

TCP

1) Abrir um socket cliente

Socket Cliente;

Cliente = new Socket("nome da maquina", porta);

2) Abrir um socket servidor

ServerSocket Servidor = new ServerSocket(porta);

3) Para o servidor esperar por uma conexão de algum cliente

Socket S = Servidor.accept();

4) Para o cliente criar o duto de entrada para receber dados do servidor

DataInputStream entrada = new

DataInputStream(Cliente.getInputStream());

5) Para o servidor criar o duto de entrada para receber dados do cliente

DataInputStream entrada = new

DataInputStream(S.getInputStream());

6) Para o ler os dados (cliente ou servidor)

A classe DataInputStream tem algumas funções para ler os dados:

entrada.read()

entrada.readChar()

entrada.readInt()

entrada.readDouble()

entrada.readLine()

7) Para o cliente criar o duto de saída para enviar dados ao servidor

Pode ser usada a classe PrintStream or DataOutputStream

PrintStream saida = new

PrintStream(Cliente.getOutputStream());

ou então

DataOutputStream output = new

DataOutputStream(Cliente.getOutputStream());

8) Para o servidor criar o duto de saída para enviar dados ao cliente

Também podem ser usadas PrintStream ou DataOutputStream.

PrintStream saida = new

```
PrintStream(S.getOutputStream());
```

ou então

```
DataOutputStream saida =  
new DataOutputStream(S.getOutputStream());
```

9) Para enviar os dados (cliente ou servidor)

A classe PrintStream tem algumas funções para enviar os dados:

```
saida.Write()
```

```
saida.println()
```

Idem para a classe DataOutputStream.

```
saida.writebytes()
```

10) Para fechar o lado cliente

```
saida.close();
```

```
entrada.close();
```

```
Cliente.close();
```

11) Para fechar o lado servidor

```
saida.close();
```

```
entrada.close();
```

```
S.close();
```

```
Servidor.close();
```

Sockets UDP

[1.] Criar um socket do tipo datagrama

```
DatagramSocket Cliente = new DatagramSocket();
```

ou

```
DatagramSocket Servidor = new DatagramSocket(porta);
```

2) Obter um endereço IP de um host

```
InetAddress endIP = InetAddress.getByName("nome do host");
```

3) Montar um datagrama UDP de saída e enviar

```
DatagramPacket enviaPacote =
```

```
new DatagramPacket(Dados, Dados.length, endIP, porta);
```

```
Cliente.send(enviaPacote);
```

```
Servidor.send(enviaPacote);
```

4) Montar um datagrama UDP de entrada e recebê-lo

```
DatagramPacket recebePacote =  
new DatagramPacket(Dados, Dados.length);  
Cliente.receive(recebePacote);  
Servidor.receive(recebePacote);
```

5) Retirar os dados, a porta e o endereço IP do pacote recebido

```
String Dados = new String(recebePacket.getData());  
InetAddress endIP = recebePacote.getAddress();  
int porta = recebePacote.getPort();
```

Resumo Sockets TCP e UDP.doc

Mac4482s2005 – BCC - Marcilio

Resumo Sockets TCP e UDP.doc

Mac2121s2004 - BMAC - Marcilio