件结构跟清晰，但也会因为微不足道的空格而破坏了层级关系。

108. spring boot 有哪些方式可以实现热部署?

SpringBoot热部署实现有两种方式：

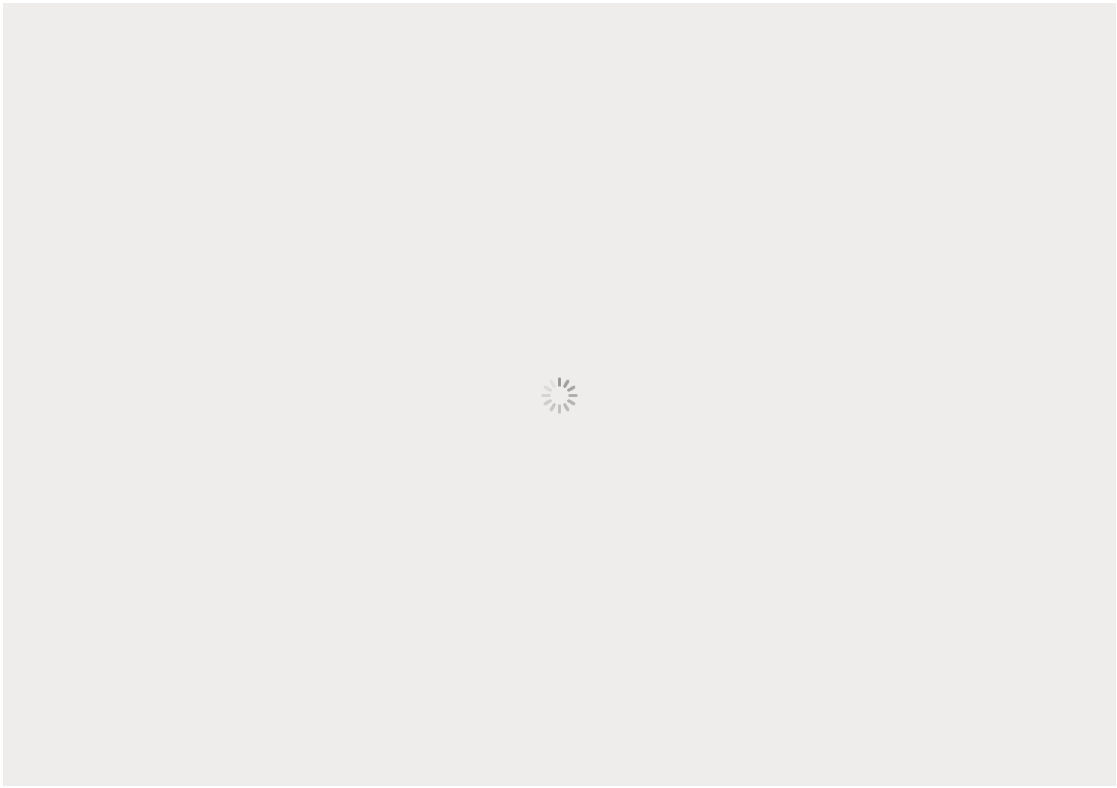
①. 使用spring loaded

在项目中添加如下代码：

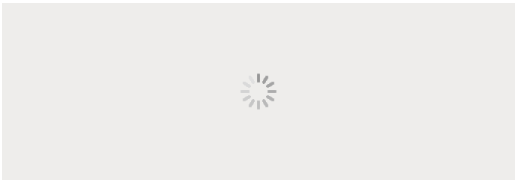
```
<build>
        <plugins>
            <plugin>
                <!-- springBoot编译插件-->
                <groupId>org.springframework.boot</groupId>
                <artifactId>spring-boot-maven-plugin</artifactId>
                <dependencies>
                    <!-- spring热部署 -->
                    <!-- 该依赖在此处下载不下来，可以放置在build标签外部下载完成后
                    <dependency>
                        <groupId>org.springframework</groupId>
                        <artifactId>springloaded</artifactId>
                        <version>1.2.6.RELEASE</version>
                    </dependency>
                </dependencies>
            </plugin>
        </plugins>
    </build>
```

添加完毕后需要使用mvn指令运行：

首先找到IDEA中的Edit configurations ,然后进行如下操作： （点击左上角的"+",然后选择maven将出现右侧面板，在红色划线部位输入如图所示指令，你可以为该指令命名(此处命名为MvnSpringBootRun)）



点击保存将会在IDEA项目运行部位出现，点击绿色箭头运行即可



②. 使用spring-boot-devtools

在项目的pom文件中添加依赖：

```
<!--热部署jar-->
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-devtools</artifactId>
</dependency>
```

然后：使用 shift+ctrl+alt+"/" （IDEA 中的快捷键） 选择 "Registry" 然后勾选 compiler.automake.allow.when.app.running

109. jpa 和 hibernate 有什么区别?

- JPA Java Persistence API，是Java EE 5的标准ORM接口，也是ejb3规范的一部分。
- Hibernate，当今很流行的ORM框架，是JPA的一个实现，但是其功能是JPA的超集。
- JPA和Hibernate之间的关系，可以简单的理解为JPA是标准接口，Hibernate是实现。那么Hibernate是如何实现与JPA的这种关系的呢。Hibernate主要是通过三个组件来实现的，及hibernate-annotation、hibernate-entitymanager和hibernate-core。
- hibernate-annotation是Hibernate支持annotation方式配置的基础，它包括了标准的JPA annotation以及Hibernate自身特殊功能的annotation。
- hibernate-core是Hibernate的核心实现，提供了Hibernate所有的核心功能。
- hibernate-entitymanager实现了标准的JPA，可以把它看成hibernate-core和JPA之间的适配器，它并不直接提供ORM的功能，而是对hibernate-core进行封装，使得Hibernate符合JPA的规范。

110. 什么是 spring cloud?

从字面理解，Spring Cloud 就是致力于分布式系统、云服务的框架。

Spring Cloud 是整个 Spring 家族中新的成员，是最近云服务火爆的必然产物。

Spring Cloud 为开发人员提供了快速构建分布式系统中一些常见模式的工具，例如：

- 配置管理
- 服务注册与发现
- 断路器
- 智能路由
- 服务间调用
- 负载均衡
- 微代理
- 控制总线
- 一次性令牌
- 全局锁
- 领导选举
- 分布式会话

- 集群状态
- 分布式消息
- ……

使用 Spring Cloud 开发人员可以开箱即用的实现这些模式的服务和应用程序。这些服务可以任何环境下运行，包括分布式环境，也包括开发人员自己的笔记本电脑以及各种托管平台。

111. spring cloud 断路器的作用是什么?

在Spring Cloud中使用了Hystrix 来实现断路器的功能，断路器可以防止一个应用程序多次试图执行一个操作，即很可能失败，允许它继续而不等待故障恢复或者浪费 CPU 周期，而它确定该故障是持久的。断路器模式也使应用程序能够检测故障是否已经解决，如果问题似乎已经得到纠正，应用程序可以尝试调用操作。

断路器增加了稳定性和灵活性，以一个系统，提供稳定性，而系统从故障中恢复，并尽量减少此故障的对性能的影响。它可以帮助快速地拒绝对一个操作，即很可能失败，而不是等待操作超时（或者不返回）的请求，以保持系统的响应时间。如果断路器提高每次改变状态的时间的事件，该信息可以被用来监测由断路器保护系统的部件的健康状况，或以提醒管理员当断路器跳闸，以在打开状态。

112. spring cloud 的核心组件有哪些?

①. 服务发现——Netflix Eureka

一个RESTful服务，用来定位运行在AWS地区（Region）中的中间层服务。由两个组件组成：Eureka服务器和Eureka客户端。Eureka服务器用作服务注册服务器。Eureka客户端是一个java客户端，用来简化与服务器的交互、作为轮询负载均衡器，并提供服务的故障切换支持。Netflix在其生产环境中使用的是另外的客户端，它提供基于流量、资源利用率以及出错状态的加权负载均衡。

②. 客服端负载均衡——Netflix Ribbon

Ribbon，主要提供客户侧的软件负载均衡算法。Ribbon客户端组件提供一系列完善的配置选项，比如连接超时、重试、重试算法等。Ribbon内置可插拔、可定制的负载均衡组件。

③. 断路器——Netflix Hystrix

断路器可以防止一个应用程序多次试图执行一个操作，即很可能失败，允许它继续而不等待故障恢复或者浪费 CPU 周期，而它确定该故障是持久的。断路器模式也使应用程序能够检测故障是否已经解决。如果问题似乎已经得到纠正，应用程序可以尝试调用操作。

④. 服务网关——Netflix Zuul

类似nginx，反向代理的功能，不过netflix自己增加了一些配合其他组件的特性。

⑤. 分布式配置——Spring Cloud Config

这个还是静态的，得配合Spring Cloud Bus实现动态的配置更新。

（完）