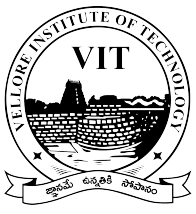
CSE3003 : Computer Networks

CN Lab 4 (Socket Programming)

Name	Gudi Varaprasad
Reg. No.	19BCE7048
Submitted to	Dr.R. NANDHA KUMAR sir
Lab Slot	L39 + L40
Department	School of Computer Science and Engineering
Email	varaprasad.19bce7048@vitap.ac.in
Date	17th February 2021



VIT®
AP

Question :

Implement a Simple chat application using DATAGRAM SOCKET / User Datagram Protocol (UDP)

UDPServer.java

```
1  // A Java program for a Server Side
2
3  import java.io.*;
4  import java.util.*;
5  import java.lang.*;
6  import java.net.*;
7
8  class UDPServer {
9      public static void main(String args[]) throws Exception {
10         // creates a datagram socket and binds it with the
               available Port Number on the localhost machine.
11         DatagramSocket serverSocket = new
               DatagramSocket(59768);
12         byte[] receivedData = new byte[1024]; // max is 1024
               charset
13         byte[] sendData = new byte[1024];
14
15         while (true) {
16             // This constructor is used to receive the packets
17             DatagramPacket receivedPacket = new
                   DatagramPacket(receivedData,
```

```
            receivedData.length);
18          serverSocket.receive(receivedPacket); // received
19
20          // convert bytes into data
21          String fromClient = new
              String(receivedPacket.getData());
22          System.out.println();
23          System.out.println("RECEIVED FROM CLIENT : \n" +
              fromClient);
24
25          InetAddress ip = receivedPacket.getAddress();
26          int port = receivedPacket.getPort();
27          // converts the sentence from client to capital
              letters
28          String capitalClient = fromClient.toUpperCase();
29          // convert it into bytes
30          sendData = capitalClient.getBytes();
31
32          // creates a datagram packet. This constructor is
              used to send the packets
33          DatagramPacket sendPacket = new
              DatagramPacket(sendData, sendData.length, ip,
              port);
34          serverSocket.send(sendPacket); // send
35      }
36  }
```

```
}

// IP address is : 192.168.100.9
// Host name : GVP
// Port used is : 59768
```

UDPClient.java

```
// A Java program for a Client Side

import java.io.*;
import java.util.*;
import java.lang.*;
import java.net.*;

class UDPClient {
    public static void main(String args[]) throws Exception {
        // InputStreamReader reads bytes and decodes them
           into characters using a specified charset
        // BufferedReader reads the stream of characters
           from the specified source
        BufferedReader userInput = new BufferedReader(new
           InputStreamReader(System.in));

        // creates a datagram socket and binds it with the
           available Port Number on the localhost machine.
        DatagramSocket clientSocket = new DatagramSocket();
```

```
16
17          // get the IP address using the host name of the
                system
18          InetAddress ip = InetAddress.getByName("GVP");
19
20          byte[] sendData = new byte[1024]; // maximum charset
                is 1024
21          byte[] receivedData = new byte[1024];
22          String sentence = userInput.readLine(); // user input
23
24          sendData = sentence.getBytes(); // convert the input
                to bytes
25
26          // creates a datagram packet. This constructor is
                used to send the packets
27          DatagramPacket sendPacket = new
                DatagramPacket(sendData, sendData.length, ip,
                59768);
28          clientSocket.send(sendPacket); // send the packet to
                server
29
30          // This constructor is used to receive the packets
31          DatagramPacket receivedPacket = new
                DatagramPacket(receivedData, receivedData.length);
32          clientSocket.receive(receivedPacket); // receive the
                packet from server
```

```
33
34          // convert bytes into data
35          String fromServer = new
               String(receivedPacket.getData());
36          System.out.println();
37          System.out.println("FROM SERVER : \n" + fromServer);
38          // close the connection if received
39          clientSocket.close();
40      }
41  }
42
43  // IP address is : 192.168.100.9
44  // Host name : GVP
45  // Port used is : 59768
```

Output for above program :