# LAB SESSION-6

COURSE:- Computer Networks Lab

Course Code:- BCSE308P

Faculty:- Anita X

Name:- Pratham Jethwani

Reg. No.:- 21BRS1593

- **TCP Client server chat application using socket programming.**

Client:

```
import java.io.*;
import java.net.*;

public class client1 {

    public static void main(String[] args) throws Exception {
        String hostName = "localhost";
        int portNumber = 8000;
        Socket clientSocket = new Socket(hostName, portNumber);
        System.out.println("Connected to server: " +
clientSocket.getInetAddress().getHostAddress() + " on port " +
clientSocket.getPort());
        BufferedReader inFromUser = new BufferedReader(new
InputStreamReader(System.in));
        BufferedReader inFromServer = new BufferedReader(new
InputStreamReader(clientSocket.getInputStream()));
        DataOutputStream outToServer = new
DataOutputStream(clientSocket.getOutputStream());
        String clientMessage, serverMessage;
        while(true) {
            System.out.print("Enter message to send: ");
            clientMessage = inFromUser.readLine();
            outToServer.writeBytes(clientMessage + "\n");
            if(clientMessage.equals("bye")) {
                System.out.println("Disconnecting from server...");
                break;
            }
            serverMessage = inFromServer.readLine();
            System.out.println("Received response from server: " +
serverMessage);
        }
        clientSocket.close();
    }
}
```

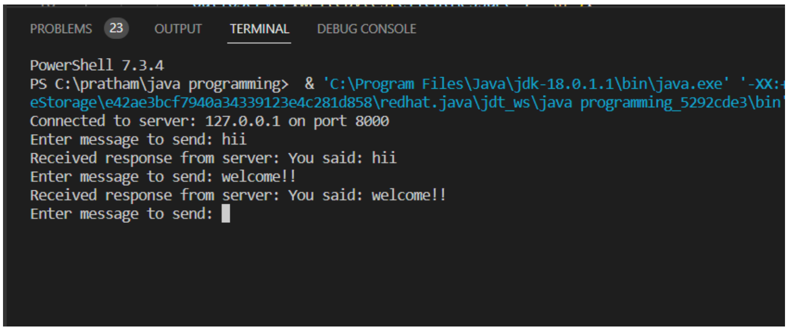
Server:

```
import java.io.*;
import java.net.*;

public class server2 {

    public static void main(String[] args) throws Exception {
        ServerSocket serverSocket = new ServerSocket(8000);
        System.out.println("Server started. Listening for connections on port 
8000...");
        Socket clientSocket = serverSocket.accept();
        System.out.println("Client connected: " + 
clientSocket.getInetAddress().getHostAddress() + " on port " + 
clientSocket.getPort());
        BufferedReader inFromClient = new BufferedReader(new 
InputStreamReader(clientSocket.getInputStream()));
        DataOutputStream outToClient = new 
DataOutputStream(clientSocket.getOutputStream());
        String clientMessage, serverMessage;
        while(true) {
            clientMessage = inFromClient.readLine();
            if(clientMessage.equals("bye")) {
                System.out.println("Client has disconnected.");
                break;
            }
            System.out.println("Received message from client: " + 
clientMessage);
            serverMessage = "You said: " + clientMessage + "\n";
            outToClient.writeBytes(serverMessage);
        }
        serverSocket.close();
    }
}
```

Output:



```
PowerShell 7.3.4
PS C:\pratham\java programming>  & 'C:\Program Files\Java\jdk-18.0.1.1\bin\java.exe' '-XX:+
eStorage\e42ae3bcf7940a34339123e4c281d858\redhat.java\jdt_ws\java programming_5292cde3\bin'
Connected to server: 127.0.0.1 on port 8000
Enter message to send: hii
Received response from server: You said: hii
Enter message to send: welcome!!
Received response from server: You said: welcome!!
Enter message to send: 
```

- **UDP Client server chat application using socket programming.**

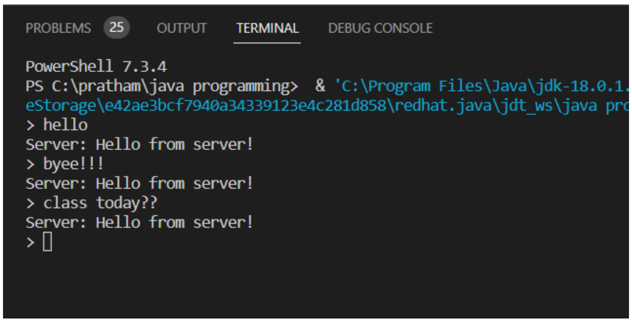
Client:

```
import java.net.*;
public class client1 {
    public static void main(String[] args) throws Exception {
        DatagramSocket socket = new DatagramSocket();
        InetAddress address = InetAddress.getByName("localhost");
        byte[] buffer;
        while (true) {
            String message = System.console().readLine("> ");
            buffer = message.getBytes();
            DatagramPacket packet = new DatagramPacket(buffer, buffer.length,
address, 4445);
            socket.send(packet);
            buffer = new byte[1024];
            packet = new DatagramPacket(buffer, buffer.length);
            socket.receive(packet);
            String response = new String(packet.getData(), 0,
packet.getLength());
            System.out.println("Server: " + response);
        }
    }
}
```

Server:

```
import java.net.*;
public class server2 {
    public static void main(String[] args) throws Exception {
        DatagramSocket socket = new DatagramSocket(4445);
        byte[] buffer;
        while (true) {
            buffer = new byte[1024];
            DatagramPacket packet = new DatagramPacket(buffer, buffer.length);
            socket.receive(packet);
            String message = new String(packet.getData(), 0,
packet.getLength());
            System.out.println("Client: " + message);
            String response = "Hello from server!";
            buffer = response.getBytes();
            packet = new DatagramPacket(buffer, buffer.length,
packet.getAddress(), packet.getPort());
            socket.send(packet);
        }
    }
}
```

Output:



PROBLEMS 25 OUTPUT TERMINAL DEBUG CONSOLE

```
PowerShell 7.3.4
PS C:\pratham\java programming>  & 'C:\Program Files\Java\jdk-18.0.1.
eStorage\e42ae3bcf7940a34339123e4c281d858\redhat.java\jdt_ws\java pro
> hello
Server: Hello from server!
> byee!!!
Server: Hello from server!
> class today??
Server: Hello from server!
> ▯
```

- **Getting the date from the server using socket programming.**

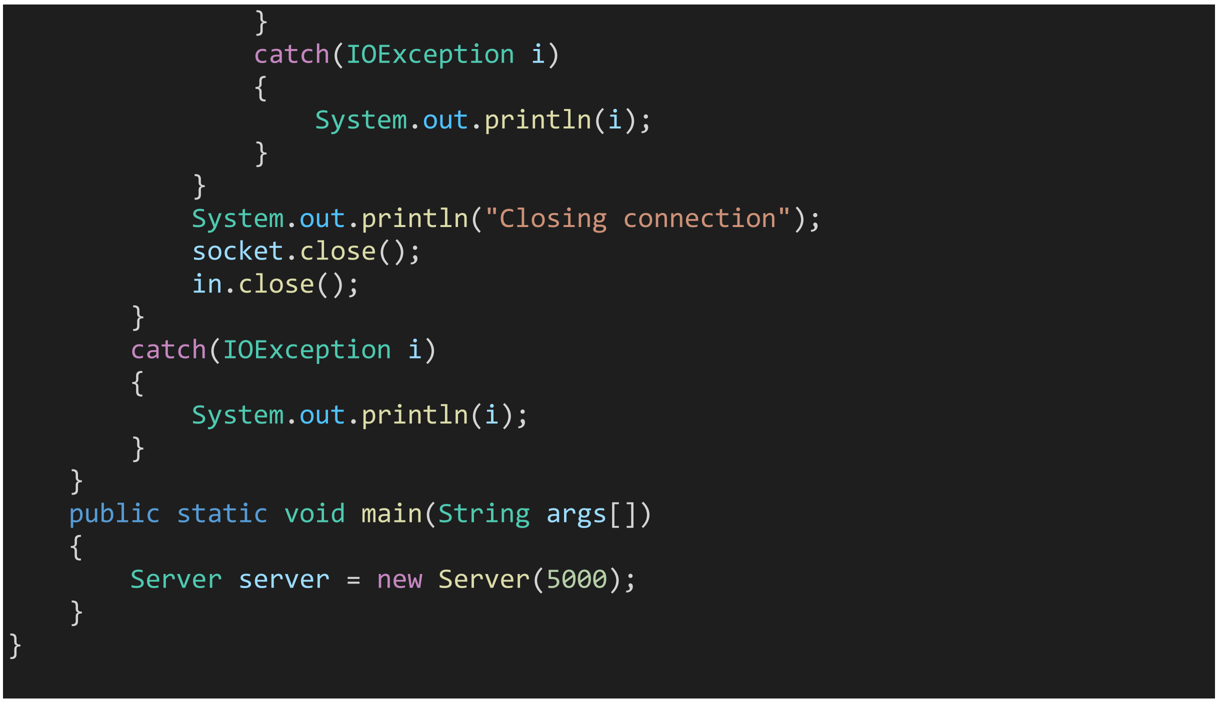
Client:

```
import java.io.*;
import java.net.*;
import java.time.LocalDateTime;
import java.time.format.DateTimeFormatter;
public class Client {
    private Socket socket = null;
    private DataInputStream input = null;
    private DataOutputStream out = null;
    public Client(String address, int port,String formattedDateTime)
    {
        try
        {
            socket = new Socket(address, port);
            System.out.println("Connected");
            System.out.println("helloo");
            System.out.println("Formatted Date and Time: " +
formattedDateTime);
            input = new DataInputStream(System.in);
            out = new DataOutputStream(
            socket.getOutputStream());
        }
        catch (UnknownHostException u)
        {
            System.out.println(u);
            return;
        }
        catch (IOException i)
        {
            System.out.println(i);
            return;
        }
        String line = "";
        while (!line.equals("Over"))
        {
            try {
                line = input.readLine();
                out.writeUTF(line);
            }
            catch (IOException i)
            {
                System.out.println(i);
            }
        }
        try {
            input.close();
            out.close();
            socket.close();
        }
        catch (IOException i) {
            System.out.println(i);
        }
    }
    public static void main(String args[])
    {
                LocalDateTime now = LocalDateTime.now();
                System.out.println("Current Date and Time: " + now);

                DateTimeFormatter formatter =
DateTimeFormatter.ofPattern("yyyy-MM-dd HH:mm:ss");
                String formattedDateTime = now.format(formatter);
        Client client = new Client("127.0.0.1", 5000,formattedDateTime);
    }
}
```

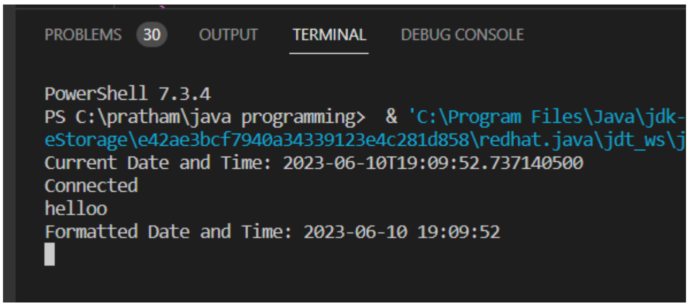
Server:

```
import java.net.*;
import java.io.*;
public class Server
{
    private Socket socket = null;
    private ServerSocket server = null;
    private DataInputStream in   = null;
    public Server(int port)
    {
        try
        {
            server = new ServerSocket(port);
            System.out.println("Server started");
            System.out.println("Waiting for a client ...");
            socket = server.accept();
            System.out.println("Client accepted");
            in = new DataInputStream(
                new BufferedInputStream(socket.getInputStream()));
            String line = "";
            while (!line.equals("Over"))
            {
                try
                {
                    line = in.readUTF();
                    System.out.println(line);
```



```
                }
                catch(IOException i)
                {
                    System.out.println(i);
                }
            }
            System.out.println("Closing connection");
            socket.close();
            in.close();
        }
        catch(IOException i)
        {
            System.out.println(i);
        }
    }
    public static void main(String args[])
    {
        Server server = new Server(5000);
    }
}
```

Output:



```
PowerShell 7.3.4
PS C:\pratham\java programming>  & 'C:\Program Files\Java\jdk-
eStorage\e42ae3bcf7940a34339123e4c281d858\redhat.java\jdt_ws\j
Current Date and Time: 2023-06-10T19:09:52.737140500
Connected
helloo
Formatted Date and Time: 2023-06-10 19:09:52
```