CSE 303L
Computer Network
Lab
Week 6
Assignment

Name: Prafull Raj

Roll no.: AP21110011016

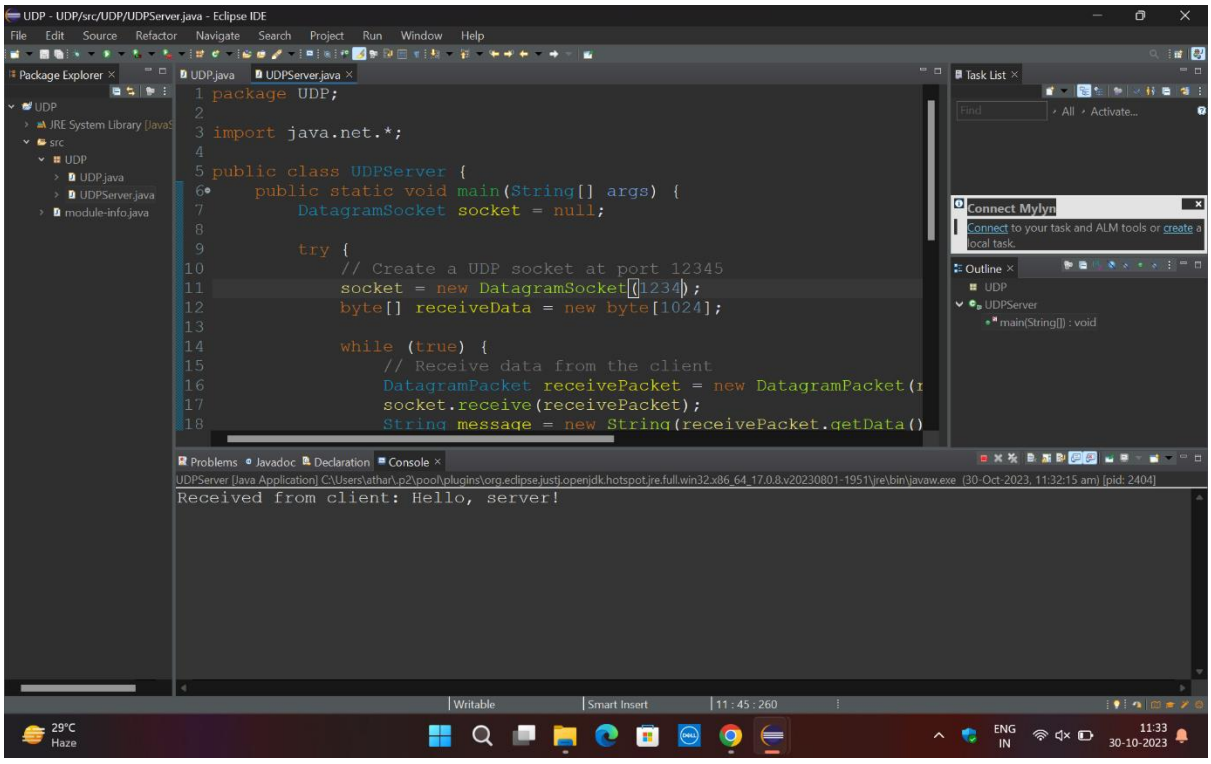
Dept: CSE **Section:** P

Q. Write a UDP server and UDP client programs. List out your observations between TCP and UDP based client/server socket API.

UDP Server

```
package UDP;

import java.net.*;


public class UDPServer {
   public static void main(String[] args) {
      DatagramSocket socket = null;


      try {
         // Create a UDP socket at port 12345
         socket = new DatagramSocket(1234);
         byte[] receiveData = new byte[1024];


         while (true) {
            // Receive data from the client
            DatagramPacket receivePacket = new DatagramPacket(receiveData, receiveData.length);
            socket.receive(receivePacket);
            String message = new String(receivePacket.getData(), 0, receivePacket.getLength());


            // Print received message
            System.out.println("Received from client: " + message);


            // Send a response to the client
            InetAddress clientAddress = receivePacket.getAddress();
            int clientPort = receivePacket.getPort();
            String responseMessage = "Hello, client!";
            byte[] sendData = responseMessage.getBytes();
            DatagramPacket sendPacket = new DatagramPacket(sendData, sendData.length, clientAddress,
clientPort);
            socket.send(sendPacket);
```

```
        }
    } catch (Exception e) {
        e.printStackTrace();
    } finally {
        if (socket != null && !socket.isClosed()) {
            socket.close();
        }
    }
  }
}
```



UDP Client

```
package ComputerNetwork;
import java.net.*;
public class UDPClient {
   public static void main(String[] args) {
      DatagramSocket socket = null;

      try {
         // Create a UDP socket
         socket = new DatagramSocket();

         // Server address and port to send data
         InetAddress serverAddress = InetAddress.getByName("localhost");
         int serverPort = 1234;

         // Message to be sent to the server
         String message = "Hello, server!";
         byte[] sendData = message.getBytes();

         // Send the message to the server
         DatagramPacket sendPacket = new DatagramPacket(sendData, sendData.length, serverAddress,
serverPort);
         socket.send(sendPacket);

         // Receive response from the server
         byte[] receiveData = new byte[1024];
         DatagramPacket receivePacket = new DatagramPacket(receiveData, receiveData.length);
         socket.receive(receivePacket);
         String responseMessage = new String(receivePacket.getData(), 0, receivePacket.getLength());

         // Print the response received from the server
         System.out.println("Received from server: " + responseMessage);
```

```
        } catch (Exception e) {
            e.printStackTrace();
        } finally {
            if (socket != null && !socket.isClosed()) {
                socket.close();
            }
        }
    }
}
```