**Date: 17/ 08/2025**

## 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:

```
// Demonstrating Server-side Programming
import java.net.*;
import java.io.*;
public class Server {
  // Initialize socket and input stream
  private Socket s = null;
  private ServerSocket ss = null;
  private DataInputStream in = null;
  // Constructor with port
  public Server(int port) {
    // Starts server and waits for a connection
    try
    {
      ss = new ServerSocket(port);
      System.out.println("Server started");

      System.out.println("Waiting for a client ...");

      s = ss.accept();
      System.out.println("Client accepted");

      // Takes input from the client socket
      in = new DataInputStream(
        new BufferedInputStream(s.getInputStream()));
```

Date: 17/ 08/2025

```
            String m = "";
            // Reads message from client until "Over" is sent
            while (!m.equals("Over"))
            {
                try
                {
                    m = in.readUTF();
                    System.out.println(m);

                }
                catch(IOException i)
                {
                    System.out.println(i);
                }
            }
            System.out.println("Closing connection");

            // Close connection
            s.close();
            in.close();
        }
        catch(IOException i)
        {
            System.out.println(i);
        }
    }
    public static void main(String args[])
    {
        Server s = new Server(5000);
    }
}
```

**Date: 17/ 08/2025**

## TCP Client Program:

```
// Demonstrating Client-side Programming
import java.io.*;
import java.net.*;

public class Client {

    // Initialize socket and input/output streams
    private Socket s = null;
    private DataInputStream in = null;
    private DataOutputStream out = null;

    // Constructor to put IP address and port
    public Client(String addr, int port)
    {
        // Establish a connection
        try {
            s = new Socket(addr, port);
            System.out.println("Connected");

            // Takes input from terminal
            in = new DataInputStream(System.in);

            // Sends output to the socket
            out = new DataOutputStream(s.getOutputStream());
        }
        catch (UnknownHostException u) {
            System.out.println(u);
            return;
        }
        catch (IOException i) {
            System.out.println(i);
            return;
        }

        // String to read message from input
        String m = "";

        // Keep reading until "Over" is input
        while (!m.equals("Over")) {
            try {
                m = in.readLine();
                out.writeUTF(m);
            }
            catch (IOException i) {
                System.out.println(i);
            }
        }

        // Close the connection
        try {
            in.close();
            out.close();
            s.close();
        }
        catch (IOException i) {
            System.out.println(i);
        }
    }
    public static void main(String[] args) {
        Client c = new Client("127.0.0.1", 5000);
    }
}
```

Date: 17/ 08/2025

## 2. For UDP Server-Client:

### UDP Server Program:

```
import java.net.*;

public class Server {
    public static void main(String[] args) {
        DatagramSocket socket = null;
        try {
            socket = new DatagramSocket(5000);
            System.out.println("UDP Server started. Waiting for messages...");

            byte[] buffer = new byte[1024];
            DatagramPacket packet = new DatagramPacket(buffer, buffer.length);

            String receivedMessage = "";
            while (!receivedMessage.equals("Over")) {
                socket.receive(packet); // Receive packet
                receivedMessage = new String(packet.getData(), 0, packet.getLength());
                System.out.println("Client: " + receivedMessage);
            }

            System.out.println("Connection ended by client.");
        } catch (Exception e) {
            System.out.println("Error: " + e);
        } finally {
            if (socket != null) socket.close();
        }
    }
}
```

**Date: 17/ 08/2025**

## UDP Client Program:

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

public class Client {
   public static void main(String[] args) {
      DatagramSocket socket = null;
      BufferedReader reader = null;

      try {
         socket = new DatagramSocket();
         InetAddress ip = InetAddress.getByName("127.0.0.1");
         reader = new BufferedReader(new InputStreamReader(System.in));

         String message = "";
         while (!message.equals("Over")) {
            System.out.print("Enter message: ");
            message = reader.readLine();

            byte[] buffer = message.getBytes();
            DatagramPacket packet = new DatagramPacket(buffer, buffer.length, ip, 5000);
            socket.send(packet);
         }
      } catch (Exception e) {
         System.out.println("Error: " + e);
      } finally {
         if (socket != null) socket.close();
         try {
            if (reader != null) reader.close();
         } catch (IOException e) {
            System.out.println(e);
```

Date: 17/ 08/2025

```
            }
        }
    }
}
```