



# Trabalho de Redes de Computadores II

## Cliente/Servidor UDP SIMPLES Exemplo de utilização de sockets UDP em Java

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

```
public class servidorUDP { //acesso a rede pode gerar excecao  
    public static void main (String[] args) throws Exception { //o correto eh usar try/catch  
        DatagramSocket conexaoServidor = new DatagramSocket(6789); //conexao nao eh o melhor nome  
        byte[] dadosEntrada = new byte[1024];  
        DatagramPacket pacoteRecebido = new DatagramPacket(dadosEntrada, dadosEntrada.length);  
        conexaoServidor.receive(pacoteRecebido);  
        String mensagemRecebida = new String (pacoteRecebido.getData());  
        System.out.println(mensagemRecebida);  
    } //fim do metodo main  
} //fim da classe servidorUDP
```

**Código similar pode ser utilizado tranquilamente no seu trabalho**

```
public class clienteUDP { //acesso a rede pode gerar excecao  
    public static void main (String[] args) throws Exception { //o correto eh usar try/catch  
        DatagramSocket conexaoCliente = new DatagramSocket(); //conexao nao eh o melhor nome  
        InetAddress enderecoIPServidor = InetAddress.getByNome("192.168.1.2");  
        byte[] dadosSaida = new byte[1024];  
        String mensagemEnviada = new String("MUITO FACIL ;-"); //substituir por APDU+mensagem transmitida  
        saida = mensagemEnviada.getBytes();  
        DatagramPacket pacoteEnviado = new DatagramPacket(saida, saida.length,  
                                                         enderecoIPServidor, 6789);  
  
        conexaoCliente.send(pacoteEnviado);  
    } //fim do metodo main  
} //fim da classe clienteUDP
```

**ATENÇÃO ao sentido da mensagem neste exemplo: cliente->servidor. Como enviar do servidor->cliente?**



# Trabalho de Redes de Computadores II

## Cliente/Servidor TCP SIMPLES Exemplo de utilização de sockets TCP em Java

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

```
public class servidorTCP { //acesso a rede pode gerar excecao  
    public static void main (String[] args) throws Exception { //o correto eh usar try/catch  
        int portaLocal = 6789;  
        ServerSocket servidor = new ServerSocket(portaLocal); //socket do tipo TCP com conexao  
        Socket conexao = null;  
        conexao = servidor.accept();  
        ObjectInputStream entrada = new ObjectInputStream(conexao.getInputStream());  
        System.out.println((String)entrada.readObject());  
    } //fim do metodo main  
} //fim da classe servidorTCP
```

**Código similar pode ser utilizado tranquilamente no seu trabalho**

```
public class clienteTCP { //acesso a rede pode gerar excecao  
    public static void main (String[] args) throws Exception { //o correto eh usar try/catch  
        int porta = 6789;  
        String host = "192.168.0.1"; //substituir pelo IP do servidor  
        Socket s = new Socket(host, porta); //socket cliente ira se conectar ao socket servidor  
        OutputStream saida1 = s.getOutputStream();  
        ObjectOutputStream saida = new ObjectOutputStream(saida1);  
        String m = new String("MUITO FACIL ;-"); //substituir por APDU+mensagem transmitida  
        saida.writeObject(m);  
        saida.flush(); //forca envio da mensagem  
    } //fim do metodo main  
} //fim da classe clienteTCP
```

**ATENÇÃO ao sentido da mensagem neste exemplo: cliente->servidor. Como enviar do servidor->cliente?**