Faculty of Information Technology
H A N O I U N I V E R S I T Y

CNE – Computer Networks

Java Socket

1. Exercise 1: TCP socket

```
import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.IOException;
import java.io.InputStreamReader;
import java.io.OutputStreamWriter;
import java.net.ServerSocket;
import java.net.Socket;

public class SimpleServerProgram {

   public static void main(String args[]) {

       ServerSocket listener = null;
       String line;
       BufferedReader is;
       BufferedWriter os;
       Socket socketOfServer = null;



       try {
           listener = new ServerSocket(9999);
       } catch (IOException e) {
           System.out.println(e);
           System.exit(1);
       }

       try {
           System.out.println("Server is waiting to accept user...");



           socketOfServer = listener.accept();
           System.out.println("Accept a client!");


           is = new BufferedReader(new
InputStreamReader(socketOfServer.getInputStream()));
           os = new BufferedWriter(new
OutputStreamWriter(socketOfServer.getOutputStream()));


           while (true) {
```

```
                line = is.readLine();

                os.write(">> " + line);
                os.newLine();
                os.flush();

                if (line.equals("QUIT")) {
                    os.write(">> OK");
                    os.newLine();
                    os.flush();
                    break;
                }
            }

        } catch (IOException e) {
            System.out.println(e);
            e.printStackTrace();
        }
        System.out.println("Sever stopped!");
    }
    }

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

public class SimpleClientDemo {

   public static void main(String[] args) {

        final String serverHost = "localhost";

        Socket socketOfClient = null;
        BufferedWriter os = null;
        BufferedReader is = null;

        try {
            socketOfClient = new Socket(serverHost, 9999);

            os = new BufferedWriter(new
OutputStreamWriter(socketOfClient.getOutputStream()));

            is = new BufferedReader(new
InputStreamReader(socketOfClient.getInputStream()));

        } catch (UnknownHostException e) {
            System.err.println("Don't know about host " + serverHost);
            return;
        } catch (IOException e) {
            System.err.println("Couldn't get I/O for the connection to " +
serverHost);
            return;
        }

        try {
            os.write("HELO");
            os.newLine();
            os.flush();
            os.write("I am Tom Cat");
            os.newLine();
```

```
            os.flush();
            os.write("QUIT");
            os.newLine();
            os.flush();

            String responseLine;
            while ((responseLine = is.readLine()) != null) {
                System.out.println("Server: " + responseLine);
                if (responseLine.indexOf("OK") != -1) {
                    break;
                }
            }

            os.close();
            is.close();
            socketOfClient.close();
        } catch (UnknownHostException e) {
            System.err.println("Trying to connect to unknown host: " + e);
        } catch (IOException e) {
            System.err.println("IOException:  " + e);
        }
    }

}
```

2. Exercise 2: UDP socket

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

class UDPEchoServer {
	public static void main(String args[]) throws Exception {
		int port = 9876;
		DatagramSocket serverSocket = new DatagramSocket(port);

		byte[] receiveData = new byte[1024];
		byte[] sendData     = new byte[1024];

		while(true) {
			DatagramPacket receivePacket = new DatagramPacket
(receiveData, receiveData.length);
			serverSocket.receive(receivePacket);
			String sentence = new String(receivePacket.getData());
			InetAddress IPAddress = receivePacket.getAddress();
			int clientPort = receivePacket.getPort();
			String capitalizedSentence = sentence.toUpperCase();
			sendData = capitalizedSentence.getBytes();
			DatagramPacket sendPacket = new DatagramPacket  (sendData,
sendData.length, IPAddress, clientPort);
			serverSocket.send(sendPacket);

		}
	}
}

import java.io.*;
```

```java
import java.net.*;

public class UDPClient {
	public static void main(String args[]) throws Exception {
		BufferedReader inFromUser = new BufferedReader(new InputStreamReader
(System.in));
		int port = 9876;
		DatagramSocket clientSocket = new DatagramSocket();
		InetAddress IPAddress = InetAddress.getByName("localhost");

		byte[] sendData = new byte[1024];
		byte[] receiveData = new byte[1024];

		while(true) {
			System.out.println("Please enter your message");
			String sentence = inFromUser.readLine();
			sendData = sentence.getBytes();
			DatagramPacket sendPacket = new DatagramPacket (sendData,
sendData.length, IPAddress, port);
			clientSocket.send(sendPacket);

			DatagramPacket receivePacket = new DatagramPacket
(receiveData, receiveData.length);
			clientSocket.receive(receivePacket);
			String modifiedSentence = new String(receivePacket.getData());
			System.out.println("FROM SERVER:" + modifiedSentence);

			//clientSocket.close();
		}

	}
}
```