

## Sockets em Java - UDP

O exemplo abaixo mostra uma aplicação cliente servidor.

O cliente envia os dados recebidos do teclado e os envia ao servidor. Em seguida recebe uma mensagem do servidor e mostra no vídeo.

O servidor recebe mensagem do cliente, transforma as letras para maiúsculas e reenvia a mesma mensagem ao cliente.

O exemplo está no livro:

Redes de Computadores e Internet – uma nova abordagem

Kurose&Ross

Addison-Wesley

Fiz algumas adaptações. Se você quiser testar cliente e servidor na mesma máquina, modifique o endereço IP do servidor (no UDPClient para **127.0.0.1**):

**UDPClient.java**

```
// Lê uma linha do teclado
// Envia o pacote (linha digitada) ao servidor
// Lê um pacote (linha) do servidor e mostra no vídeo

import java.io.*; // classes para input e output streams e
import java.net.*; // DatagramaSocket, InetAddress, DatagramaPacket

class UDPClient {
    public static void main(String args[]) throws Exception
    {
        // cria o stream do teclado
        BufferedReader inFromUser =
            new BufferedReader(new InputStreamReader
                (System.in));

        // declara socket cliente
        DatagramSocket clientSocket = new DatagramSocket();

        // obtém endereço IP do servidor com o DNS
        InetAddress IPAddress =
            InetAddress.getByName("hostname");

        byte[] sendData = new byte[1024];
        byte[] receiveData = new byte[1024];

        // lê uma linha do teclado
        String sentence = inFromUser.readLine();
        sendData = sentence.getBytes();

        // cria pacote com o dado, o endereço do server e porta do servidor
        DatagramPacket sendPacket =
            new DatagramPacket(sendData, sendData.length, IPAddress, 9876);

        //envia o pacote
        clientSocket.send(sendPacket);
```

```
        // declara o pacote a ser recebido
        DatagramPacket receivePacket =
            new DatagramPacket(receiveData, receiveData.length);

        // recebe pacote do servidor
        clientSocket.receive(receivePacket);

        // separa somente o dado recebido
        String modifiedSentence =
            new String(receivePacket.getData());

        // mostra no vídeo
        System.out.println("FROM SERVER:" + modifiedSentence);

        // fecha o cliente
        clientSocket.close();
    }
}
```

## UDPServer.java

```
// Recebe um pacote de algum cliente
// Separa o dado, o endereço IP e a porta deste cliente
// Transforma em maiúscula
// Envia ao cliente, usando o endereço IP e a porta recebidos
// Volta ao início

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

class UDPServer {
    public static void main(String args[]) throws Exception
    {
        // cria socket do servidor com a porta 9876
        DatagramSocket serverSocket = new
            DatagramSocket(9876);

        byte[] receiveData = new byte[1024];
        byte[] sendData = new byte[1024];
        while(true)
        {
            // declara o pacote a ser recebido
            DatagramPacket receivePacket =
                new DatagramPacket(receiveData,
                    receiveData.length);

            // recebe o pacote do cliente
            serverSocket.receive(receivePacket);

            // pega os dados, o endereço IP e a porta do cliente
            // para poder mandar a msg de volta
            String sentence = new String(
                receivePacket.getData());
            InetAddress IPAddress =
                receivePacket.getAddress();
            int port = receivePacket.getPort();

            // transforma em maiúsculas
            String capitalizedSentence =
                sentence.toUpperCase();
            sendData = capitalizedSentence.
                getBytes();

            // monta o pacote com endereço IP e porta
            DatagramPacket sendPacket =
                new DatagramPacket(sendData, sendData.length,
                    IPAddress, port);

            // envia ao cliente
            serverSocket.send(sendPacket);
        }
    }
}
```