**Python SMTP**

He, Ze(zh700@nyu.edu)

Code

|  |
| --- |
| from socket import \*  import base64  import ssl  msg = "\r\n I love computer networks!"  endmsg = "\r\n.\r\n"  #Fill in start  mailserver = 'smtp.qq.com'  mailport = 465  #Fill in end  # Fill in start  clientSocket = socket(AF\_INET, SOCK\_STREAM)  sslsocket = ssl.wrap\_socket(clientSocket)  sslsocket.connect((mailserver, mailport))  #Fill in end  recv1 = sslsocket.recv(1024).decode()  print recv1  if recv1[:3] != '220':  print '220 reply not received from server.'  # Send HELO command and pritn server response  heloCommand = 'HELO Ze\r\n'  sslsocket.send(heloCommand)  recv2 = sslsocket.recv(1024).decode()  print recv2  if recv2[:3] != '250':  print '250 reply not received from server.'  # Send MAIL FROM command and print server response.  # Fill in start  au\_login = "AUTH LOGIN\r\n"  sslsocket.send(au\_login.encode("utf-8"))  recv3 = sslsocket.recv(1024).decode()  print(recv3)  au\_user = base64.b64encode("87631834@qq.com".encode("utf-8"))  sslsocket.send(au\_user + "\r\n".encode("utf-8"))  recv4 = sslsocket.recv(1024).decode()  print(recv4)  au\_pw = base64.b64encode("Kataku20062339".encode("utf-8"))  sslsocket.send(au\_pw + "\r\n".encode("utf-8"))  recv5 = sslsocket.recv(1024).decode()  print(recv5)  mailFrom = 'MAIL FROM: <87631834@qq.com>\r\n'  sslsocket.send(mailFrom)  recv6 = sslsocket.recv(1024).decode()  print recv6  if recv6[:3] != '250':  print '250 reply not received from server.'  # Fill in end  # Send RCPT TO command and print server response.  # Fill in start  recipientTo = 'RCPT TO: <zh700@nyu.edu>\r\n'  sslsocket.send(recipientTo)  recv7 = sslsocket.recv(1024).decode()  print recv7  if recv7[:3] != '250':  print '250 reply not received from server.'  # Fill in end  # Send DATA command and print server response.  # Fill in start  sendingDataCommand = 'DATA\r\n'  print sendingDataCommand  sslsocket.send(sendingDataCommand)  recv8 = sslsocket.recv(1024).decode()  print recv8  if recv8[:3] == '250':  print '250 reply not received from server.'  # Fill in end  # Send message data.  # Fill in start  sslsocket.send( msg + endmsg )  recv9 = sslsocket.recv(1024).decode()  print recv9  if recv9[:3] != '250':  print '250 reply not received from server.'  # Fill in end  # Send QUIT command and get server response.  # Fill in start  quitMessage = 'QUIT\r\n'  print quitMessage  sslsocket.send(quitMessage)  recv10 = sslsocket.recv(1024).decode()  print recv10  if recv10[:3] != '250':  print '250 reply not received from server.'  # Fill in end |

Screemshots:



