**一、内容**

**1.1邮件服务设计**

阿里云邮件推送服务提供了发送邮件良好的接口，只需下载其sdk用代码调用即可。

**1.2 web service设计**

web service服务器端用阿里云邮件推送服务实现发送邮件及验证邮箱的函数，提供给客户端调用。

**1.3 java GUI设计**

一个简单的GUI发送邮箱界面作为客户端，上面实现了发送邮件和验证邮箱的功能，直接调用web service服务器端的函数。

**1.4 网页端设计**

网页端设计发送邮件和验证邮箱的网页作为客户端，jsp后端直接调用web service服务。

**二、实现与实验**

**2.1 软件实现**

采用阿里云邮件服务，用web service实现 sendEmail(String\_url,String\_payload)发送单个邮件、sendEmailBatch(String[]\_url,String\_payload)批量发送邮件、validateEmailAddress(String\_url)验证邮件有效性三个函数；写2个客户端，一个javaGUI，一个网页，客户端远程调用web service的函数实现邮件服务。

**2.2 实验环境**

开发工具：eclipse

开发语言：java

实验架构：web service

第三方服务：阿里云邮件推送服务

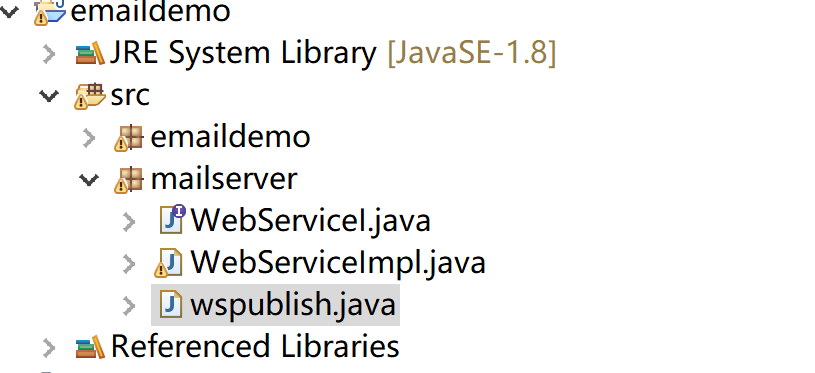
**2.3 实验步骤**

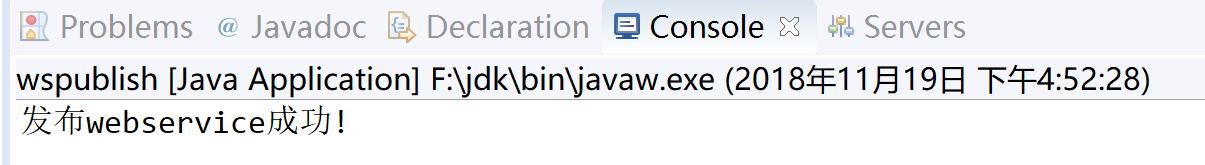
1. 运行服务器程序emaildemo中的wspublish.java发布web service服务
2. 登陆<http://127.0.0.1:8989/Webservice?wsdl>查看服务是否发布成功，若出现xml格式文件，则发布成功
3. 运行java GUI客户端程序mailclient中的wsclient.java，出现界面测试发送邮件和验证邮箱功能
4. 用tomcat运行网页端程序Emailclient，访问网页测试发送邮件和验证邮箱

**三、结果与讨论**

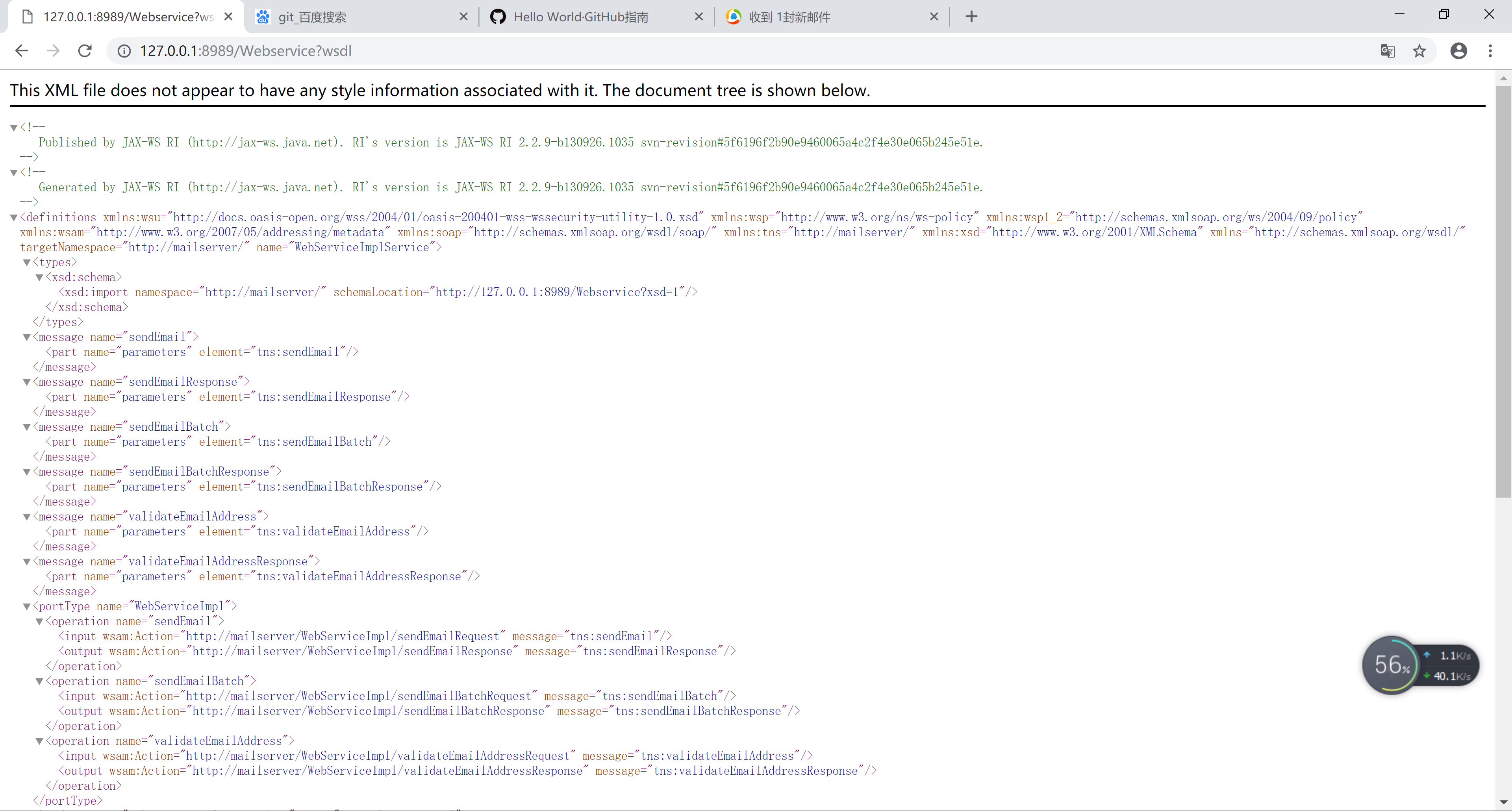
**3.1 结果**

（1）运行服务器程序emaildemo中的wspublish.java发布web service服务



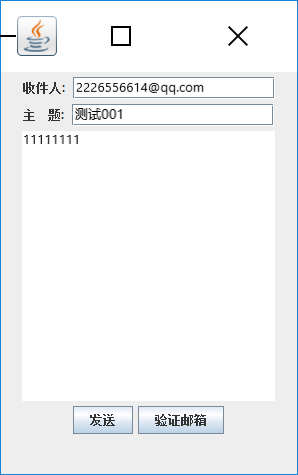


（2）登陆http://127.0.0.1:8989/Webservice?wsdl查看服务是否发布成功

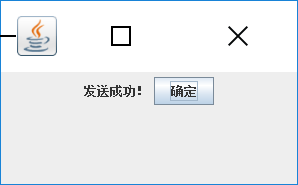


出现xml格式文件，说明发布web service服务成功。

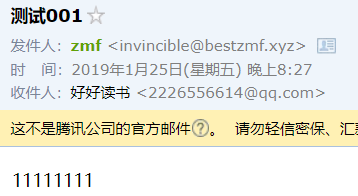
（3）运行java GUI客户端程序mailclient中的wsclient.java并测试



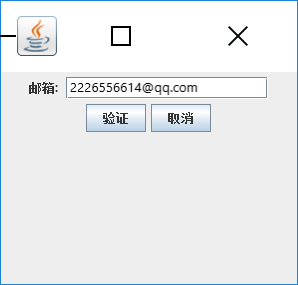
点击发送



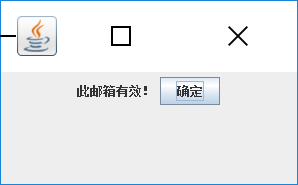
在qq邮箱中接收到对应邮件，说明发送成功



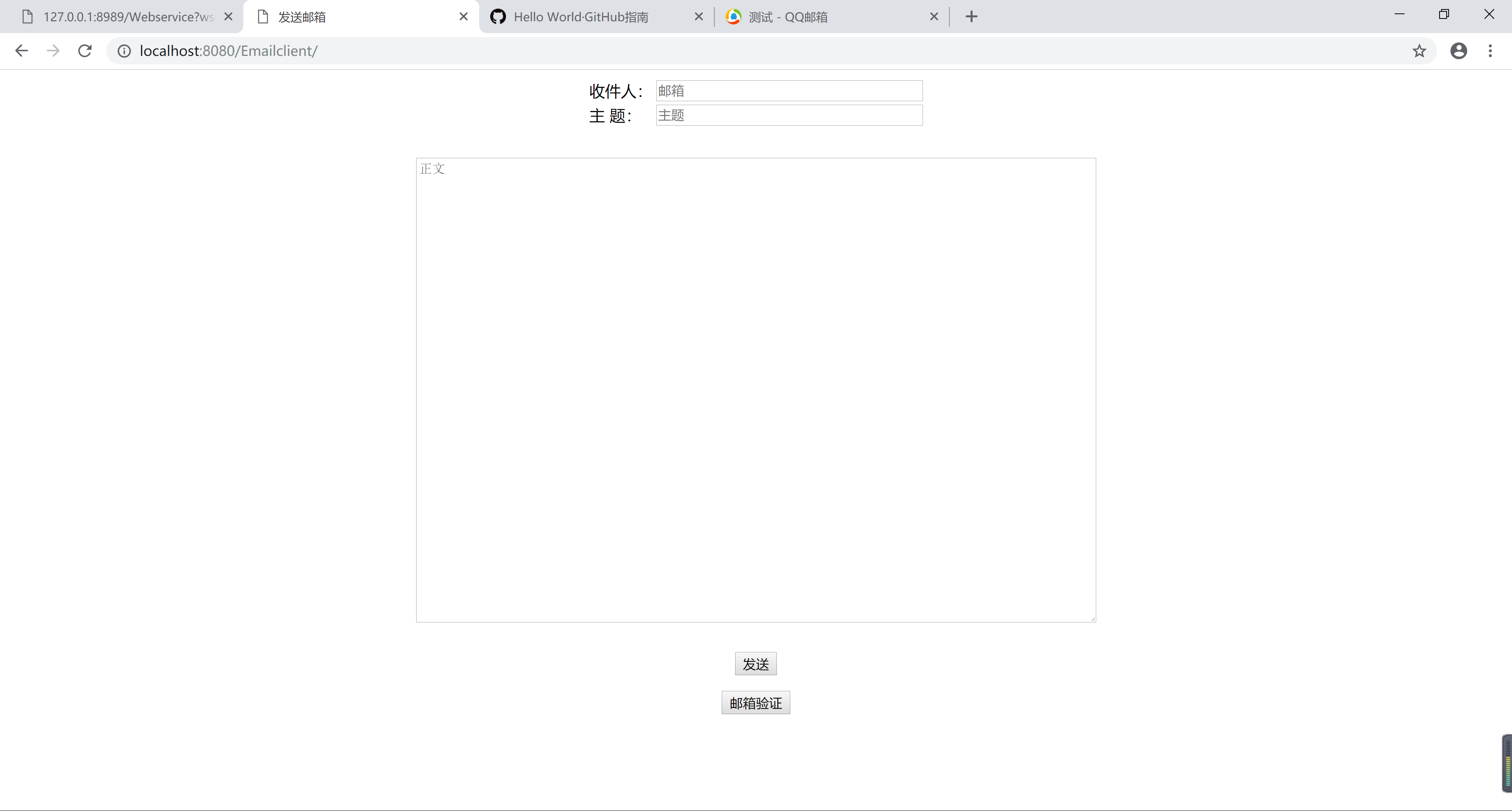
点击验证邮箱测试出现验证界面



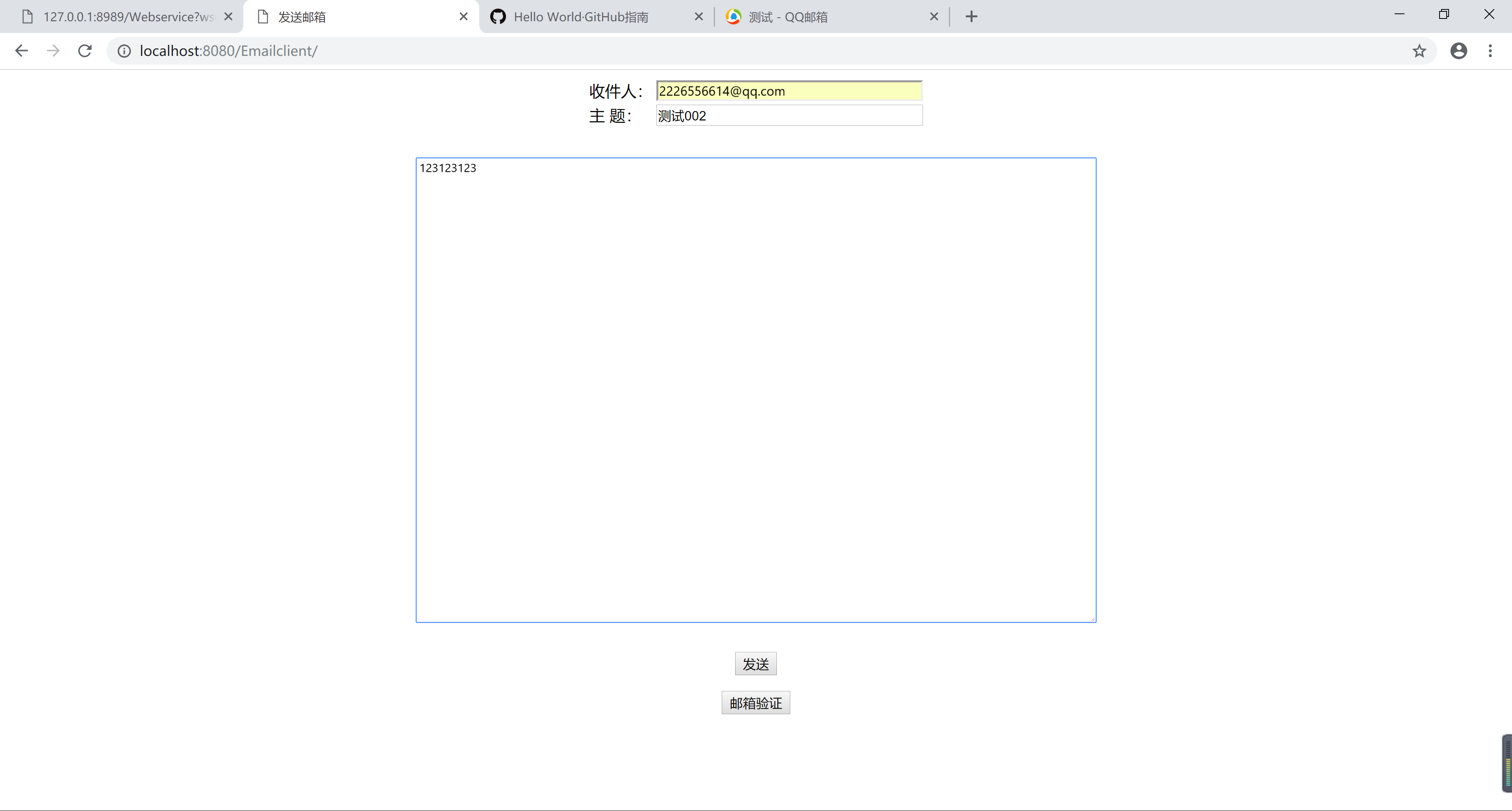
点击验证



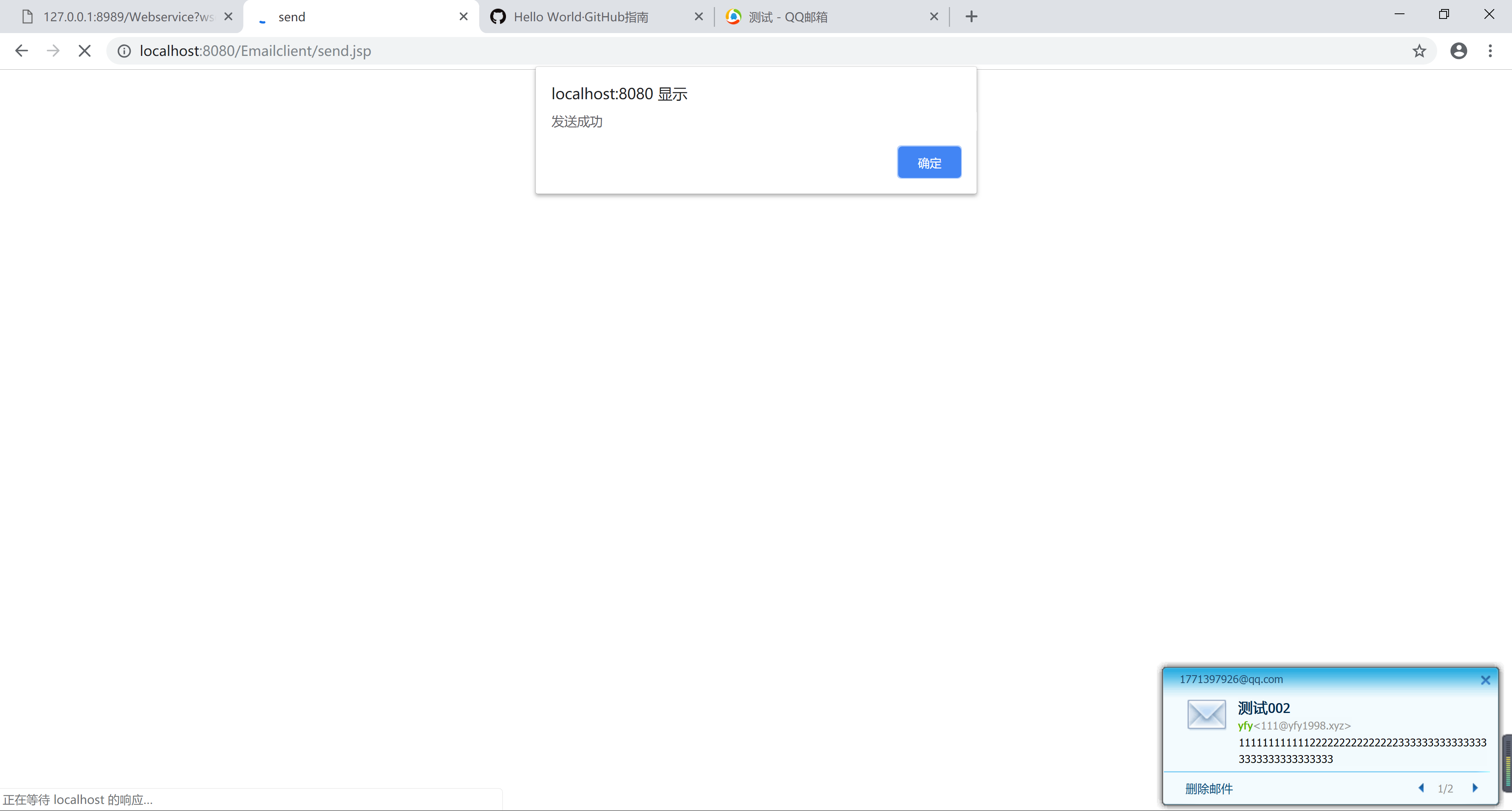
（4）用tomcat运行网页端程序Emailclient并测试



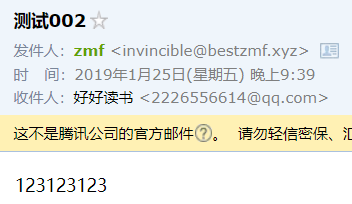
填写邮件内容发送邮件测试



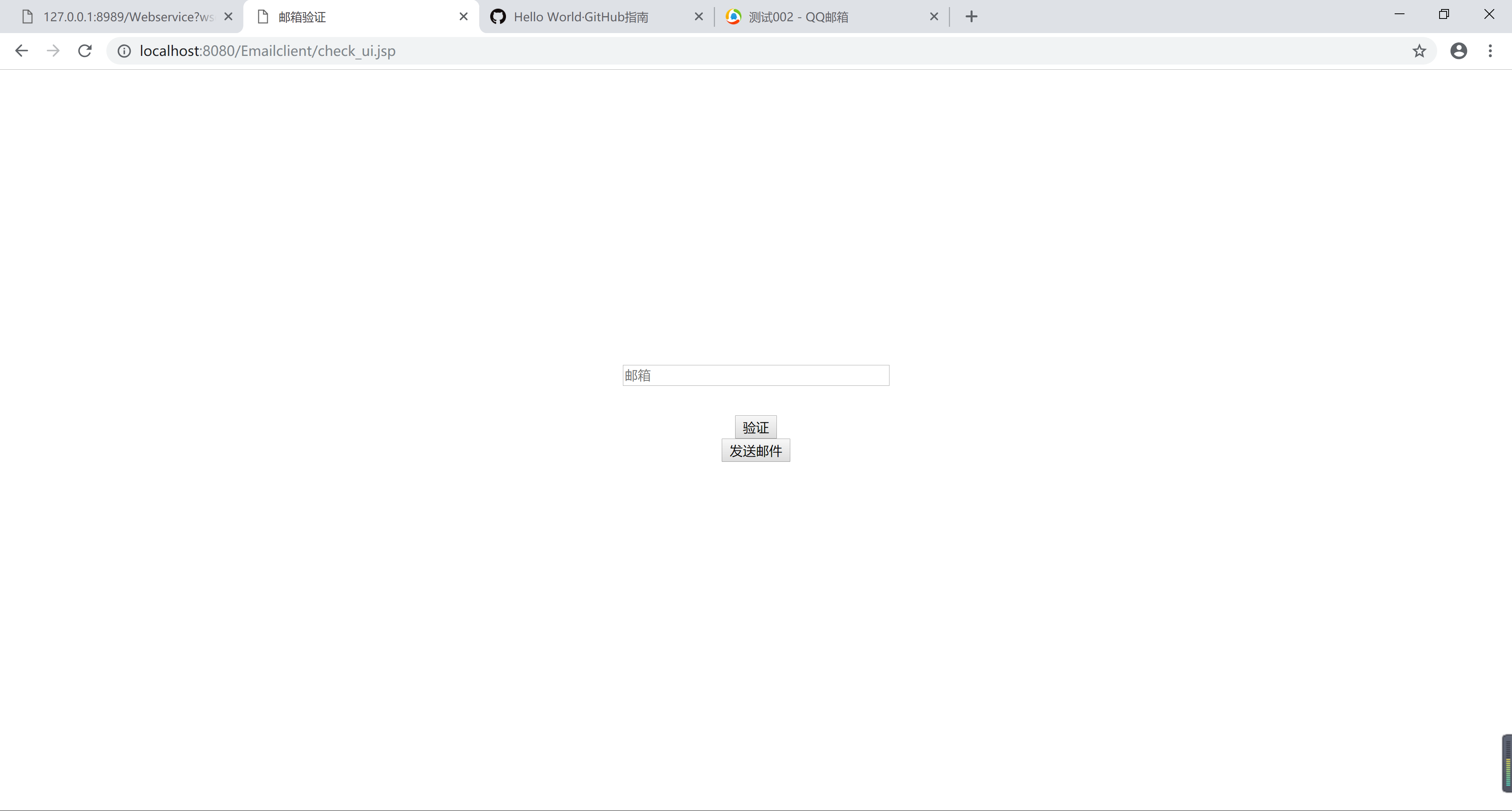
点击发送



查看邮箱的邮件



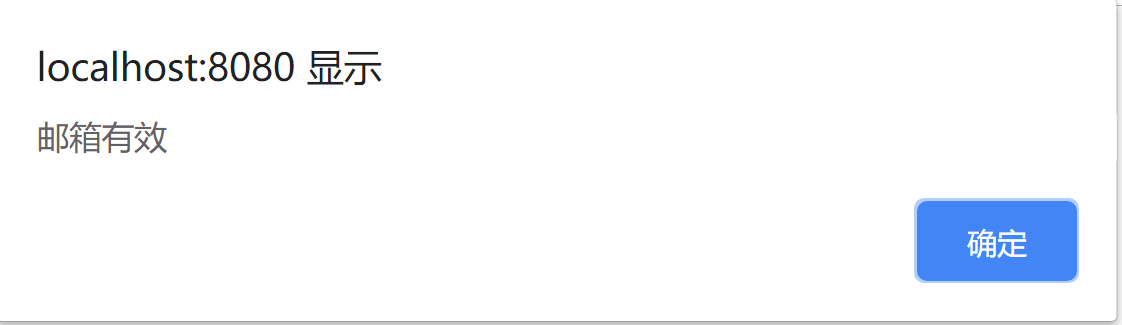
点击验证邮箱进入验证邮箱界面



填写一个邮箱并验证

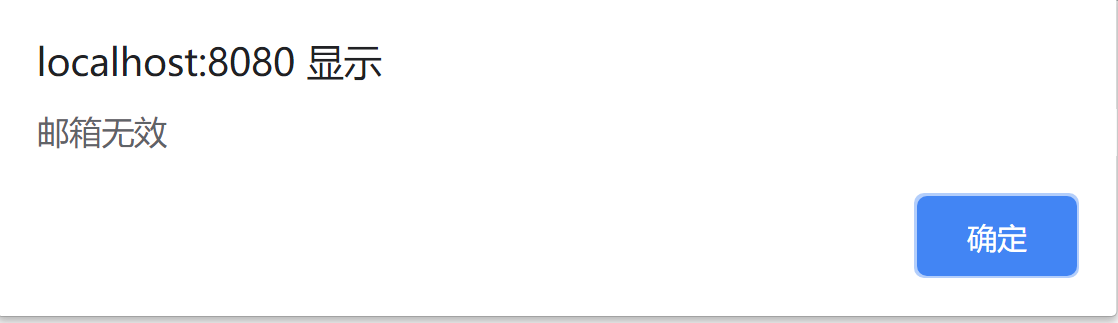


点击验证



换一个无效的邮箱验证





**四、附件**

（emaildemo为服务器程序、emailclient为javaGUI客户端程序、Emailclient为网页端程序）

**核心代码：**

//发送单一邮件

@Override

**public** **boolean** sendEmail(String s1, String s2, String s3) {

// **TODO** Auto-generated method stub

**if**(!validateEmailAddress(s1)) **return** **false**;

IClientProfile profile = DefaultProfile.*getProfile*("cn-hangzhou", "LTAIhCugHZ2ov8tb", "LQPJxqqXXzTVN5mnYYMnAXB1wbNZJY");

IAcsClient client = **new** DefaultAcsClient(profile);

SingleSendMailRequest request = **new** SingleSendMailRequest();

**try** {

request.setAccountName("invincible@bestzmf.xyz");

request.setFromAlias("zmf");

request.setAddressType(1);

request.setTagName("控制台创建的标签");

request.setReplyToAddress(**true**);

request.setToAddress(s1);

//可以给多个收件人发送邮件，收件人之间用逗号分开，批量发信建议使用BatchSendMailRequest方式

//request.setToAddress("邮箱1,邮箱2");

request.setSubject(s2);

request.setHtmlBody(s3);

SingleSendMailResponse httpResponse = client.getAcsResponse(request);

} **catch** (ServerException e) {

**return** **false**;

// e.printStackTrace();

}

**catch** (ClientException e) {

**return** **false**;

//e.printStackTrace();

}

**return** **true**;

}

//发送批量邮件

@Override

**public** **boolean** sendEmailBatch(String s1, String s2, String s3) {

// **TODO** Auto-generated method stub

IClientProfile profile = DefaultProfile.*getProfile*("cn-hangzhou", "LTAIhCugHZ2ov8tb", "LQPJxqqXXzTVN5mnYYMnAXB1wbNZJY");

IAcsClient client = **new** DefaultAcsClient(profile);

SingleSendMailRequest request = **new** SingleSendMailRequest();

**try** {

request.setAccountName("invincible@bestzmf.xyz");

request.setFromAlias("zmf");

request.setAddressType(1);

request.setTagName("控制台创建的标签");

request.setReplyToAddress(**true**);

//可以给多个收件人发送邮件，收件人之间用逗号分开，批量发信建议使用BatchSendMailRequest方式

request.setToAddress(s1);

request.setSubject(s2);

request.setHtmlBody(s3);

SingleSendMailResponse httpResponse = client.getAcsResponse(request);

} **catch** (ServerException e) {

**return** **false**;

// e.printStackTrace();

}

**catch** (ClientException e) {

**return** **false**;

//e.printStackTrace();

}

**return** **true**;

}

//邮箱验证

@Override

**public** **boolean** validateEmailAddress(String str) {

// **TODO** Auto-generated method stub

String pat="\\w+@\\w+(\\.\\w{2,3})\*\\.\\w{2,3}";

Pattern p=Pattern.*compile*(pat);

Matcher m=p.matcher(str);

**if**(m.matches()){

**return** **true**;

}**else**{

**return** **false**;

}

}

1. **相关链接**
2. 开通

<https://www.aliyun.com/product/directmail?spm=5176.10695662.1174764.1.26661d09hoP5Gr>

1. 参考文档

<https://help.aliyun.com/document_detail/29414.html?spm=5176.10695662.1174764.3.26661d09hoP5Gr>

1. maven依赖

<https://developer.aliyun.com/sdk/java.html?spm=5176.7946885.J_8058803260.1259.42d2fa4bQGIAHw>