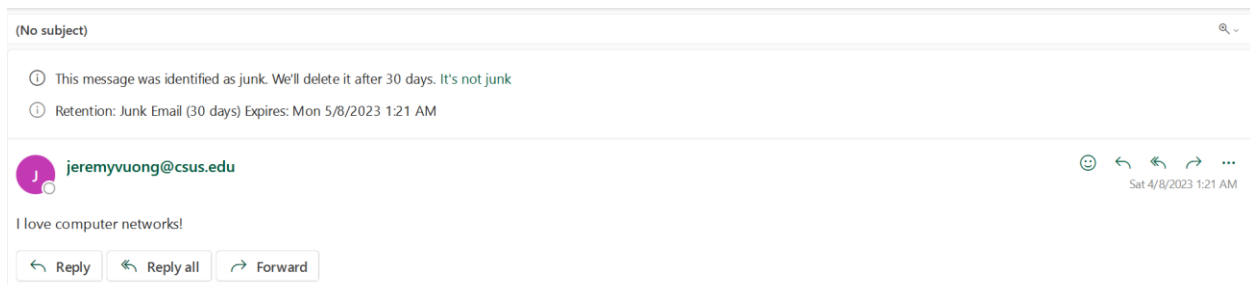


**Author:** Jeremy Vuong  
**Course:** CSC 138-03  
**Professor:** Jun Dai  
**Date:** 04/08/2023

## Socket Programming Assignment 2 – Mail Client

```
jeremyvuong@ecs-pa-coding1:~  
[jeremyvuong@ecs-pa-coding1 ~]$ python SMTPClient.py  
220 smtp.saclink.csus.edu Microsoft ESMTMP MAIL Service ready at Sat, 8 Apr 2023 01:21:04 -0700  
  
250 smtp.saclink.csus.edu Hello [130.86.188.33]  
  
250 2.1.0 Sender OK  
  
250 2.1.5 Recipient OK  
  
354 Start mail input; end with <CRLF>.<CRLF>  
  
250 2.6.0 <160182d2-5335-4c47-bbe2-524886a9d66f@irt-pa-e16mbx01.saclink.csus.edu> [InternalId=35519379538049, Hostname=irt-pa-e16mbx02.saclink.csus.edu] 1613 bytes in 0.135, 11.591 KB/sec Queued mail for delivery  
  
221 2.0.0 Service closing transmission channel  
  
[jeremyvuong@ecs-pa-coding1 ~]$
```



```
jeremyvuong@ecs-pa-coding1:~  
1 from socket import *  
2  
3 msg = "\r\nI love computer networks!"  
4 endmsg = "\r\n.\r\n"  
5  
6 # Choose a mail server (e.g. Google mail server) and call it mailserver  
7 mailserver = "smtp.csus.edu"  
8 mailserver_port = 25  
9  
10 # Create socket called clientSocket and establish a TCP connection with mailserver  
11 clientSocket = socket(AF_INET, SOCK_STREAM)  
12 clientSocket.connect((mailserver, mailserver_port))  
13 recv = clientSocket.recv(1024)  
14 print(recv)  
15 if recv[:3] != b'220':  
16     print('220 reply not received from server.')  
17  
18 # Send HELO command and print server response.  
19 helloCommand = 'HELO Alice\r\n'  
20 clientSocket.send(helloCommand.encode())  
21 recv1 = clientSocket.recv(1024)  
22 print(recv1)  
23 if recv1[:3] != b'250':  
24     print('250 reply not received from server.')  
25  
26 # Send MAIL FROM command and print server response.  
27 mailFromCommand = 'MAIL FROM: <jeremyvuong@csus.edu>\r\n'  
28 clientSocket.send(mailFromCommand.encode())  
29 recv2 = clientSocket.recv(1024)  
30 print(recv2)  
31 if recv2[:3] != b'250':  
32     print('250 reply not received from server.')  
33  
34 # Send RCPT TO command and print server response.  
35 rcptToCommand = 'RCPT TO: <jeremyvuong@csus.edu>\r\n'  
36 clientSocket.send(rcptToCommand.encode())  
37 recv3 = clientSocket.recv(1024)  
38 print(recv3)  
39 if recv3[:3] != b'250':  
40     print('250 reply not received from server.')  
41  
42 # Send DATA command and print server response.  
43 dataCommand = 'DATA\r\n'  
44 clientSocket.send(dataCommand.encode())  
45 recv4 = clientSocket.recv(1024)  
46 print(recv4)  
47 if recv4[:3] != b'354':  
48     print('354 reply not received from server.')  
49  
50 # Send message data.  
51 clientSocket.send(msg.encode())  
52  
53 # Message ends with a single period.  
54 clientSocket.send(endmsg.encode())  
55 recv5 = clientSocket.recv(1024)  
56 print(recv5)  
57 if recv5[:3] != b'250':  
58     print('250 reply not received from server.')  
59  
60 # Send QUIT command and get server response.  
61 quitCommand = 'QUIT\r\n'  
62 clientSocket.send(quitCommand.encode())  
63 recv6 = clientSocket.recv(1024)  
64 print(recv6)  
65 if recv6[:3] != b'221':  
66     print('221 reply not received from server.')  
67  
68 # Close the client socket  
69 clientSocket.close()  
70
```

```

from socket import *

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

# Choose a mail server (e.g. Google mail server) and call it mailserver
mailserver = "smtp.csus.edu"
mailserver_port = 25

# Create socket called clientSocket and establish a TCP connection with mailserver
clientSocket = socket(AF_INET, SOCK_STREAM)
clientSocket.connect((mailserver, mailserver_port))
recv = clientSocket.recv(1024)
print(recv)
if recv[:3] != b'220':
    print('220 reply not received from server.')

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

# Send MAIL FROM command and print server response.
mailFromCommand = 'MAIL FROM: <jeremyvuong@csus.edu>\r\n'
clientSocket.send(mailFromCommand.encode())
recv2 = clientSocket.recv(1024)
print(recv2)
if recv2[:3] != b'250':
    print('250 reply not received from server.')

# Send RCPT TO command and print server response.
rcptToCommand = 'RCPT TO: <jeremyvuong@csus.edu>\r\n'
clientSocket.send(rcptToCommand.encode())
recv3 = clientSocket.recv(1024)
print(recv3)
if recv3[:3] != b'250':
    print('250 reply not received from server.')

```

```
# Send DATA command and print server response.
```

```
dataCommand = 'DATA\r\n'
```

```
clientSocket.send(dataCommand.encode())
```

```
recv4 = clientSocket.recv(1024)
```

```
print(recv4)
```

```
if recv4[:3] != b'354':
```

```
    print('354 reply not received from server.')
```

```
# Send message data.
```

```
clientSocket.send(msg.encode())
```

```
# Message ends with a single period.
```

```
clientSocket.send(endmsg.encode())
```

```
recv5 = clientSocket.recv(1024)
```

```
print(recv5)
```

```
if recv5[:3] != b'250':
```

```
    print('250 reply not received from server.')
```

```
# Send QUIT command and get server response.
```

```
quitCommand = 'QUIT\r\n'
```

```
clientSocket.send(quitCommand.encode())
```

```
recv6 = clientSocket.recv(1024)
```

```
print(recv6)
```

```
if recv6[:3] != b'221':
```

```
    print('221 reply not received from server.')
```

```
# Close the client socket
```

```
clientSocket.close()
```