



Date: 08/07/2025

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.**

1. For TCP Server-Client:

TCP Server Program:

```
1 import java.io.*;
2 import java.net.*;
3
4 public class Server {
5     public static void main(String args[]) throws IOException
6     {
7         // create a server socket on port number 9090
8         ServerSocket serverSocket = new ServerSocket(9090);
9         System.out.println("Server is running and waiting for client connection...");
10
11        // Accept incoming client connection
12        Socket clientSocket = serverSocket.accept();
13        System.out.println("Client connected!");
14
15        // Setup input and output streams for communication with the client
16        BufferedReader in = new BufferedReader(new
17           InputStreamReader(clientSocket.getInputStream()));
18        PrintWriter out = new PrintWriter(clientSocket.getOutputStream(), true);
19
20        // Read message from client
21        String message = in.readLine();
22        System.out.println("Client says: " + message);
23
24        // Send response to the client
25        out.println("Message received by the server.");
26
27        // Close the client socket
28        clientSocket.close();
29        // Close the server socket
30        serverSocket.close();
31    }
32 }
```



Date: 08/07/2025

TCP Client Program:

```
1 import java.io.*;
2 import java.net.*;
3
4 public class Client {
5     public static void main(String args[]) throws IOException
6     {
7         // create a socket to connect to the server running on localhost at port
8         // number 9090
9         Socket socket = new Socket("localhost", 9090);
10
11         // Setup output stream to send data to the server
12         PrintWriter out = new PrintWriter(socket.getOutputStream(), true);
13
14         // Setup input stream to receive data from the server
15         BufferedReader in = new BufferedReader(new
16             InputStreamReader(socket.getInputStream()));
17
18         // Send message to the server
19         out.println("Hello from client!");
20
21         // Receive response from the server
22         String response = in.readLine();
23         System.out.println("Server says: " + response);
24
25         // Close the socket
26         socket.close();
27     }
28 }
```



Date: 08/07/2025

2. For UDP Server-Client:

UDP Server Program:

```
1 import java.io.IOException;
2 import java.net.DatagramPacket;
3 import java.net DatagramSocket;
4
5 public class udpBaseServer_2
6 {
7     public static void main(String[] args) throws IOException
8     {
9         // Step 1 : Create a socket to listen at port 1234
10        DatagramSocket ds = new DatagramSocket(1234);
11        byte[] receive = new byte[65535];
12
13        DatagramPacket DpReceive = null;
14        while (true)
15        {
16
17            // Step 2 : create a DatagramPacket to receive the data.
18            DpReceive = new DatagramPacket(receive, receive.length);
19
20            // Step 3 : review the data in byte buffer.
21            ds.receive(DpReceive);
22
23            System.out.println("Client:-" + data(receive));
24
25            // Exit the server if the client sends "bye"
26            if (data(receive).toString().equals("bye"))
27            {
28                System.out.println("Client sent bye.....EXITING");
29                break;
30            }
31
32            // Clear the buffer after every message.
33            receive = new byte[65535];
34        }
35    }
36
37    // A utility method to convert the byte array
38    // data into a string representation.
39    public static StringBuilder data(byte[] a)
40    {
41        if (a == null)
42            return null;
43        StringBuilder ret = new StringBuilder();
44        int i = 0;
45        while (a[i] != 0)
46        {
47            ret.append((char) a[i]);
48            i++;
49        }
50        return ret;
51    }
52 }
```



Date: 08/07/2025

UDP Client Program:

```
1 import java.io.IOException;
2 import java.net.DatagramPacket;
3 import java.net.DatagramSocket;
4 import java.net.InetAddress;
5 import java.util.Scanner;
6
7 public class udpBaseClient_2
8 {
9     public static void main(String args[]) throws IOException
10    {
11        Scanner sc = new Scanner(System.in);
12
13        // Step 1: Create the socket object for
14        // carrying the data.
15        DatagramSocket ds = new DatagramSocket();
16
17        InetAddress ip = InetAddress.getLocalHost();
18        byte buf[] = null;
19
20        // loop while user not enters "bye"
21        while (true)
22        {
23            String inp = sc.nextLine();
24
25            // convert the String input into the byte array.
26            buf = inp.getBytes();
27
28            // Step 2 : Create the datagramPacket for sending
29            // the data.
30            DatagramPacket DpSend =
31                new DatagramPacket(buf, buf.length, ip, 1234);
32
33            // Step 3 : invoke the send call to actually send
34            // the data.
35            ds.send(DpSend);
36
37            // break the loop if user enters "bye"
38            if (inp.equals("bye"))
39                break;
40        }
41    }
42 }
```