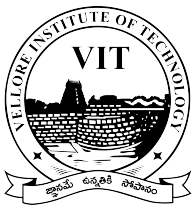
CSE3003 : Computer Networks

CN Lab 3 (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	10th February 2021



VIT®
AP

Question :

Implement a Simple Chat Application using STREAM SOCKET (TCP/IP)

ServerSide.java

```java
// A Java program for a Serverside

import java.net.*;
import java.io.*;
public class ServerSide {
    //initialize socket and input stream
    private Socket socket = null;
    private ServerSocket server = null;
    private DataInputStream in = null;

    // constructor with port
    public ServerSide(int port) {
        // starts server and waits for a connection
        try {
            // creates new server using port number
            server = new ServerSocket(port);
            System.out.println("Server has started !! ");
            System.out.println("Waiting for a client
               response...");

            // accept client request
            socket = server.accept();
```

```
22              System.out.println("Client request accepted");
23
24              // takes input from the client socket
25              in = new DataInputStream(
26                  new
                       BufferedInputStream(socket.getInputStream()));
27              String line = "";
28
29              // reads message from client until "END" is sent
30              while (!line.equals("END")) {
31                  try {
32                      // convert the string into UTF format
33                      line = in .readUTF();
34                      System.out.println(line);
35                  } catch (IOException i) {
36                      System.out.println(i);
37                  }
38              }
39              System.out.println("Closing the connection !! ");
40
41              // close connection
42              socket.close(); in .close();
43          } catch (IOException i) {
44              // print if any exception occured
45              System.out.println(i);
46          }
```

```
47      }
48
49      public static void main(String args[]) {
50          // contructor initialised with port number as
                argument
51          ServerSide server = new ServerSide(59768);
52      }
53  }
54
55  // IP address is : 192.168.100.9
56  // Port used is : 59768
```

ClientSide.java

```
1   // A Java program for a ClientSide
2
3   import java.net.*;
4   import java.io.*;
5   public class ClientSide {
6       // initialize socket and input output streams
7
8       private Socket socket = null;
9       private DataInputStream input = null;
10      private DataOutputStream out = null;
11
12      // constructor to put ip address and port
13      public ClientSide(String address, int port) {
```

```
14          // establish a connection
15          try {
16              socket = new Socket(address, port);
17              System.out.println("Connected to Server.");
18              System.out.println("Type your message.... Type
                    END to stop.");
19              // takes input from terminal
20              input = new DataInputStream(System.in);
21
22              // sends output to the socket
23              out = new
                    DataOutputStream(socket.getOutputStream());
24
25          } catch (UnknownHostException u) {
26              // print if any exceptions there
27              System.out.println(u);
28          } catch (IOException i) {
29              System.out.println(i);
30          }
31          // string to read message from input
32          String line = "";
33
34          // keep reading until "END" is input
35          while (!line.equals("END")) {
36              try {
37                  // read the input
```

```
38                  line = input.readLine();
39                  // convert the input to UTF format
40                  out.writeUTF(line);
41              } catch (IOException i) {
42                  System.out.println(i);
43              }
44          }
45          // close the connection
46          try {
47              input.close();
48              out.close();
49              socket.close();
50          } catch (IOException i) {
51              System.out.println(i);
52          }
53      }
54      public static void main(String args[]) {
55          ClientSide client = new ClientSide("192.168.100.9",
               59768);
56      }
57  }
58  // IP address is : 192.168.100.9
59  // Port used is : 59768
```

Output for above program :