from socket import \*

# Choose a mail server (e.g. Google mail server) and call it mailserver

mailserver = 'mail.smtp2go.com' #Fill in start #Fill in end

mailport = 2525

# Create socket called clientSocket and establish a TCP connection with mailserver

#Fill in start

clientSocket = socket(AF\_INET,SOCK\_STREAM)

clientSocket.connect((mailserver, mailport))

#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 MAIL FROM command and print server response.

# Fill in start

mailFromCommand = 'MAIL FROM: <mail@mail.com>\r\n'

clientSocket.send(mailFromCommand.encode())

recv2 = clientSocket.recv(1024).decode()

print(recv2)

if recv2[:3] != '250':

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

# Fill in end

# Send RCPT TO command and print server response.

# Fill in start

rcptToCommand = 'RCPT TO: <mail@mail.com>\r\n'

clientSocket.send(rcptToCommand.encode())

recv3 = clientSocket.recv(1024).decode()

print(recv3)

if recv3[:3] != '250':

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

# Send DATA command and print server response.

# Fill in start

dataCommand = 'DATA\r\n'

clientSocket.send(dataCommand.encode())

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

print(recv4)

if recv4[:3] != '250':

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

# Fill in end

# Send message data.

# Fill in start

message = input('Enter message here: ')

# Fill in end

# Message ends with a single period.

# Fill in start

endMailMessage = '\r\n.\r\n'

byteMessage = (message + endMailMessage).encode()

clientSocket.send(byteMessage)

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

print(recv5)

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

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

# Fill in end

# Send QUIT command and get server response.

# Fill in start

quitCommand = 'QUIT\r\n'

clientSocket.send(quitCommand.encode())

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

print(recv6)

if recv6[:3] != '250':

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

# Fill in end

# Close the socket

clientSocket.close()

