WireShark实验 - 套接字编程作业 2：SMTP

信息安全智息技术班　杨乙　　21307130076

完整的SMTP邮件客户端的代码：

```python
from socket import *
import base64

msg = "I love computer networks!"
endmsg = "\r\n.\r\n"

# Choose a mail server
mailserver = "smtp.163.com"

# Create socket called clientSocket and establish a TCP connection with mailserver
# Fill in start
clientSocket = socket(AF_INET, SOCK_STREAM)
clientSocket.connect((mailserver, 25))
# Fill in end

recv = clientSocket.recv(1024).decode()
print(recv)
if recv[:3] != '220':
    print('220 reply not received from server.')

# Send HELO command and print server response.
heloCommand = 'HELO Alice\r\n'
clientSocket.send(heloCommand.encode())
recv1 = clientSocket.recv(1024).decode()
print(recv1)
if recv1[:3] != '250':
    print('250 reply not received from server.')

# send AUTH LOGIN command
# Fill in start
LoginCommand = 'AUTH LOGIN\r\n'
clientSocket.send(LoginCommand.encode())
recv2 = clientSocket.recv(1024).decode()
print(recv2)
if recv2[:3] != '334':
    print('Login failed!')

clientSocket.send(base64.b64encode('18816576169'.encode()) + b'\r\n')
recv3 = clientSocket.recv(1024).decode()
print(recv3)
if recv3[:3] != '334':
    print('Wrong ID! Please check your input.')
```

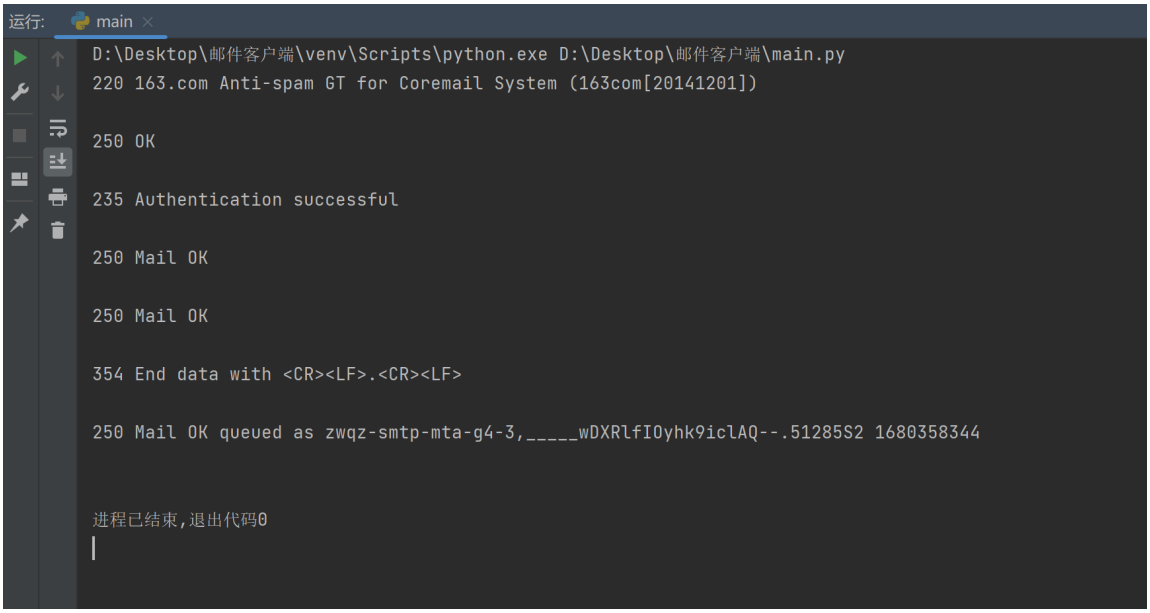
```
43
44 clientSocket.send(base64.b64encode('UKMGZMWAFBKWUFAH'.encode()) + b'\r\n')
45 recv4 = clientSocket.recv(1024).decode()
46 print(recv4)
47 if recv4[:3] != '235':
48     print('Wrong password! Please try again.')
49 # Fill in end
50
51 # Send MAIL FROM command and print server response.
52 # Fill in start
53 send_mail_addr = '18816576169@163.com'
54 MailFromCommand = 'MAIL FROM: <' + send_mail_addr + '>\r\n'
55 clientSocket.send(MailFromCommand.encode())
56 recv5 = clientSocket.recv(1024).decode()
57 print(recv5)
58 if recv5[:3] != '250':
59     print('Something wrong with MAIL FROM server.')
60 # Fill in end
61
62 # Send RCPT TO command and print server response.
63 # Fill in start
64 rec_mail_addr = '3182443151@qq.com'
65 RcptToCommand = 'RCPT TO: <' + rec_mail_addr + '>\r\n'
66 clientSocket.send(RcptToCommand.encode())
67 recv6 = clientSocket.recv(1024).decode()
68 print(recv6)
69 if recv6[:3] != '250':
70     print('Something wrong with RCPT TO server.')
71 # Fill in end
72
73 # Send DATA command and print server response.
74 # Fill in start
75 DataCommand = 'DATA\r\n'
76 clientSocket.send(DataCommand.encode())
77 recv7 = clientSocket.recv(1024).decode()
78 print(recv7)
79 if recv7[:3] != '354':
80     print('Something wrong with data_send_begin.')
81 # Fill in end
82
83 # Send message data.
84 # Fill in start
85 subject = 'I love computer networks!'
86 mailmsg = 'From: ' + send_mail_addr + '\r\n'
87 mailmsg += 'To: ' + rec_mail_addr + '\r\n'
88 mailmsg += 'Subject: ' + subject + '\r\n'
89 mailmsg += msg
90
91 clientSocket.send(mailmsg.encode())
92 clientSocket.send(endmsg.encode())
93 # Fill in end
94
95 # Fill in start
96 recv8 = clientSocket.recv(1024).decode()
97 print(recv8)
```

```python
if recv8[:3] != '250':
    print('Something wrong with data transport.')
# Fill in end

# Send QUIT command and get server response.
# Fill in start
QuitCommand = 'QUIT\r\n'
clientSocket.send(QuitCommand.encode())
recv9 = clientSocket.recv(1024).decode()
print(recv9)
if recv9[:3] != '221':
    print('Something wrong with QUIT server.')
# Fill in end

# close
clientSocket.close()
```

代码运行成功：



发送方邮箱：

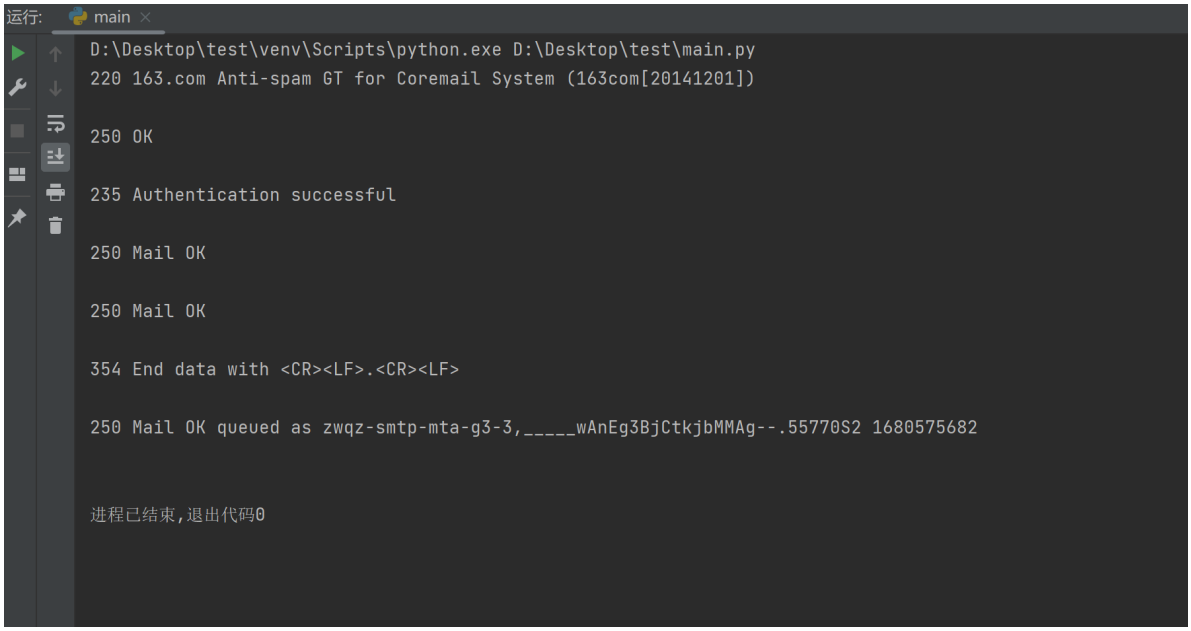
I love computer networks!

接收方邮箱：

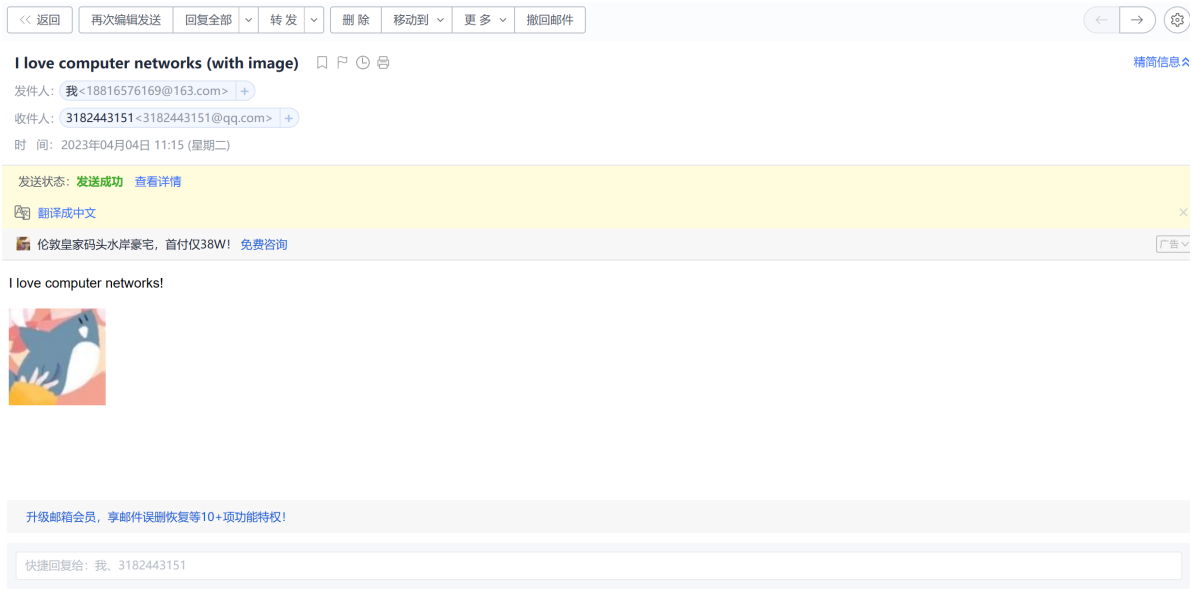
I love computer networks!

可选练习

代码运行成功：



发送方邮箱：



接收方邮箱：

由于 QQ 邮箱的安全机制（为防止邮件打开跟踪等攻击）不会直接显示图片。需要点击 “信任此发件人” 以查看图片：

客户端代码：

```
from socket import *
import base64

head = ("From: 18816576169@163.com\r\n" +
        "To: 3182443151@qq.com\r\n" +
        "Subject: I love computer networks (with image)\r\n" +
        "MIME-Version: 1.0\r\n" +
        "Content-Type: multipart/related;
boundary='separator'\r\n\r\n").encode()

htmlHead = ("--separator\r\n" +
            "Content-Type: text/html\r\n" +
            "Content-Transfer-Encoding: base64\r\n\r\n").encode()

htmlData = b"""
<html>
  <head>
    <title>Send an email with image</title>
  </head>

  <body>
    <p>I love computer networks!</p>
    <p><image src="cid:image1" title="This is a picture"></p>
  </body>
</html>
"""


htmlData_base64 = base64.b64encode(htmlData)

imageHead = ("\r\n\r\n--separator\r\n" +
             "Content-Type: image/jpeg\r\n" +
             "Content-Transfer-Encoding: base64\r\n" +
             "Content-ID: image1\r\n\r\n").encode()

with open("picture.jpg", "rb") as f:
    imageData = base64.b64encode(f.read())
```

```
37
38  headEnd = "\r\n\r\n--separator--\r\n".encode()
39
40  endmsg = "\r\n.\r\n"
41
42  msg = head
43  msg += htmlHead
44  msg += htmlData_base64
45  msg += imageHead
46  msg += imageData
47  msg += headEnd
48
49  # Choose a mail server
50  mailserver = "smtp.163.com"
51
52  # Create socket called clientSocket and establish a TCP connection with
    mailserver
53  # Fill in start
54  clientSocket = socket(AF_INET, SOCK_STREAM)
55  clientSocket.connect((mailserver, 25))
56  # Fill in end
57
58  recv = clientSocket.recv(1024).decode()
59  print(recv)
60  if recv[:3] != '220':
61      print('220 reply not received from server.')
62
63  # Send HELO command and print server response.
64  heloCommand = 'HELO Alice\r\n'
65  clientSocket.send(heloCommand.encode())
66  recv1 = clientSocket.recv(1024).decode()
67  print(recv1)
68  if recv1[:3] != '250':
69      print('250 reply not received from server.')
70
71  # send AUTH LOGIN command
72  # Fill in start
73  LoginCommand = 'AUTH LOGIN\r\n'
74  clientSocket.send(LoginCommand.encode())
75  recv2 = clientSocket.recv(1024).decode()
76  if recv2[:3] != '334':
77      print('Login failed!')
78
79  clientSocket.send(base64.b64encode('18816576169'.encode()) + b'\r\n')
80  recv3 = clientSocket.recv(1024).decode()
81  if recv3[:3] != '334':
82      print('Wrong ID! Please check your input.')
83
84  clientSocket.send(base64.b64encode('UKMGZMWAFBKWUFAH'.encode()) + b'\r\n')
85  recv4 = clientSocket.recv(1024).decode()
86  print(recv4)
87  if recv4[:3] != '235':
88      print('Wrong password! Please try again.')
89  # Fill in end
90
```

```python
# Send MAIL FROM command and print server response.
# Fill in start
send_mail_addr = '18816576169@163.com'
MailFromCommand = 'MAIL FROM: <' + send_mail_addr + '>\r\n'
clientSocket.send(MailFromCommand.encode())
recv5 = clientSocket.recv(1024).decode()
print(recv5)
if recv5[:3] != '250':
    print('Something wrong with MAIL FROM server.')
# Fill in end

# Send RCPT TO command and print server response.
# Fill in start
rec_mail_addr = '3182443151@qq.com'
RcptToCommand = 'RCPT TO: <' + rec_mail_addr + '>\r\n'
clientSocket.send(RcptToCommand.encode())
recv6 = clientSocket.recv(1024).decode()
print(recv6)
if recv6[:3] != '250':
    print('Something wrong with RCPT TO server.')
# Fill in end

# Send DATA command and print server response.
# Fill in start
DataCommand = 'DATA\r\n'
clientSocket.send(DataCommand.encode())
recv7 = clientSocket.recv(1024).decode()
print(recv7)
if recv7[:3] != '354':
    print('Something wrong with data_send_begin.')
# Fill in end

# Send message data.
# Fill in start
clientSocket.send(msg)
clientSocket.send(endmsg.encode())
# Fill in end

# Fill in start
recv8 = clientSocket.recv(1024).decode()
print(recv8)
if recv8[:3] != '250':
    print('Something wrong with data transport.')
# Fill in end

# Send QUIT command and get server response.
# Fill in start
QuitCommand = 'QUIT\r\n'
clientSocket.send(QuitCommand.encode())
recv9 = clientSocket.recv(1024).decode()
if recv9[:3] != '221':
    print('Something wrong with QUIT server.')
# Fill in end

# close
```

```
clientSocket.close()

```

对代码的解释：

练习要求 “发送包含文本和图像的电子邮件”，说明需要将图片嵌入邮件消息内，而不是作为附件发送。为实现上述要求，选择发送 html，并将图片嵌入 html 当中

因此在这一封邮件中，包含了 html 数据、图片数据两部分。因此我们要将邮件的 `Content-Type` 域指定为 `multipart/related` ，将图片以内嵌资源的方式存储在邮件中

上述两部分数据要有各自的消息头。其中 html 数据的 `Content-Type` 域为 `text/html`；图片数据的 `Content-Type` 域为 `image/jpeg`。并且各个数据部分需要通过 `boundary` 在开始部分和结尾部分进行封装

同时，为嵌入图片，需要将图片数据的 `Content-ID` 域设置为 `image1` 并且在 html 的 `img` 标签中引用 `src="cid:image1"`。图片数据直接读取图片获得