1. 创建一个普通的Spring Boot项目，叫做config-server,添加依赖

<dependency>  
 <groupId>org.springframework.cloud</groupId>  
 <artifactId>spring-cloud-config-server</artifactId>  
</dependency>

1. 在入口类上添加@EnableConfigServer注解，表示开启配置中心服务端功能
2. 在application.properties中配置一下git仓库的信息，使用GitHub（当然也可以使用码云），这里需要先在Github上创建一个名为scConfig的项目，创建好之后，再做如下配置：

spring.application.name=config-server

server.port=2007

spring.cloud.config.server.git.uri=https://github.com/lenve/scConfig.git

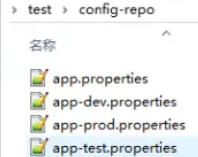
spring.cloud.config.server.git.search-paths=config-repo

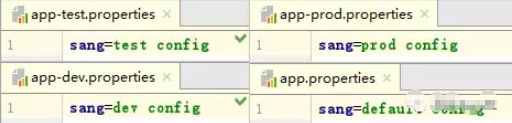
spring.cloud.config.server.git.username=username

spring.cloud.config.server.git.password=password

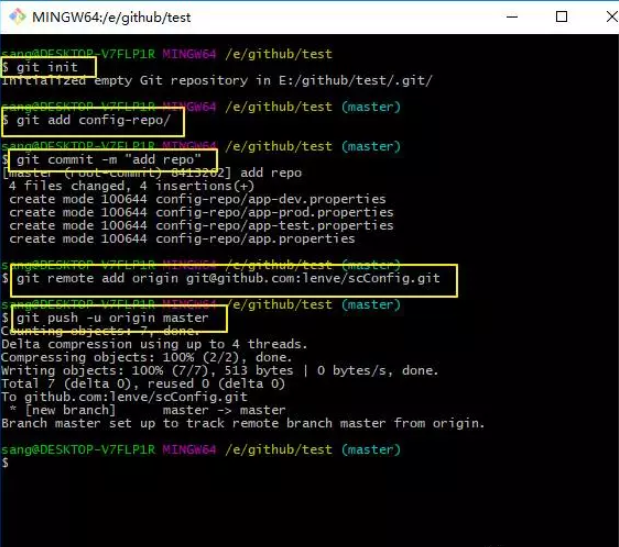
1.uri表示配置中心所在仓库的位置  
2.search-paths表示仓库下的子目录  
3.username表示你的GitHub用户名  
4.password表示你的GitHub密码

4.在github上设置好配置中心，首先我在本地找一个空文件夹，在该文件夹中创建一个文件夹叫config-repo，然后在config-repo中创建四个配置文件，如下：





5.执行如下命令将本地文件同步到Github仓库中



6. 如此之后，我们的配置文件就上传到GitHub上了。此时启动我们的配置中心，通过/{application}/{profile}/{label}就能访问到我们的配置文件了，其中application表示配置文件的名字，对应我们上面的配置文件就是app，profile表示环境，我们有dev、test、prod还有默认，label表示分支，默认我们都是放在master分支上,访问<http://localhost:2007/app/prod/master>

**客户端配置**

1. 创建一个普通的Spring Boot工程config-client，添加依赖

<dependency>

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

       <artifactId>spring-boot-starter-web</artifactId>

   </dependency>

   <dependency>

       <groupId>org.springframework.cloud</groupId>

       <artifactId>spring-cloud-starter-config</artifactId>

</dependency>

2. 创建bootstrap.properties文件，来获取配置信息，注意**这些信息一定要放在bootstrap.properties文件中才有效**

spring.application.name=app

spring.cloud.config.profile=dev

spring.cloud.config.label=master

spring.cloud.config.uri=http://localhost:2007/

server.port=2008

这里的name对应了配置文件中的application部分，profile对应了profile部分，label对应了label部分，uri则表示配置中心的地址。

1. 配置完成之后创建一个测试Controller

@RestController

@RefreshScope

public class HelloController {

   @Value("${sang}")

   String sang;

   @Autowired

   Environment env;

   @RequestMapping("/sang")

   public String sang() {

       return this.sang;

   }

   @RequestMapping("/sang2")

   public String sang2() {

       return env.getProperty("sang", "未定义");

   }

}

访问http://localhost:2008/sang