CPSC 471 – Programming Assignment 2

Mail Client Code:

```
from socket import *
msg = "\r\n I love computer networks!"
endmsg = "\r\n.\r\n"  # Choose a mail server (e.g. Google mail server) and
call it mailserver
mailserver = "mail.smtp.com"#Fill in start   #Fill in end  # Create socket
called clientSocket and establish a TCP connection with mailserver
clientSocket = socket(AF_INET, SOCK_STREAM)
serverPort = 80
clientSocket.connect((mailserver, serverPort))
#Fill in start
#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.')
clientSocket.sendall('AUTH LOGIN\r\n'.encode())
print(clientSocket.recv(1024).decode())
clientSocket.sendall(('mail1.smtp.com\r\n').encode())
print(clientSocket.recv(1024).decode())
clientSocket.sendall(('*****\r\n').encode())
print(clientSocket.recv(1024).decode())
# Send MAIL FROM command and print server response.
mailFrom = 'mail1.smtp.com'
mailCommand = 'MAIL FROM: <'+mailFrom+'>\r\n'
clientSocket.send(mailCommand.encode())
recv2 = clientSocket.recv(1024).decode()
print(recv2)
# Fill in start
# Fill in end
# Send RCPT TO command and print server response.
clientSocket.sendall(('RCPT TO: xyz - Programming Assignment 2 SMTP Socket
Programming\r\n').encode())
recv3 = clientSocket.recv(1024).decode()
print(recv3)
# Fill in start
# Fill in end
# Send DATA command and print server response.
clientSocket.sendall(('DATA: Programming Assignment 2 SMTP Socket
Programming\r\n').encode())
recv4 = clientSocket.recv(1024).decode()
print(recv4)
# Fill in start
```

```
# Fill in end
# Send message data.
message = 'Hello! This is my second Programming Assignment SMTP Socket
Programming!\r\n'
clientSocket.sendall(message.encode())
recv5 = clientSocket.recv(1024).decode()
print(recv5)
# Fill in start
# Fill in end
# Message ends with a single period.
clientSocket.sendall(endmsg.encode())
recv6 = clientSocket.recv(1024).decode()
print(recv6)
# Fill in start
# Fill in end
# Send QUIT command and get server response.
clientSocket.sendall('QUIT\r\n'.encode())
recv7 = clientSocket.recv(1024).decode()
print(recv7)
# Fill in start
# Fill in end
```

Screenshot of output received:

```
prateeti@Prateetis-MacBook-Air Socket Programming % python3 smtp_programming.py
220 mtl-mtsp-mta02-in1 ESMTP (SMTP.com v6)

250 2.0.0 mtl-mtsp-mta02-in1 says HELO to 98.148.185.85:49522

334 VXNlcm5hbWU6

334 UGFzc3dvcmQ6

535 5.7.8 Sorry.

501 5.5.2 MAIL FROM syntax error
```

TLS or SSL Addition:

```
import socket
from socket import *
import ssl

msg = "\r\n I love computer networks!"
```

```
endmsg = "\r\n.\r\n"  # Choose a mail server (e.g. Google mail server) and
call it mailserver
mailserver = "mail.smtp.com"#Fill in start   #Fill in end  # Create socket
called clientSocket and establish a TCP connection with mailserver
# clientSocket = socket(AF_INET, SOCK_STREAM)

context = ssl.SSLContext(ssl.PROTOCOL_TLS_CLIENT)
# context.load_verify_locations('path/to/cabundle.pem')

with socket.socket(socket.AF_INET, socket.SOCK_STREAM, 0) as sock:
  with context.wrap_socket(sock, server_hostname=mailserver) as clientSocket:
    print(clientSocket.version())

serverPort = 80
clientSocket.connect((mailserver, serverPort))
#Fill in start
#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.')
clientSocket.sendall('AUTH LOGIN\r\n'.encode())
print(clientSocket.recv(1024).decode())
clientSocket.sendall(('mail1.smtp.com\r\n').encode())
print(clientSocket.recv(1024).decode())
clientSocket.sendall(('*****\r\n').encode())
print(clientSocket.recv(1024).decode())



# Send MAIL FROM command and print server response.
mailFrom = 'mail1.smtp.com'
mailCommand = 'MAIL FROM: <'+mailFrom+'>\r\n'
clientSocket.send(mailCommand.encode())
recv2 = clientSocket.recv(1024).decode()
print(recv2)
# Fill in start
# Fill in end
# Send RCPT TO command and print server response.
clientSocket.sendall(('RCPT TO: xyz - Programming Assignment 2 SMTP Socket
Programming\r\n').encode())
recv3 = clientSocket.recv(1024).decode()
print(recv3)
# Fill in start
# Fill in end
# Send DATA command and print server response.
clientSocket.sendall(('DATA: Programming Assignment 2 SMTP Socket
Programming\r\n').encode())
recv4 = clientSocket.recv(1024).decode()
print(recv4)
```

```
# Fill in start
# Fill in end
# Send message data.
message = 'Hello! This is my second Programming Assignment SMTP Socket
Programming!\r\n'
clientSocket.sendall(message.encode())
recv5 = clientSocket.recv(1024).decode()
print(recv5)
# Fill in start
# Fill in end
# Message ends with a single period.
clientSocket.sendall(endmsg.encode())
recv6 = clientSocket.recv(1024).decode()
print(recv6)
# Fill in start
# Fill in end
# Send QUIT command and get server response.
clientSocket.sendall('QUIT\r\n'.encode())
recv7 = clientSocket.recv(1024).decode()
print(recv7)
# Fill in start
# Fill in end
```