

Servlet 发送电子邮件

使用 Servlet 发送一封电子邮件是很简单的，但首先您必须在您的计算机上安装 **JavaMail API** 和 **Java Activation Framework (JAF)**。

- 您可以从 Java 网站下载最新版本的 **JavaMail**，打开网页右侧有个 **Downloads** 链接，点击它下载。
- 您可以从 Java 网站下载最新版本的 **JAF (版本 1.1.1)**。

你也可以使用本站提供的下载链接：

- [JavaMail mail.jar 1.4.5](#)
- [JAF \(版本 1.1.1 \) activation.jar](#)

下载并解压缩这些文件，在新创建的顶层目录中，您会发现这两个应用程序的一些 jar 文件。您需要把 **mail.jar** 和 **activation.jar** 文件添加到您的 CLASSPATH 中。

发送一封简单的电子邮件

下面的实例将从您的计算机上发送一封简单的电子邮件。这里假设您的**本地主机**已连接到互联网，并支持发送电子邮件。同时确保 Java Email API 包和 JAF 包的所有的 jar 文件在 CLASSPATH 中都是可用的。

```
// 文件名 SendEmail.java
import java.io.*;
import java.util.*;
import javax.servlet.*;
import javax.servlet.http.*;
import javax.mail.*;
import javax.mail.internet.*;
import javax.activation.*;

public class SendEmail extends HttpServlet{

    public void doGet(HttpServletRequest request,
                      HttpServletResponse response)
        throws ServletException, IOException
    {
        // 收件人的电子邮件 ID
        String to = "abcd@gmail.com";

        // 发件人的电子邮件 ID
        String from = "web@gmail.com";

        // 假设您是从本地主机发送电子邮件
        String host = "localhost";
```

```
// 获取系统的属性
Properties properties = System.getProperties();

// 设置邮件服务器
properties.setProperty("mail.smtp.host", host);

// 获取默认的 Session 对象
Session session = Session.getDefaultInstance(properties);

// 设置响应内容类型
response.setContentType("text/html;charset=UTF-8");
PrintWriter out = response.getWriter();

try{
    // 创建一个默认的 MimeMessage 对象
    MimeMessage message = new MimeMessage(session);
    // 设置 From: header field of the header.
    message.setFrom(new InternetAddress(from));
    // 设置 To: header field of the header.
    message.addRecipient(Message.RecipientType.TO,
        new InternetAddress(to));
    // 设置 Subject: header field
    message.setSubject("This is the Subject Line!");
    // 现在设置实际消息
    message.setText("This is actual message");
    // 发送消息
    Transport.send(message);
    String title = "发送电子邮件";
    String res = "成功发送消息...";
    String docType = "<!DOCTYPE html> \n";
    out.println(docType +
        "<html>\n" +
        "<head><title>" + title + "</title></head>\n" +
        "<body bgcolor=\"#f0f0f0\">\n" +
        "<h1 align=\"center\">" + title + "</h1>\n" +
        "<p align=\"center\">" + res + "</p>\n" +
        "</body></html>");
}catch (MessagingException mex) {
    mex.printStackTrace();
}
}
```

现在让我们来编译上面的 Servlet，并在 web.xml 文件中创建以下条目：

```
....
<servlet>
```

```
<servlet-name>SendEmail</servlet-name>
<servlet-class>SendEmail</servlet-class>
</servlet>

<servlet-mapping>
<servlet-name>SendEmail</servlet-name>
<url-pattern>/SendEmail</url-pattern>
</servlet-mapping>
....
```

现在通过访问 URL `http://localhost:8080/SendEmail` 来调用这个 Servlet。这将会发送一封电子邮件到给定的电子邮件 ID `abcd@gmail.com`，并将显示下面所示的响应：

发送电子邮件

成功发送消息...

如果您想要发送一封电子邮件给多个收件人，那么请使用下面的方法来指定多个电子邮件 ID：

```
void addRecipients(Message.RecipientType type,
                  Address[] addresses)
throws MessagingException
```

下面是对参数的描述：

- **type**：这将被设置为 TO、CC 或 BCC。在这里，CC 代表抄送，BCC 代表密件抄送。例如 `Message.RecipientType.TO`。
- **addresses**：这是电子邮件 ID 的数组。当指定电子邮件 ID 时，您需要使用 `InternetAddress()` 方法。

发送一封 HTML 电子邮件

下面的实例将从您的计算机上发送一封 HTML 格式的电子邮件。这里假设您的**本地主机**已连接到互联网，并支持发送电子邮件。同时确保 Java Email API 包和 JAF 包的所有的 jar 文件在 CLASSPATH 中都是可用的。

本实例与上一个实例很类似，但是这里我们使用 `setContent()` 方法来设置第二个参数为 "text/html" 的内容，该参数用来指定 HTML 内容是包含在消息中的。

使用这个实例，您可以发送内容大小不限的 HTML 内容。

```
// 文件名 SendEmail.java
import java.io.*;
import java.util.*;
import javax.servlet.*;
import javax.servlet.http.*;
import javax.mail.*;
import javax.mail.internet.*;
import javax.activation.*;
```

```
public class SendEmail extends HttpServlet{

    public void doGet(HttpServletRequest request,
                      HttpServletResponse response)
        throws ServletException, IOException
    {
        // 收件人的电子邮件 ID
        String to = "abcd@gmail.com";

        // 发件人的电子邮件 ID
        String from = "web@gmail.com";

        // 假设您是从本地主机发送电子邮件
        String host = "localhost";

        // 获取系统的属性
        Properties properties = System.getProperties();

        // 设置邮件服务器
        properties.setProperty("mail.smtp.host", host);

        // 获取默认的 Session 对象
        Session session = Session.getDefaultInstance(properties);

        // 设置响应内容类型
        response.setContentType("text/html;charset=UTF-8");
        PrintWriter out = response.getWriter();

        try{
            // 创建一个默认的 MimeMessage 对象
            MimeMessage message = new MimeMessage(session);
            // 设置 From: header field of the header.
            message.setFrom(new InternetAddress(from));
            // 设置 To: header field of the header.
            message.addRecipient(Message.RecipientType.TO,
                                new InternetAddress(to));
            // 设置 Subject: header field
            message.setSubject("This is the Subject Line!");

            // 设置实际的 HTML 消息, 内容大小不限
            message.setContent("<h1>This is actual message</h1>",
                               "text/html" );

            // 发送消息
            Transport.send(message);
            String title = "发送电子邮件";
            String res = "成功发送消息...";
            String docType = "<!DOCTYPE html> \n";
            out.println(docType +
```

```
"<html>\n" +
"<head><title>" + title + "</title></head>\n" +
"<body bgcolor=\"#f0f0f0\">\n" +
"<h1 align=\"center\">" + title + "</h1>\n" +
"<p align=\"center\">" + res + "</p>\n" +
"</body></html>");
}catch (MessagingException mex) {
    mex.printStackTrace();
}
}
```

编译并运行上面的 Servlet，在给定的电子邮件 ID 上发送 HTML 消息。

在电子邮件中发送附件

下面的实例将从您的计算机上发送一封带有附件的电子邮件。这里假设您的**本地主机**已连接到互联网，并支持发送电子邮件。同时确保 Java Email API 包和 JAF 包的所有的 jar 文件在 CLASSPATH 中都是可用的。

```
// 文件名 SendEmail.java
import java.io.*;
import java.util.*;
import javax.servlet.*;
import javax.servlet.http.*;
import javax.mail.*;
import javax.mail.internet.*;
import javax.activation.*;

public class SendEmail extends HttpServlet{

    public void doGet(HttpServletRequest request,
                       HttpServletResponse response)
        throws ServletException, IOException
    {
        // 收件人的电子邮件 ID
        String to = "abcd@gmail.com";

        // 发件人的电子邮件 ID
        String from = "web@gmail.com";

        // 假设您是从本地主机发送电子邮件
        String host = "localhost";

        // 获取系统的属性
        Properties properties = System.getProperties();

        // 设置邮件服务器
        properties.setProperty("mail.smtp.host", host);
```

```
// 获取默认的 Session 对象
Session session = Session.getDefaultInstance(properties);

// 设置响应内容类型
response.setContentType("text/html;charset=UTF-8");
PrintWriter out = response.getWriter();

try{
    // 创建一个默认的 MimeMessage 对象
    MimeMessage message = new MimeMessage(session);

    // 设置 From: header field of the header.
    message.setFrom(new InternetAddress(from));

    // 设置 To: header field of the header.
    message.addRecipient(Message.RecipientType.TO,
        new InternetAddress(to));

    // 设置 Subject: header field
    message.setSubject("This is the Subject Line!");

    // 创建消息部分
    BodyPart messageBodyPart = new MimeBodyPart();

    // 填写消息
    messageBodyPart.setText("This is message body");

    // 创建一个多部分消息
    Multipart multipart = new MimeMultipart();

    // 设置文本消息部分
    multipart.addBodyPart(messageBodyPart);

    // 第二部分是附件
    messageBodyPart = new MimeBodyPart();
    String filename = "file.txt";
    DataSource source = new FileDataSource(filename);
    messageBodyPart.setDataHandler(new DataHandler(source));
    messageBodyPart.setFileName(filename);
    multipart.addBodyPart(messageBodyPart);

    // 发送完整的消息部分
    message.setContent(multipart );

    // 发送消息
    Transport.send(message);
    String title = "发送电子邮件";
    String res = "成功发送电子邮件...";
```

```
String docType = "<!DOCTYPE html> \n";
out.println(docType +
"<html>\n" +
"<head><title>" + title + "</title></head>\n" +
"<body bgcolor=\"#f0f0f0\">\n" +
"<h1 align=\"center\">" + title + "</h1>\n" +
"<p align=\"center\">" + res + "</p>\n" +
"</body></html>");
}catch (MessagingException mex) {
    mex.printStackTrace();
}
}
```

编译并运行上面的 Servlet，在给定的电子邮件 ID 上发送带有文件附件的消息。

用户身份认证部分

如果需要向电子邮件服务器提供用户 ID 和密码进行身份认证，那么您可以设置如下属性：

```
props.setProperty("mail.user", "myuser");
props.setProperty("mail.password", "mypwd");
```

电子邮件发送机制的其余部分与上面讲解的保持一致。

[← Servlet 自动刷新页面](#)

[Servlet 包 →](#)

[点我分享笔记](#)