

# 创建消费rest的工程

- 参考这个：<https://spring.io/guides/gs/consuming-rest/>
- 我从这里开始创建工程的：<https://start.spring.io/>
- 我选了打包格式是war，另外选择了“web”和“actuator”
- 我用的工程代码在这里：<https://github.com/caishu/rs-consumer.git>
- 注意QuoteController.java的22行，用的地址是刚才部署的服务的内部域名；端口是8080
- 另外注意代码里我添加了WEB-INF/jboss-web.xml这个文件。这个文件是把这个war映射到/的访问；否则wildfly自动部署war会用war文件名做uri的路径
- 这一步的验证办法是，mvn clean package看到BUILD SUCCESS就可以
- 另外注意这里我还定义了一个新的rest，访问这个时候，会调用刚才部署的greeting

# 部署rest消费服务

- `oc new-app wildfly:latest~. --name wildfly-rs-consumer` 这里用developer用户执行
- 把刚才`mvn clean package`编译出来的war文件（在target目录里）用scp传到linux上，假设在/root/目录里
- 在/root/目录执行：`oc start-build wildfly-rs-consumer --from-file=rsConsumer.war`
- `oc expose svc wildfly-rs-consumer` 。这个是创建route，使外部可以访问这个新的服务
- 在mac用浏览器访问<http://wildfly-rs-consumer-myproject.127.0.0.1.nip.io/health> 应该可以看到status up的字样。这样，就是部署成功了