



“proyecto ”

(JESUS NICOMEDES DE LA CRUZ SANCHEZ)

(ING. SISTEMAS COMPUTACIONALES Y DISEÑO EN SOFTWARE), Instituto
Universitario de Yucatán

(2303040748): (REDES)

(PERLA ALEJANDRA LANDERO HEREDIA)

(19/10/2025)

```
Start Page x Cliente.java x servidor.java x
Source History
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/licenses/license-default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Main.java to edit this template
 */
package cliente;
import java.net.DatagramPacket;
import java.net.DatagramSocket;
import java.net.InetAddress;
import java.nio.charset.StandardCharsets;
/**
 *
 * @author Jesus
 */
public class servidor {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // TODO code application logic here
        final int PORT = 9876;
        final int BUFFER_SIZE = 1024;

        try (DatagramSocket serverSocket = new DatagramSocket(port: PORT)) {
            System.out.println("Servidor UDP iniciado en el puerto " + PORT + ". Esperando paquetes...");

            byte[] receiveBuffer = new byte[BUFFER_SIZE];

            while (true) {
                DatagramPacket receivePacket = new DatagramPacket(buf: receiveBuffer, length: receiveBuffer.length);
                serverSocket.receive(p: receivePacket); // espera paquete

                // Leer mensaje entrante
                String message = new String(bytes: receivePacket.getData(), offset: 0, length: receivePacket.getLength(), charset: StandardCharsets.UTF_8);
                InetAddress clientAddress = receivePacket.getAddress();
                int clientPort = receivePacket.getPort();
            }
        }
    }
}
```

```
Start Page x Cliente.java x servidor.java x
Source History
4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40
/*
package cliente;
import java.net.DatagramPacket;
import java.net.DatagramSocket;
import java.net.InetAddress;
import java.nio.charset.StandardCharsets;
import java.util.Scanner;
/**
 *
 * @author Jesus
 */
public class Cliente {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        // TODO code application logic here
        final String SERVER_HOST = "localhost";
        final int SERVER_PORT = 9876;
        final int BUFFER_SIZE = 1024;

        try (DatagramSocket socket = new DatagramSocket();
            Scanner scanner = new Scanner(source: System.in, charset: StandardCharsets.UTF_8)) {

            InetAddress serverAddress = InetAddress.getByName(host: SERVER_HOST);
            System.out.println("Cliente UDP. Escribe mensajes para enviar al servidor (escribe 'salir' para terminar).");

            while (true) {
                System.out.print(s: "> ");
                String line = scanner.nextLine();
                if (line == null) break;
                if ("salir".equalsIgnoreCase(anotherString: line.trim())) {
                    System.out.println(s: "Saliendo...");
                    break;
                }
            }
        }
    }
}
```

Ciente (run)

run:

Servidor UDP iniciado en el puerto 9876. Esperando paquetes...

salir

paquete|

The screenshot shows an IDE interface. At the top, a file explorer displays a project structure with 'Test Packages' and 'Libraries'. Below it, a code editor shows the following Java code:

```
14  L  */
15  public class Cliente {
```

Below the code editor, an 'Output' window is open, showing the execution of the 'Cliente' class. The output window has three tabs: 'Cliente (run)', 'Cliente (run) #3', and 'Cliente (run) #5'. The 'Cliente (run)' tab is active, displaying the following text:

```
run:
Cliente UDP. Escribe mensajes para enviar al servidor (escribe 'salir' para terminar).
> salir
Saliendo...
BUILD SUCCESSFUL (total time: 8 seconds)
```