Name : Bramha Nimbalkar

Roll no : 7

Srn : 202100381

ASSIGMENT 8

Socket Programming for UDP Client, UDP Server.

Client

```
import java.net.DatagramPacket;
import java.net.DatagramSocket;
import java.net.InetAddress;

public class UDPClient {
    public static void main(String[] args) {
        try {
            DatagramSocket socket = new DatagramSocket();

            String message = "Hello, UDP Server!";
            byte[] sendData = message.getBytes();
            InetAddress serverAddress = InetAddress.getByName("localhost");
// Change to the server's IP address if not local

            DatagramPacket sendPacket = new DatagramPacket(sendData,
sendData.length, serverAddress, 9876);
            socket.send(sendPacket);

            System.out.println("Sent message: " + message);

            byte[] receiveData = new byte[1024];
            DatagramPacket receivePacket = new DatagramPacket(receiveData,
receiveData.length);
            socket.receive(receivePacket);

//            String acknowledgment = new String(receivePacket.getData(),
0, receivePacket.getLength());
//            System.out.println("Received acknowledgment: " +
acknowledgment);

            socket.close();
        } catch (Exception e) {
            e.printStackTrace();
        }
```

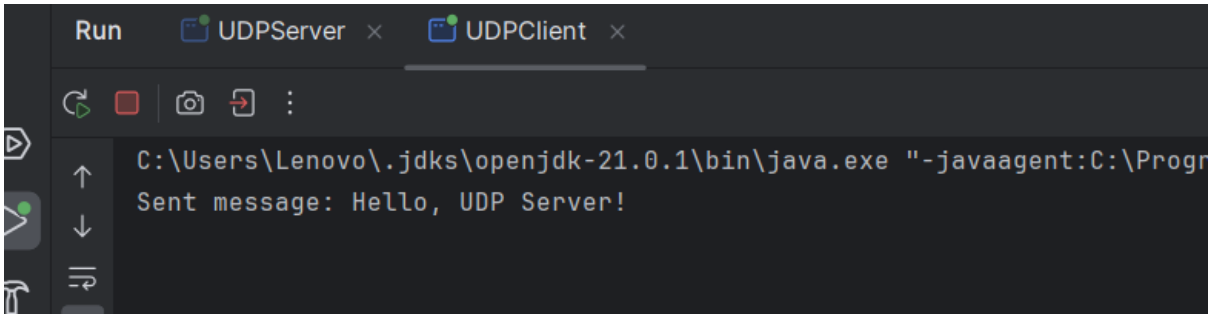
```
    }
}
```

Server

```
import java.net.DatagramPacket;
import java.net.DatagramSocket;

public class UDPServer {
    public static void main(String[] args) {
        try {
            DatagramSocket socket = new DatagramSocket(9876); // Create a
UDP socket on port 9876

            byte[] receiveData = new byte[1024];

            while (true) {
                DatagramPacket receivePacket = new
DatagramPacket(receiveData, receiveData.length);
                socket.receive(receivePacket);

                String message = new String(receivePacket.getData(), 0,
receivePacket.getLength());
                System.out.println("Received message: " + message);

                // Add your logic here to process the received message.

                // Send an acknowledgment back to the client
//                String acknowledgment = "Message received by server";
//                byte[] sendData = acknowledgment.getBytes();
//                DatagramPacket sendPacket = new DatagramPacket(sendData,
sendData.length, receivePacket.getAddress(), receivePacket.getPort());
//                socket.send(sendPacket);
            }
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

Output

Client



Server