INSTITUTE OF COMPUTER TECHNOLOGY

B-TECH COMPUTER SCIENCE ENGINEERING 2025-26

SUBJECT:COMPUTER NETWORKS

NAME: Rahul Prajapati

ENRLL. NO: 23162171020

BRANCH: CYBER SECURITY

BATCH: 52

PRACTICAL_07

Aim: To implement Socket Programming Scenario: An organization named Albert Enterprise has established two departments for better performance of the company, as each department will be having some specific set of tasks to perform. So, this will reduce the time and increase the efficiency of the work. As both the departments are dependent on each other, they need to communicate more frequently. To solve the problem, the IT department has suggested the option to create a chat application using socket programming which will work only in the office premises. So, help the IT professionals to create the chat application.

Make sure that the application has the below mentioned features:

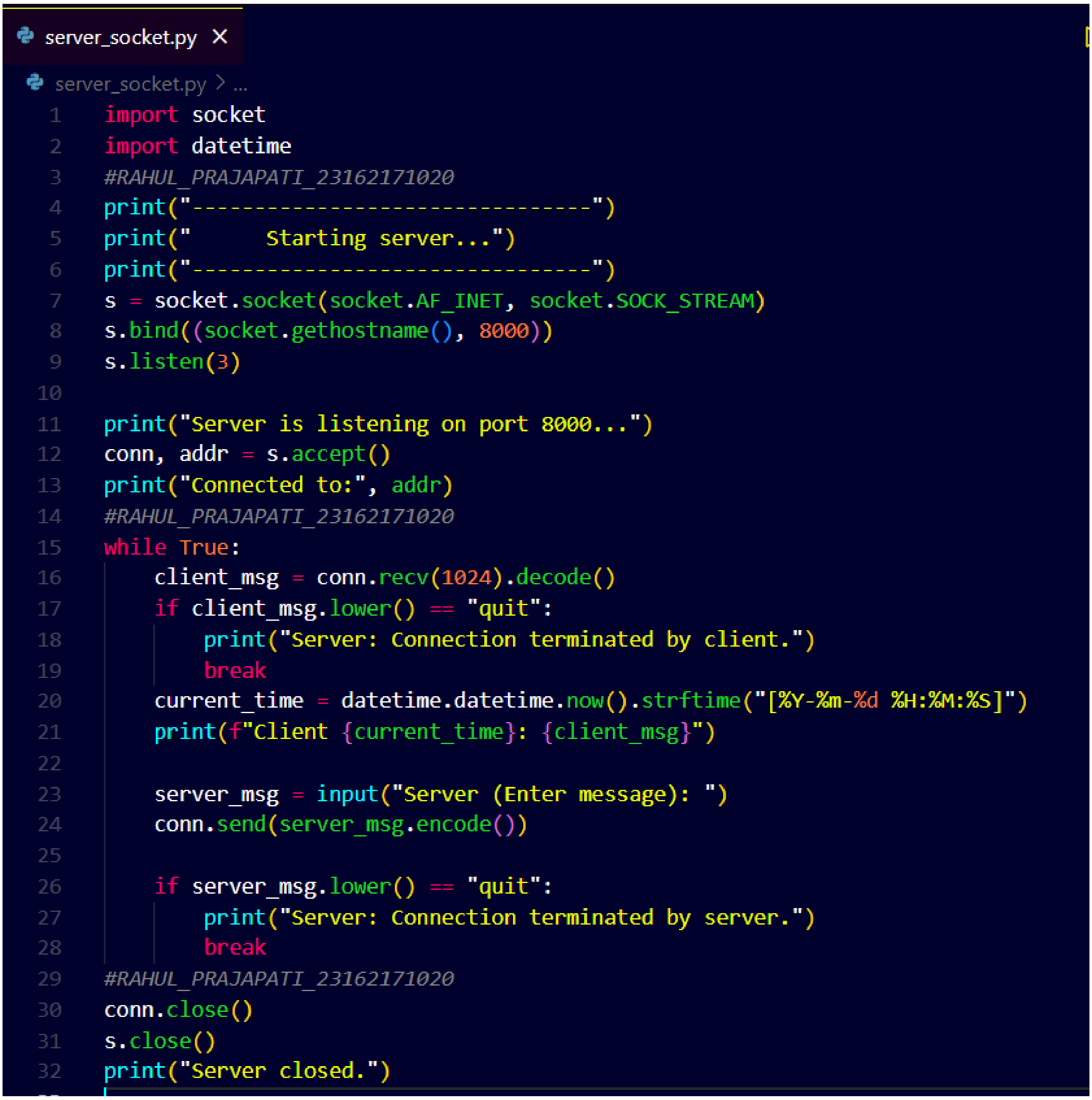
1) Department 1 will be set as the SERVER while department 2 will be set as a CLIENT device.

2) The message received by CLIENT or SERVER must be displayed with time stamp.

3) If any of the device irrespective of CLIENT or SERVER has sent the message that the “quit”, then connection should be closed on both the ends.

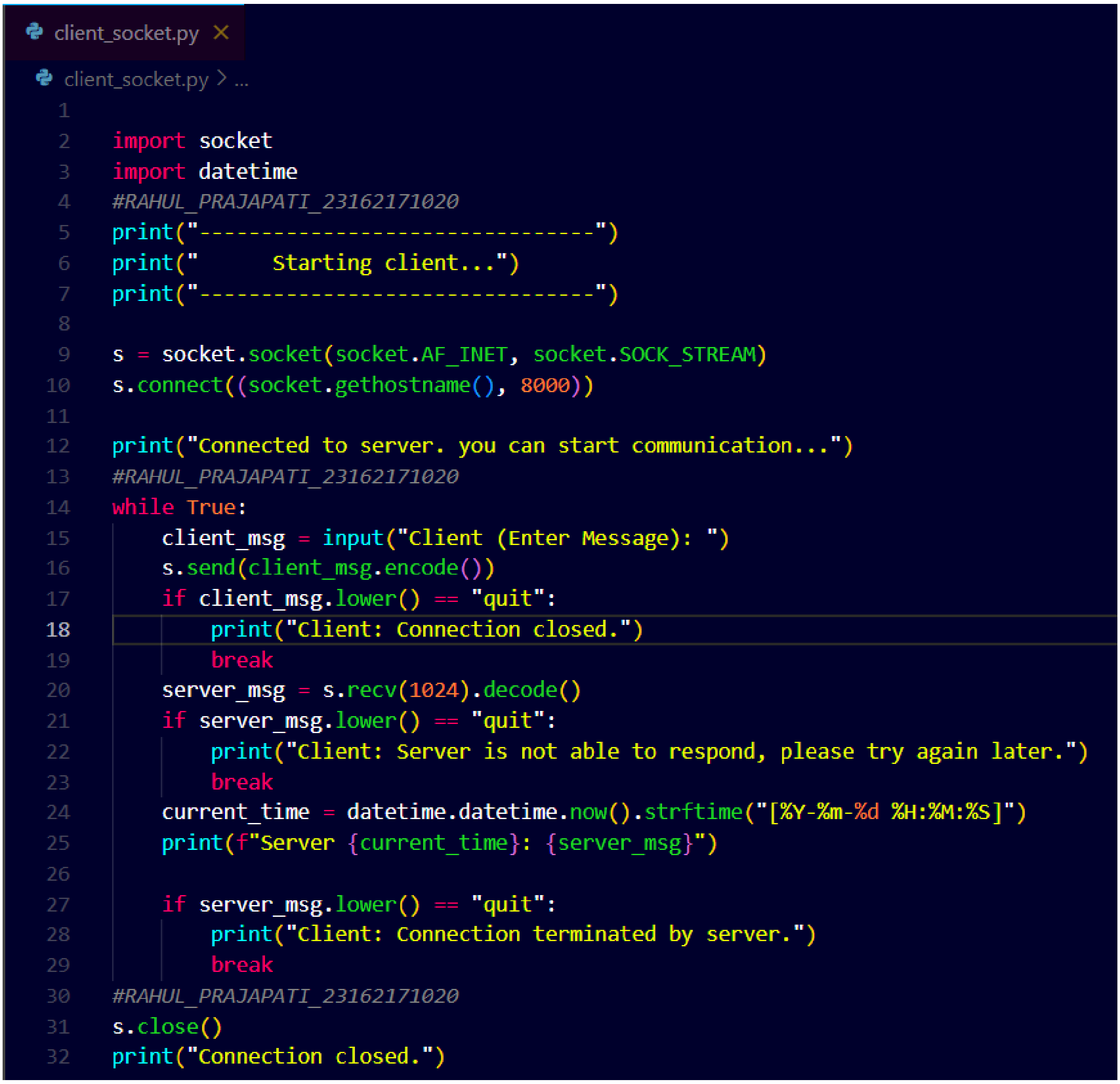
4) There is no restriction on the protocol selection, you can use UDP or TCP. Justify the reason for selection of the specific protocol.

1. SERVER PROGRAM:



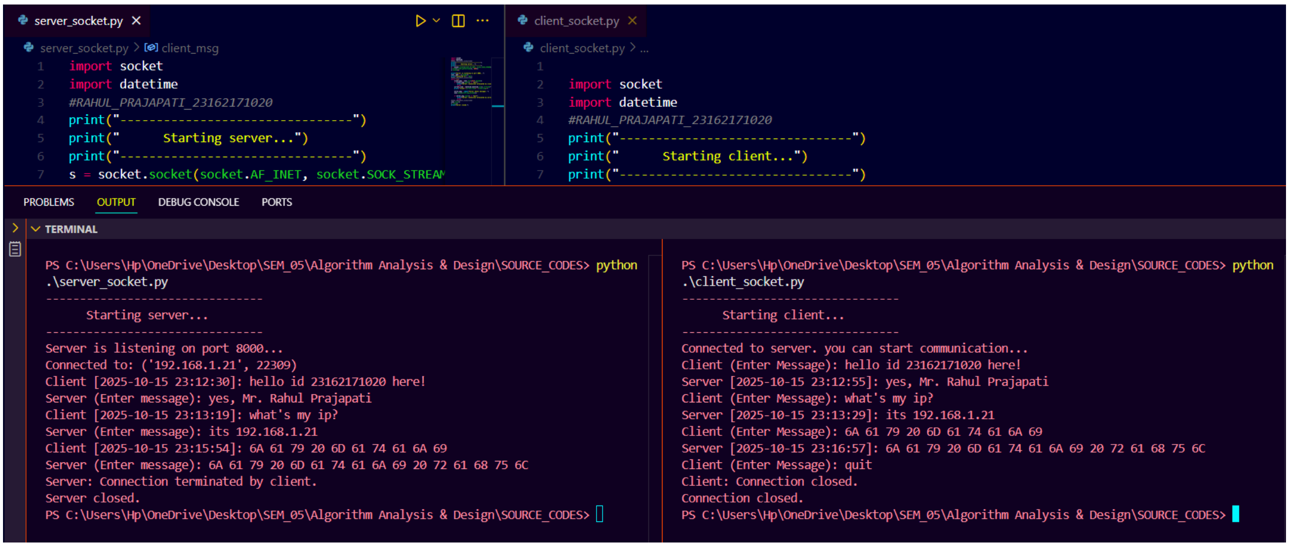
```python
import socket
import datetime
#RAHUL_PRAJAPATI_23162171020
print("--------------------------------")
print("      Starting server...")
print("--------------------------------")
s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
s.bind((socket.gethostname(), 8000))
s.listen(3)

print("Server is listening on port 8000...")
conn, addr = s.accept()
print("Connected to:", addr)
#RAHUL_PRAJAPATI_23162171020
while True:
    client_msg = conn.recv(1024).decode()
    if client_msg.lower() == "quit":
        print("Server: Connection terminated by client.")
        break
    current_time = datetime.datetime.now().strftime("[%Y-%m-%d %H:%M:%S]")
    print(f"Client {current_time}: {client_msg}")

    server_msg = input("Server (Enter message): ")
    conn.send(server_msg.encode())

    if server_msg.lower() == "quit":
        print("Server: Connection terminated by server.")
        break
#RAHUL_PRAJAPATI_23162171020
conn.close()
s.close()
print("Server closed.")
```

2. CLIENT PROGRAM:



```python

import socket
import datetime
#RAHUL_PRAJAPATI_23162171020
print("--------------------------------")
print("      Starting client...")
print("--------------------------------")

s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
s.connect((socket.gethostname(), 8000))

print("Connected to server. you can start communication...")
#RAHUL_PRAJAPATI_23162171020
while True:
    client_msg = input("Client (Enter Message): ")
    s.send(client_msg.encode())
    if client_msg.lower() == "quit":
        print("Client: Connection closed.")
        break
    server_msg = s.recv(1024).decode()
    if server_msg.lower() == "quit":
        print("Client: Server is not able to respond, please try again later.")
        break
    current_time = datetime.datetime.now().strftime("[%Y-%m-%d %H:%M:%S]")
    print(f"Server {current_time}: {server_msg}")

    if server_msg.lower() == "quit":
        print("Client: Connection terminated by server.")
        break
#RAHUL_PRAJAPATI_23162171020
s.close()
print("Connection closed.")
```

OUTPUT:



```python
import socket
import datetime
#RAHUL_PRAJAPATI_23162171020
print("--------------------------------")
print("      Starting server...")
print("--------------------------------")
s = socket.socket(socket.AF_INET, socket.SOCK_STREAM
```

```python

import socket
import datetime
#RAHUL_PRAJAPATI_23162171020
print("--------------------------------")
print("      Starting client...")
print("--------------------------------")
```

```
PS C:\Users\Hp\OneDrive\Desktop\SEM_05\Algorithm Analysis & Design\SOURCE_CODES> python
.\server_socket.py
--------------------------------
      Starting server...
--------------------------------
Server is listening on port 8000...
Connected to: ('192.168.1.21', 22309)
Client [2025-10-15 23:12:30]: hello id 23162171020 here!
Server (Enter message): yes, Mr. Rahul Prajapati
Client [2025-10-15 23:13:19]: what's my ip?
Server (Enter message): its 192.168.1.21
Client [2025-10-15 23:15:54]: 6A 61 79 20 6D 61 74 61 6A 69
Server (Enter message): 6A 61 79 20 6D 61 74 61 6A 69 20 72 61 68 75 6C
Server: Connection terminated by client.
Server closed.
PS C:\Users\Hp\OneDrive\Desktop\SEM_05\Algorithm Analysis & Design\SOURCE_CODES>
```

```
PS C:\Users\Hp\OneDrive\Desktop\SEM_05\Algorithm Analysis & Design\SOURCE_CODES> python
.\client_socket.py
--------------------------------
      Starting client...
--------------------------------
Connected to server. you can start communication...
Client (Enter Message): hello id 23162171020 here!
Server [2025-10-15 23:12:55]: yes, Mr. Rahul Prajapati
Client (Enter Message): what's my ip?
Server [2025-10-15 23:13:29]: its 192.168.1.21
Client (Enter Message): 6A 61 79 20 6D 61 74 61 6A 69
Server [2025-10-15 23:16:57]: 6A 61 79 20 6D 61 74 61 6A 69 20 72 61 68 75 6C
Client (Enter Message): quit
Client: Connection closed.
Connection closed.
PS C:\Users\Hp\OneDrive\Desktop\SEM_05\Algorithm Analysis & Design\SOURCE_CODES>
```

Conclusion: The chat application works well for communication between both departments using **TCP** for reliable message delivery. Messages show the time they are sent, and the chat closes safely when "quit" is typed. It helps both teams share information quickly and easily.