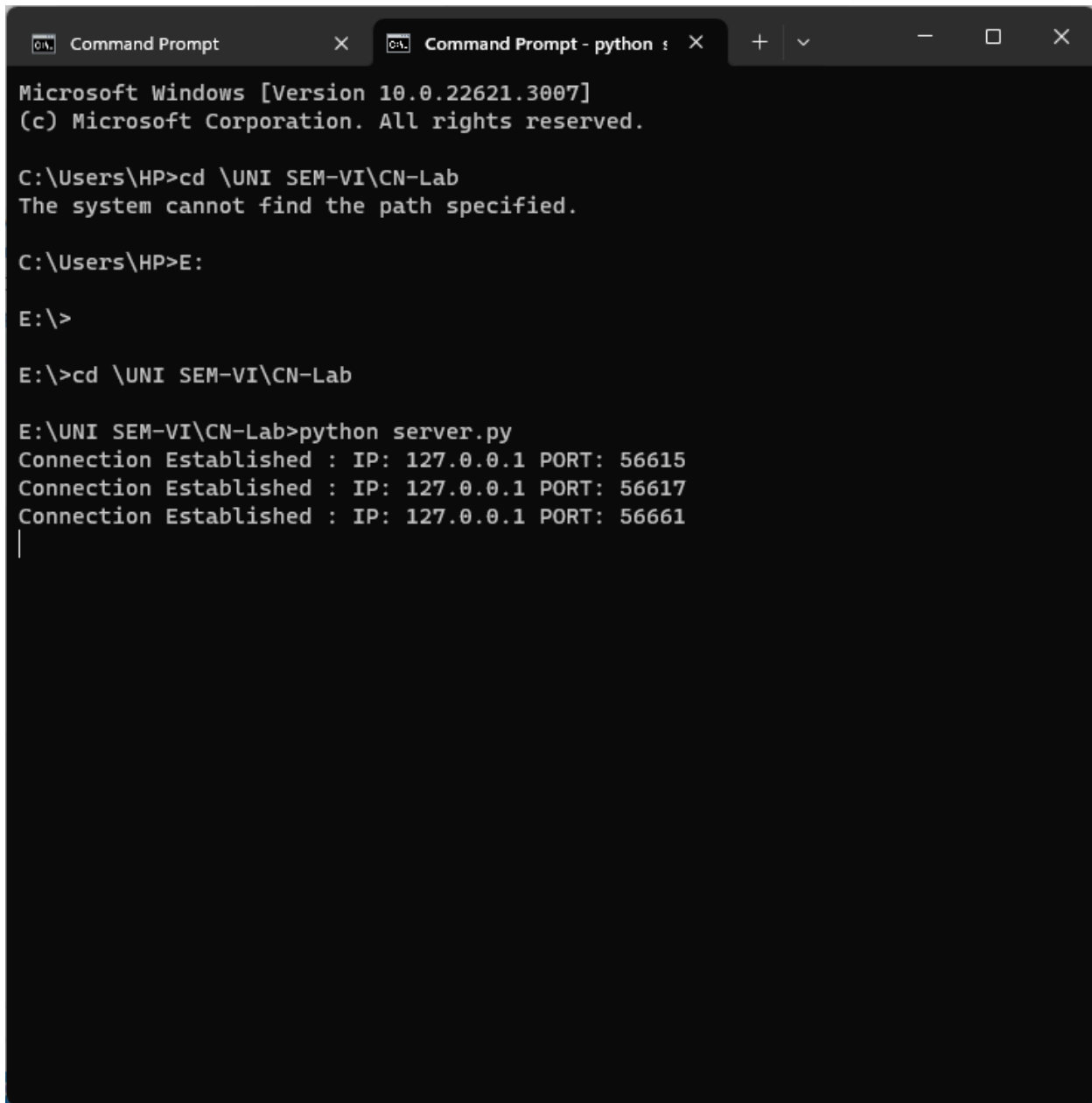


K214553 CN LAB 03

SERVER:



```
Microsoft Windows [Version 10.0.22621.3007]
(c) Microsoft Corporation. All rights reserved.

C:\Users\HP>cd \UNI SEM-VI\CN-Lab
The system cannot find the path specified.

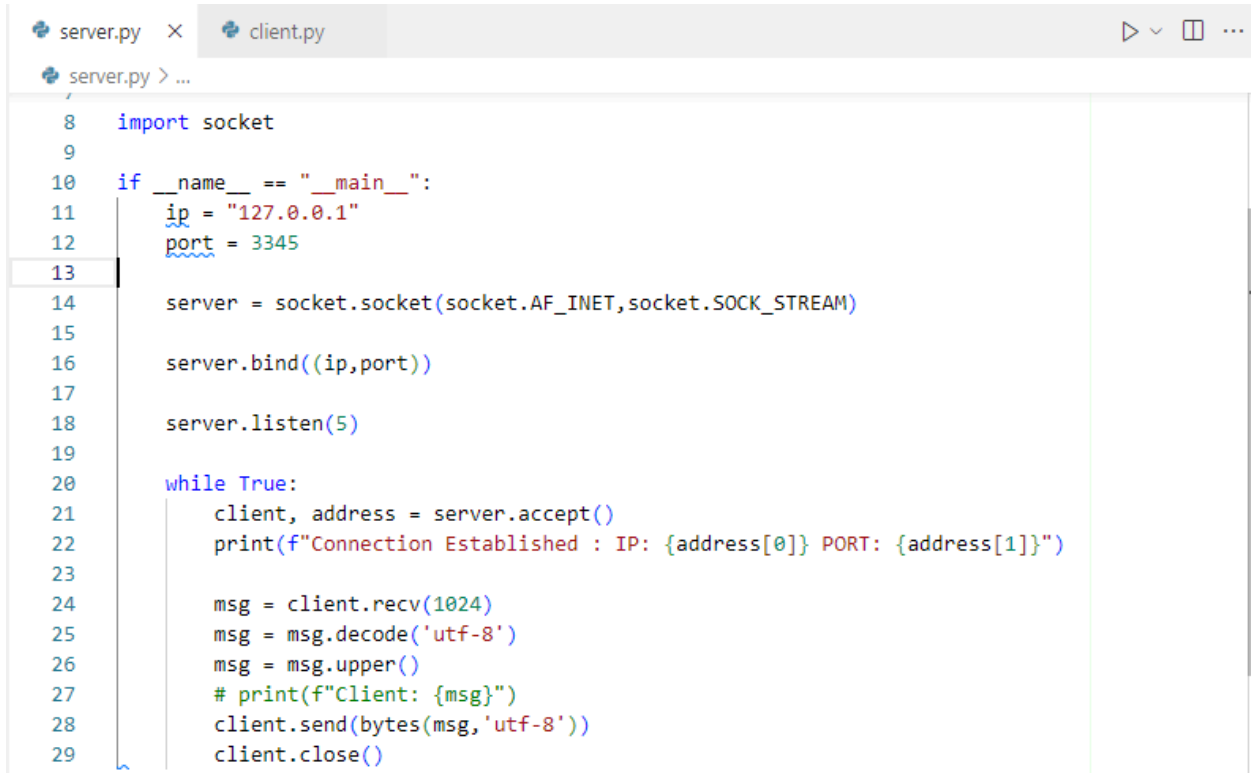
C:\Users\HP>E:

E:\>

E:\>cd \UNI SEM-VI\CN-Lab

E:\UNI SEM-VI\CN-Lab>python server.py
Connection Established : IP: 127.0.0.1 PORT: 56615
Connection Established : IP: 127.0.0.1 PORT: 56617
Connection Established : IP: 127.0.0.1 PORT: 56661
|
```

K214553 CN LAB 03



The image shows a code editor window with two tabs: 'server.py' and 'client.py'. The 'server.py' tab is active, displaying a Python script for a server. The script uses the 'socket' module to create a server socket, bind it to the IP address '127.0.0.1' on port '3345', and listen for connections. It then enters a 'while True' loop where it accepts connections, prints the connection details, receives a message, decodes it, converts it to uppercase, and sends it back to the client before closing the connection.

```
server.py  x  client.py
server.py > ...
8  import socket
9
10 if __name__ == "__main__":
11     ip = "127.0.0.1"
12     port = 3345
13
14     server = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
15
16     server.bind((ip, port))
17
18     server.listen(5)
19
20     while True:
21         client, address = server.accept()
22         print(f"Connection Established : IP: {address[0]} PORT: {address[1]}")
23
24         msg = client.recv(1024)
25         msg = msg.decode('utf-8')
26         msg = msg.upper()
27         # print(f"Client: {msg}")
28         client.send(bytes(msg, 'utf-8'))
29         client.close()
```

K214553 CN LAB 03

CLIENT:

```
Command Prompt
Microsoft Windows [Version 10.0.22621.3007]
(c) Microsoft Corporation. All rights reserved.

C:\Users\HP>E:

E:\>cd \UNI SEM-VI\CN-Lab

E:\UNI SEM-VI\CN-Lab>python client.py
Enter: abcd

E:\UNI SEM-VI\CN-Lab>
E:\UNI SEM-VI\CN-Lab>python client.py
Enter: abdc

E:\UNI SEM-VI\CN-Lab>python client.py
Enter: abcd
Server: ABCD

E:\UNI SEM-VI\CN-Lab>python client.py
Enter: Rabia
Server: RABIA

E:\UNI SEM-VI\CN-Lab>python client.py
Enter: apple
Server: APPLE

E:\UNI SEM-VI\CN-Lab>|
```

K214553 CN LAB 03



The image shows a code editor window with two tabs: 'server.py' and 'client.py'. The 'client.py' tab is active, displaying a Python script. The script imports the 'socket' module and defines a main function. It sets the IP address to '127.0.0.1' and the port to 3345. It then creates a socket object, connects it to the specified IP and port, and enters a loop where it receives data from the server, decodes it, and prints it. The line numbers 1 through 16 are visible on the left side of the editor.

```
client.py > ...
1  import socket
2
3  if __name__ == "__main__":
4      ip = "127.0.0.1"
5      port = 3345
6
7      server = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
8
9      server.connect((ip, port))
10
11     string = input("Enter: ")
12     server.send(bytes(string, 'utf-8'))
13     msg = server.recv(1024)
14     msg = msg.decode('utf-8')
15     print(f"Server: {msg}")
16
```