

spring-cloud-config client 配置客户端

作者: [ws](#)

原文链接: <https://ld246.com/article/1536733279191>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

Maven导入

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

1.增加bootstrap.properties

```
# 配置服务器url
spring.cloud.config.uri = http://localhost:9090/
# 远程地址properties 前缀 [ws.properties]
spring.cloud.config.name = ws
# 远程地址properties -后缀 [ws-dev.properties]
spring.cloud.config.profile = default
# git仓库分支
spring.cloud.config.label = master
```

2.修改application.properties

```
spring.application.name = client
#springcloud-Finchley.SR1版本中,意思是:使用包含暴露的服务端口
management.endpoints.web.exposure.include = env,refresh
```

3.修改Application.java

```
package com.gupao.springcloudclientconfig;

import org.omg.CORBA.Environment;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.beans.factory.annotation.Value;
import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.RestController;

@SpringBootApplication
@RestController
public class SpringCloudClientConfigApplication {

    @Value("${my.name}")
    public String name;
    public static void main(String[] args) {
```

```
    SpringApplication.run(SpringCloudClientConfigApplication.class, args);
}

@GetMapping("/config")
public String getConfig(){
    return name;
}
}
```

4.测试

<http://localhost:8080/config>,成功显示default.

5.修改配置生效

<http://localhost:8080/actuator/env> -> [search ws.properties](#) -> 通过POST方法请求localhost:8080/actuator/refresh 刷新配置文件,发现已变更则成功!

6.修改后刷新bean

在需要修改的类中增加@RefreshScope注解,就会开启刷新@Value的值