

Client -

```
package act4_1;

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

public class Client {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        try {
            // Crée une nouvelle socket UDP pour le client
            DatagramSocket clientSocket = new DatagramSocket();
            // Adresse du serveur
            InetAddress serverAddress = InetAddress.getByName("localhost");
            // Ici, "localhost" signifie que le serveur est sur la même machine
            int serverPort = 1235;
            Scanner scanner = new Scanner(System.in);
            System.out.println("Donnez votre prénom :\n");
            String message = scanner.nextLine();
            byte[] sendData = message.getBytes();
            // Convertit la saisie de l'utilisateur en tableau de bytes
            // Crée un paquet contenant les données à envoyer, l'adresse du serveur et le port
            DatagramPacket sendPacket = new DatagramPacket(sendData, sendData.length, serverAddress, serverPort);
            // Envoie le paquet au serveur
            clientSocket.send(sendPacket);
            // Réception de la réponse du serveur
            byte[] receiveData = new byte[1024];
            DatagramPacket receivePacket = new DatagramPacket(receiveData, receiveData.length);

            clientSocket.receive(receivePacket);
            // Attends la réponse du serveur
            // Convertit la réponse en une chaîne de caractères
            String serverResponse = new String(receivePacket.getData(), 0, receivePacket.getLength());
            // Affiche la réponse du serveur
            System.out.println(serverResponse);

        } catch (SocketException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        } catch (UnknownHostException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        } catch (IOException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }

    }
}
```

Server -

```
package act4_1;

import java.net.DatagramSocket;
import java.io.IOException;
import java.net.DatagramPacket;
import java.net.InetAddress;
import java.net.SocketException;
import java.text.SimpleDateFormat;
import java.util.Date;

public class Server {

    public static void main(String[] args) {
        // TODO Auto-generated method stub

        try {
            int serverPort = 1235;
            DatagramSocket serverSocket = new DatagramSocket(serverPort);
            // Crée une socket pour le serveur

            System.out.println("Le serveur est en attente de connexions...");

            while (true) {
                byte[] receiveData = new byte[1024];
                // Crée un tableau de bytes pour stocker les données reçues
                DatagramPacket receivePacket = new DatagramPacket(receiveData, receiveData.length);
                // Crée un paquet pour recevoir des données
                serverSocket.receive(receivePacket);
                // Attend la réception de données depuis un client
                String clientMessage = new String(receivePacket.getData(), 0, receivePacket.getLength());
                // Convertit les données reçues en une chaîne de caractères
                InetAddress clientAddress = receivePacket.getAddress();
                int clientPort = receivePacket.getPort();
                // Obtient le port du client et l'adresse IP du client

                // Réponse au client
                String welcomeMessage = "Bienvenu " + clientMessage;
                // Crée un message de réponse
                byte[] sendData = welcomeMessage.getBytes();
                // Convertit le message de réponse en tableau de bytes
                DatagramPacket sendPacket = new DatagramPacket(sendData, sendData.length, clientAddress, clientPort);
                // Crée un paquet pour envoyer la réponse au client
                serverSocket.send(sendPacket);
                // Envoie la réponse au client
            }

        } catch (IOException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }

    }
}
```