



Date: 14/08/2024

Lab Practical #07:

Study Client-Server Socket programming - TCP & UDP

Practical Assignment #07:

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:

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

class Server {

    public static void main(String args[])
        throws Exception
    {

        // Create server Socket
        ServerSocket ss = new ServerSocket(888);

        // connect it to client socket
        Socket s = ss.accept();

        System.out.println("Connection established");

        // to send data to the client
        PrintStream ps = new PrintStream(s.getOutputStream());

        // to read data coming from the client
        BufferedReader br =
            new BufferedReader( new InputStreamReader( s.getInputStream()));

        // to read data from the keyboard
        BufferedReader kb
            = new BufferedReader( new InputStreamReader(System.in));

        // server executes continuously
        while (true) {

            String str, str1;

            // repeat as long as the client
```

Date: 14/08/2024

```
// does not send a null string
// read from client
while ((str = br.readLine()) != null) {
    System.out.println(str);
    str1 = kb.readLine();
    // send to client
    ps.println("server :"+str1);
}
// close connection
ps.close();
br.close();
kb.close();
ss.close();
s.close();
// terminate application
System.exit(0);
} // end of while
}
}
```

TCP Client Program:

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

class Client {

    public static void main(String args[])
        throws Exception
    {
        // Create client socket
        Socket s = new Socket("localhost", 888);
        // to send data to the server
        DataOutputStream dos = new DataOutputStream(s.getOutputStream());
        // to read data coming from the server
```

Date: 14/08/2024

```
BufferedReader br =
new BufferedReader(new InputStreamReader(s.getInputStream()));

// to read data from the keyboard
BufferedReader kb =
new BufferedReader( new InputStreamReader(System.in));
String str, str1;
// repeat as long as exit
// is not typed at client
while (!(str = kb.readLine()).equals("exit")) {
    // send to the server
    dos.writeBytes("client :"+str + "\n");
    // receive from the server
    str1 = br.readLine();
    System.out.println(str1);
}

// close connection.
dos.close();
br.close();
kb.close();
s.close();
}
}
```

```
D:\01 darshan\B.Tech semester 5\computer network\Socket>java server.java
Connection established
client :hii
hello
client :how are you???
i am fine how can i help you.
client :what is 2+2??
ans of 2+2 is 4
client :thank you exit
welcome

D:\01 darshan\B.Tech semester 5\computer network\Socket>
```

```
D:\01 darshan\B.Tech semester 5\computer network\Socket>java client.java
hii
server :hello
how are you???
server :i am fine how can i help you.
what is 2+2??
server :ans of 2+2 is 4
thank you exit
server :welcome
exit

D:\01 darshan\B.Tech semester 5\computer network\Socket>
```

2. For UDP Server-Client:

UDP Server Program:

```
import java.io.IOException;
import java.net.DatagramPacket;
import java.net.DatagramSocket;
import java.net.InetAddress;
import java.net.SocketException;

public class udpBaseServer_2
{
    public static void main(String[] args) throws IOException
    {
        // Step 1 : Create a socket to listen at port 1234
        DatagramSocket ds = new DatagramSocket(1234);
        byte[] receive = new byte[65535];

        DatagramPacket DpReceive = null;
        while (true)
        {
            // Step 2 : create a DatagramPacket to receive the data.
            DpReceive = new DatagramPacket(receive, receive.length);

            // Step 3 : receive the data in byte buffer.
            ds.receive(DpReceive);

            System.out.println("Client:-" + data(receive));

            // Exit the server if the client sends "bye"
            if (data(receive).toString().equals("bye"))
            {
                System.out.println("Client sent bye.....EXITING");
                break;
            }

            // Clear the buffer after every message.
            receive = new byte[65535];
        }
    }

    // A utility method to convert the byte array
    // data into a string representation.
    public static StringBuilder data(byte[] a)
```

Date: 14/08/2024

```
{
    if (a == null)
        return null;
    StringBuilder ret = new StringBuilder();
    int i = 0;
    while (a[i] != 0)
    {
        ret.append((char) a[i]);
        i++;
    }
    return ret;
}
```

UDP Client Program:

```
import java.io.IOException;
import java.net.DatagramPacket;
import java.net.DatagramSocket;
import java.net.InetAddress;
import java.util.Scanner;

public class udpBaseClient_2
{
    public static void main(String args[]) throws IOException
    {
        Scanner sc = new Scanner(System.in);

        // Step 1: Create the socket object for carrying the data.
        DatagramSocket ds = new DatagramSocket();

        InetAddress ip = InetAddress.getLocalHost();
        byte buf[] = null;

        // loop while user not enters "bye"
        while (true)
        {
            String inp = sc.nextLine();

            // convert the String input into the byte array.
            buf = inp.getBytes();
        }
    }
}
```



Date: 14/08/2024

```
// Step 2 : Create the datagramPacket for sending the data.
DatagramPacket DpSend =
    new DatagramPacket(buf, buf.length, ip, 1234);

// Step 3 : invoke the send call to actually send the data.
ds.send(DpSend);

// break the loop if user enters "bye"
if (inp.equals("bye"))
    break;
    }
    }
}
```

```
D:\01 darshan\B.Tech semester 5\computer network\Socket>java UDPServer.java
Client:-hello
Client:-hihi
Client:-how are you??
Client:-exit
Client:-bye
Client sent bye.....EXITING

D:\01 darshan\B.Tech semester 5\computer network\Socket>
```

```
D:\01 darshan\B.Tech semester 5\computer network\Socket>java UDPClient.java
hello
hihi
how are you??
exit
bye

D:\01 darshan\B.Tech semester 5\computer network\Socket>
```