# 消息队列(MQ)的使用说明

## 说明

mq模块封装了框架ActiveMQ,

* 使用配置化,比较方便,
* 在activemq两种基本模式queue和topic上实现了第三种request-response模式

粗略介绍可以参考 [activemq的三种基本通信方式总结](http://blog.csdn.net/wilsonke/article/details/42776057)

源代码位于: <https://github.com/okMyMan/learning/tree/master/alvin-mq>

需要打包或者找我

## 步骤一

添加依赖包

<dependency>  
 <groupId>com.utils</groupId>  
 <artifactId>ooteco-mq</artifactId>  
 <version>1.0-SNAPSHOT</version>  
</dependency>

## 步骤二

定义生产者

<!--生产者-->  
<bean id="testMQHandler" class="com.utils.mq.impl.active.ActiveMQClientHandler">  
 <constructor-arg index="0" ref="testConfig" />  
</bean>  
<bean class="com.utils.mq.bean.MessageServiceConfig" id="testConfig">  
 <property name="mqUrl" value="${MQ\_URL}" />  
 <property name="serviceType" value="OOTECO\_MQ\_TEST" /> <!-- 生产者,消费者保持一致-->  
 <property name="serviceName" value="${NODE\_NAME}" /> <!-- 建议写自己的ip地址,方便本地调试 -->  
 <property name="broadcastFlag" value="false" /><!-- 是否广播 -->  
 <property name="responseFlag" value="true" /> <!-- 是否同步异步 true是同步 -->  
</bean>

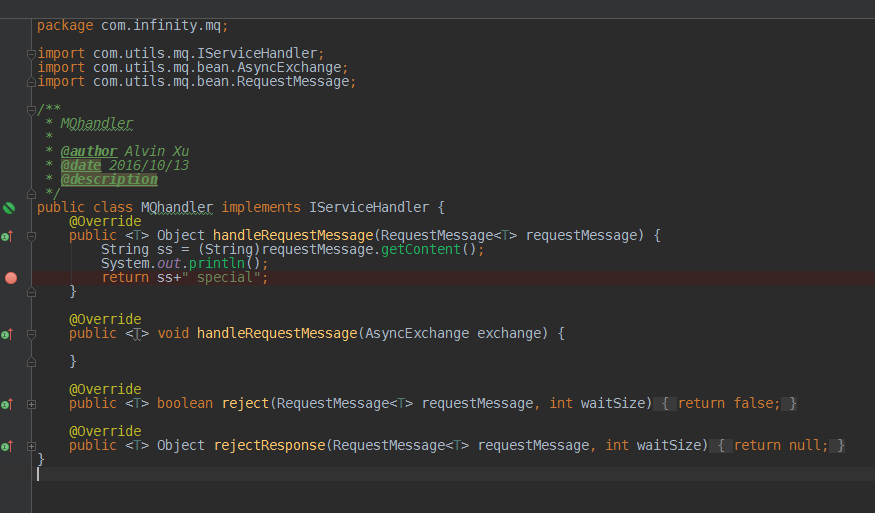
## 步骤三

定义消费者

<!-- 消费者配置 -->  
<bean class="com.utils.mq.bean.MessageServiceConfig" id="ticketConfig">  
 <property name="mqUrl" value="${MQ\_URL}" />  
 <property name="serviceType" value="OOTECO\_MQ\_TEST" /><!-- -->  
 <property name="serviceName" value="${NODE\_NAME}" />  
 <property name="broadcastFlag" value="false" /><!-- 是否广播 -->  
 <property name="responseFlag" value="true" /> <!-- 是否同步异步 true是同步 -->  
</bean>  
  
<!-- 消费者-->  
<bean id="autoTicketService" class="com.utils.mq.impl.active.ActiveMQServerHandler">  
 <constructor-arg index="0" ref="ticketConfig" />  
 <constructor-arg index="1" ref="mqHandler" />  
</bean>  
  
<!-- 消费者者处理方法-->  
<bean id="mqHandler" class="com.infinity.mq.MQhandler">  
</bean>

## 步骤四:

定义处理方法类MQhandler



**这部分可以忽略**

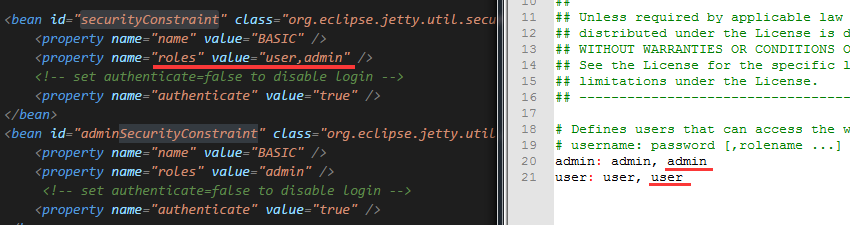
后台管理页面

<http://192.168.118.4:8161/admin/>可以查看queue,topic,consumer,producer的详情

**管理后台的口令密码设置**

文件jetty.xml设置角色

文件jetty-realm.properties设置角色用户名,密码



**ACTIVEMQ主题、队列设置用户名密码**

(1)修改activemq.xml配置,需要新增一个插件,在<broker>节点里面<systemUsage>节点前面添加如下

**<plugins>              <simpleAuthenticationPlugin>                  <users>                      <authenticationUser username="${activemq.username}" password="${activemq.password}" groups="users,admins"/>                  </users>              </simpleAuthenticationPlugin>           </plugins>**

(2)用户名密码文件为:credentials.properties

activemq.username=ooteco

activemq.password=ooteco

guest.password=guest