**Lab Practical #06:**

Study Client-Server Socket programming - TCP & UDP

**Practical Assignment #06:**

1. **Write a C/Java code for TCP Server-Client Socket Programming.**
2. **Write a C/Java code for UDP Server-Client Socket Programming.**
3. **For TCP Server-Client:**

**TCP Server Program:**

import java.io.\*;

import java.net.\*;

public class TCPServer {

public static void main(String[] args) {

int port = 8080;

try (ServerSocket serverSocket = new ServerSocket(port)) {

System.out.println("TCP Server is waiting for connection on port " + port + "...");

// Accept client connection

Socket socket = serverSocket.accept();

System.out.println("Client connected: " + socket.getInetAddress());

// Input stream (from client)

BufferedReader in = new BufferedReader(new InputStreamReader(socket.getInputStream()));

// Output stream (to client)

PrintWriter out = new PrintWriter(socket.getOutputStream(), true);

// Read message from client

String clientMessage = in.readLine();

System.out.println("Client: " + clientMessage);

// Send reply to client

out.println("Hello from TCP Server");

socket.close();

} catch (IOException e) {

e.printStackTrace();

}

}

}

**TCP Client Program:**

import java.io.\*;

import java.net.\*;

public class TCPClient {

public static void main(String[] args) {

String serverAddress = "127.0.0.1"; // localhost

int port = 8080;

try (Socket socket = new Socket(serverAddress, port)) {

// Output stream (to server)

PrintWriter out = new PrintWriter(socket.getOutputStream(), true);

// Input stream (from server)

BufferedReader in = new BufferedReader(new InputStreamReader(socket.getInputStream()));

// Send message to server

out.println("Hello from TCP Client");

// Read reply from server

String serverMessage = in.readLine();

System.out.println("Server: " + serverMessage);

} catch (IOException e) {

e.printStackTrace();

}

}

}

1. **For UDP Server-Client:**

**UDP Server Program:**

import java.net.\*;

public class UDPServer {

public static void main(String[] args) {

int port = 8080;

try (DatagramSocket socket = new DatagramSocket(port)) {

byte[] receiveBuffer = new byte[1024];

byte[] sendBuffer;

System.out.println("UDP Server is waiting for message on port " + port + "...");

// Receive data from client

DatagramPacket receivePacket = new DatagramPacket(receiveBuffer, receiveBuffer.length);

socket.receive(receivePacket);

String clientMessage = new String(receivePacket.getData(), 0, receivePacket.getLength());

System.out.println("Client: " + clientMessage);

// Send reply to client

String reply = "Hello from UDP Server";

sendBuffer = reply.getBytes();

InetAddress clientAddress = receivePacket.getAddress();

int clientPort = receivePacket.getPort();

DatagramPacket sendPacket = new DatagramPacket(sendBuffer, sendBuffer.length, clientAddress, clientPort);

socket.send(sendPacket);

} catch (Exception e) {

e.printStackTrace();

}

}

}

**UDP Client Program:**

import java.net.\*;

public class UDPClient {

public static void main(String[] args) {

String serverAddress = "127.0.0.1"; // localhost

int port = 8080;

try (DatagramSocket socket = new DatagramSocket()) {

byte[] sendBuffer;

byte[] receiveBuffer = new byte[1024];

// Send message to server

String message = "Hello from UDP Client";

sendBuffer = message.getBytes();

InetAddress serverIP = InetAddress.getByName(serverAddress);

DatagramPacket sendPacket = new DatagramPacket(sendBuffer, sendBuffer.length, serverIP, port);

socket.send(sendPacket);

System.out.println("Message sent to server.");

// Receive reply from server

DatagramPacket receivePacket = new DatagramPacket(receiveBuffer, receiveBuffer.length);

socket.receive(receivePacket);

String serverMessage = new String(receivePacket.getData(), 0, receivePacket.getLength());

System.out.println("Server: " + serverMessage);

} catch (Exception e) {

e.printStackTrace();

}

}

}