



Proyecto

Jorge Alberto Garduza Palacios

LICENCIATURA EN INGENIERIA EN SISTEMAS COMPUTACIONALES Y DISEÑO
DE SOFTWARE, Instituto Universitario de Yucatán

2303040781: REDES

Perla Judith Landero Heredia

19 de octubre del 2025

```
ClienteUDP.java X ServidorUDP.java
C:\Users\jorge> OneDrive > Documentos > NetBeansProjects > ServidorUDP > src > servidorudp > ClienteUDP.java
12 public class ClienteUDP {
13     public static void main(String[] args) throws InterruptedException {
14         final int PUERTO_SERVIDOR = 6000;
15
16         try {
17             // IP local para el cliente (puede ser 0.0.0.0 o la IP específica de la interfaz)
18             InetAddress direccionLocalCliente = InetAddress.getByName("0.0.0.0");
19
20             // Creamos el socket cliente en un puerto específico
21             DatagramSocket socketCliente = new DatagramSocket(0, direccionLocalCliente); // puerto aleatorio
22
23             // Nos "conectamos" al servidor (UDP no establece conexión real, pero fija IP/puerto destino)
24             InetAddress direccionServidor = InetAddress.getByName(HOST_SERVIDOR);
25             socketCliente.connect(direccionServidor, PUERTO_SERVIDOR);
26
27             // Mensaje a enviar
28             String mensaje = "Hola desde el cliente UDP";
29             byte[] bufferEnvio = mensaje.getBytes();
30
31             // Creamos y enviamos el paquete
32             DatagramPacket paqueteEnvio = new DatagramPacket(bufferEnvio, bufferEnvio.length);
33             socketCliente.send(paqueteEnvio);
34
35             // Recibir respuesta
36             byte[] bufferRecepcion = new byte[1024];
37             DatagramPacket paqueteRespuesta = new DatagramPacket(bufferRecepcion, bufferRecepcion.length);
38
39             socketCliente.receive(paqueteRespuesta);
40             String respuesta = new String(paqueteRespuesta.getData(), 0, paqueteRespuesta.getLength());
41
42             System.out.println("Respuesta del servidor: " + respuesta);
43
44             socketCliente.close();
45
46         } catch (IOException e) {
47             System.err.println("IOException en el cliente: " + e.getMessage());
48         }
49     }
50 }
```

```
2 Users > jorge > OneDrive > Documentos > NetBeansProjects > ServidorUDP > src > servidorudp > ServidorUDP.java
11
12 public class ServidorUDP {
13
14     public static void main(String[] args) {
15         final int PUERTO_SERVIDOR = 6000;
16         byte[] buffer = new byte[1024];
17
18         try {
19             // Asociamos el socket al puerto y a la IP local específica (0.0.0.0 para todas las interfaces)
20             InetAddress direccionLocal = InetAddress.getByName("0.0.0.0");
21             DatagramSocket socketServidor = new DatagramSocket(PUERTO_SERVIDOR, direccionLocal);
22
23             System.out.println("Servidor UDP escuchando en el puerto " + PUERTO_SERVIDOR);
24
25             while (true) {
26                 // Preparamos el paquete para recibir datos
27                 DatagramPacket paqueteRecibido = new DatagramPacket(buffer, buffer.length);
28                 socketServidor.receive(paqueteRecibido);
29
30                 String mensaje = new String(paqueteRecibido.getData(), 0, paqueteRecibido.getLength());
31                 System.out.println("Mensaje recibido: " + mensaje);
32
33                 // Enviar respuesta al cliente
34                 String respuesta = "Mensaje recibido correctamente";
35                 byte[] bufferRespuesta = respuesta.getBytes();
36                 InetAddress direccionCliente = paqueteRecibido.getAddress();
37                 int puertoCliente = paqueteRecibido.getPort();
38
39                 DatagramPacket paqueteRespuesta = new DatagramPacket(bufferRespuesta, bufferRespuesta.length, direccionCliente, puertoCliente);
40                 socketServidor.send(paqueteRespuesta);
41             }
42
43         } catch (IOException e) {
44             System.err.println("IOException en el servidor: " + e.getMessage());
45         }
46     }
47 }
```

```
run:
Cliente UDP conectado a localhost/127.0.0.1:3306
Escribe un mensaje (o 'salir' para terminar): hOLA
Mensaje enviado al servidor.
Servidor dice: Mensaje recibido: hOLA
Escribe un mensaje (o 'salir' para terminar):
```

```
run:
Servidor UDP iniciado en el puerto 3306
Cliente dice: hOLA
Respuesta enviada al cliente.
AD}
```