# SpringCould[Server]

|  |
| --- |
| /\*\*  \* 配置了git远程连接话,可以使用域名+端口号+git配置文件的键,即可访问数据  \* 例如：http://localhost:8888/foo/dev  \* 访问的是https://github.com/forezp/SpringcloudConfig/里面的所有配置文件,键为foo  \* **@author** zengfanjie  \*  \*/  @SpringBootApplication  @EnableConfigServer //开启配置服务  **public** **class** ConfigServerApplication {  **public** **static** **void** main(String[] args) {  SpringApplication.*run*(ConfigServerApplication.**class**, args);  }  } |
| spring.application.name=config-server  server.port=8888  ## 远程地址  spring.cloud.config.server.git.uri=https://github.com/forezp/SpringcloudConfig/  ## 名为respo的库  spring.cloud.config.server.git.searchPaths=respo  ## master 分支  spring.cloud.config.label=master  ## 如果是公开的库的话,可以不用填写,如果是私人的话,那么要填写  spring.cloud.config.server.git.username=  spring.cloud.config.server.git.password= |

# SpringCould[Client]

|  |
| --- |
| /\*\*  \* config-client从config-server获取了foo的属性，而config-server是从git仓库读取的  \* **@author** zengfanjie  \*  \*/  @SpringBootApplication  @RestController  **public** **class** ConfigClientApplication {  **public** **static** **void** main(String[] args) {  SpringApplication.*run*(ConfigClientApplication.**class**, args);  }  @Value("${foo}")  String foo;  @RequestMapping(value = "/hi")  **public** String hi(){  **return** foo;  }  } |
| spring.application.name=config-client  ## 指明远程仓库的分支  spring.cloud.config.label=master  ## dev开发环境配置文件 test测试环境 pro正式环境  spring.cloud.config.profile=dev  ## 指明配置服务中心的网址。  spring.cloud.config.uri= http://localhost:8888/  server.port=8881 |