Sockets Programming

REDES 2025-2026

UDP socket programming

UDP communication: no "connection" between client and server

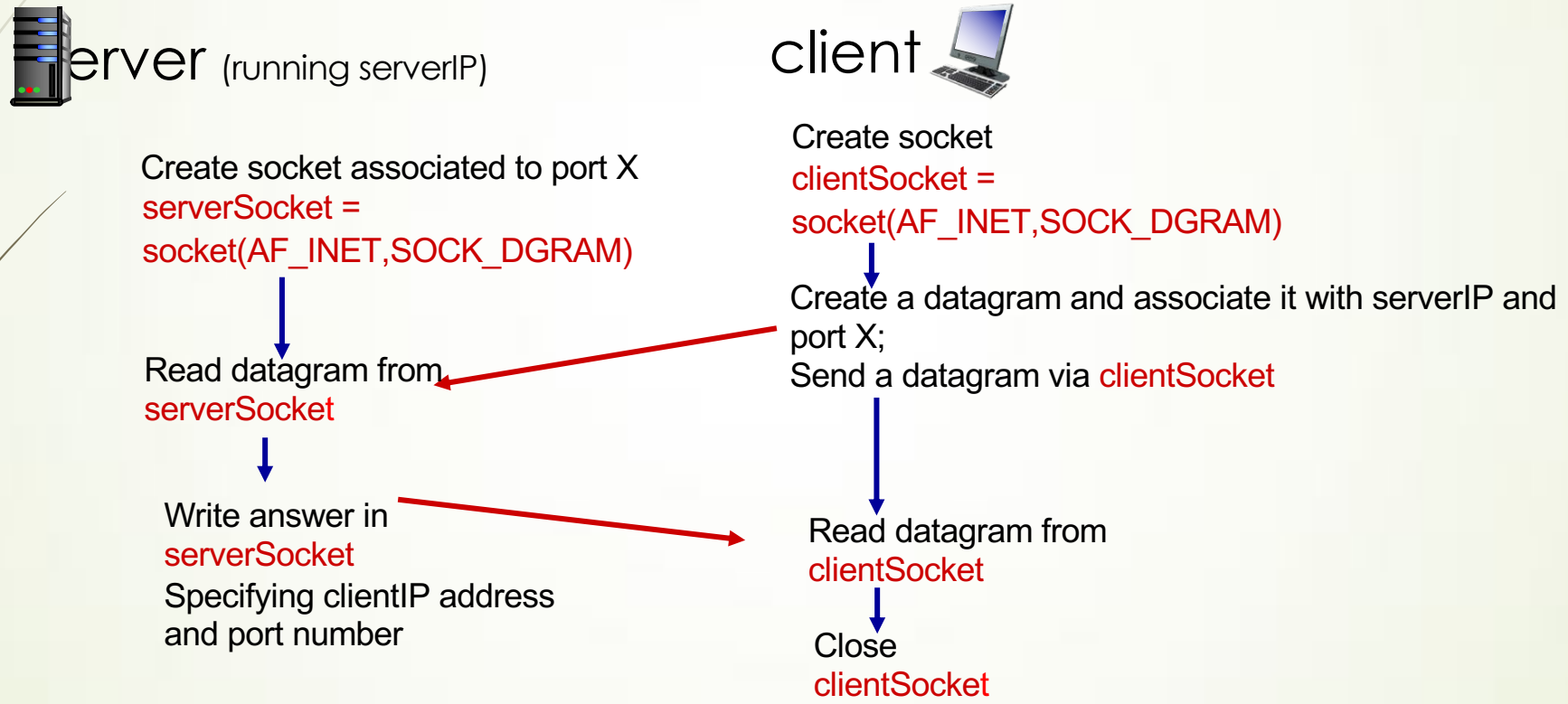
- No handshaking before starting data transmission
- The sender must explicitly places the destination IP address and port number in each datagram
- The receiver extracts the return address and port from the received datagram

UDP communication: data can be lost or received out of order

From an application perspective:

- UDP provides unreliable transfer of groups of bytes (datagrams or messages) between client and server

UDP socket – client/server interaction



UDP Client

Python client - UDPClient

Include the Python sockets library →
```
from socket import *
serverName = 'hostname'
serverPort = 12000
```
Create UDP socket →
```
clientSocket = socket(AF_INET, SOCK_DGRAM)
```
Get input from keyboard →
```
message = input('Input lowercase sentence:')
```
Define receiver IP address and port; Send through socket →
```
clientSocket.sendto(message.encode(),
                    (serverName, serverPort))
```
Get response to string →
```
modifiedMessage, serverAddress =
                    clientSocket.recvfrom(2048)
```
Print the received message; Close port →
```
print (modifiedMessage.decode())
clientSocket.close()
```

message.encode() converts the Python string variable to an array of bytes to be placed in the datagram

modifiedMessage.decode() does the reverse

UDP Server

Python server - UDPServer

```
from socket import *
serverPort = 12000
serverSocket = socket(AF_INET, SOCK_DGRAM)
serverSocket.bind(('' , serverPort))
print ("The server is ready to receive")
while True:
    message, clientAddress = serverSocket.recvfrom(2048)
    modifiedMessage = message.decode().upper()
    serverSocket.sendto(modifiedMessage.encode(),
                        clientAddress)
```

Create UDP socket →

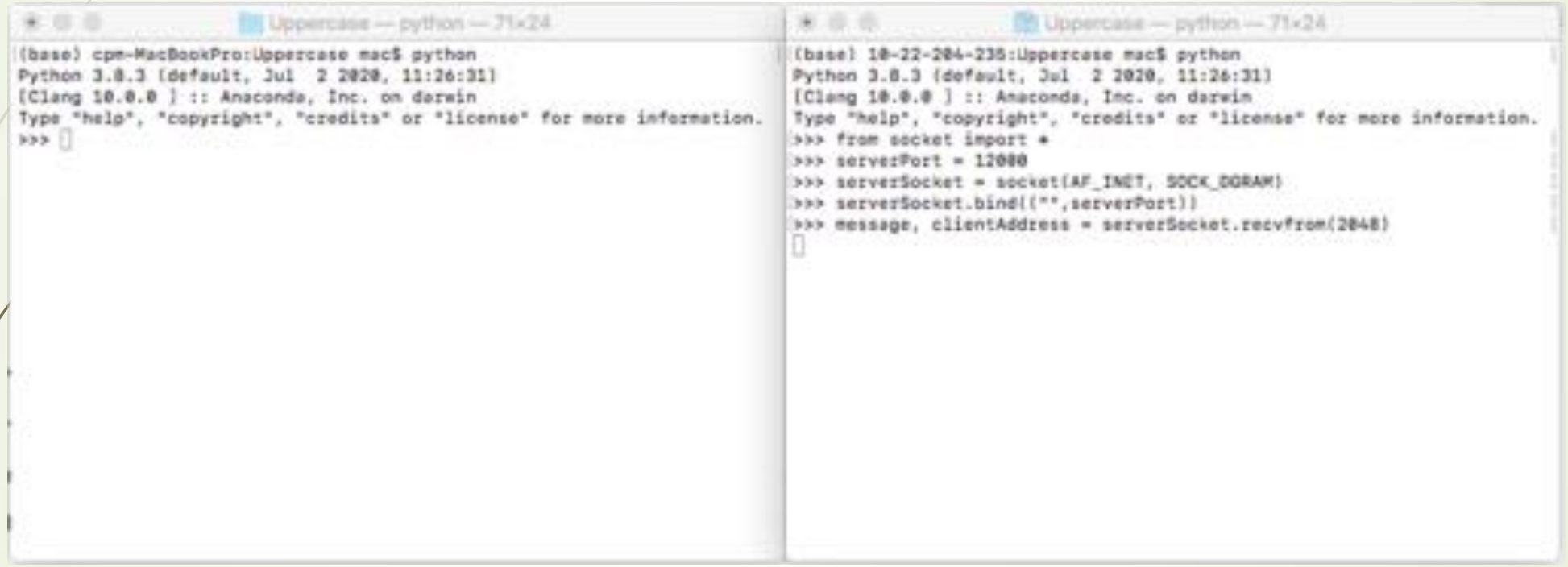
Associate to local port 12000 →

Server cycle →

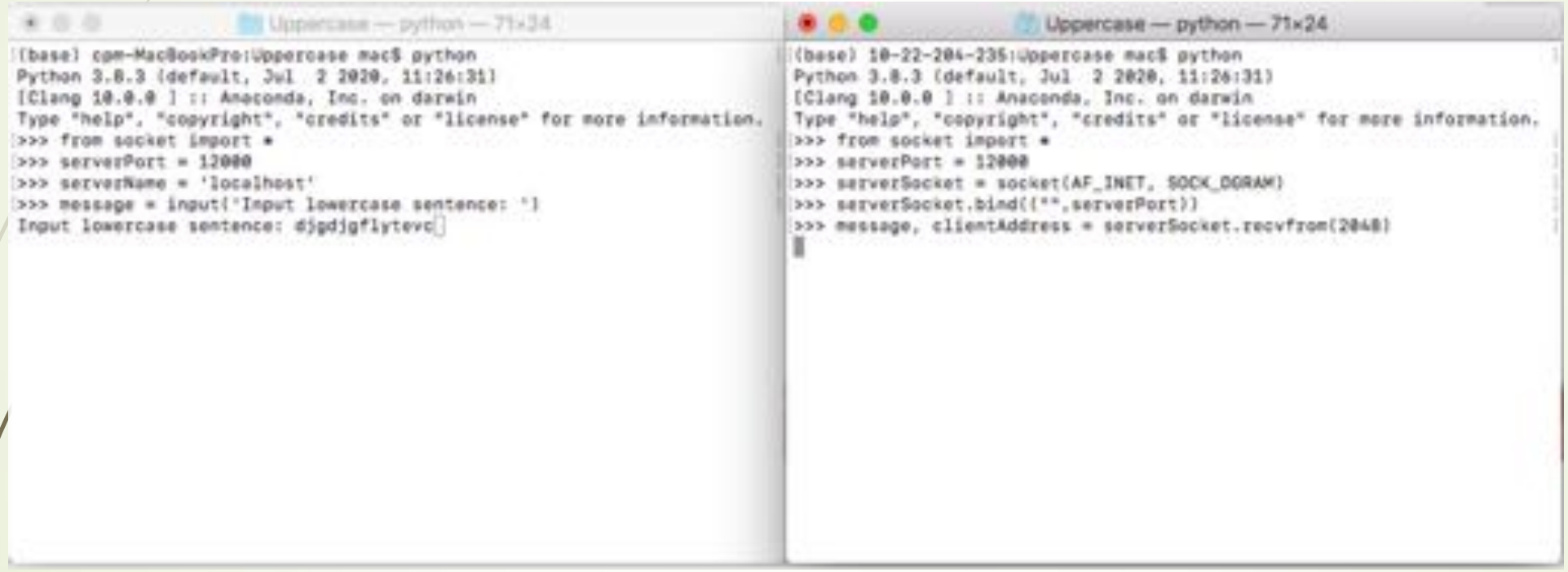
Read from UDP socket for message, get client address (client IP and port) →

Send message to client →

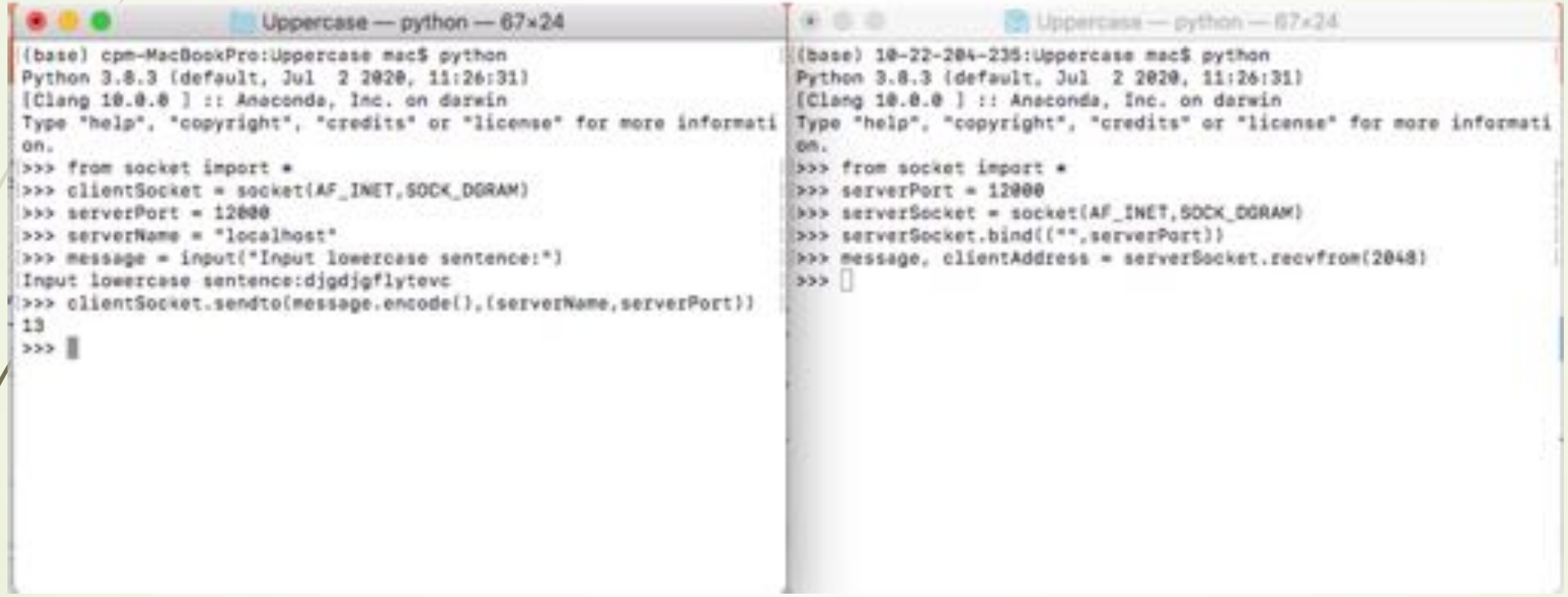
Server starts ⟶ is available



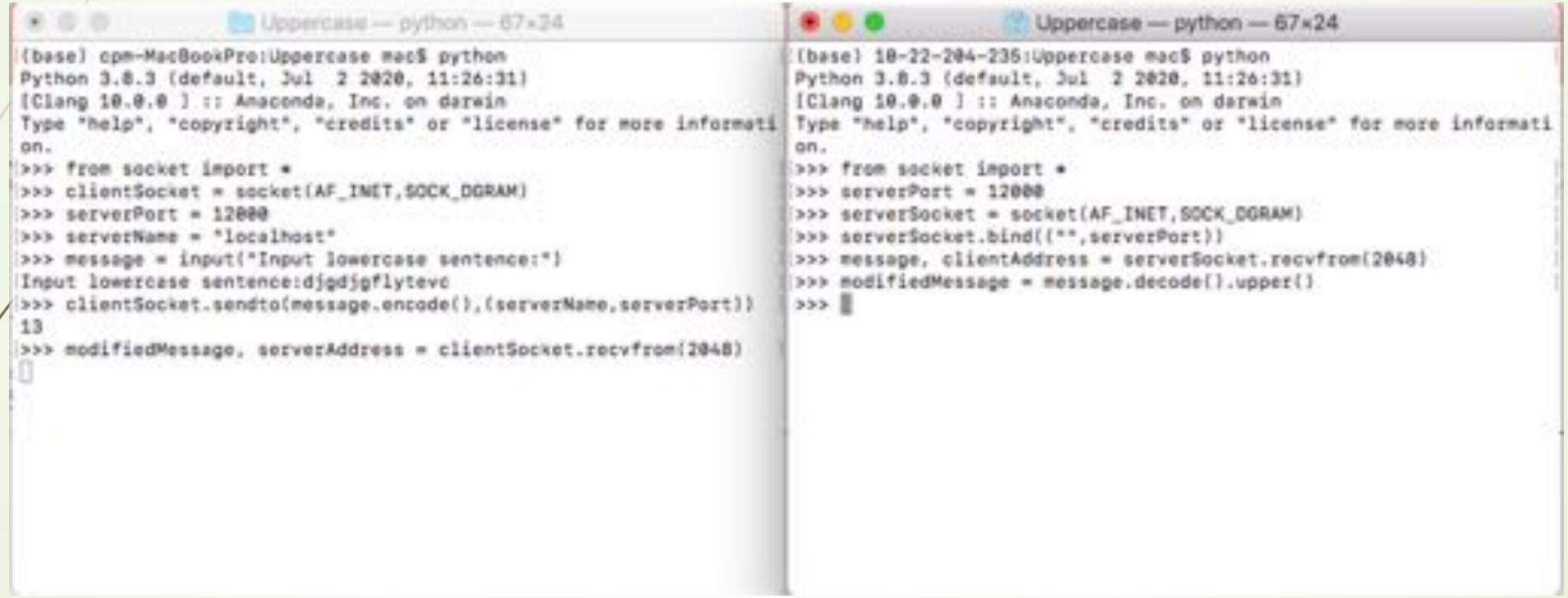
Client before sending datagram



After client sent and server received



Client receives modified message



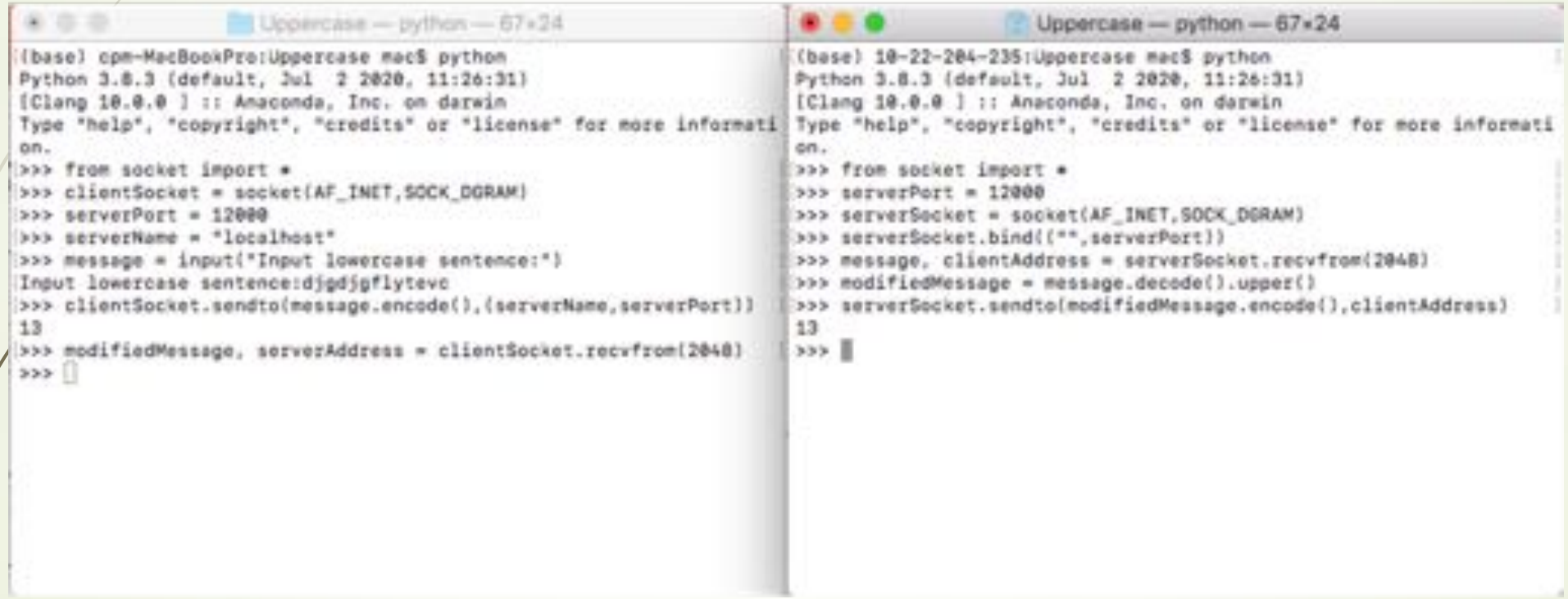
Uppercase — python — 67×24

```
(base) cpm-MacBookPro:Uppercase mac$ python
Python 3.8.3 (default, Jul  2 2020, 11:26:31)
[Clang 10.0.0 ] :: Anaconda, Inc. on darwin
Type "help", "copyright", "credits" or "license" for more informati
on.
>>> from socket import *
>>> clientSocket = socket(AF_INET,SOCK_DGRAM)
>>> serverPort = 12000
>>> serverName = "localhost"
>>> message = input("Input lowercase sentence:")
Input lowercase sentence:djgdjgflytevc
>>> clientSocket.sendto(message.encode(),(serverName,serverPort))
13
>>> modifiedMessage, serverAddress = clientSocket.recvfrom(2048)
```

Uppercase — python — 67×24

```
(base) 10-22-204-235:Uppercase mac$ python
Python 3.8.3 (default, Jul  2 2020, 11:26:31)
[Clang 10.0.0 ] :: Anaconda, Inc. on darwin
Type "help", "copyright", "credits" or "license" for more informati
on.
>>> from socket import *
>>> serverPort = 12000
>>> serverSocket = socket(AF_INET,SOCK_DGRAM)
>>> serverSocket.bind(("",serverPort))
>>> message, clientAddress = serverSocket.recvfrom(2048)
>>> modifiedMessage = message.decode().upper()
>>>
```

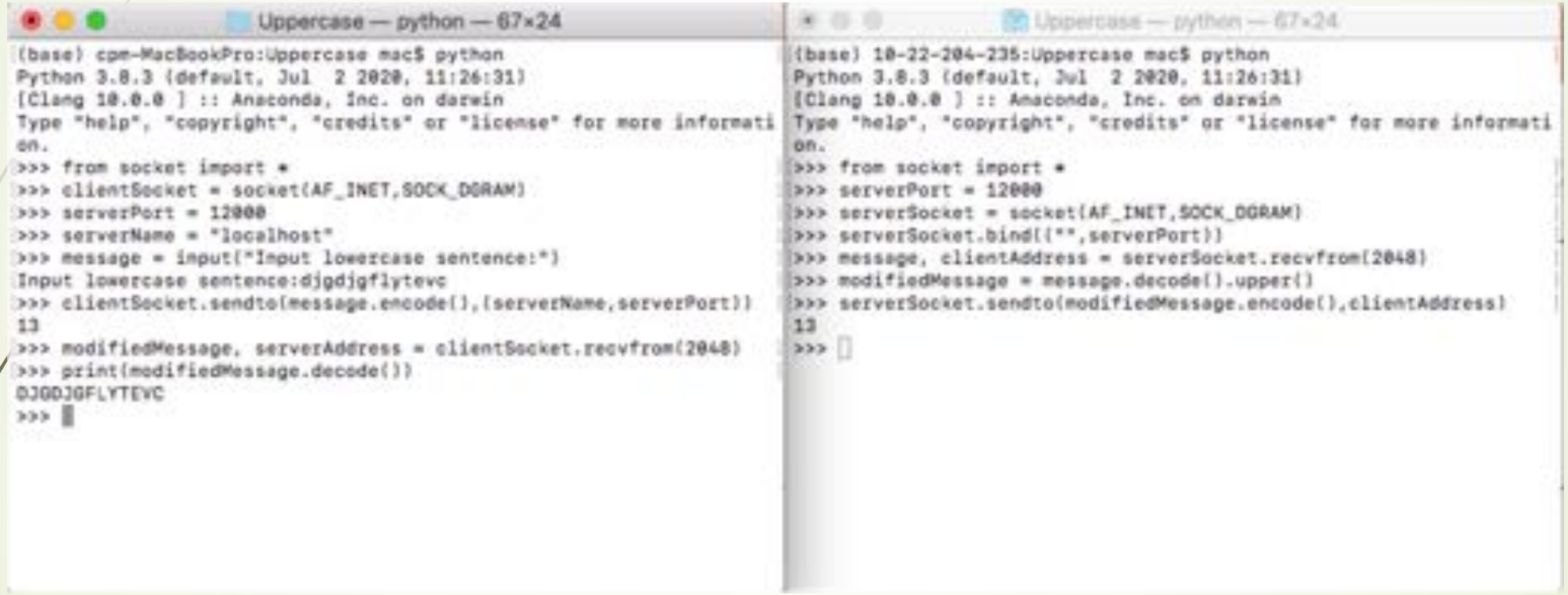
Client receives modified message



```
(base) cpm-MacBookPro:Uppercase mac$ python
Python 3.8.3 (default, Jul  2 2020, 11:26:31)
[Clang 10.0.0 ] :: Anaconda, Inc. on darwin
Type "help", "copyright", "credits" or "license" for more informati
on.
>>> from socket import *
>>> clientSocket = socket(AF_INET,SOCK_DGRAM)
>>> serverPort = 12000
>>> serverName = "localhost"
>>> message = input("Input lowercase sentence:")
Input lowercase sentence:djgdjgflytevc
>>> clientSocket.sendto(message.encode(),(serverName,serverPort))
13
>>> modifiedMessage, serverAddress = clientSocket.recvfrom(2048)
>>>
```

```
(base) 10-22-204-235:Uppercase mac$ python
Python 3.8.3 (default, Jul  2 2020, 11:26:31)
[Clang 10.0.0 ] :: Anaconda, Inc. on darwin
Type "help", "copyright", "credits" or "license" for more informati
on.
>>> from socket import *
>>> serverPort = 12000
>>> serverSocket = socket(AF_INET,SOCK_DGRAM)
>>> serverSocket.bind(("",serverPort))
>>> message, clientAddress = serverSocket.recvfrom(2048)
>>> modifiedMessage = message.decode().upper()
>>> serverSocket.sendto(modifiedMessage.encode(),clientAddress)
13
>>>
```

Client receives modified message



Client and server

```python
from socket import *

clientSocket = socket(AF_INET,SOCK_DGRAM)

serverName="localhost"
serverPort= 12000

message=input("Input lowercase sequence: ")

clientSocket.sendto( message.encode(),(serverName, serverPort))

modifiedMessage, serverAddres = clientSocket.recvfrom(2048)

print(modifiedMessage.decode())

clientSocket.close()
```

```python
from socket import *

serverSocket = socket(AF_INET,SOCK_DGRAM)

serverPort= 12000

serverSocket.bind(("",serverPort))

while  True:
    message, clientAddress = serverSocket.recvfrom( 2048)
    modifiedMessage = message.decode().upper()
    serverSocket.sendto( modifiedMessage.encode(), clientAddress)

serverSocket.close()
```