

JavaMail
JavaMail API □□
JavaMail API □□□□
JavaMail API □□□
JavaMail API □□□□□
JavaMail API □□□□□□□
JavaMail API □□□□□□□□
JavaMail API □□□□HTML□□□□
JavaMail □□□□□□□□□□
JavaMail □□□□□□□
JavaMail □□□□□□□
JavaMail □□□□□□□
JavaMail □□□□□□/□□
JavaMail □□□□□□/□□
JavaMail □□□□□□□
JavaMail Gmail SMTP□□□□
JavaMail □□□□□□□□
JavaMail □□□□□
JavaMail □□□□□
JavaMail SMTP□□□□
JavaMail IMAP□□□□
JavaMail POP3□□□□

### JavaMail 退回邮件

¥ 我要打赏 作者：张得山 JavaMail 评论：0 条 Windows技术QQ群：227270512



消息可以被退回的几个原因。这个问题是在rfc1211深度讨论。只有服务器可确定特定的邮箱或用户名的不存在。当服务器检测到错误，它会返回一个消息，指出失败的原邮件的发件人的原因。

有覆盖发送状态通知许多互联网标准，但大量的服务器不使用特殊技术来返还这些失败消息支持这些新的标准代替。因此，它变得非常难以关联与导致问题的原始邮件的退回邮件。

JavaMail 包括用于解析传递状态通知的支持。有许多技术和试探法来处理这个问题。其中一个是可变的信封返回路径的技术。可以设置的返回路径中的封装器，如下面的例子。这是在弹跳邮件被发送到的地址。您可能需要将其设置为一个通用的地址，而不同：头，这样您就可以处理远程退回。这可以通过设置mail.smtp.from属性在JavaMail中。

### 创建Java类

创建一个Java类文件SendEmail，是其内容如下：

```
import java.util.Properties;

import javax.mail.Message;
import javax.mail.PasswordAuthentication;
import javax.mail.Session;
import javax.mail.Transport;
import javax.mail.internet.InternetAddress;
import javax.mail.internet.MimeMessage;

public class SendEmail {
    public static void main(String[] args) throws Exception {
        String smtpServer = "smtp.gmail.com";
        int port = 587;
        final String userid = "youraddress";//change accordingly
        final String password = "*****";//change accordingly
        String contentType = "text/html";
        String subject = "test: bounce an email to a different address " +
            "from the sender";

        String from = "youraddress@gmail.com";
        String to = "bouncere@fauxmail.com";//some invalid address
        String bounceAddr = "toaddress@gmail.com";//change accordingly
        String body = "Test: get message to bounce to a separate email address";

        Properties props = new Properties();

        props.put("mail.smtp.auth", "true");
        props.put("mail.smtp.starttls.enable", "true");
        props.put("mail.smtp.host", smtpServer);
        props.put("mail.smtp.port", "587");
        props.put("mail.transport.protocol", "smtp");
        props.put("mail.smtp.from", bounceAddr);

        Session mailSession = Session.getInstance(props,
            new javax.mail.Authenticator() {
                protected PasswordAuthentication getPasswordAuthentication() {
                    return new PasswordAuthentication(userid, password);
                }
            });

        MimeMessage message = new MimeMessage(mailSession);
        message.addFrom(InternetAddress.parse(from));
        message.setRecipients(Message.RecipientType.TO, to);
        message.setSubject(subject);
        message.setContent(body, contentType);

        Transport transport = mailSession.getTransport();
        try {
            System.out.println("Sending ....");
            transport.connect(smtpServer, port, userid, password);
            transport.sendMessage(message,
                message.getRecipients(Message.RecipientType.TO));
            System.out.println("Sending done ...");
        } catch (Exception e) {
            System.err.println("Error Sending: ");
            e.printStackTrace();
        }

        transport.close();
    } // end function main()
}
```

在这里，我们可以看到，该属性mail.smtp.from设距离来源地址不同。

### 编译并运行

现在，我们的类是准备好了，让我们编译上面的类。我已经保存了类SendEmail.java到目录：/home/manisha/JavaMailAPIExercise。我们需要 javax.mail.jar 和 activation.jar 在classpath中。执行下面的命令从命令提示符编译类（两个jar文件 被放置在/home/manisha/目录下）：



少儿教育

留学考试

中小学

小语种

大学考试

职场英语

培训就选北京新东方

GO

javac -cp /home/manisha/activation.jar:/home/manisha/javax.mail.jar SendEmail.java



现在, 这个类被编译, 执行下面的命令来运行:

```
java -cp /home/manisha/activation.jar:/home/manisha/javax.mail.jar: SendEmail
```

### 验证输出

应该看到下面的消息命令控制台上:

```
Sending .....  
Sending done ....
```

**本站所有代码下载** : 请扫描本页面底部(右侧)二维码并关注微信公众号, 回复:" 代码下载 " 获取。

上一篇 : [JavaMail 限额管理](#)

下一篇 : [JavaMail SMTP服务器](#)



登录

畅言  
CHANGYAN



畅言一下

还没有评论, 快来抢沙发吧!

易百教程正在使用畅言



### 热评话题

1. [Git创建存储库 - Git教程™](#)
2. [MongoDB更改用户密码和自定义数据 - MongoDB教程™](#)
3. [MongoDB启用身份验证 - MongoDB教程™](#)
4. [MongoDB添加用户 - MongoDB教程™](#)
5. [MongoDB管理用户和角色 - MongoDB教程™](#)
6. [Git快速入门 - Git教程™](#)
7. [\[Ubuntu\]error: C compiler cannot create executables - 易百教程™](#)

### 推荐/最新教程

- [Spring JDBC](#)
- [Kotlin](#)
- [CouchDB](#)
- [JSoup](#)
- [NumPy](#)

### 最新更新

- [MongoDB](#)
- [MongoDB](#)
- [MongoDB](#)
- [MongoDB](#)
- [MongoDB](#)

### 站点信息

- [站点信息](#)
- [站点信息](#)
- [站点信息](#)
- [站点信息](#)
- [站点信息](#)

### QQ学习群

[QQ群:227270512](#)  
[QQ群:418407075](#)  
[JavaQQ群:172393525](#)

关注微信公众号

· JSF

· Cassandra

· Java

· MongoDB

· MongoDB

· MongoDB

