

Python3 SMTP发送邮件

SMTP (Simple Mail Transfer Protocol) 即简单邮件传输协议,它是一组用于由源地址到目的地址传送邮件的规则,由它来控制信件的中转方式。

python的smtpplib提供了一种很方便的途径发送电子邮件。它对smtp协议进行了简单的封装。

Python创建 SMTP 对象语法如下：

```
import smtplib

smtpObj = smtplib.SMTP( [host [, port [, local_hostname]]] )
```

参数说明：

- host: SMTP 服务器主机。你可以指定主机的ip地址或者域名如:runoob.com,这个是可选参数。
- port: 如果你提供了 host 参数,你需要指定 SMTP 服务使用的端口号,一般情况下SMTP端口号为25。
- local_hostname: 如果SMTP在你的本机上,你只需要指定服务器地址为 localhost 即可。

Python SMTP对象使用sendmail方法发送邮件,语法如下：

```
SMTP.sendmail(from_addr, to_addrs, msg[, mail_options, rcpt_options])
```

参数说明：

- from_addr: 邮件发送者地址。
- to_addrs: 字符串列表,邮件发送地址。
- msg: 发送消息

这里要注意一下第三个参数,msg是字符串,表示邮件。我们知道邮件一般由标题,发信人,收件人,邮件内容,附件等构成,发送邮件的时候,要注意msg的格式。这个格式就是smtp协议中定义的格式。

实例

以下是一个使用Python发送邮件简单的实例：

实例

```
#!/usr/bin/python3
import smtplib
from email.mime.text import MIMEText
from email.header import Header
sender = 'from@runoob.com'
receivers = ['429240967@qq.com'] # 接收邮件,可设置为你的QQ邮箱或者其他邮箱
# 三个参数: 第一个为文本内容,第二个 plain 设置文本格式,第三个 utf-8 设置编码
```

```
message = MIMEText('Python 邮件发送测试...', 'plain', 'utf-8')
message['From'] = Header("菜鸟教程", 'utf-8') # 发送者
message['To'] = Header("测试", 'utf-8') # 接收者
subject = 'Python SMTP 邮件测试'
message['Subject'] = Header(subject, 'utf-8')
try:
    smtpObj = smtplib.SMTP('localhost')
    smtpObj.sendmail(sender, receivers, message.as_string())
    print ("邮件发送成功")
except smtplib.SMTPException:
    print ("Error: 无法发送邮件")
```

我们使用三个引号来设置邮件信息，标准邮件需要三个头部信息：**From**, **To**, 和 **Subject**，每个信息直接使用空行分割。

我们通过实例化 smtplib 模块的 SMTP 对象 *smtpObj* 来连接到 SMTP 访问，并使用 *sendmail* 方法来发送信息。

执行以上程序，如果你本机安装sendmail，就会输出：

```
$ python3 test.py
邮件发送成功
```

查看我们的收件箱(一般在垃圾箱)，就可以查看到邮件信息：


Python SMTP 邮件测试 ☆

发件人：菜鸟教程@iZ23mtq8bs1Z <> 

(由 from@runoob.com 代发) 

时 间：2016年4月5日(星期二) 下午4:30

收件人：测试@iZ23mtq8bs1Z

这是一封垃圾箱中的邮件。请勿轻信中奖、汇款等虚假信息，勿轻易拨打陌生电话。  举报垃圾邮件 移回收件箱

Python 邮件发送测试...

如果我们本机没有 sendmail 访问，也可以使用其他服务商的 SMTP 访问（QQ、网易、Google等）。

实例

```
#!/usr/bin/python3
import smtplib
from email.mime.text import MIMEText
from email.header import Header
# 第三方 SMTP 服务
mail_host="smtp.XXX.com" #设置服务器
mail_user="XXXX" #用户名
mail_pass="XXXXXX" #口令
sender = 'from@runoob.com'
receivers = ['429240967@qq.com'] # 接收邮件，可设置为你的QQ邮箱或者其他邮箱
message = MIMEText('Python 邮件发送测试...', 'plain', 'utf-8')
message['From'] = Header("菜鸟教程", 'utf-8')
message['To'] = Header("测试", 'utf-8')
```

```
subject = 'Python SMTP 邮件测试'
message['Subject'] = Header(subject, 'utf-8')
try:
    smtpObj = smtplib.SMTP()
    smtpObj.connect(mail_host, 25) # 25 为 SMTP 端口号
    smtpObj.login(mail_user,mail_pass)
    smtpObj.sendmail(sender, receivers, message.as_string())
    print ("邮件发送成功")
except smtplib.SMTPException:
    print ("Error: 无法发送邮件")
```

使用Python发送HTML格式的邮件

Python发送HTML格式的邮件与发送纯文本消息的邮件不同之处就是将MIMEText中_subtype设置为html。具体代码如下：

实例

```
#!/usr/bin/python3
import smtplib
from email.mime.text import MIMEText
from email.header import Header
sender = 'from@runoob.com'
receivers = ['429240967@qq.com'] # 接收邮件，可设置为你的QQ邮箱或者其他邮箱
mail_msg = """
<p>Python 邮件发送测试...</p>
<p><a href="http://www.runoob.com">这是一个链接</a></p>
"""


message = MIMEText(mail_msg, 'html', 'utf-8')
message['From'] = Header("菜鸟教程", 'utf-8')
message['To'] = Header("测试", 'utf-8')
subject = 'Python SMTP 邮件测试'
message['Subject'] = Header(subject, 'utf-8')
try:
    smtpObj = smtplib.SMTP('localhost')
    smtpObj.sendmail(sender, receivers, message.as_string())
    print ("邮件发送成功")
except smtplib.SMTPException:
    print ("Error: 无法发送邮件")
```


执行以上程序，如果你本机安装sendmail，就会输出：

```
$ python3 test.py
邮件发送成功
```

查看我们的收件箱(一般在垃圾箱)，就可以查看到邮件信息：

Python SMTP 邮件测试 ☆

发件人: 菜鸟教程@iZ23mtq8bs1Z <> 
(由 from@runoob.com 代发) 
时 间: 2016年4月5日(星期二) 下午4:45
收件人: 测试@iZ23mtq8bs1Z

这是一封垃圾箱中的邮件。请勿轻信中奖、汇款等虚假信息，勿轻易拨打陌生电话。  举报垃圾邮件 移回收件箱

Python 邮件发送测试...

[这是一个链接](#)

Python 发送带附件的邮件

发送带附件的邮件，首先要创建MIMEMultipart()实例，然后构造附件，如果有多个附件，可依次构造，最后利用smtplib.smtp发送。

实例




```
#!/usr/bin/python3
import smtplib
from email.mime.text import MIMEText
from email.mime.multipart import MIMEMultipart
from email.header import Header
sender = 'from@runoob.com'
receivers = ['429240967@qq.com'] # 接收邮件，可设置为你的QQ邮箱或者其他邮箱
#创建一个带附件的实例
message = MIMEMultipart()
message['From'] = Header("菜鸟教程", 'utf-8')
message['To'] = Header("测试", 'utf-8')
subject = 'Python SMTP 邮件测试'
message['Subject'] = Header(subject, 'utf-8')
#邮件正文内容
message.attach(MIMEText('这是菜鸟教程Python 邮件发送测试.....', 'plain', 'utf-8'))
# 构造附件1，传送当前目录下的 test.txt 文件
att1 = MIMEText(open('test.txt', 'rb').read(), 'base64', 'utf-8')
att1["Content-Type"] = 'application/octet-stream'
# 这里的filename可以任意写，写什么名字，邮件中显示什么名字
att1["Content-Disposition"] = 'attachment; filename="test.txt"'
message.attach(att1)
# 构造附件2，传送当前目录下的 runoob.txt 文件
att2 = MIMEText(open('runoob.txt', 'rb').read(), 'base64', 'utf-8')
att2["Content-Type"] = 'application/octet-stream'
att2["Content-Disposition"] = 'attachment; filename="runoob.txt"'
message.attach(att2)
try:
    smtpObj = smtplib.SMTP('localhost')
    smtpObj.sendmail(sender, receivers, message.as_string())
    print ("邮件发送成功")
```


```
except smtplib.SMTPException:  
    print ("Error: 无法发送邮件")
```

```
$ python3 test.py  
邮件发送成功
```

查看我们的收件箱(一般在垃圾箱), 就可以查看到邮件信息:

Python SMTP 邮件测试 ☆

发件人: 菜鸟教程@iZ23mtq8bs1Z <> 
(由 from@runoob.com 代发) 
时 间: 2016年4月5日(星期二) 下午5:27
收件人: 测试@iZ23mtq8bs1Z
附 件: 2 个 ( test.txt...)

垃圾邮件中的附件可能包含木马病毒等有害内容。为了您的帐号安全, 请勿轻易打开附件。  举报垃圾邮件 移回收件箱

这是菜鸟教程Python 邮件发送测试.....

附件(2 个)

普通附件 ↓ 全部下载

 test.txt (29字节)
下载 预览 转存 ▼

 runoob.txt (4字节)
下载 预览 转存 ▼

在 HTML 文本中添加图片

邮件的 HTML 文本中一般邮件服务商添加外链是无效的, 正确添加图片的实例如下所示:

实例

```
#!/usr/bin/python3  
import smtplib  
from email.mime.image import MIMEImage  
from email.mime.multipart import MIMEMultipart  
from email.mime.text import MIMEText  
from email.header import Header  
sender = 'from@runoob.com'  
receivers = ['429240967@qq.com'] # 接收邮件, 可设置为你的QQ邮箱或者其他邮箱  
msgRoot = MIMEMultipart('related')  
msgRoot['From'] = Header("菜鸟教程", 'utf-8')  
msgRoot['To'] = Header("测试", 'utf-8')  
subject = 'Python SMTP 邮件测试'  
msgRoot['Subject'] = Header(subject, 'utf-8')  
msgAlternative = MIMEMultipart('alternative')  
msgRoot.attach(msgAlternative)  
mail_msg = ""  
<p>Python 邮件发送测试...</p>  
<p><a href="http://www.runoob.com">菜鸟教程链接</a></p>
```

```
<p>图片演示: </p>
<p></p>
"""

msgAlternative.attach(MIMEText(mail_msg, 'html', 'utf-8'))
# 指定图片为当前目录
fp = open('test.png', 'rb')
msgImage = MIMEImage(fp.read())
fp.close()
# 定义图片 ID, 在 HTML 文本中引用
msgImage.add_header('Content-ID', '<image1>')
msgRoot.attach(msgImage)
try:
    smtpObj = smtplib.SMTP('localhost')
    smtpObj.sendmail(sender, receivers, msgRoot.as_string())
    print ("邮件发送成功")
except smtplib.SMTPException:
    print ("Error: 无法发送邮件")
```

```
$ python3 test.py
```

```
邮件发送成功
```

查看我们的收件箱(如果在垃圾箱可能需要移动到收件箱才可正常显示), 就可以查看到邮件信息:



Python SMTP 邮件测试

菜鸟教程@iZ23mtq8bs1Z <> 详情

Python 邮件发送测试...

[菜鸟教程链接](#)

图片演示:

RUNOOB.COM

使用第三方 SMTP 服务发送

这里使用了 QQ 邮箱(你也可以使用 163, Gmail等)的 SMTP 服务, 需要做以下配置:



QQ 邮箱通过生成授权码来设置密码：

POP3/IMAP/SMTP/Exchange/CardDAV/CalDAV服务

开启服务：	POP3/SMTP服务 (如何使用 Foxmail 等软件收发邮件?)	已开启 关闭
	IMAP/SMTP服务 (什么是 IMAP, 它又是如何设置?)	已开启 关闭
	Exchange服务 (什么是Exchange, 它又是如何设置?)	已开启 关闭
	CardDAV/CalDAV服务 (什么是CardDAV/CalDAV, 它又是如何设置?)	已开启 关闭
	(POP3/IMAP/SMTP/CardDAV/CalDAV服务均支持SSL连接。如何设置?)	

温馨提示：登录第三方客户端时，密码框请输入“授权码”进行验证？

生成授权码

生成授权码作为密码

QQ 邮箱 SMTP 服务器地址：smtp.qq.com，ssl 端口：465。

以下实例你需要修改：发件人邮箱（你的QQ邮箱），密码，收件人邮箱（可发给自己）。

QQ SMTP

```
#!/usr/bin/python3
import smtplib
from email.mime.text import MIMEText
from email.utils import formataddr
my_sender='429240967@qq.com' # 发件人邮箱账号
my_pass = 'xxxxxxxxxx' # 发件人邮箱密码
my_user='429240967@qq.com' # 收件人邮箱账号，我这边发送给自己
def mail():
    ret=True
    try:
        msg=MIMEText('填写邮件内容','plain','utf-8')
        msg['From']=formataddr(["FromRunoob",my_sender]) # 括号里的对应发件人邮箱昵称、发件人邮箱账号
        msg['To']=formataddr(["FK",my_user]) # 括号里的对应收件人邮箱昵称、收件人邮箱账号
        msg['Subject']="菜鸟教程发送邮件测试" # 邮件的主题，也可以说是标题
        server=smtplib.SMTP_SSL("smtp.qq.com", 465) # 发件人邮箱中的SMTP服务器，端口是25
        server.login(my_sender, my_pass) # 括号中对应的是发件人邮箱账号、邮箱密码
        server.sendmail(my_sender,[my_user,],msg.as_string()) # 括号中对应的是发件人邮箱账号、收件人邮箱账号、发送邮件
        server.quit() # 关闭连接
    except Exception: # 如果 try 中的语句没有执行，则会执行下面的 ret=False
        ret=False
    return ret
ret=mail()
if ret:
    print("邮件发送成功")
```



```
else:  
    print("邮件发送失败")
```

```
$ python test.py  
邮件发送成功
```

发送成功后，登陆收件人邮箱即可查看：

菜鸟教程发送邮件测试 ☆

发件人: 7@qq.com >

时间: 2017年3月17日(星期五) 下午2:14

收件人:

这是菜鸟教程的邮件测试内容

更多内容请参阅：<https://docs.python.org/3/library/email-examples.html>。

← Python3 网络编程

Python3 多线程 →



2 篇笔记

写笔记



```
message['From'] = Header("xxxxx@163.com", 'utf-8')  
message['To'] = Header("xxxxx@qq.com,xxxxx@qq.com", 'utf-8')  
smtpObj.sendmail("xxxxx@163.com", ["xxxxx@qq.com", "xxxxx@qq.com"], message.as_string())
```

注意点：

From、To 要与 sendmail 中的前两个参数保持一致

To 多个邮箱用逗号隔开，sendmail 中的第二个参数要用 list

```
subject = 'Python SMTP 邮件测试'  
message['Subject'] = Header(subject, 'utf-8')
```

注意点：subject 中不能含有 "test" 关键字，否则会被视为垃圾邮件

```
smtpObj.login(mail_user, mail_pass)
```

注意点：如果使用授权码登录第三方邮件客户端，则此处的密码(mail_pass)使用授权码

一生默默守护你 8个月前 (07-29)



在发送带附件的邮件的时候，例程中有一行代码如下：

```
att1 = MIMEText(open('test.txt', 'rb').read(), 'base64', 'utf-8')
```

这里的 test.txt 必须是在当前的工作目录下的，所以要时想发送这个附件，需要将这个附件移动到工作目录下。

或者还有一种方法就是，指定文件的路径：

```
path='D:/test/test.txt'  
file=open(path,'rb')  
att1=MIMEText(file.read(),'base64','utf-8')
```

也可以直接写成如下形式：

```
att1=MIMEText(open('D:/test/test.txt', 'rb').read(),'base64','utf-8')
```

yan 7个月前 (08-09)