

# Atividade 2

**Monitoria InfraCom**

Josenildo Vicente(jva)

**Atenção:** Para as questões de 1 a 3, será necessário entregar a atividade pelo classroom sendo necessário “upar” para cada questão 2 (dois) arquivos .java (um cliente e outro servidor) nomeados indicando a questão (ex: questao1Cliente.java, questao1Servidor.java). **Não enviar arquivo zipado.** Para a questão 4, responda nesse mesmo documento e o envie também pelo classroom.

Nesta atividade, vocês produzirão programas do tipo Cliente-Servidor, explanando e exercitando os conceitos de sockets vistos na aula.

1. Faça um programa Cliente-Servidor utilizando o protocolo TCP através da porta 8877 em que o servidor deverá apresentar no console qualquer mensagem recebida do cliente durante os primeiros 10 segundos após o estabelecimento da conexão, depois disso a conexão com o cliente será encerrada. Após o tempo do cliente expirar ou caso ele caia, o servidor deve ficar apto novamente a receber uma outra conexão pela mesma porta.
2. Faça um programa Cliente-Servidor utilizando o protocolo TCP através da porta 8877 em que o cliente possa enviar e receber mensagens, lidas do teclado, do servidor (na ordem de, primeiro envia e depois fica esperando uma resposta do servidor para poder enviar outra mensagem) e o servidor possa também enviar e receber mensagens do cliente (na ordem de, primeiro fica esperando uma mensagem do cliente, para depois enviar uma resposta) podendo ter inúmeras trocas de mensagens enquanto estiver ativo/online.
3. Faça um programa Cliente-Servidor utilizando o protocolo UDP através da porta 8888 em que o servidor deverá apresentar no console todas as mensagens que recebe do cliente até que receba uma mensagem específica (“Tchau”), em que o servidor detecta automaticamente a mensagem e envia uma mensagem ao cliente (“Adeus”) para ser impresso no console do cliente esta mensagem e em seguida a conexão deverá ser encerrada.
4. Execute o programa a seguir e responda:



```

package lista;

import java.io.*;
import java.net.*;
import javax.swing.*;

public class Myprogram {

    String ip;
    String aux;
    int port;

    public void ExecuteTarefa() {
        ip = JOptionPane.showInputDialog(null, "entre com um IP:");
        aux = JOptionPane.showInputDialog(null, "entre com o
número das portas desejadas:");
        port = Integer.parseInt(aux);

        System.out.println("fazendo ...., aguarde" + "\n");

        try {
            InetAddress addr = InetAddress.getByName(ip);
            String hostname = addr.getHostName();

            for(int i=0;i < port; i++) {
                Socket s = null;

                try {
                    s = new Socket(addr,i);

                    System.out.println("Porta: " + i + " está
xxxx no IP " + hostname);
                }
                catch(IOException ex){
                    System.out.println("Porta: "+ i + " está
yyyy no IP" +hostname);
                }
                finally {
                    try {
                        if(s!=null) {
                            s.close();

```



```

        }
    }
    catch(IOException ex) {}
}
}
}
catch(UnknownHostException ex) {
    System.out.println(ex);
}

System.out.println("\n"+"Fim!");

}

public static void main(String[] args) {

    Myprogram sp = new Myprogram();
    sp.ExecuteTarefa();

}

}

```

- Para o que serve o programa acima? Explique qual seria o melhor significado para xxxx e yyyy.
- Esse tipo de programa pode ser usado para fins maliciosos? (Sim, Não). Explique.
- Execute o programa informando o IP 127.0.0.1. Pesquise para o que serve esse IP e explique o que acontece quando é usado no programa.

